International Journal of Computer Vision 70(2), 151–163, 2006

DOI: 10.1007/s11263-006-7936-3

Automatic Segmentation of the Left Ventricle in Cardiac MR and CT Images

MARIE-PIERRE JOLLY
Imaging and Visualization Department, Siemens Corporate Research, Princeton, NJ, USA
marie-pierre.jolly@siemens.com

Received April 3, 2003; Revised January 21, 2005; Accepted January 21, 2005

Abstract. This paper describes a segmentation technique to automatically extract the myocardium in 4D cardiac MR and CT datasets. The segmentation algorithm is a two step process. The global localization step roughly localizes the left ventricle using techniques such as maximum discrimination, thresholding and connected component analysis. The local deformations step combines EM-based region segmentation and Dijkstra active contours using graph cuts, spline fitting, or point pattern matching. The technique has been tested on a large number of patients and both quantitative and qualitative results are presented.

1. Introduction

Cardiovascular disease is the leading cause of death in the United States even if mortality has been declining over the years thanks to the development of new cardiac imaging technologies. To help in the diagnosis of disease, physicians are interested in identifying the heart chambers, the endocardium and epicardium, and measuring the change in ventricular blood volume (ejection fraction) and wall thickening properties over the cardiac cycle. The left ventricle is of particular interest since it pumps oxygenated blood out to distant tissue in the entire body. There are a number of techniques available to the physician to diagnose heart diseases and conditions. Both magnetic resonance (MR) and computer tomography (CT) imaging provide the physician with excellent quality images which allow a detailed analysis of the organ.

Argus is a cardiac analysis package developed by Siemens and sold on both MR and CT platforms which offers a complete system of drawing tools and automatic segmentation algorithms to allow the physician to outline the myocardium in each image in the patient data set, compute volumes, ejection fraction, and perform wall thickening analysis. In this paper, we present the segmentation module in Argus, which combines edge, region, and shape information in a deformable template approach. This is an extension of the work presented in Jolly (2001).

An exact description of the MR and CT acquisition processes is beyond the scope of this paper, but a brief description will help understand the entire system. An MR scanner acquires one image at a time. For each slice, multiple images (typically, 20 to 30) are acquired while the patient holds his/her breath. Since the system is connected to an electrocardiogram (ECG), it knows exactly how the heart beats, so it can combine all the images for one slice to reconstruct the equivalent of one heart cycle. Typically, 10 to 15 slices are acquired from base to apex. A CT scanner acquires a whole volume at a time. Again, the system is connected to an ECG and multiple volumes are acquired during multiple phases (about 10 with the new multi-array scanners). Currently, the volumes are reformatted manually into short axis slices.

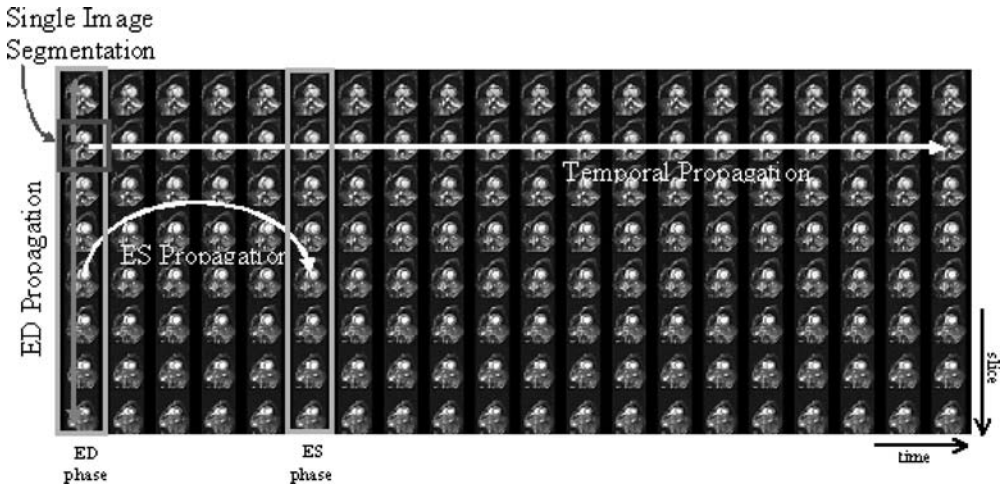

Figure 1. Typical workflow to automatically segment the myocardium.

The typical workflow is depicted in Fig. 1. The images are displayed in a matrix where the rows represent different slices and the columns different time instances. The first column corresponds to the end-diastole (ED) phase, when the heart is fully expanded. The end-systole (ES) phase is usually about one third down the temporal phases. The first slice is at the base of the heart (the highest part of the heart in the body) while the last slice is at the apex (bottom tip) of the left ventricle. The physician starts by segmenting one image at the base of the ventricle during ED phase and propagates the segmented contours to all the slices in the ED phase. This is called ED propagation. Then, all the ED contours are propagated to the ES phase during what is called ES propagation. These three steps are enough to compute the ejection fraction. For a more detailed analysis, the user can also propagate all the ED contours to all the other phases using temporal propagation. At all stages, the physician can correct the results of the automatic segmentation using the drawing tools.

There is a long history of publications on the segmentation of cardiac MR images. The system proposed by Fleagle et al. (1991) was able to delineate the borders of the myocardium using a minimum cost path graph search algorithm after the user indicated the center of the left ventricular cavity and the area of interest with a few mouse clicks. Geiger et al. (1995) used a dynamic programming approach to refine the contours specified by the user. Goshtasby and Turner (1995) proposed a two step algorithm combining intensity thresholding to recover the bright blood and local gradient to outline the strong edges using elastic curves. Weng et al. (1997) thresholded the image based on parameters estimated during a learning phase to get a good approximate of the segmentation. Although we have not directly used any of these techniques, all of these papers have greatly influenced our work. Another very interesting contribution to the segmentation of cardiac images is the work of a team at the Leiden University Medical Center (Mitchell et al., 2001). They use active appearance models to segment both the left and right ventricles. The algorithm that we propose is divided into two steps (see Fig. 2). In the global localization step, the system roughly localizes the left ventricle in the image. The local deformations step combines EM-based region segmentation and Dijkstra active contours using graph cuts, spline fitting, or point pattern matching.

The challenges in segmenting cardiac MR or CT images are multiple. First, the endocardium is not necessarily the sharpest boundary between the myocardium and the blood pool. In many cases, the physician wants to include the papillary muscles and trabeculations in the blood pool, resulting in an endocardium border slightly larger than expected by a novice. Second, the epicardium is even more difficult to recover because there are no image edges between the myocardium and the liver. Also, the myocardium splits around the left and the right ventricle. Finally, if the heart is surrounded by fat, the edge between the fat and the lungs is much stronger than the desired edge between the myocardium and the fat.

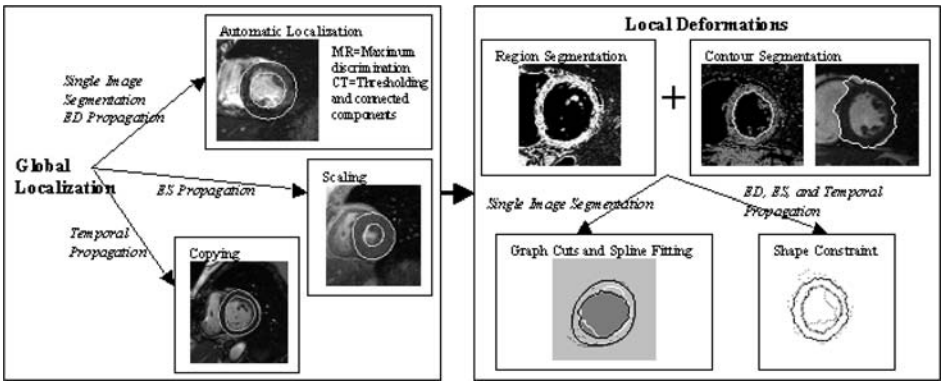

Figure 2. Flowchart of the overall algorithm. Note that different techniques are used for different segmentation tasks.

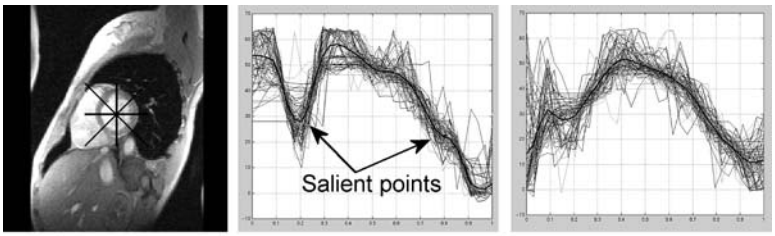

Figure 3. The feature set defining the heart left ventricle. (a) The four cross sections through the ventricle used to extract the features; Aligned training profiles along with their average for the (b) horizontal cross section and (c) vertical cross section.

2. Global Localization of the Left Ventricle

As for any deformable template algorithm, our method requires an approximate delineation of the object of interest to be provided. We have developed various strategies to perform this task automatically depending on the segmentation situation. As shown in the left part of Fig. 2, in the case of single image segmentation and ED propagation, we use an automatic localization technique that is based on maximum discrimination for MR images (Section 2.1) and thresholding for CT images (Section 2.2). For ES and temporal propagation, we simply copy or scale the contour from the previous image (Section 2.3).

2.1. *Maximum Discrimination for MR Images*

The gray level appearance of various tissues in MR varies greatly from patient to patient, acquisition to acquisition, and even image to image. However, in bright blood sequences (which are the most common for function assessment), the left ventricle roughly ressembles a dark donut. We have developed a method to encode this appearance and quickly locate instances in new images using the maximum discrimination method proposed by Colmenarez and Huang (1997). We will summarize the method here, but more details are given in Duta et al. (1999b) and Jolly et al. (2001).

Due to the relative symmetry of the left ventricle and computational constraints, we have only used the gray values of the pixels along the main four cross sections through the ventricle to derive the feature set (see Fig. 3(a)). Each cross section was sampled to contain 25 points, so that the donut texture is encoded as a 100-dimensional feature vector $x = x_1, \ldots, x_{100}$. To train the system, we used 101,250 positive examples and 123,096 negative examples. The system models the feature vector as a Markov process. The goal of the training was then to find the ordering of this Markov process that maximized the separation between positive and negative examples or minimized the Kullback

distance:

$$H_{P\|N} = \sum_{x\in\Omega} P(X = x) \log \frac{P(X = x)}{N(X = x)} \qquad (1)$$

between their respective distributions P and N. The task of minimizing the Kullback distance is equivalent to an asymmetric traveling salesman problem and can be solved using simulated annealing. As a result, the system determined the proper ordering $s_1, \ldots, s_{100}$ of the pixels along the 4 cross sections through the left ventricle as well as the Markov transition probabilities for the positive and negative distributions.

In addition, we collected 84 training sample profiles from 14 patients and computed the average profile for each of the 4 cross sections. Note that if the training profiles are not aligned, the punctual average is meaningless. Therefore, we used the curve warping method proposed by Ramsay and Li (1998) to align the profiles (see Figs. 3(b) and (c)). This allowed us to define 8 salient points in the gray level profiles as the intersection of the 4 cross sections with the myocardium medial axis.

During the detection stage a pixel is classified as left ventricle if the log-likelihood ratio of the feature vector $O = (o_1, \ldots, o_n)$ around it is positive:

$$L(o) = \ln \frac{P(X_{s_i} = o_{s_i})}{N(X_{s_i} = o_{s_i})} + \sum_{i=2}^{n} \ln \frac{P(X_{s_i} = o_{s_i} | X_{s_{i-1}} = o_{s_{i-1}})}{N(X_{s_i} = o_{s_i} | X_{s_{i-1}} = o_{s_{i-1}})} \qquad (2)$$

Each position is tested with multiple scales of the template. Since several neighboring positions can be classified as left ventricle, we partition them into clusters. The final choice among the cluster candidates is made using a Hough based voting procedure on the individual profiles warped onto their corresponding average profiles. The locations of the salient points in the image are accumulated in an array to vote for the most likely center and radius for the myocardium centerline. Since the automatic localization process returns a circle, the initial contours are then generated as two concentric circles by simply setting the thickness to 10 pixels.

The same automatic localization algorithm is used for ED propagation because different slices might have been acquired during a different breath hold position and the location of the left ventricle might have shifted quite a bit between slices. However, since the ventricle size is approximately known from the template image, we limit the scale search to 0.85 to 1.15 times the size of the template. We also limit the search space for the location of the ventricle to 30 pixels around the location of the template.

This process detects the left ventricle in 73% of the cases. However, most of the failures are in the ES phase (where the process is usually not used). In the ED phase, the left ventricle is correctly detected 92% of the time and the failures only occur around the apex. In many cases, we can limit the search space using the fact that the user cropped and zoomed the image to see details better and achieve 100% detection. Thus, in practical use, this process is very reliable.

2.2. *Thresholding for CT Images*

For CT images, the detection task is a lot easier because the gray level of a tissue is approximately known through the physics of the acquisition (X-rays). The gray levels represent the standard Hounsfield Unit which is proportional to the attenuation coefficient. In this representation, water is assigned a value of zero and air a value of -1000. We estimated from training examples that the blood pixels could be isolated by simply thresholding the image at 187. After removing small connected components, we compute the following features for each remaining connected components. The eccentricity is the ratio between the minimum and the maximum radii on the connected component. The convexity is the ratio between the area of the connected component and the area of its convex hull. Finally, the circularity of the convex hull is defined as 4π times its area over the square of its perimeter. We are looking for the connected component that is least eccentric, most circular, and most convex. The winning connected component is approximated by a circle and corresponds to the endocardium. The epicardium is generated as a circle with the same center and a larger radius. This technique is only used for single CT image segmentation.

2.3. *Scaling and Copying*

For temporal propagation, the contours are simply copied from one frame to the next because the position of the left ventricle does not change over time. Also, the aspect of the left ventricle barely changes

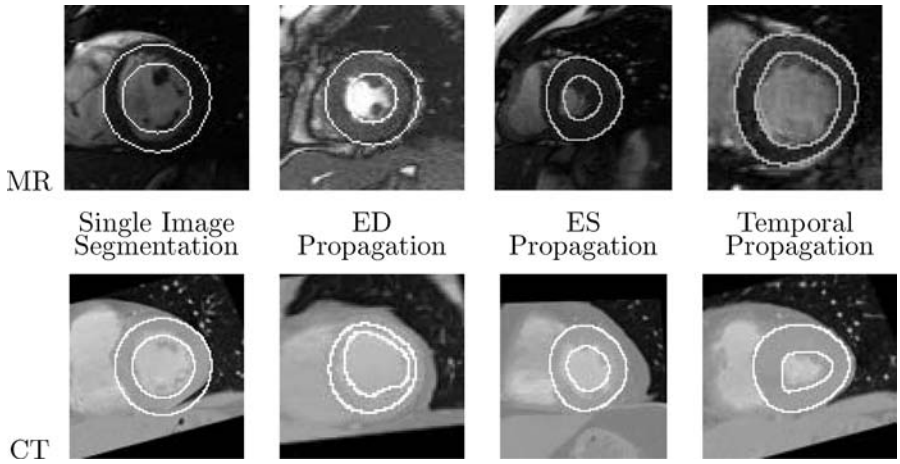

Figure 4. Examples of contours after global localization.

since there are typically less than 50ms between time frames. For ES propagation, the heart fully contracts between ED and ES, so the contours are scaled. Since the endocardium is known to contract more than the epicardium, the endocardium is scaled by 0.6 and the epicardium by 0.9. We can also use scaling for ED propagation in CT images, since the whole volume was acquired at once and we know there will not be any shift between slices. In this case, both the endocardium and epicardium are scaled by 0.8 down the slices and by 1.25 up the slices.

2.4. Examples

Figure 4 shows some localization examples either through automatic localization or propagation. It can be seen that in all cases, the starting point to the local deformations process is pretty good. Even in case where the global localization starting point is not great (as in CT ED propagation in Fig. 4), the local deformations can usually recover.

3. Local Deformations

The local deformations first combine region segmentation (Section 3.1) and active contours (Section 3.2), as shown in the right part of Fig. 2. In the case of propagation, we use a shape matching algorithm (Section 3.3) because the shape of the left ventricle is known from the previous image. In the case of single image segmentation, we use graph cuts and spline fitting (Section 3.4).

3.1. Region Segmentation

In most medical imaging modalities, the intensity of a pixel depends on the properties of the tissue being imaged. In our MR sequences, both blood and fat are very bright and inhomogeneous, the myocardium is somewhat dark, but not as dark as the air-filled lungs. This fact can be verified by looking at the histogram of a region around the myocardium (see Fig. 5). We use the Expectation-Maximization (EM) algorithm (Redner and Walker, 1984) to fit a mixture of 3 Gaussians to

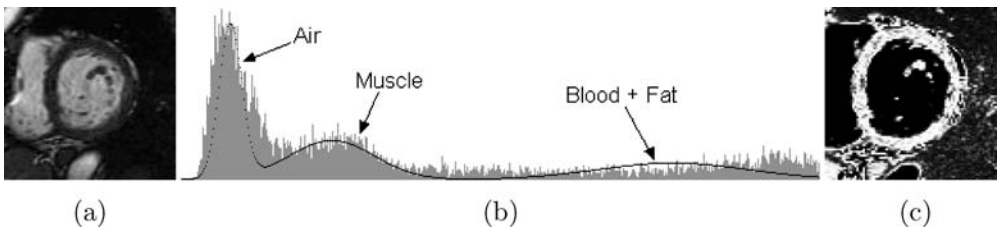

Figure 5. Region segmentation for MR images. (a) input image; (b) histogram; (c) myocardium response image.

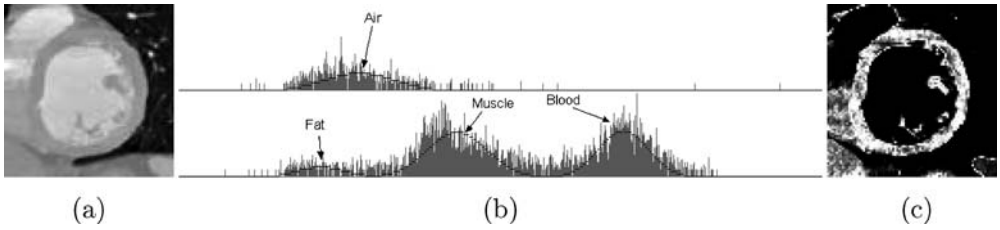

Figure 6. Region segmentation for CT images. (a) input image; (b) histogram (shown in two parts); (c) myocardium response image.

the histogram. We then create a myocardium response image by computing the probability that a pixel belongs to the middle Gaussian distribution which corresponds to the myocardium. It can be seen from Fig. 5(c) that the left ventricle myocardium is nicely highlighted.

The gray level distribution for CT images is pretty different as can be seen from Fig. 6. The range of the gray levels in much higher, typically, from −900 to 600 (for MR images, it is more like 0 to 700). Also, the histogram shows four peaks: the lungs are very dark and well separated from the other organs. Among the brighter pixels, there is fat and parts of the liver, muscle, and blood. Again, we use the EM algorithm, but now, fit 4 Gaussians to the histogram. The physics of the CT acquisition allowed us to determine the average myocardium gray level empirically. We simply choose the Gaussian peak closest to this average of 84. The myocardium response image is then computed in the same way. Again, Fig. 6(c) shows that the myocardium is nicely highlighted.

3.2. *Active Contours*

To complement the results of region segmentation, we use an active contour formulation similar to (Geiger et al., 1995) or (Mortensen and Barrett, 1998) approach. The advantage of these graph theoretic methods over the traditional gradient descent approach proposed by Kass et al. (1988) is that they are able to recover the global optimum of the energy function and are therefore insensitive to the initial contour position. The energy function (or cost) of a contour defined by the sequence of pixels $p_1, p_2, \ldots, p_N$ is defined as:

$$E(p_1, p_2, \ldots, p_N) = \sum_{i=1}^{N} e(p_i, p_{i+1}) \qquad (3)$$

Mortensen and Barrett (1998) showed that this energy function can be minimized using Dijkstra's algorithm when we define a graph where nodes correspond to pixels and $e(p, q)$ is the cost of the edge connecting pixels p and q. Dijkstra's algorithm was designed to find the shortest path between one source and all the other nodes in a graph. To recover a closed contour, we use two passes of the algorithm. Given an approximate contour in an image as in Fig. 7(a), we place a symmetrical search space around it and define a line of source nodes (all connected to a pseudo source node) and sink nodes as in Fig. 7(b). Dijkstra's algorithm then finds the shortest path between the pseudo source node and one of the sink nodes as seen in Fig. 7(c). Since there is no guarantee that this contour will be closed, in the second pass we define a new single source point half way around the recovered contour as in Fig. 7(d)

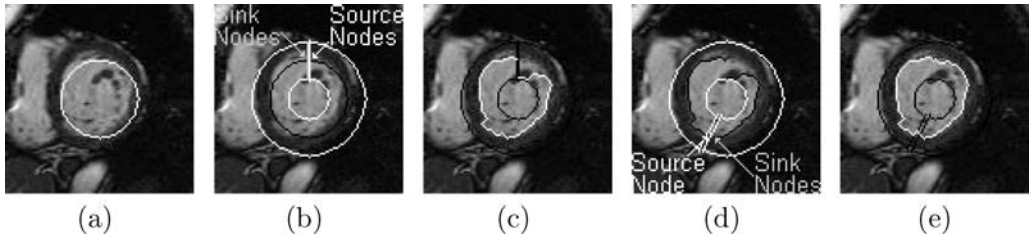

Figure 7. Two passes of Dijkstra's algorithm are used to recover a closed contour. In the first pass (a)–(c), the source and sink nodes are defined along the entire width of the search space. In the second pass (d)–(e), the source is the contour point half way around the contour.

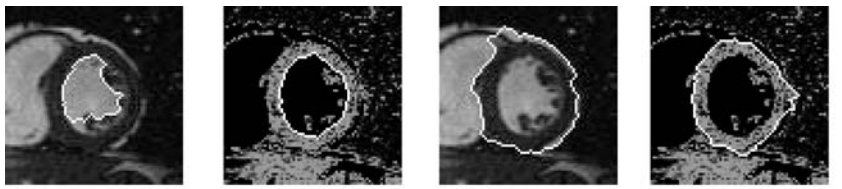

Figure 8. Various energy functions can be used to recover a contour using Dijkstra's algorithm.

and run Dijkstra's algorithm again to produce the final closed contour as in Fig. 7(e).

To combine the information provided by both the image and the myocardium response image, we have chosen to run Dijkstra's algorithm independently on both images. Each run gives different candidate points for the contours. For each contour point, we also define its confidence as the inverse of its contribution to the total energy function:

$$w(p_i) = \frac{1}{E(p_i) - E(p_{i-1})} = \frac{1}{e(p_{i-1}, p_i)} \quad (4)$$

The simplest energy function defines the cost of a link between two pixels p and q as:

$$e_1(I, p, q) = \frac{1}{\|\nabla I(q)\|^2 + \epsilon} \quad (5)$$

where $\|\nabla I(q)\|^2$ is the image gradient magnitude at pixel q and ϵ is a small constant (0.001) to bound the energy function. Even though the images look fairly noisy, the Sobel edge detector is sufficient to compute the gradient magnitude and direction. We also define another energy function using:

$$e_2(I, z > 0, p, q) = \begin{cases} \dfrac{1}{\|\nabla I(q)\|^2 + \epsilon} & \text{if} \quad z > 0 \\ 1/\epsilon & \text{otherwise} \end{cases} \quad (6)$$

where z is the Z-component of the cross product between $\vec{pq}$ and $\vec{\nabla} I(q)$. This function is designed so that the resulting contour encloses a bright region and is surrounded by a dark region. This is used to recover the endocardium in the input image. To recover the endocardium in the myocardium response image however, the sign of the test on the cross product has to be reversed, so we use $e_2(H, z < 0, p, q)$. To recover the epicardium in the myocardium response image, we use $e_2(H, z > 0, p, q)$. In the input image however, the epicardium is surrounded by a bright region along the septum and either a bright region (fat) or a dark region (lungs) along the lateral wall so, we simply use $e_1(I, p, q)$.

In Fig. 8 it can be seen that different energy functions highlight different features of the myocardium. In particular, for the endocardium $e_2(I, z > 0, p, q)$ outlines the papillary muscles while $e_2(H, z < 0, p, q)$ does not. Also, for the epicardium $e_1(I, p, q)$ outlines the fat while $e_2(H, z > 0, p, q)$ stays closer to the myocardium.

3.3. *Shape Constraint*

When the contours are propagated from one image to the next, either spatially or temporally, the shape of the contours does not change drastically. We thus define a template consisting of all the points on the contours in the previous image. All points on contours resulting from Dijkstra's algorithm in the current image are candidate points for the final segmentation. The goal is then to align the template to the data points. We use the shape alignment method proposed by Duta et al. (1999a) to establish the correspondence between a subset A' of the template points $A = \{A_j\}$, $j = 1, \ldots, n$ and a subset B' of the candidate points $B = \{B_k\}$, $k = 1, \ldots, m$.

The goal is to find the coefficients (a, b, c, d) of the similarity transform which minimize the distance $f(N_c)$.

$$f(N_c) = \frac{1}{N_c^2} \sum_{j=1}^{N_c} w(B_j)\big[\big(x_{A_j} - ax_{B_j} - cy_{B_j} - b\big)^2 + \big(y_{A_j} - ay_{B_j} - cx_{B_j} - d\big)^2\big] + \frac{2}{N_c} \quad (7)$$

where N_c is the number of established correspondences. We use the term $2/N_c$ to avoid the trivial solution with no correspondences. We also weigh the distance by the confidence $w(B_j)$ of candidate point B_j

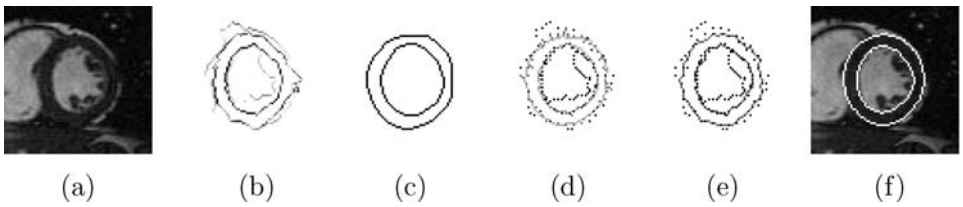

Figure 9. Applying shape constraints to recover the myocardium contours: (a) input image; (b) the 4 recovered contours where darker points show higher confidence; (c) template from contours in the previous image; (d) best similarity transform; (e)warped template; (f) final segmentation.

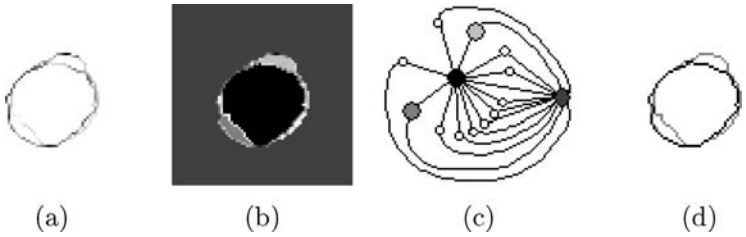

Figure 10. Best cycle for the endocardium: (a) confidence image; (b) connected regions between boundary parts; (c) corresponding graph; (d) the final segmentation is shown in dark.

as defined in Eq. (4). Given two pairs of corresponding points, the similarity transform is computed and applied to the entire template. Neighboring template and data points become matching pairs and $f(N_c)$ can be evaluated. We examine the 10% of candidate points with largest confidence to find the similarity transform which minimizes $f(N_c)$. After warping the template by moving its points to their corresponding data points, the contours are smoothed using the method proposed by Xu et al. (2000) that minimizes shrinkage.

3.4. No Shape Constraint

In the case of single image segmentation, there is no shape information available to the system. To obtain the final contour, we use the technique from our previous version of the algorithm presented in Jolly et al. (2001). The endocardium and epicardium are treated separately.

In the case of the endocardium, since Dijkstra's algorithm favors smaller contours and we want to make sure that the final contour goes behind the papillary muscles, we modify the confidence value at every contour point as follows:

$$w_{\text{endo}}(p) = w(p)\left(\frac{d(p,\Omega)}{d_{\max}}\right)^3 \tag{8}$$

where Ω is the centroid of the candidate contour points, $d(p,\Omega)$ is the distance between a candidate point p and the centroid, and $d_{\max}$ is the maximum distance over all candidate points (to normalize the weighting coefficient). This will emphasize points that are far away from the centroid to try and make the contour as large as possible.

We would like to find the best closed contour that separates the center of the data points from the border of the image. Figure 10 illustrates the algorithm. We define a graph where the nodes correspond to regions inside the boundary pixels in the data points. The edges between two nodes encode the confidence of the boundary between two regions, more precisely, it is the sum of the confidence w_{endo} of all the points on the boundary between the two regions. We then use a graph cut algorithm based on the max flow algorithm (Ford and Fulkerson, 1962) to cut the graph. The cut edges in the graph form a cycle equivalent to a closed contour.

The segmentation for the epicardium is more difficult because there is usually no clear edge between the myocardium and the liver, and the right ventricle merges into the left ventricle. First, we update the confidence measure to emphasize points that are closer to the endocardium so that we do not segment the fat and the boundary does not get lost where contours are very faint.

$$w_{\text{epi}}(p) = w(p)\left(\frac{D_{\max} - D(p,\text{endo})}{D_{\max}}\right)^3 \tag{9}$$

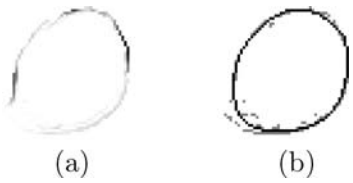

Figure 11. Spline fit for the epicardium: (a) confidence image; (b) the final segmentation is shown in dark.

where $D(p, \text{endo})$ is the distance between candidate point p and the closest point on the endocardium and $D_{\max}$ is the maximum distance for all candidate points. To keep the result smooth, we then fit a spline through the candidate points using the least squares spline fitting technique described in Bartles et al. (1987) (Section 21.5). More details are given in Jolly et al. (2001). Figure 11 shows the result of spline smoothing.

4. Results

The algorithm was tested on both MR and CT images independently. Data was obtained from various institutions, where for each dataset, an expert manually outlined the myocardium in all slices of ED and ES. We were not able to gather more than one manual tracing per dataset to evaluate inter- and intra-expert variability. Instead, we compared our contours with manual tracings in the following manner. We ran our algorithm to automatically segment the ED and ES phases of all the datasets. To compare an automatic contour A with a true contour B, we computed the distances $d(a, B) = \min_{b \in B} \|a - b\|$ for all points a in the automatic contour. Similarly, we computed $d(b, A)$ for all points b in the true contour. We then plotted the cumulative distribution of these distances for all contour points, all images, and all patients. We believe that this evaluation method provides more information about the actual accuracy of the algorithm. The ejection fraction (EF) or the volumes are much more forgiving, in the sense that the contours could be consistently wrong and still produce a reasonable ejection fraction. The automatic segmentation was also evaluated in clinical settings. Crowe et al. (2002) studied a previous version of the algorithm and showed that the method estimated volumes within 5%, mass within 10%, and EF within 2.5% of manually drawn contours. More recently, Francois et al. (2004) demonstrated that the myocardial mass computed from automatically generated contours was within 5% of the true myocardial mass.

The algorithm is relatively fast: it takes about half a second per image on a 1GHz Pentium 4. Thus, the ejection fraction can be obtained in less than 10 seconds after automatic segmentation of both ED and ES phases. An entire dataset can be segmented in about a minute and a half after which volume/time curves can be displayed to the user. The slowest part of the algorithm is the shape matching component.

4.1. Results on MR Images

Our algorithm was first tested on MR images. We collected 29 patient data sets along with a manual segmentation of the ED and ES phases, for a total of 482 segmented images. The images were acquired on Siemens MAGNETOM systems using two different pulse sequences. FLASH pulse sequences were traditionally used for MR, but Siemens has recently pionneered the TrueFISP pulse sequences for cardiac cine imaging which present higher contrast-to-noise ratio without affecting temporal or spatial resolution. We collected 22 TrueFISP patients and 7 FLASH patients. Our database presents a great variety of heart shapes, image contrast, and edge crispness. The difficulty with FLASH images (Figs. 15 or 16(b)) is that the edges are very blurred. The challenge with TrueFISP images is that the papillary muscles are so well defined that it can be difficult to avoid outlining them.

Figure 12 shows the cumulative distribution of the error distances over all ED and ES frames of all

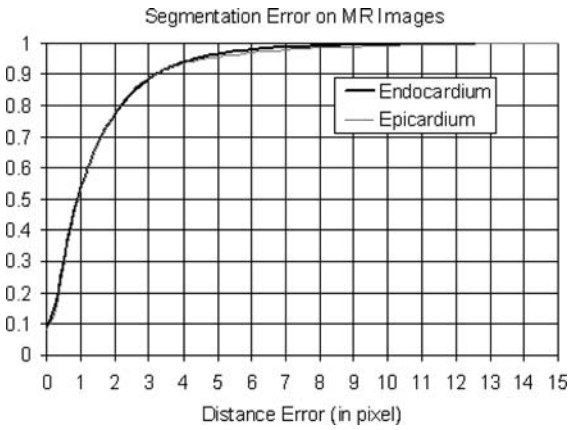

Figure 12. Cumulative distribution of error distances between the true contours and the segmented contours, over all points, all images, and all patients for MR images.

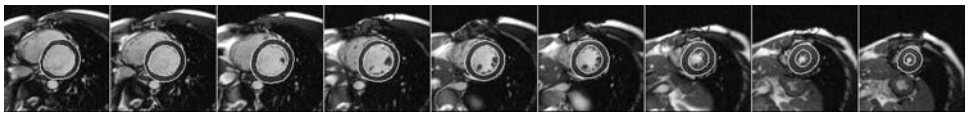

Figure 13. Automatic localization followed by ED propagation for all the slices of the ED phase.

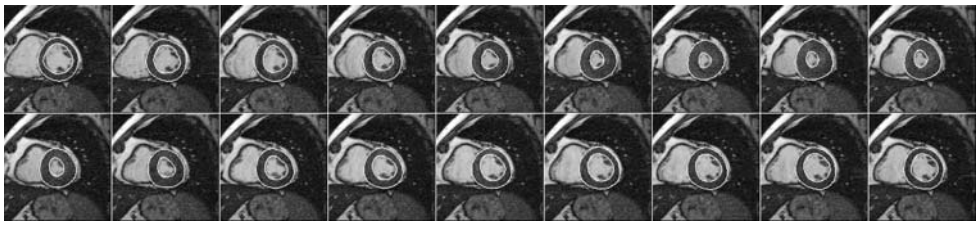

Figure 14. Temporal propagation to all the phases in one slice.

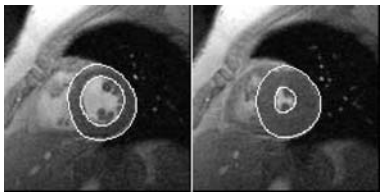

Figure 15. ES propagation.

29 datasets. It can be seen that, on the average, the error is less than 1 pixel and an error of 5 pixels or more is made less than 5% of the time. Figure 13 shows the segmentation of all the slices in an ED phase. Figure 14 shows a temporal propagation for all the phases of a particular slice. It can be seen that the papillary muscles are kept inside the blood pool for all phases so that the shape of the endocardium is close to the shape defined in the ED phase. Figure 15 shows an example of ES propagation. Figure 16 shows other segmentation examples which demonstrate the strength of our algorithm in handling different cases.

Figures 17 and 18 illustrate the importance of the shape constraint. In Figure 17(a) the contours, obtained after single image segmentation, are wrong. the correct segmentation is shown in Fig. 17(b). This patient has suffered an infarct in the past and the lower right side of the myocardium is very thin. In addition, there is no clear edge in the image between the myocardium and the liver. Figure 17(c) shows the previous slice in the volume where we have defined the desired contour manually. When the contour is propagated to the next slice, the shape constraint is enforced and the segmentation result is correct (as seen in Fig. 17(d)). Figure 18 demonstrates that whether the user (a) outlined the papillary muscles or (c) did not, the contour propagated to the next image retains the proper shape (b) with or (d) without papillary muscles.

4.2. Results on CT Images

We also tested our algorithm on CT images. We acquired 18 patient datasets on the Siemens SOMATOM

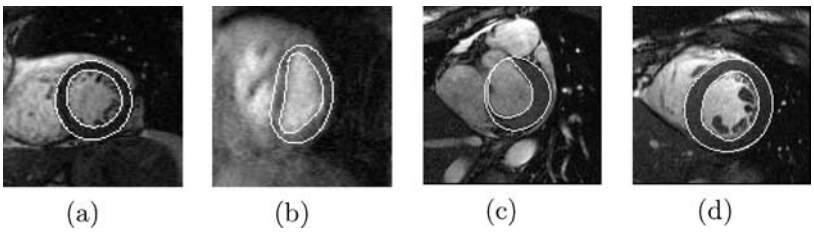

Figure 16. Other segmentation examples: (a) large amount of fat around the myocardium; (b) elongated cross section of the left ventricle; (c) endocardium and epicardium coincide at the valve plane; (d) thick myocardium and large papillary muscles.

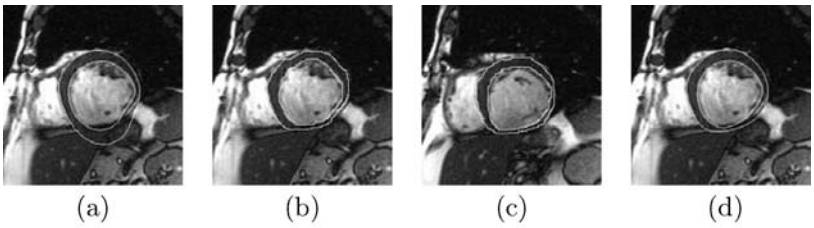

Figure 17. Shape constraint in propagation: (a) wrong result from single image segmentation; (b) correct segmentation; (c) manual tracing in the previous frame; (d) correct result from propagation.

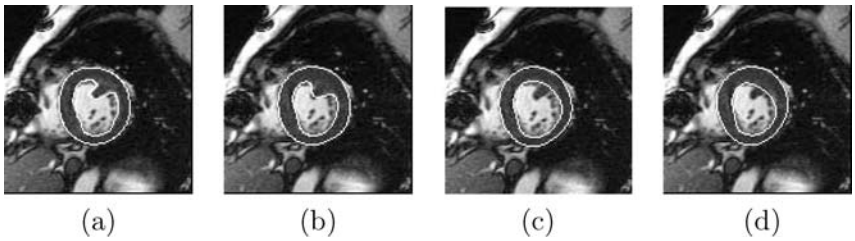

Figure 18. Shape constraint in propagation: (a) manual tracing including papillary muscles; (b) propagation to the next frame; (c) manual tracing excluding papillary muscles; (d) propagation to the next frame.

Sensation 4 and Sensation 16 CT scanners using helical scanning. The Sensation 16 scanner is a newer scanner, it is faster, more versatile and produces sharper images with less jagged edges. We had 6 datasets from a Sensation 16 scanner and 12 datasets from a Sensation 4. Since a CT scanner acquires an entire volume of data, we had to reformat each volume into slices. We chose to use multi-planar reformatting (MPR) to generate 8 mm thick slices, which is the thickness of a typical MR slice. Again, we collected ground truth contours for the ED and ES slices. This gave us 338 segmented images.

Figure 19 shows the cumulative distributions for the error distances. It can be seen that the system makes an average error of 2.5 pixels. And for 95% of the contour points, the error is less than 15 pixels. It might seem that the errors in CT images are much worse than the errors in MR images. However, typically, the physical size of a pixel in an MR image is about 3 times larger than in a CT image, so overall, the performance of the algorithm is about the same on MR and CT images.

Figure 20 shows some examples of the segmentation on CT images. For all examples, we show a particular slice with the ED phase on the left and the ES phase on the right. Both were segmented automatically. Since the slice images are obtained from the whole CT volumes, there might be cases where the right ventricle is actually on the right side of the left ventricle (see Fig. 20(f)). Nevertheless, the algorithm handles the situation well

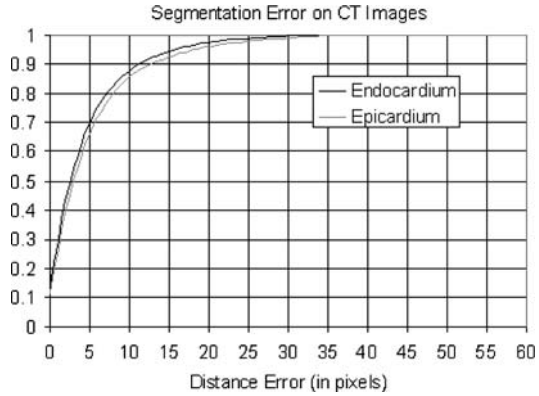

Figure 19. Cumulative distribution of error distances between the true contours and the segmented contours, over all points, all images, and all patients for CT images.

because the energy function for the epicardium does not use gradient direction.

5. Conclusions

We have presented an algorithm to segment the left ventricle in cardiac MR and CT datasets. The algorithm combines EM-based region segmentation, Dijkstra active contours, and shape information through a point pattern matching strategy. This algorithm was first developed to segment the left ventricle on short axis

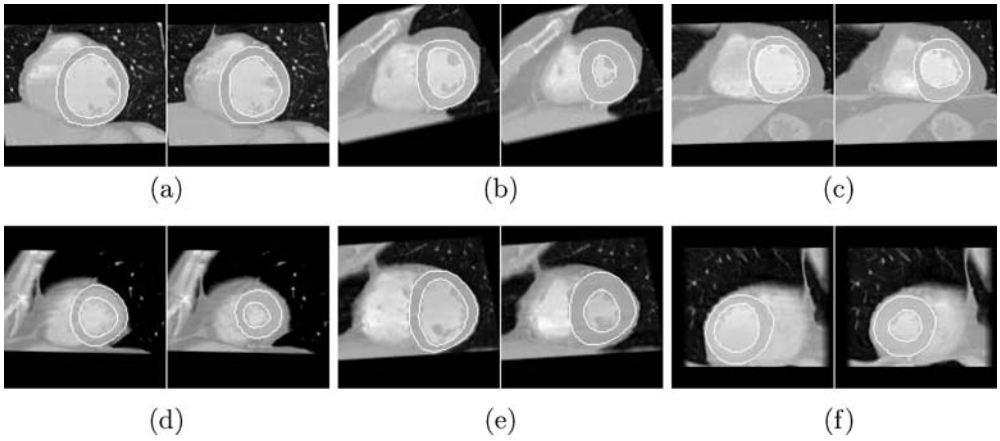

Figure 20. Examples of segmentation results for CT images.

slices in MR. It was then extended to CT with minimal adjustment to produce similar performance accuracy. We have obtained excellent results with this technique and integrated this algorithm with the latest version of the cardiac analysis package Argus commercialized by Siemens. Argus is now being widely used in clinical settings because the algorithm is fast and accurate enough. Currently, the performance of the algorithm is limited by its 2D nature. However, MR datasets are getting denser spatially and include long axis slices, and CT datasets are getting denser temporally. Thus, the next step is to explore a 3D or 4D approach for the segmentation of the left ventricle.

Acknowledgments

We would like to thank our student collaborators, Nicolae Duta for designing the automatic localization for MR images and Ying Sun for implementing the automatic localization for CT images. Of course, this work would not have been possible without the support of our collaborators at Siemens Medical Solutions, namely, Orlando Simonetti and Jeffrey Bundy who provided the MR data with ground truth and Matthias Niethammer who provided the CT data with ground truth. We would also like to acknowledge our colleagues at Siemens Corporate Research and in particular Gareth Funka-Lea for fruitful discussions.

References

Bartles, R.H., Beatty, J.C., and Barsky, B.A.. 1987. *An Introduction to Splines for Use in Computer Graphics & Geometric Modeling*. Morgan Kaufmann Publishers.

Colmenarez, A. and Huang, T. 1997. Face detection with information-based maximum discrimination. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*. San Juan, Puerto Rico, pp. 782–787.

Crowe, M., Jolly, M.-P., Vargas, J., and Simonetti, O. 2002. Assessment of a method to generate myocardial contours for left ventricle analysis without user input. In *Society for Cardiovascular Magnetic Resonance Annual Meeting*.

Duta, N., Jain, A.K., and Dubuisson-Jolly, M.-P. 1999a. Learning 2D shape models. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Vol. II. Fort Collins, CO, pp. 8–14.

Duta, N., Jain, A.K., and Dubuisson-Jolly, M.-P. 1999b. Learning-based object detection in cardiac MR images. In *Proc. International Conference on Computer Vision*. Corfu, Greece, pp. 1210–1216.

Fleagle, S.R., Thedens, D.R., Ehrhardt, J.C., Scholz, T.D., and Skorton, D.J. 1991. Automated identification of left ventricular borders from spin-echo resonance images. *Investigative Radiology* 26:295–303.

Ford, L. and Fulkerson, D. 1962. *Flow in networks*. Princeton University Press.

Francois, C.J., Fieno, D.S., Shors, S.M., and Finn, J.P. 2004. Left ventricular mass: Manual and automatic segmentation of trueFISP and FLASH cine MR images in dogs and pigs. *Radiology* 230(2):389–395.

Geiger, D., Gupta, A., Costa, L.A., and Vlontzos, J. 1995. Dynamic programming for detecting, tracking, and matching deformable contours. *IEEE Trans. Pattern Analysis and Machine Intelligence* 17(3):294–302.

Goshtasby, A. and Turner, D.A. 1995. Segmentation of cardiac cine MR images for extraction of right and left ventricular chambers. *IEEE Trans. Medical Imaging* 14(1):56–64.

Jolly, M.-P. 2001. Combining edge, region, and shape information to segment the left ventricle in cardiac MR images. In *Proc. Medical Image Computing and Computer-Assisted Intervention*. Utrecht, The Netherlands, pp. 482–490.

Jolly, M.-P., Duta, N., and Funka-Lea, G. 2001. Segmentation of the left ventricle in cardiac MR images. In *Proc. International Conference on Computer Vision*, Vol. I. Vancouver, Canada, pp. 501–508.

Kass, M., Witkin, A., and Terzopoulos, D. 1988. Snakes: Active contour models. *International Journal of Computer Vision* 2:321–331.

Mitchell, S.C., Lelieveldt, B.P.F., van der Geest, R.J., Bosch, H.G., Reiber, J.H.C., and Sonka, M. 2001. Multistage hybrid active appearance model matching: Segmentation of left and right ventricles in cardiac MR images. *IEEE Trans. Medical Imaging* 20(5):415–423.

Mortensen, E.N. and Barrett, W.A. 1998. Interactive segmentation with intelligent scissors. *Graphical Models and Image Processing* 60:349–384.

Ramsay, J.O. and Li, X. 1998. Curve registration. *Journal of the Royal Statistical Society Series B* 60:351–363.

Redner, R.A. and Walker, H.F. 1984. Mixture densities, maximum likelihood and the EM algorithm. *SIAM Review* 26:195–239.

Weng, J., Singh, A., and Chiu, M.Y. 1997. Learning-based ventricle detection from cardiac MR and CT images. *IEEE Trans. Medical Imaging* 16(4):378–391.

Xu, C., Jr., A.Y., and Prince, J.L. 2000. On the relationship between parametric and geometric active contours. In *Asilomar Conf. Signals, Systems, and Computers*. pp. 483–489.