

Automatic cardiac MRI myocardium segmentation using graphcut

Gunnar Kedenburg^{*a}, Chris A. Cocosco^b, Ullrich Köthe^c, Wiro J. Niessen^d,
Evert-jan P.A. Vonken^e and Max A. Viergever^f

^{a,c}Univ. of Hamburg, CS Dept., Vogt-Kölln-Str. 30, 22527 Hamburg, Germany;

^bPhilips Research Labs, Technical Sys. Div., Röntgenstr. 24-26, 22335 Hamburg, Germany;

^dErasmus Med. Center, PO Box 1738, 300 DR Rotterdam, The Netherlands;

^{e,f}Univ. Med. Center Utrecht, Heidelberglaan 100, 3584 CX Utrecht, The Netherlands

ABSTRACT

Segmentation of the left myocardium in four-dimensional (space-time) cardiac MRI data sets is a prerequisite of many diagnostic tasks. We propose a fully automatic method based on global minimization of an energy functional by means of the graphcut algorithm. Starting from automatically obtained segmentations of the left and right ventricles and a cardiac region of interest, a spatial model is constructed using simple and plausible assumptions. This model is used to learn the appearance of different tissue types by non parametric robust estimation. Our method does not require previously trained shape or appearance models. Processing takes 30-40s on current hardware. We evaluated our method on 11 clinical cardiac MRI data sets acquired using cine balanced fast field echo. Linear regression of the automatically segmented myocardium volume against manual segmentations (performed by a radiologist) showed an RMS error of about 12ml.

Keywords: Cardiac MRI, Myocardium Segmentation, Automatic Segmentation, Graphcut

1. INTRODUCTION

Cardiovascular diseases are the main cause of death in western countries. Through magnetic resonance imaging (MRI), images of the cardiac region can be obtained for early and non-intrusive diagnosis of cardiovascular diseases. The objective of this work is the fully automatic segmentation of the left myocardium from axial cine cardiac MRI (three space dimensions plus time) volumes, where the in-plane direction is chosen such that a two chamber short-axis view is present. Our method requires a segmentation of the left and right ventricles, and a cardiac region of interest.

1.1. Clinical relevance

A segmentation of the myocardium enables computing clinical measures such as myocardium volume and wall thickening during the cardiac cycle. Myocardium thickness is a useful indicator of heart problems. For example, a thick wall can indicate a chronically stiffened myocardium, caused by high blood pressure. Conversely, a thin wall can also indicate problems. There is a counter mechanism that causes a weak heart to inflate further, in order to provide more blood. Aside from the original cause of the heart weakness, which needs to be determined, a stretched myocardium is a problem in itself, because the great stress on the thin tissue can cause the muscle to tear. A segmentation of the myocardium is also interesting for visual inspection.

1.2. Advantages of automatic segmentation

A full manual segmentation of cine cardiac MR is impractical, as it takes an expert approximately two hours to segment a typical dataset of 150 slices. Therefore, hand segmentation is usually only done for the end systole and end diastole phases. To utilize all imaging data that is acquired in a cardiac MR examination for quantification, automatic methods are needed. Another problem with hand segmentation is that it is subject to intra- and inter-operator variability. Automatic segmentation makes it much easier to compare segmentations done at different times or different places, because it is the result of a fully defined and deterministic process. Finally, our method produces a segmentation that is consistent along all four axes of the scan. This would be very tedious to do by hand.

* 9kedenbu@informatik.uni-hamburg.de

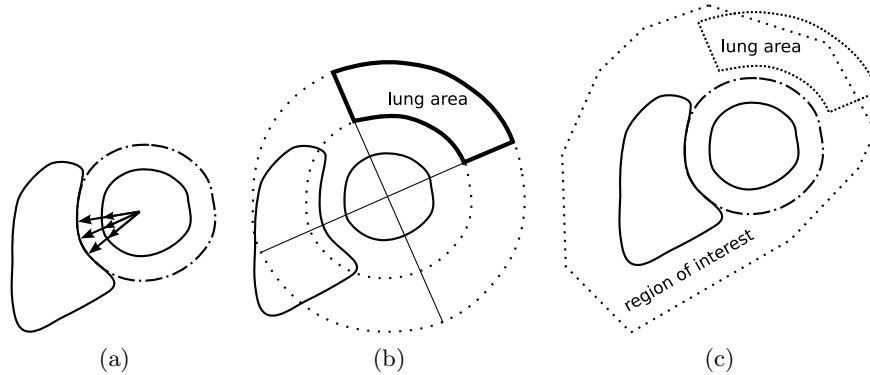


Figure 1. (a) Rays are cast from the left to the right ventricle center in order to estimate the myocardium thickness (by convention, the left ventricle appears on the right).

(b) The lung area is constructed from the estimated myocardium thickness and the ventricle center points.

(c) Finished model. The lung is usually not completely contained in the the (cardiac) region of interest.

1.3. Advantages of the proposed method

The method we will describe here is “image driven”. Contrary to some other cardiac segmentation methods, which need a-priori gray-level appearance models, this method computes a gray-level model from the image to be segmented, making the method more robust against changes in image contrast and against artifacts. It also does not use a-priori shape models which might fail especially for sick patients with an atypical heart. Our method estimates the myocardium shape from the ventricle shape instead.

We use the graphcut algorithm to compute a *global* minimum of an energy functional that is derived from the scan. The energy functional combines edge and region information. Other methods, like for example the level set¹ method, compute local minima. This makes these methods more sensitive to the choice of start values and complicates the interpretation of segmentation results.

2. SPATIAL MODEL

In case of MR images, it is difficult to define an a-priori appearance model for the important tissue types. Contrary to CT, image intensities in MR do not have a standardized physical meaning. Because of that, our method tries to learn the appearance of muscle tissue, blood and the lung from the scan itself.

To find suitable training samples for the learning process, we need a simple spatial model. It is constructed from a segmentation of the left and right ventricles, and a cardiac region of interest. There are fully automatic methods for obtaining these, we used the method of Cocosco et al.² The assumptions made during the construction are simple and plausible, for example: The myocardium surrounds the left ventricle and is bounded by the right ventricle, the myocardium thickness between the two ventricles is a reasonable estimation of its overall thickness, and the endocardium contour provides a good first approximation of the epicardium contour.

2.1. Left myocardium estimation

An estimation of the left myocardium is constructed by casting rays from the left ventricle to the right ventricle. Several rays are cast for each slice, starting from the left ventricle center. The rays fan out 120 degrees, centering at the right ventricle center. This is depicted in figure 1a. For each ray that crosses both the left and the right ventricle borders in the ventricle segmentation, the ratio of the two border distances to the ray origin is computed. If at least a sixth (20 degrees) of the rays that were cast for a single slice cross both borders, the median of the ratios is used as a scaling factor for this slice. For those slices with fewer successful ray casts, the median of the successfully established slice scaling factors of the same phase is used. The left ventricle border is scaled around the left ventricle center by the slice scaling factor, to form an estimation of the left epicardium contour. The area between these two borders is the left myocardium estimation. The estimated epicardium contour is dash-dotted in figure 1a.

2.2. Lung estimation

Further scaling is done to find a part of the lung. The learning process is robust against outliers, therefore the estimation does not need to be accurate. As shown on figure 1b, the estimated left epicardium contour is scaled by fixed factors to form inner and outer borders of a lung area. Of the resulting ring-shaped area, just a quarter is taken. The ad-hoc factors 1.25 and 2.25 were sufficient for the images we tried. The completed spatial model is shown in figure 1c.

3. APPEARANCE MODEL

A training set is constructed by uniformly drawing image intensity samples from the MR image, using the spatial model constructed in the previous section to label the samples. We use a fuzzy k-nearest-neighbors (kNN) classifier trained on these samples to assign muscle tissue probability values to all voxels of the image.

3.1. Fuzzy kNN

kNN is a widely-used and simple classification algorithm. The idea is that, given a training set and a natural number k , a pattern is assigned to the class to which the majority of the k nearest patterns belong. The distance between patterns is a metric in the feature space. We use a “fuzzy” variant of this algorithm: Each of the two training sets is represented by a histogram of image intensities. Starting from the sample to be classified, an equidistant wave is simultaneously propagated over a foreground and a background histogram until the sum of covered histogram values is at least k for the first time. Typically, the sum of covered histogram values will exceed k . It is important to allow this, because otherwise it would be unclear which of several possible training samples at the same distance should be chosen to complete the set of k nearest samples. The fuzzy kNN function would not be well-defined. But if the number of covered samples can exceed k , the function value can be (and is) defined to be the ratio of foreground training samples in the set of covered samples to the total number of covered samples. This histogram-based method is fast, but to lose even less time on the classifier evaluation, the function values are computed once for each of the values that can occur in the MR image and then just looked up when the function is actually applied to the whole volume.

In order not to introduce a bias into the classifier, the same amount of training samples is drawn for foreground and background. A histogram of the training set image intensities is shown in figure 2a. It shows that there are two decision boundaries, one between lung and muscle, and one between blood and muscle. The lung has the lowest intensity, the blood the highest. To represent both background classes equally, the same amount of training samples is drawn for blood and lung. k is set to the square root of the number of myocardium training samples. This choice is common for kNN classifiers and provided very acceptable results in our experiments.

We do not use the classical kNN because some smaller structures are often not preserved in the bi-level output of kNN. Figure 2a shows some overlap around the decision boundaries. In the overlap regions, a classical kNN would still return background (0) or foreground (1), even if the decision had been made with a difference of a single training pattern. Another undesirable property of classical kNN is that the function graph can switch between 0 and 1 more than once in overlap regions. This might introduce strong additional edges into the probability image. In figure 2b, the values of the kNN function and the fuzzy kNN function are shown for all image intensities of a phase of an example MR image.

3.2. Pruning

Although the kNN algorithm is quite robust against outliers, it is helpful to eliminate training samples that were incorrectly labeled because of the imperfect fit of the spatial model, or are intensity outliers, for example because of partial volume or image artifacts. To do this, the training samples are themselves evaluated by the fuzzy kNN function. Incorrectly labeled training samples are discarded. This is a pruning/editing³ operation. Our method does not use a (symmetric) pruning threshold of 0.5, because such a severe threshold eliminates significant image detail. Rather, the foreground training samples are required to have a fuzzy kNN value of at least 0.15, and background training samples must have a value of under 0.85. Choosing these thresholds leads to a lower probability of falsely discarding a pattern. After pruning, the fuzzy kNN is trained again from the remaining samples. By pruning, a large part of the foreground reaches the estimated probability 1. This is

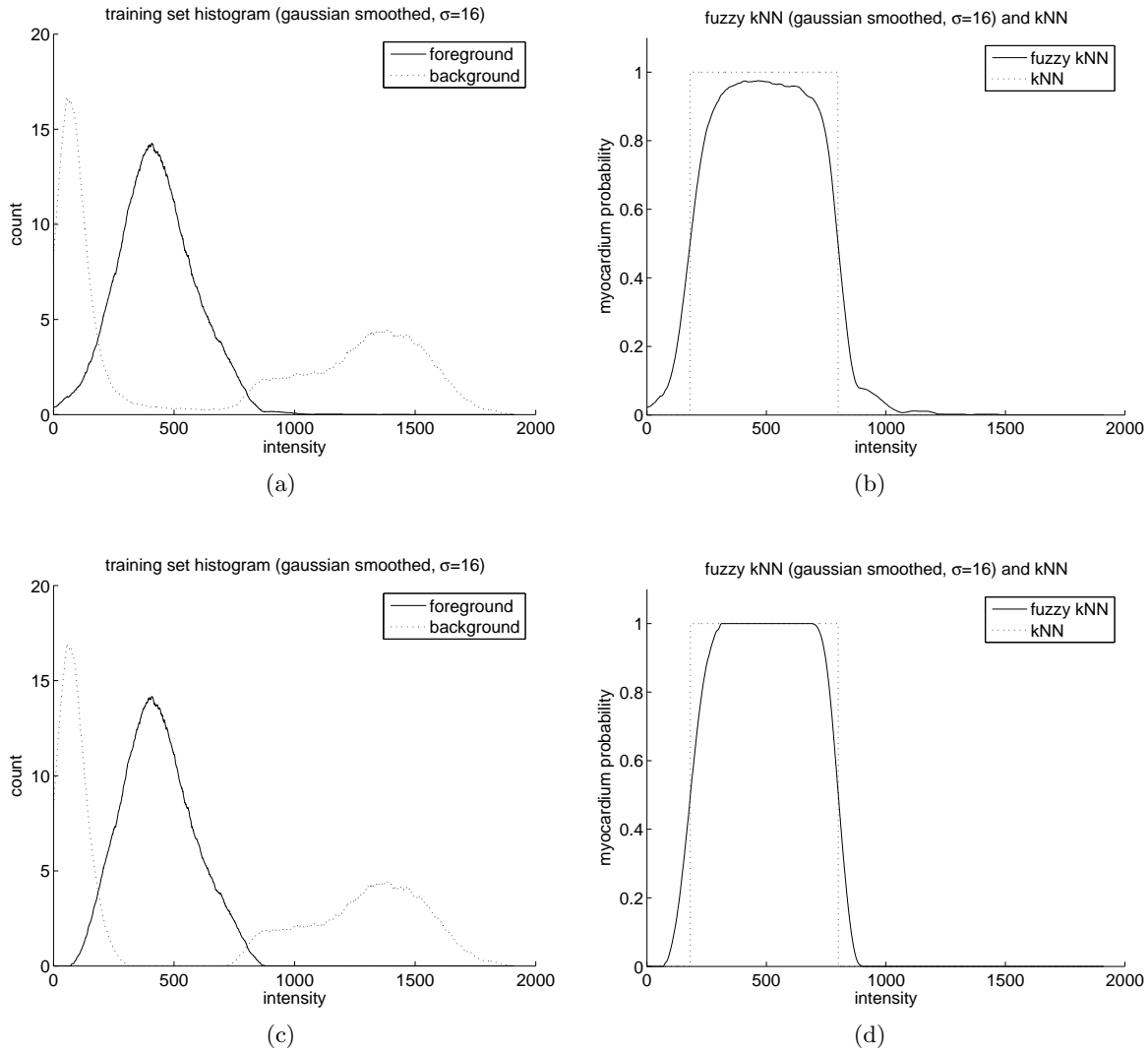


Figure 2. (a) A training set histogram before pruning is applied.

(b) The fuzzy kNN and kNN function graphs computed from the histogram in (a).

(c) The training set histogram shown in (a), after two pruning rounds. Pruning removed intensity outliers (for example partial volume between myocardium and ventricles) and training samples that were mislabeled because of the imperfect fit of the model or an imperfect ventricle segmentation.

(d) The fuzzy kNN and kNN function graphs computed from the histogram in (c).

Note: The smoothing was done for illustration purposes, our method uses non-smoothed histograms and functions.

important for the choice of class penalty functions that will be explained in 4.4. Pruning is done twice in our software. Experience has shown that only very few, if any, samples are pruned in the second run. The feature space and the newly trained fuzzy kNN are shown in figure 2c and 2d.

The appearance model is built separately for each phase of the scan. If there are very few training samples available for a certain phase, training samples are taken from all phases instead (for use in this phase). It is then applied to the MRI scan to give a tissue probability image. The input and output images are shown in figure 3.

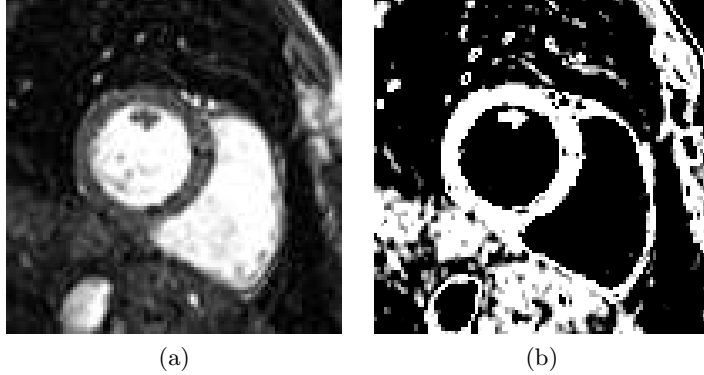


Figure 3. (a) The magnetic resonance image (acquired using a Philips Intera 1.5T scanner). (b) The result of applying the fuzzy kNN classifier to each voxel of the magnetic resonance image.

4. SEGMENTATION

The interactive graphcut method is adapted to fully automatically compute a segmentation of the left myocardium, based on the tissue probability image and seed points obtained from the spatial model. Graphcut⁴ was developed by Boykov and Jolly.

4.1. The graphcut idea

Graphcut is a two-class segmentation algorithm. A special graph is constructed: Each voxel in the region of interest is represented by a node in the graph. Additionally, there are two special nodes, called terminal nodes. One terminal node, the source node, represents the segmentation foreground. The other, called sink node, represents the background. Each of the voxel nodes has edges, called n-edges, connecting it to its neighborhood. We chose the eight-neighborhood for three spatial dimensions plus time. Except for the n-edges, there are two t-edges for each voxel node, connecting the node to the two terminal nodes. The graph is a flow network. For flow networks, there are efficient algorithms for computing a minimal cut that separates the two terminal nodes. The idea of the graphcut algorithm is that this minimal cut defines a segmentation.

A cut that separates the two terminal nodes must leave each node connected to at most one of the terminal edges. A voxel belongs to the class that the still connected terminal node represents. If neither of the terminal edges is connected, both possibilities represent a minimum cut.

4.2. Energy functional

Which segmentation corresponds to a minimal cut depends on the choice of edge weights. In the graphcut algorithm, two different types of edge weights are used to encode two different penalty function types.

D_p is a function that defines a penalty for the class membership (foreground or background) of point p . It takes two different values, one for each possible label. V_{pq} defines a penalty for assigning different classes to neighboring points p and q . Separating similar points should be penalized stronger than separating dissimilar points. The penalty functions are then combined into the graphcut energy functional:

$$E(L) = \lambda \sum_{p \in P} D_p(L_p) + (1 - \lambda) \sum_{(p,q) \in N} V_{pq}(L_p, L_q)$$

N is the neighborhood relation on the voxels, P is the set of voxels. In our application, only the voxels in the cardiac region of interest are part of P . As mentioned before, we use the four dimensional eight-neighborhood to make the segmentation consistent along all four axes of the image for better accuracy in the presence of noisy slices or phases. But any neighborhood can be used: two-dimensional (xy), three-dimensional (xyt or xyz) and four-dimensional ($xyzt$). $L = \{L_p \mid p \in P\}$ denotes the function that returns for a given voxel of a segmentation

the value 1 for foreground and 0 for background. $\lambda \in [0, 1]$ is a weighting factor that can be used to shift between the importance of class membership optimality and neighborhood optimality. The choice of λ is explained in subsection 4.5.

The penalty functions are implemented by setting the source node t-edge weights to $\lambda \cdot D_p(1)$ for all nodes p , the sink node t-edge weights to $\lambda \cdot D_p(0)$ and the n-edge weights to $(1 - \lambda) \cdot V_{pq}(0, 1)$ ($= (1 - \lambda) \cdot V_{pq}(1, 0)$) for all neighboring nodes p and q .

Each n-edge in the graph is represented by exactly one V -type summand of this functional. Every pair of t-edges is represented by exactly one D -type summand. Because a minimal cut minimizes the sum of edge weights (penalties), it globally minimizes the graphcut energy functional.

4.3. Seed points

Seed points are points that have a fixed label. Graphcut can be constrained to produce an optimal segmentation among only those with the prescribed labels at the seed points. For our choice of penalty functions (explained in 4.4), foreground seed points are necessary. Otherwise, labeling all voxels as background functions would be optimal. The seed points are placed two dimensionally onto the slices.

As foreground seed points, we chose a line between the left ventricle border and the estimated epicardium contour. It is constructed by scaling the left ventricle border around the left ventricle center. The scaling factor is $0.6 + 0.4 \cdot s$, where s denotes the scaling factor used for the myocardium estimation on the slice. Additionally, there is a single background seed point, the left ventricle center. Taking the right ventricle center as a second seed point turned out to be not robust enough, because on some images, the right ventricle has almost split into two parts near the valve plane and the ventricle center falls in between these two parts, on muscle tissue.

Seed points are implemented in the graphcut algorithm by using a t-link weight (to the terminal node of the prescribed label) that is higher than the sum of weights of all n-links incident at the seed point. The other t-link weight is set to zero. Then, the high weight t-link cannot be part of a minimum cut, because it would be better to cut all the incident n-links and the zero t-link instead. So the seed point will indeed have the prescribed label.

4.4. Choice of penalty functions

Our penalty functions are similar to those in the work of Boykov and Jolly,⁴ but they are used differently in our method.

4.4.1. Neighborhood penalty function

$$V_{pq}(0, 1) = V_{pq}(1, 0) := \frac{1}{l_{pq}} \cdot \exp\left(-\frac{(I_p - I_q)^2}{2 \cdot \sigma^2}\right), \quad V_{pq}(0, 0) = V_{pq}(1, 1) := 0,$$

Here, I is the tissue probability image, l_{pq} is a neighborhood edge length, and $\sigma > 0$ is a parameter.

As a function of the intensity difference $I_p - I_q$, the V -type penalty functions have an inflection point at σ . They are scaled versions of the normal distribution. Similar image intensities are penalized much more than dissimilar penalties. The choice of σ is explained in subsection 4.5.

Let T be the number of phases in the scan, and let s_x , s_y and s_z denote the physical spacings along x, y and z. We define the edge length by $l_{pq} := \sqrt{s_x^2(p_x - q_x)^2 + s_y^2(p_y - q_y)^2 + s_z^2(p_z - q_z)^2 + s_t^2(p_t - q_t)^2}$. For s_t , we use the ad-hoc formula $s_t := \frac{20}{T} \cdot \frac{1}{2}(s_x + s_y)$. So for 20 phases, s_t is the same as the mean of s_x and s_y , and higher time resolution leads to a lower time spacing s_t . This has worked well for us.

In general, the length of a step along the time axis can be defined based on the ejection fraction and the size of the left ventricle at a certain phase, for example the end diastole phase. This information is available from the ventricle segmentation. We did not follow this idea, but it could be interesting for future work.

4.4.2. Class penalty function

For the class penalty function, we use a probabilistic approach.

$$D_p(0) := -\frac{1}{3} \cdot \log(1 - J_p), \quad D_p(1) := -\frac{1}{3} \cdot \log(J_p), \quad \text{with } J_p := \max(0.5 \cdot I_p, 5\%)$$

The scaling factor $\frac{1}{3}$ and the cutoff at 5% have the effect that the resulting functions map into the interval $[0, 1]$. Without a cutoff, D would be undefined for probability 0 and unbounded, which would lead to segmentations that are dominated by very few points or even a single point.

As a motivation for this choice of class penalty functions, consider the special case of $\lambda = 1$ in the graphcut energy functional:

$$E(L) = \sum_{p \in P} D_p(L_p) = -\frac{1}{3} \left(\sum_{\substack{p \in P \\ L_p=0}} \log(1 - J_p) + \sum_{\substack{p \in P \\ L_p=1}} \log(J_p) \right)$$

Minimizing the energy functional maximizes the compound likelihood of the segmentation over the set of possible segmentations, under the assumption of stochastic independence:

$$\min_L E(L) = \max_L \prod_{\substack{p \in P \\ L_p=0}} (1 - J_p) \prod_{\substack{p \in P \\ L_p=1}} J_p$$

Using the image J instead of I for computing class penalties makes sense, because even if the fuzzy kNN evaluation on a sample yields 1, it is unclear whether or not the voxel is part of the myocardium. It might also be muscle tissue that is not part of this particular muscle. The method should not be biased to include the voxel. But if the fuzzy kNN value is low, this indicates that the voxel is not part of any muscle area. J combines the muscle probability estimated by the fuzzy kNN with the probability of a voxel being part of the myocardium, given that it is a muscle voxel. For the latter probability, we use 0.5.

With the class penalty function chosen like this, the foreground “grows” from the foreground seed line into the myocardium. It becomes a compromise between the cost of labeling voxels that have a myocardium probability of less than 0.5 as foreground, and the cost of the cut to the background.

4.5. Parameter choice

We fixed the two most influential parameters of the graphcut, λ and σ , at the value 0.5. Fixing λ at 0.5 means that both types of penalty functions are equally represented in the graphcut energy functional, and fixing σ at 0.5 means that the inflection point of the neighbor edge penalty function is located at probability difference 0.5. It is unlikely that two samples of the same class have a probability difference of 0.5 or more, while two samples of different classes will most likely differ by more than 0.5. This is apparent from the histograms and graphs shown in figure 2.

For a quantitative measure of segmentation performance, depending on the choice of the two parameters, we computed the *Kappa*⁵ measure against manual segmentations of the end systole and end diastole phases. Kappa is a chance-corrected similarity measure between two segmentations, defined (for two classes) as follows: Given a set of N patterns. Let a_i denote the total number of samples labeled as class i in both segmentations, and by s_i, t_i the total number of samples labeled as i in the first, respectively, in the second segmentation ($\forall i = \{1, 2\}$). Then Kappa is defined as:

$$\kappa := \frac{P_o - P_e}{1 - P_e}, \quad \text{with } P_o := \frac{a_1 + a_2}{N} \quad \text{and } P_e := \frac{s_1 t_1 + s_2 t_2}{N^2}.$$

P_o is the observed proportion of agreement (“accuracy”), and P_e is the expected (due to chance) proportion of agreement. The maximum Kappa value of 1 corresponds to perfect agreement, while 0 corresponds to agreement due to chance alone.

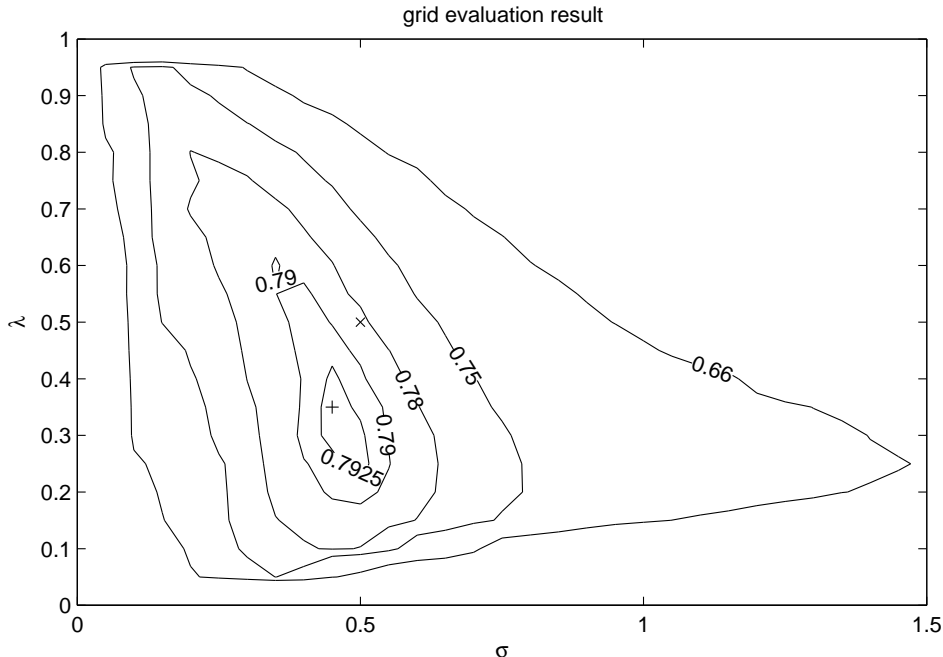


Figure 4. Segmentations of all 11 datasets were computed using (σ, λ) parameter pairs from a grid covering the box $(0, 1.5] \times [0, 1]$. The contour plot shows curves of equal mean Kappa values. For each segmented volume, there is one end diastole and one end systole Kappa value.

The Kappa measure was computed for all parameters on the grid $(\sigma, \lambda) = (0, 1.5] \times [0, 1]$, using the complete method as described in this work on a set of 11 MR images, against manual segmentations of the end systole and end diastole phases (the same data sets as in the linear regression). The mean Kappa values, depending on the parameters, are shown in figure 4. Our parameter choice is supported by the grid evaluation. A cross marks the point $(\sigma, \lambda) = (0.5, 0.5)$. The mean Kappa value improves only slightly if the (for this set of images) optimal parameter combination $(\sigma, \lambda) = (0.45, 0.35)$ is used instead. It is marked by a plus.

5. BOOSTING

To improve the segmentation, one iteration of boosting is employed. After the graphcut stage, the spatial model is updated: The lung area is replaced by its intersection with the background of the segmentation, and the estimated myocardium area is replaced by its intersection with the foreground of the segmentation. The blood area stays the same. This makes the model more correct, although some correct training samples might be lost. In our experiments, the results were always improved.

Using the updated spatial model, a new appearance model is computed. It is used to build a new tissue probability image and finally a new segmentation.

6. POST PROCESSING

After boosting, each slice of every phase of the segmentation produced by the graphcut algorithm is post processed individually (two-dimensionally) in three steps. In the first and most important step, the convex hull of the endocardium (as determined in the segmentation stage) is marked as background. This removes papillary muscles on the inside of the myocardium that are intensity-wise indistinguishable from the myocardium and therefore end up in the foreground of the segmentation. The second post processing step deletes all foreground regions except the (two-dimensionally) largest connected component. Finally, as a third step, morphological closing with single pixel radius is applied to remove isolated holes (intensity outliers, for example noise in the MR image).

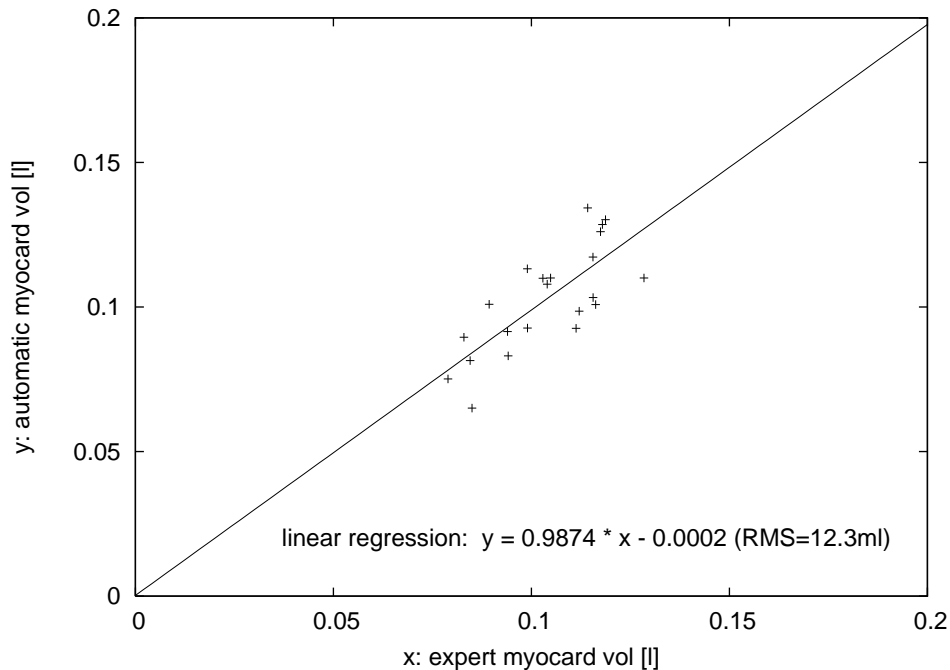


Figure 5. The linear regression of expert segmentation volume to automatic segmentation volume shows a slope near one, a y-intercept near zero, and a root mean square error of about 12ml.

7. RESULTS

The segmentation of the left myocardium was evaluated on 11 clinical balanced fast field echo (FFE) cine cardiac MRI data sets (image size $256 \times 256 \times 10 \times 20$ and spatial resolution $1.5\text{mm} \times 1.5\text{mm} \times 8\text{mm}$ for most of the images). We compared the automatically segmented volume to the volume of manual segmentations performed by a radiologist for two cardiac phases per data set, this is shown in figure 5a. The linear regression shows a slope near one and a y-intercept near zero, meaning that the results of the proposed method are not biased towards a larger or smaller volume. The RMS was 12ml for a myocardium volume of about 0.1l.

It is expected that the inter-operator RMS or even the retry of the same operator is in the same RMS range. Indeed, preliminary inter-rater measurements on four datasets, two phases each, show an RMS error of about 11ml. A more detailed study is still required.

Processing took about 30-40s on one processor of an Intel Xeon 2.4 GHz machine, for segmenting a complete time series (this is significantly faster than any prior art that we are aware of). Our method is implemented in C++ and uses the Max-Cut algorithm made available by Kolmogorov.⁶

Figures 6, 7 and 8 show example segmentations computed by our method in comparison to the expert manual segmentations.

8. CONCLUSION

An extension of the graphcut⁴ method is introduced for use in automatic myocardium segmentation from cine cardiac MRI. Compared to prior art in myocardium segmentation (Boykov and Jolly,⁷ Kauss et al,⁸ Mitchell et al⁹), our new method requires no user interaction and exploits the four dimensional character of the image (xyz and time). Instead of requiring previously trained appearance or shape models, an appearance model is trained on the data itself. The quantitative regression results, as well as the short processing time are very encouraging for clinical use.

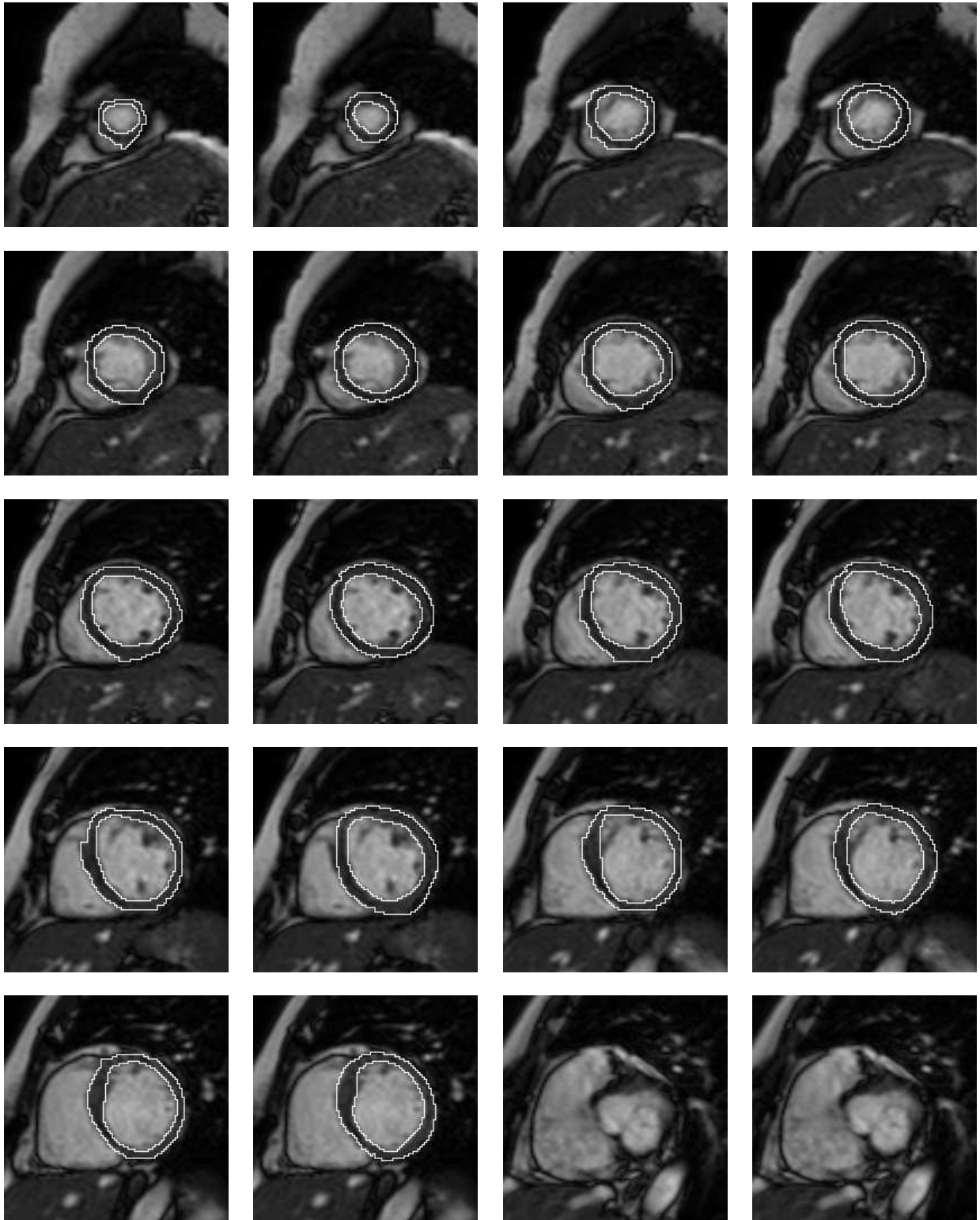


Figure 7. Comparison of automatic to manual segmentation for the entire end diastole phase of an example data set. Automatic and manual segmentations are shown in alternating order. Each row begins with an automatic segmentation.

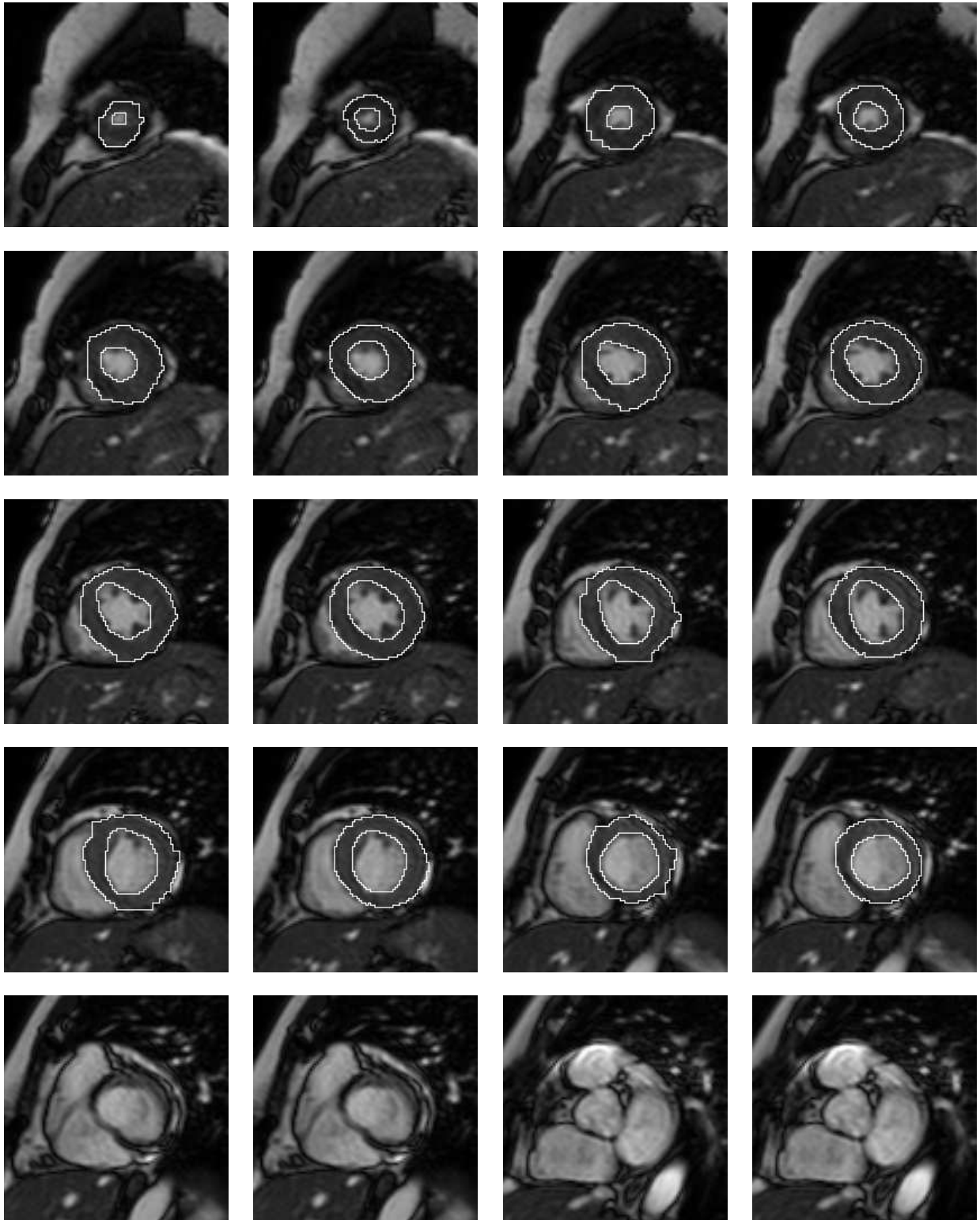


Figure 8. Comparison of automatic to manual segmentation for the entire end systole phase of an example data set. Automatic and manual segmentations are shown in alternating order. Each row begins with an automatic segmentation.