Skeleton-Based Myocardium Segmentation

André Neubauer and Rainer Wegenkittl

VRVis Research Center, Vienna, Austria

ABSTRACT

Computer-aided analysis of four-dimensional tomography data plays an increasingly vital role in the diagnosis and treatment of heart function deficiencies. A key task for understanding the dynamics involved within a recorded cardiac cycle is to segment the acquired data to identify objects of interest, like the heart muscle (or myocardium) and the left ventricle.

In this paper, a new robust and fast semi-automatic algorithm for segmentation of the myocardium from a CT data set is presented. The user marks the myocardium by placing a poly-line onto one slice of the data volume. This poly-line forms a skeleton representing the cross-section of the myocardium on this slice. This skeleton is then automatically propagated and adjusted to the other slices in order to create a skeleton of the entire heart muscle. Then a cost function is applied, which calculates for each voxel the cost of the cheapest path from the voxel to the skeleton, with the cost of a path being determined by its length and the data it passes through. The boundaries of the myocardium can then be extracted as an iso-surface in the volume generated by this cost function.

Keywords: Manuscript format, template, SPIE Proceedings, LaTeX

1. INTRODUCTION

With the introduction of four-dimensional tomographic imaging techniques, it has become possible to evaluate important properties of a patient's heart function by generating images at a number of time points distributed over one cardiac circle. Abnormalities in myocardial motion, which can indicate deficits in blood perfusion, can be extracted by analyzing the resulting image sequence. Interest of cardiologists is usually focussed on the heart muscle (myocardium). It encloses the left ventricle (LV) of the heart and, by contracting and expanding, makes blood diffuse out of the ventricle into the entire body. Properties of the myocardium and the LV as a function of both time and space reflect the state of the patient's cardiac system very well. Crucial properties include the change of myocardial wall thickness over time as well as the volume of the blood perfused during one cardiac cycle. Image segmentation is an important sub-task in the process of computer-aided analysis and assessment of cardiac activity. Objects of interest (here: the myocardium and the LV) must be identified and segmented within each 3D volume in order to be able to exract properties of their motion. Since a correct segmentation of the myocardium leads to trivial segmentation of the LV, which is simply the interior of the heart muscle, the important first step in the automatic or semi-automatic processing of 4D cardiac tomography data is to segment the myocardium.

Both Magnetic Resonance Imaging (MRI) and Computer Tomography (CT) are widely used to acquire 4D cardiac data. Both have their advantages and disadvantages. The most significant advantage of MRI in cardiac imaging is that MRI scanners expose the patients solely to magnetic fields instead of radiation, while a CT scanner uses x-ray for data aquisition and thus imposes some risk to the patient's health. On the other hand, CT can generate images of higher quality: Recent developments in the area of Computer Tomography allow for ultra-fast high quality scanning at high resolutions. Electronic Beam Computer Tomography (EBCT), a very fast CT-based method was introduced in 1984. However, the high cost of the systems limits its availability to relatively few sites. Multi-Slice Computer Tomography (MSCT) was developed in 1998.[1] Multi-slice technology accelerates the scanning process by simultaneously acquiring 4 (or, since recently, even 16) slices. Therefore it became possible to generate 4D CT images of high quality, which are highly resolved in all three spatial

Further author information: (Send correspondence to A.A.A.)
A.A.A.: E-mail: aaa@tbk2.edu, Telephone: 1 505 123 1234
B.B.A.: E-mail: bba@cmp.com, Telephone: +33 (0)1 98 76 54 32, Address: A Very Long Address, City, Country

dimensions *and* the time dimension. This is currently not possible with MRI. The better quality of CT images, compared to MRI scans, allows for a higher level of accuracy in image segmentation, which can be crucial for the task of analysis of cardiac function.

This paper deals with segmentation of the myocardium from a three-dimensional CT scan. Section 2 presents previous approaches. Section 3 describes the new technique, skeleton-based myocardium segmentation.

2. RELATED WORK

Myocardium segmentation techniques mostly work in a semi-automatic fashion: Approaches with a high level of interactivity include the intelligent scissors and live-wire[2, 3] techniques. Here, the user specifies points of the boundary, which are then automatically connected by the segmentation algorithm. Sometimes, only simple segmentation techniques, like, for instance, thresholding and object extraction using mathematical morphology, are applied and the results are manually edited.[4] However, in order to increase the efficiency in the radiologists' work, manual interaction in the segmentation process is required to be minimized. The most widely used methods for myocardium segmentation are deformable model techniques[5]: A deformable boundary is initialized and, by applying a set of different forces, aligned to significant gradients in the image. However, often those techniques require the initialized boundary being very close to the final expected result, which often increases the level manual interaction. Popular deformable models include, for instance, energy-minimizing splines called *snakes*.[6] Methods used to initialize and refine deformable models are still subject to research: Jones and Metaxas[7] use pixel-affinity (a measure of probability that two neighboring pixels belong to the same object) and balloon forces[8] to initialize a deformable model. The LV boundary is initialized in the middle of the LV and iteratively enlarged into all directions until pixels of low affinity are reached. The problem is that noise and papillary muscles (small muscles inside the LV) might lead to bad initialization. Chakraborty and Duncan introduced an approach to refine an initialized boundary by methods based on game theory.[9] This technique attempts to combine both region-based and boundary-based segmentation to achieve to best possible results. Two modules, or 'players', with different strategies, one region-based, the other boundary-based work in a coordinated fashion towards the common goal, the correct object segmentation. Techniques have been proposed, which segment the LV and in each slice initialize the outer boundary of the myocardium (the epicardium) as a shape similar to the boundary of the LV (the endocardium) and then refine this estimation to better fit the data,[10] using a deformable model. Usually these techniques miss a significant part of the myocardium, since not every slice that intersects the myocardium also intersects the LV. In recent years, methods have been proposed, which perform automatic segmentation of the heart based on atlases.[11] Data sets acquired from a number of patients or volunteers form a training set. They are abstracted to common features which are collected to an atlas and aid the automatic segmentation of new data sets. Research in this area, however, is still young. Current systems are either restricted to 2D images or have trouble segmenting data sets where structures are significantly different to those encountered in the training set. An interesting atlas-based technique that robustly segments the LV is the deformable template method by Rueckert and Burger.[12] Here, the concept of deformable models is enriched by using atlas-based a-priori knowledge about the shapes of 2D cross-sections of the ventricle. Another technique for fully automatic segmentation has been proposed by Spreeuwers and Breeuwer.[13] The myocardium boundaries in MR cardio perfusion scans are detected by exploiting changes in the relation of contrast agent concentration between the myocardium and surrounding tissue over time. This technique, however, can only be applied to MRI data. Also the use of methods from artificial intelligence has been attempted. Drobics et al. report promising results of experiments employing Self-Organizing Maps (SOM) to segmentation of the myocardium in CT scans.[14] Currently, however, segmentation results have to be manually improved.

3. SKELTON-BASED MYOCARDIUM SEGMENTATION

This section describes the new algorithm for myocardium segmentation from a 3D cardiac data-set. The basic structure of the technique is outlined in figure 1 From the figure can be seen that a large part of the algorithm (steps 1 through 6) is performed in 2D space on a slice-per-slice basis. Only steps 7 through 10 are done in 3D. The 2D part of the algorithm is aimed at finding out, which slices of the volume intersect the myocardium and marking the myocardial cross-sections on those slices by center lines. The algorithm is initiated by the user by selecting a slice and manually placing a poly-line (the 2D *skeleton*) on it, which approximates the center

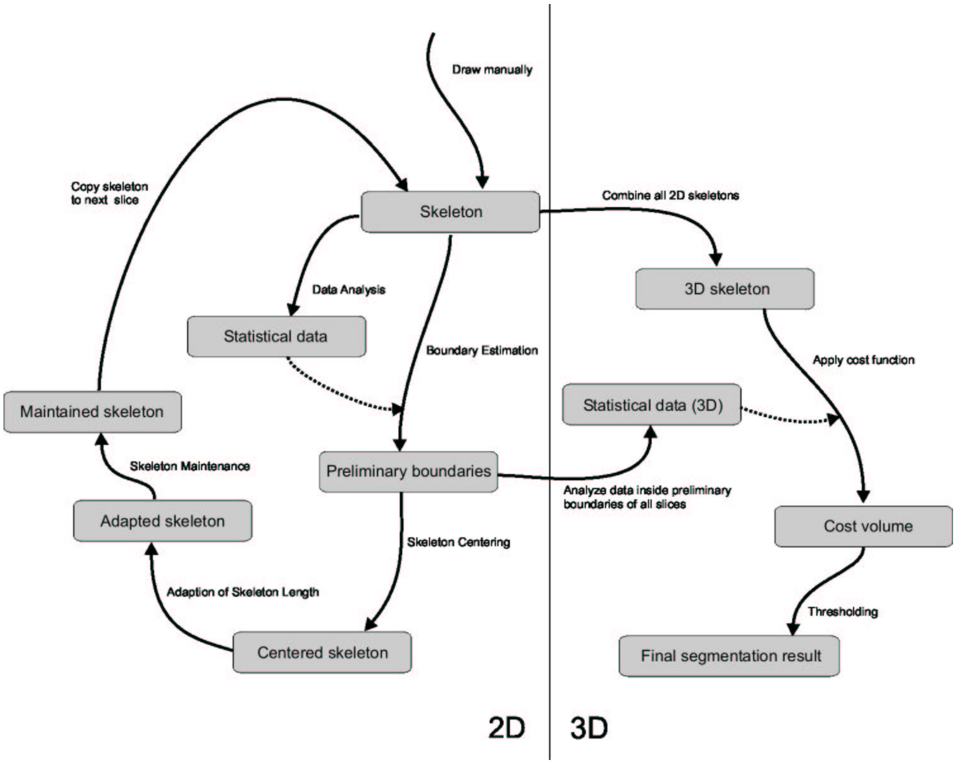

Figure 1. Skeleton-Based Myocardium Segmentation: algorithm overview

line of the myocardial cross-section on the slice (step 1). This skeleton is then automatically propagated to the other slices (steps 2,3,4,5 and 6). Then the 3D part of the algorithm is started. The collected 2D skeletons are combined to a 3D skeleton of the heart muscle (step 7). Then each voxel in the data set is assigned a cost value denoting the cost of the cheapest path from the voxel to the skeleton (steps 8 and 9). Thresholding within the resulting volume of cost values yields the final segmentation result (step 10). The 10 steps are described in more detail in the following paragraphs:

3.1. Step 1: Manual Marking

This part of the algorithm is the only part which requires user interaction. In the first step the user selects a slice of the data set and marks the cross-section of the myocardium in this slice by manually approximating its center line. This is done by setting a number of representative ”skeleton knots” which are then automatically connected by lines to form a skeleton of the myocardial cross-section in the current slice. Since this step influences the complete segmentation algorithm, the skeleton should be a sound representation of the myocardial cross-section in the selected slice and should be placed such that it is entirely located inside the myocardium. An example skeleton is depicted in the leftmost image of figure 3. Here 14 skeleton points were manually placed. Usually, this user-interaction part of the algorithm takes less than 30 seconds.

3.2. Skeleton Propagation

After the skeleton has been placed, the automatic propagation-algorithm (steps 2 through 6) is initiated. Due to the high spatial resolution of CT images, there is a big amount of inter-slice coherency. This means that a skeleton representing a myocardial cross-section well in one slice is usually also a good representation of the myocardial cross-section in each of the two neighboring slices. Thus, a skeleton can be propagated from one slice (then source slice) to another (the destination slice) by first refining it to give a better representation of the cross-section of the myocardial cross-section in the source slice and then copying it to the destination slice.

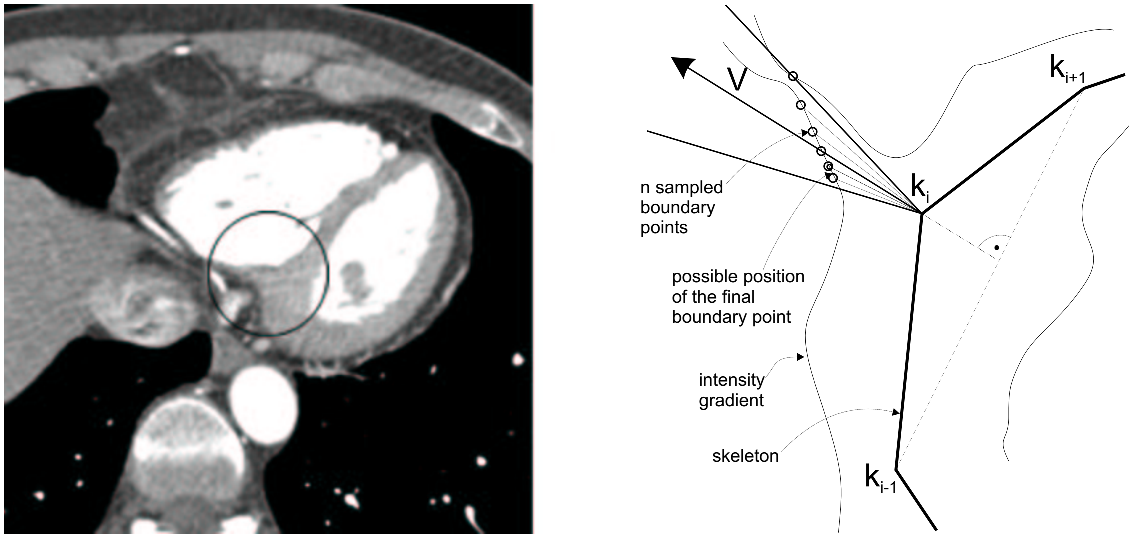
V
k_{i+1}
n sampled boundary points
k_i
possible position of the final boundary point
intensity gradient
skeleton
k_{i-1}

Figure 2. missing gradients (left), robust boundary point estimation (right)

Refinement itself consists of the following steps: First, preliminary boundaries of the myocardial cross-section in the source slice are found (steps 2 and 3), then the skeleton is centered by placing it halfway between those preliminary boundaries (step 4). After that a crucial maintenance process is applied to the skeleton in order to further ensure stable propagation (step 5). Finally the skeleton can be copied to the neighboring slice.

3.2.1. Step 2: Data Analysis

In order to be able to approximate the boundaries of the myocardial cross-section, statistical data about the distribution of data values representing the heart muscle are extracted. The average values (Ei and Eg) and standard deviations (Si and Sg) both for data values and gradient sizes are calculated by taking evenly spaced samples all over the skeleton.

3.2.2. Step 3: Boundary Estimation

The statistical properties calculated in step 2 are used for estimating boundary points. For each skeleton knot k_i, a $primaryvector$ v which is parallel to the current slice and normal to the line connecting the currently processed skeleton knot's immediate neighbors k_{i-1} and k_{i+1} is calculated. Now, starting from the skeleton knot, data samples are taken in direction of v. The distance between samples is equal to the sampling distance used in the data analysis step. As soon as data values and/or gradients are encountered, which diverge significantly from the calculated average values (with respect to the standard deviations) the algorithm assumes that a boundary point has been found. The formulas that are used to decide whether a point is located on the boundary are closely related to the pixel-affinity model proposed by Udupa[15] and Jones[7]: Let $S_1, S_2, ..., S_6$ be the intensity values at 6 successive sample locations, and E_i, E_g, S_i and S_g the values calculated in step 2. Then the probability of the boundary being located between S_3 and S_4 is indirectly proportional to the value u with:

$$u = k \cdot h_i + (1-k) \cdot h_g \,, \tag{1}$$

where h_i takes into account the deviations of the sampled values from E_i, with:

$$h_i = e^{-\frac{((V_1-E_i)/S_i)^2}{-2}} \cdot (1 - e^{-\frac{((V_2-E_i)/S_i)^2}{-2}}) \,, \tag{2}$$

and h_g takes into account the deviation of the encountered gradient from Eg, with:

$$h_g = e^{-(\frac{|V_1-V_2|-E_g}{S_g})^2} \,, \tag{3}$$

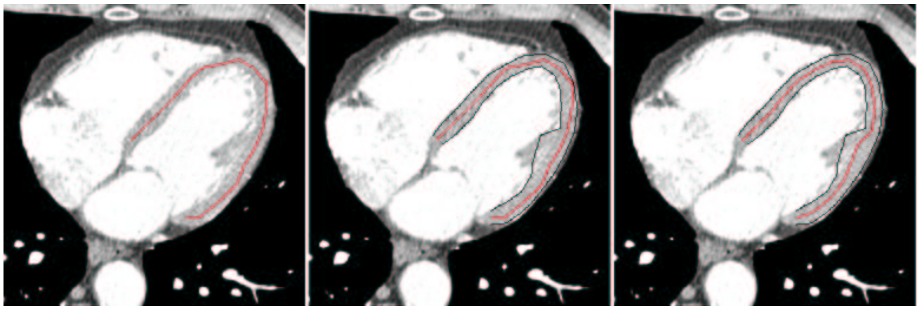

Figure 3. manually drawn skeleton (left), estimated boundaries (middle) and centered skeleton (right)

V_1 and V_2 are the results of low-pass filtering the sampled values in order to tolerate some level of noise: $V_1 = (S_1 + 2S_2 + 5S_3)/8$ and $V_2 = (S_6 + 2S_5 + 5S_4)/8$. Coefficient k, with $0 \leq k \leq 1$ controls the weighting of the intensity- and gradient-related terms and can be adapted to the properties of the data. A boundary point is assumed to be found, if u is below a certain threshold T. Good results are achieved with $T = 0.1$. Not all parts of the myocardial boundary are well recognizable by a significant change of intensity (see left image in figure 2), especially boundaries between the myocardium and other muscles such as papillary muscles. A sampling process as described above might encounter a gradient too late, already well inside neighboring tissue. So, in order to increase stability, the process of boundary-point detection is repeated n times, using n different sampling vectors evenly distributed within an angle of 30 degrees around v (see figure). Out of the resulting n potential boundary points, those m (in the current implementation, n is 11 and m is 3) points, which are closest to the currently processed skeleton point, are selected. The final boundary point is then calculated by averaging these m points. Another boundary point is then searched in the opposite direction (-v).

3.2.3. step 4: Skeleton Centering

The preliminary boundaries found in step 3 are used to center the skeleton in order to make it a better representation of the myocardial cross-section in the current slice. The result of step 3 is a set of $2p$ boundary points, with p being the number of skeleton points. In step 4, each skeleton point is moved to the exact center between two opposite boundary points. The result can be seen in the rightmost image of figure 3.

3.2.4. step 5: Adaptation of Skeleton Length

A skeleton must have the correct length in order to provide a sound representation of the underlying myocardial cross-section. Thus, during skeleton propagation, it is important to check in each slice whether the skeleton has to be extended or shortened. Figure .. shows examples for a too short skeleton, a too long skeleton and a skeleton of correct length. A skeleton, defined by the knots $(k_0, k_1, ..., k_{n-1})$ and consisting of the segments $(s_{0,1}, s_{1,2}, ..., s_{n-2,n-1})$ with $s_{i,j}$ being the segment between the knots k_i and k_j, is assumed to be too long, if, for a given $0 \leq m < n - 2$ all segments $(s_{0,1}, ..., s_{m,m+1})$, or $(s_{m,m+1}, ..., s_{n-2,n-1})$, traverse data values that significantly diverge from the statistical data collected in step 2. In that case, knots $(k_0, ..., k_m)$, or $(k_{m+1}, ..., k_{n-1})$, are removed from the skeleton. The algorithm used to extend a too short skeleton is illustrated in figure ...: A boundary point p_0 is searched from knot k_0 using the method described in section 3.2.2 with the primary vector v being parallel to the vector from knot k_1 to knot k_0. For further processing, point p_0 is moved slightly away from the boundary, using for instance: $p'_0 = k_0 + 0.85 * (p_0 - k_0)$. A potential new skeleton knot n can then be detected by centering p'_0: Boundary points p_1 and p_2 are searched using vector v' with v' being orthogonal to v and $-v'$, respectively, as primary vectors and p'_0 as origin. The new knot n is the center point between p_1 and p_2. Then, another boundary point, p_3 is searched using k_0 as the origin and a primary vector parallel to the vector from knot k_0 to n. If the distance between k_0 and p_3 exceeds a predefined threshold, knot n is added to the skeleton and the complete operation is repeated (with k_0 now being the new found n) iteratively until the boundary point p_3 is close enough to the skeleton. Analogous computation is performed at the other end of the skeleton.

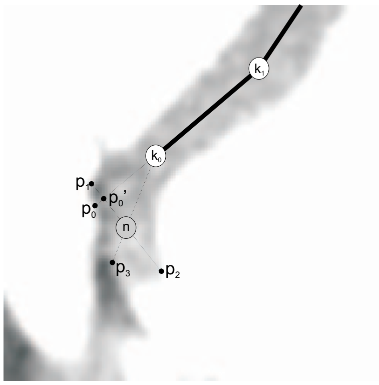

Figure 4. extension of a too short skeleton

3.2.5. step 5: Skeleton Maintenance

A very important sub-task of skeleton propagation is to enforce constraints on the development of the skeleton in order to keep it in shape for further stable propagation. Certain properties of the skeleton must be watched and controlled in every slice to better adapt it to the cross-section of the heart muscle in each slice. One trivial but still very important point is to watch the order of skeleton knots, since it is desireable to prevent self-intersections of the skeleton, which could cause its further development become chaotic. Furthermore, skeleton knots might have to be added or removed in order to prevent too long or too short distances between neighboring skeleton knots. Also some less trivial measures have to be applied, which will be explained in the following paragraphs:

Adaptation of Skeleton Geometry The geometry of the skeleton must reflect the geometry of the underlying myocardial cross-section well. So, if the geometry of the cross-section changes, the geometry of the skeleton must do so too. Among all slices of the volume, there are always some slices whose intersection with the myocardium has a circular (closed) shape and some where this is not the case, regardless of the oprientation of the slices. If during the propagation of the skeleton, the shape of the myocardial cross-section changes from non-circular to circular, the skeleton must be closed as well. The check that has to be performed is simple: The skeleton has to be closed, if step 4 has led to self-intersection of the skeleton or there is a path from the first to the last skeleton knot which does not traverse any data values which statistically significantly differ from usual myocardium data, as derived in step 2.

Skeleton-Segment Improvement After the skeleton has been centered, the probability that it is entirely located on the cross-section of the heart-muscle is high. However, there is still the possibility that, although all skeleton knots are located on the myocardial cross-section, one or more skeleton segments partly move through voxel data not representing the heart muscle. This causes problems, since all voxels being part of the skeleton will be part of the final segmented object. Also, a skeleton segment traveling through tissue other than the myocardium will have a negative impact on the correctness of the statistical data calculated in step 2. Thus, such segments have to be identified and replaced. Identification is straightforward: Each segment of the skeleton is sampled for data values unlikely to represent a part of the myocardium. Once a problematic segment is identified, it is replaced by two segments, using the following algorithm: Let s be the problematic segment of the skeleton connecting the skeleton knots k_0 and k_1, E_i the average data value of the skeleton and S_i the standard deviation of skeleton data values. Now, a path is searched from knot k_0 to knot k_1, which traverses only myocardium voxels. The two segments replacing s will be separated by a knot located exactly half way on the identified path.

An initially empty sorted list of voxels is initialized. Each voxel is assigned a cost and a distance value, both initially 0. In the list, voxels are sorted by the cost value, the voxel with lowest cost being first in the list.

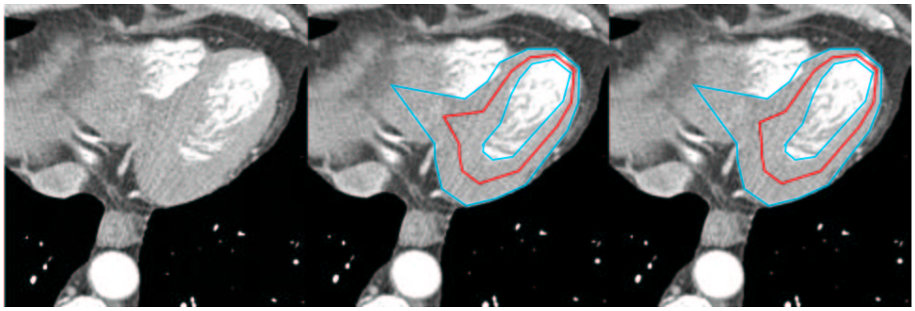

Figure 5. skeleton outside myocardium (middle), skeleton restricted (right))

The voxel containing knot k_0 is first added to the sorted list. Then the following steps are performed iteratively:

1. Take the first voxel with cost c and distance d from the list.
2. Add each immediate neighbor voxel, which has not been assigned a cost value other than 0 yet, to the sorted list and assign distance $d' = d + 1$ and cost c' to the neighbor voxel with:

$$c' = c + 1 - e^{-(\frac{((v-E_i)/S_i)^2}{2})}, \tag{4}$$

   with v being the data value at the neighbor voxel.
3. If one of the neighbor voxels contains the skeleton knot k_1, stop the iteration.
4. return to 1.

A new skeleton knot n is then initialized at the position of k_1 and iteratively moved into the direction of decreasing distance values until a voxel has been reached whose distance value is about half the distance value of the voxel containing k_1. Finally, n is centered. The skeleton segment s is then replaced by two new segments which are separated by the new skeleton knot n.

3.2.6. Coping with missing gradients

Often, tissues surrounding the myocardium are of very similar intensity as the heart muscle. Therefore, usually, some parts of the myocardial boundary are not well recognizable even by the human eye (see figure 5, left image). In some slices, the algorithm as described so far would have trouble detecting the correct boundaries, since data sampling as proposed would not encounter any significant gradients and data values significantly diverging from the myocardial intensity level. Thus, completely wrong preliminary boundary points might be found, leading to the skeleton, through centering, moving outside the myocardial cross-section (see figure 5). The solution for this problem is to make use of the high level of inter-slice coherency in CT data sets: The intersections of two neighboring slices with the myocardial boundary differ only slightly. So, also the skeletons of two neighboring slices must be similar. Thus significant deviations of the skeleton on the current slice from the skeleton on the previously processed slice must be recognized and corrected. There are various ways to do this:

One possibility is to 'peer into the future'. As was already pointed out, the most important goal of skeleton maintenance is to make sure that it will, when copied to the next slice, still be located on the myocardium. So, if S_c is the slice currently processed and propagation is in direction of increasing slice number, movement of a skeleton knot could be restricted to a (two-dimensional) connected region whose voxel values conform to the statistical data from step 2 on all slices from S_c up to S_{c+i}. The value i is not fixed, but adapted depending on the situation. So, for instance, if the region of possible skeleton knot destinations is still large enough after checkin S_{c+j}, then i can be chosen to be higher than j. The center image in Figure 5 shows a skeleton which is partly outside the myocardium. The right image shows the skeleton whose movement has been restricted.

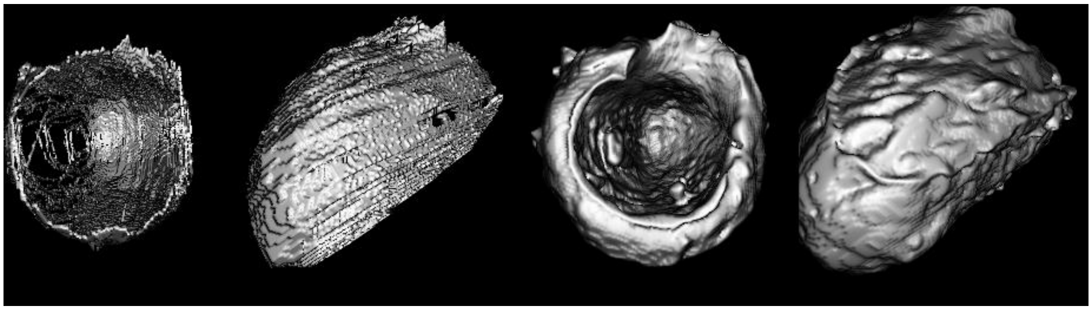

Figure 6. 3D skeleton of a myocardium (left) and final segmentation result (right), viewed from two different angles

Another possibility is to prevent a skeleton knot from moving too close towards the preliminary boundaries identified on the previously processed slice (S_{c-1}). The idea is to leave as much space between the skeleton knot and the boundary as the boundary is likely to travel between slice S_{c-1} and S_{c+1}. The problem is that this technique is, of course, based on assumptions about inter-slice coherency. Thus the first proposed method seems to be the more stable one.

A very simple and still stable method is to compare the distance d_1 between the centered skeleton knot k_i and the outer preliminary boundary on slice S_{c-1} and the distance d_2 between the knot and the outer preliminary boundary on S_c. If d_2 is significantly larger than d_1, i.e., if $d_2/d_1 > T$ for a given threshold T, the skeleton knot is discarded and replaced by a new knot n searched between k_{i-1} and k_{i+1} using the algorithm described in the paragraph about skeleton-segment improvement.

3.3. Final 3D Segmentation

As soon as the 2D skeleton has been propagated to each slice of the data set, which intersects the myocardium, the 3D part of the algorithm is started. First, the 2D skeletons of all slices are combined to form a three-dimensional skeleton of the myocardium (step 8, see figure 6). Then a volume is generated holding for each voxel a value denoting the cost of the cheapest path from the voxel to the skeleton (steps 9 and 10). Finally the myocardium is extracted from this volume using thresholding (step 11). Steps 10 and 11 are explained in more detail in the following paragraphs:

3.3.1. Step 10: Generation of the Cost Volume

The goal of this step of the algorithm is to assign a value v to each voxel of the data-set, denoting the probability that the voxel represents a part of the myocardium. Calculation of these cost values is based upon the result of the skeleton-propagation phase, the three-dimensional skeleton of the myocardium as well as information about the intensity level of voxels belonging to the myocardium. To obtain the latter, the preliminary boundaries identified in step 3 can be used: The average intensity value of the voxels located inside those boundaries is calculated as well as a standard deviation (step 9). Then, for each voxel, the cost value v is obtained by calculating the minimal cost of a path from the voxel to the skeleton, with the cost of a path being determined by its length and how significantly the values that it passes through deviate from the statistical values acquired in step 9. The algorithm used is similar to the one used for skeleton-segment improvement described in section 3.2.4: Again, a list of voxels is initialized, sorted by cost values assigned to the voxels. Initially, the list holds all skeleton voxels. Their cost values are, obviously, zero. Then, iteratively, the following steps are performed:

1. The voxel V with lowest cost is taken from the list. If the cost value of the voxel is higher than a threshold T, the algorithm is stopped. Threshold T is not equal to the threshold used in step 11. Its only purpose is to reduce computation time. Let c be the cost value and v the intensity value of voxel V.

2. A new cost value c_n is calculated for each neighbor voxel of V, using the formula:

$$c_n = 1 - ((1-c) - C_1 - C_2 \cdot (e^{-\frac{((v-E_i)/S_i)^2}{2}})^{C_3}, \qquad (5)$$

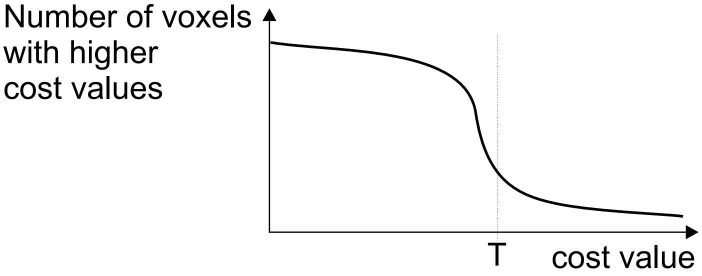

Figure 7. threshold determination

with E_i and S_i describing the statistical distribution of myocardial intensity. The constant values C_1, C_2 and C_3 control the dynamics of cost propagation. C_1 determines the degree of penalization of path length. C_2 determines the degree of penalization of intensity deviation and C_3 controls the steepness of the function mapping cost increase to intensity deviation. For usual CT images, good results can be achieved by using the values $C_1 = 0.03, C_2 = 3$ and $C_3 = 2$. If the neighbor voxel has not yet been assigned a cost value, or its cost value is higher than c_n, c_n is its new cost value.

3. return to 1.

3.3.2. Step 11: Thresholding

The final segmented object is identified by performing thresholding in the volume generated in step 10. Therefore, the final crucial task is to automatically find a suitable threshold. As already pointed out, not all parts of the boundary of the myocardium are well recognizable, since surrounding tissue is often at a very similar intensity level as the heart muscle. Parts of the boundary, which manifest themselves in recognizable intensity gradients can be easily found by extracting the highest gradients in the cost function. The function mapping to each possible cost value c the number of voxels with a cost value $c' > c$ usually looks similar to the function depicted in the left image in figure 7. The changes in steepness are due to the sharp edges mentioned above. Therefore, in order to best approximate those sharp edges, the threshold should be inside (and possibly near the upper bound of) the interval of steep decrease of the aforementioned function, similar to the threshold T proposed in figure.

Since radiologists are usually interested in the basic shape of the myocardium and not in every detail of its surface, it is better to low-pass filter the cost volume before thresholding. In order to keep those boundaries which have to be estimated due to missing gradients smooth, filtering can be performed adaptively with respect to gradient sizes.

4. RESULTS

Skeleton-Based Myocardium Segmentation was tested on 15 cardiac CT data-sets of differing quality levels. It proved to be a robust and accurate technique for myocardium segmentation. Further advantages include little user interaction and, most important, a high degree of flexibility with respect to the shape and orientation of the segmented myocardium and with respect to the shapes of myocardial cross-sections. Segmentation of a 3D volume currently takes about 60 seconds. Figure 8 shows examples of two-dimensional and three-dimensional segmentation results.

5. CONCLUSION AND FUTURE WORK

Skeleton-Based Myocardium Segmentation, a new technique for segmenting the heart muscle in a CT volume, was introduced. Although the technique works well, some improvements are desired: Current research focusses on the extension of the technique to semi-automatic segmentation of four-dimensional data sets. Another possible improvement would be to further reduce the need for user interaction, possibly even to extend the semi-automatic technique to a fully automatic one. This would include automatic detection of the LV in one slice followed by estimation of the myocardial cross-section. Another important matter of future research will be the adaptation of the algorithm to process also MRI images.

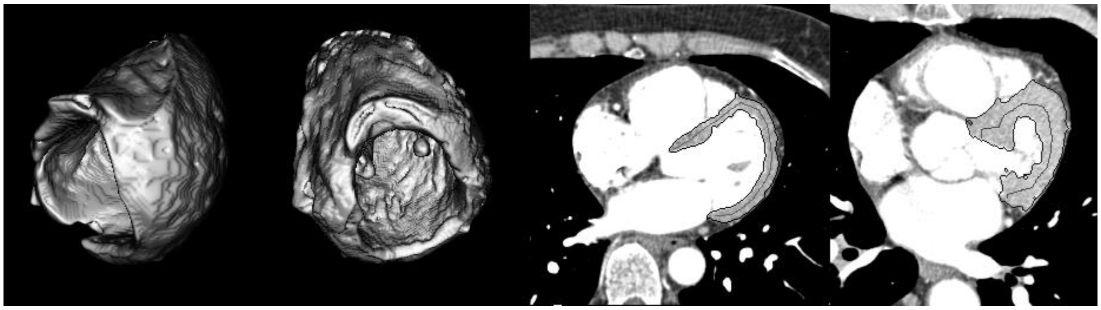

Figure 8. segmentation results

ACKNOWLEDGMENTS

This work has been carried out as part of the basic research project 'Virtual Reality for Scientific Applications' at the VRVis research center in Vienna, Austria, which is funded by the Austrian governmental research project Kplus.

REFERENCES

1. A. F. Kopp, K. Klingenbeck-Regn, M. Heuschmid, A. Kúttner, B. Ohnesorge, T. Flohr, S. Schaller, and C. D. Claussen, "Multislice computed tomography: Basic principles and clinical applications," *electromedica* **68**(2), pp. 94–105, 2000.
2. E. N. Mortensen, B. S. Morse, W. A. Barrett, and J. K. Udupa, "Adaptive boundary detection using 'live wire' two-dimensional dynamic programming," *Proc. Computers in Cardiology (CIC)*, pp. 635–638, 1992.
3. M. Urschler, H. Mayer, R. Bolter, and F. Leberl, "The live-wire approach for the segmentation of left ventricle Electron-Beam CT images," *26th Workshop of the Austrian Association for Pattern Recognition (AGM/AAPR)*, 2002.
4. W. Lin and R. A. Robb, "Visualization of cardiac dynamics using physics-based deformable model," *Proc. SPIE Medical Imaging*, 2000.
5. T. McInerney and D. Terzopoulos, "Deformable models in medical image analysis: A survey," *Medical Image Analysis* **1**(2), pp. 91–108, 1996.
6. M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Actived contour models," *Proc. First International Conference on Computer Vision*, 1987.
7. T. N. Jones and D. N. Metaxas, "Segmentation using deformable models with affinity-based localization," *Proc. CVRMed-MRCAS*, pp. 53–62, 1997.
8. L. D. Cohen, "On active contour models and balloons," *CVGIP: Image Understanding* **53**(2), pp. 211–218, 1991.
9. A. Chakraborty and J. S. Duncan, "Integration of boundary finding and region-based segmentation using game theory," *Proc. XIVth International Conference on Information*, pp. 189–200, 1995.
10. G. I. Sánchez-Ortiz and P. Burger, "Vector field analysis of the dynamics of the heart using velocity encoded cine mr imaging," *9th International Symposium on Computer Assisted Radiology (CAR)*, pp. 228–233, 1995.
11. M. Lorenzo-Valdés, G. I. Sanchez-Ortiz, R. Mohiaddin, and D. Rueckert, "Atlas-based segmentation and tracking of 3d cardiac MR images using non-rigid registration," *Proc. MICCAI*, pp. 642–650, 2002.
12. D. Rueckert, "Geometrically deformable templates for shape-based segmentation and tracking in cardiac MR images," *Proc. EMMCVPR*, pp. 83–98, 1997.
13. L. Spreeuwers and M. Breeuwer, "Automatic detection of the myocardial boundaries of the right and left ventricle," *Proc. SPIE Conference on Medical Imaging* **4322**, pp. 1207–1217, 2001.
14. M. Drobics, P. Hobelsberger, and R. Wegenkittl, "Analysis and visualization of 4D medical images using self-organizing maps and clustering," *Proc. 26th Workshop of the Austrian Association for Pattern Recognition (AAPR)*, 2002.

15. J. K. Udupa and S. Samarasekera, "Fuzzy connectedness and object definition: theory, algorithms and applications in image segmentation," *CVGIP: Graphical Models and Image Processing* **58**(3), pp. 246–261, 1996.