

# 4D Cardiac Segmentation of the Epicardium and Left Ventricle

G. Pons Moll<sup>1</sup>, G. Tadmor<sup>2</sup>, R.S. MacLeod<sup>3</sup>, B. Rosenhahn<sup>1</sup> and D.H. Brooks<sup>2</sup>

<sup>1</sup> Leibniz Universität /Institut für Informationsverarbeitung (TNT),Hannover, Germany

<sup>2</sup> Northeastern University/Department of Electrical and Computer Engineering, Boston, MA, USA

<sup>3</sup> University of Utah/Nora Eccles Harrison Cardiovascular Research and Training Institute (CVRTI), Salt Lake City, Utah,USA

*Abstract*— We present an efficient pipeline for the segmentation and tracking of the Epicardium (Epi) and Left Ventricle (LV) from a set of 3D cardiac MRI sequences. The surface structure is handled as an array of planar active contours, interconnected between adjacent slices and frames, providing spatial and temporal consistency. In a given cardiac phase, the stacking of slice contours constitute a 3D mesh with a cylindrical topology. In a first stage the heart region is automatically localized by making use of the time variance of the heart region. Extraction of heart border is performed by means of energy minimization. Finally, we construct a 3D tensor array from all time surfaces in cylindrical coordinates and time and space consistency of the segmentation is enforced by fitting an m-variate tensor smoothing spline to the final 3D array.

*Keywords*— Cardiac Segmentation, Tracking, Active Contour

## I. INTRODUCTION

The lack of a proper 3D visualization during the training, planning and guidance stages of cardiac surgery makes segmentation of cardiac structures from magnetic resonance (MRI) images an interesting and very important field of research. In addition, early identification of myocardium dysfunction through quantitative analysis, permits a reliable and fast diagnosis of heart diseases. The problem of automatic cardiac segmentation has been extensively investigated. Numerous methods in the literature rely on a previous learning of the shape and appearance of cardiac structures from training examples [1, 2, 3, 4, 5]. These algorithms can encode a higher level knowledge about the appearance and shape of the heart and are hence more robust. The main concerns related to these models are data collection and manual segmentation and registration of the training samples. Therefore this training process is usually very time consuming. Non parametric evolving curves like levelsets [6] have also been used for segmentation of cardiac volumes. However, levelset based methods are computationally very expensive and the behavior of the evolving interface is difficult to control without any shape prior [7, 8]. Levelsets can naturally handle topological changes but this feature is not needed for our application.

### A. Our contribution

We propose an efficient pipeline for the automatic segmentation and tracking of the Epicardium and left ventricle based on active contours [9] that doesn't require previous information learned from a database. Compared to previous methods that segment cardiac contours, our method offers the following advantages: 1) The heart is quickly localized with the distinctive time variance of the heart region. In contrast to other methods that detect the heart in single images [10, 11], our method detects the whole heart volume giving spatial consistency to the initialization. 2) Temporal and spatial consistency is jointly enforced in the segmentation by fitting an m-tensor smoothing spline to the radial basis functions of the time-surfaces. 3) Our heart localization method can also be easily integrated in Appearance and Shape Model based approaches to initialize the position and orientation of the learned model. Moreover, methods based on Active Contours and Deformable Models can benefit from the Time-Space consistency module. Our overall heart localization and segmentation framework is outlined by the flow chart in Fig. 1. In a first stage, we roughly localize the heart volume by find-

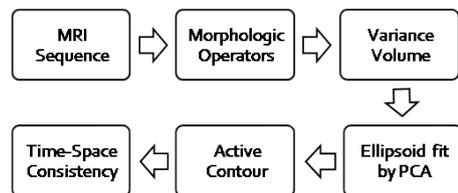


Fig. 1: The flow chart of our approach

ing the bounding ellipsoid that encloses the heart in all the cardiac phases (section III. ). In a second stage, with this initial curve we find the border of the heart by means of energy minimization (section IV. ). The resulting segmented cardiac volume is propagated to the next frame and the algorithm is iterated until the segmentations for all the cardiac phases are obtained. In a final stage, time and space consistency is enforced in the planar segmentations by fitting an m-tensor smoothing spline to all time surfaces in a single optimization step (section V. ).

## II. PRE-PROCESSING

MRI images are noisy and contain a lot of distractors like little veins that appear in the cross sections as bright spots around the contour. We want to clean the images but at the same time preserve the heart shape and appearance. To this end, we use morphological operators to erase the bright spots from the image. First, we perform an opening using a disk shaped structuring element of radius 2 pixels  $se[n]$  that is going to erase all the bright spots smaller than the disk.

$$\gamma_{se}(I[n]) = (I[n] \ominus se[n]) \oplus se[n] \quad (1)$$

Where  $\gamma_{se}$  is the opening operator and  $\ominus, \oplus$  the erosion and dilation operators respectively. Second, we binarize the output from the opening and use it as a mask  $B[n]$  to the original images; the choice of the threshold is not important since the resulting background pixels are close to zero.

The process is illustrated in Fig. 2. With this process we eliminate the background noise and more important the little veins surrounding the heart region are removed from the images while preserving the heart appearance as can be seen in Fig. 3d.

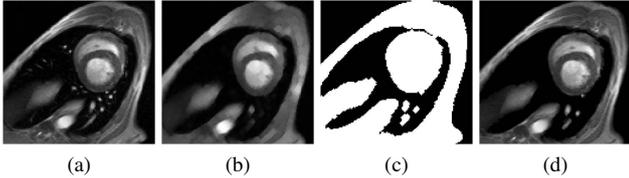


Fig. 2: (a) Example image, (b) Output of opening  $\gamma_{se}(I[n])$ , (c) Binary mask from opening  $B[n]$ ; (d) Final image,  $I[n]$

## III. INITIALIZATION

Pixels belonging to the heart will have in general a higher variance than background pixels due to the wall motion during contraction and due to the heavy blood flow inside the heart [11]. By exploiting this distinctive feature and enforcing spatial consistency across the slices we are able to localize the heart volume. To this end, we compute the time variance volume  $V_\sigma$ :

$$V_\sigma = \frac{1}{T} \sum_{t=1}^T (V_t - V_\mu)^2 \quad \text{where} \quad V_\mu = \frac{1}{T} \sum_{t=1}^T V_t \quad (2)$$

where  $V_\mu$  is the mean time volume. We then binarize the variance volume  $V_\sigma$  and find the largest connected region. The threshold is selected so that as many pixels as the heart volume are preserved. In order to initialize our active contour we

find the minimum enclosing ellipsoid by Principal Component Analysis (PCA) Fig. 3. This provides an initial estimate of the volume for the first frame. Although this method provides fairly well estimates for the mid-ventricular slices Fig. 3e, some problems appear in the apex cross-sections because the variance is not discriminative enough. In our experiments the apex cross-section initialization is manually corrected.

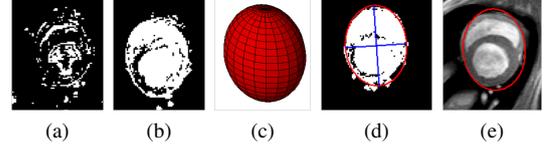


Fig. 3: Initialization steps: (a) Simple Background Subtraction; (b) Variance volume cross-section; (c) Ellipsoid fit; (d) Example of cross-sections of the Variance Volume and the Ellipsoid; (e) Initialization curve

## IV. SEGMENTATION

### A. Energy minimization

For each slice in the MRI volume we can use the cross-sections of the ellipsoid 3e as the initial estimates for our active contour. The segmented volume for the current frame is used as the initialization for the next frame. The boundary of the epicardium is found by moving every point in the curve in the radial direction so that the energy functional in equation (3) is minimized. In order to avoid mesh resampling during optimization we obtain a time invariant parameterization by sampling the curve with respect to its centroid clockwise at every 5 degrees starting from north obtaining a total of 72 particles. Then for each point in the curve, we choose the candidate that minimizes the cost function in (3). We use radial windows of 11 pixels, centered at the corresponding curve point. The energy function can be expressed as:

$$\operatorname{argmin}_{x \in \text{Window}} w_1 \frac{d^2 y}{dx^2} - w_2 \langle \nabla I(x), \vec{r}(\theta_i) \rangle - w_3 y - w_4 |r(\theta_{i-1}) - r(x, \theta_i)| \quad (3)$$

where  $x = [x, y]^T$  denotes the image locations in the search window,  $y$  is the image intensity curve in the radial direction,  $\nabla I(x)$  is the image gradient found using the sobel filters in the horizontal and vertical directions,  $r(x, \theta_{i-1})$  is the radius of the candidate with respect to the centroid and  $r(\theta_{i-1})$  is the radius of the previous candidate minimizer in the curve. Note that all the external forces of the functional in equation (3) derive from image features computed in the radial direction. Therefore, the search space is constrained to elliptically shaped objects.

## B. Curve parameterization

Even though smoothness is enforced in the energy functional (3) the curve we obtain is not perfectly smooth yet. Because of the elliptical shape of the heart cross-sections, Elliptical Fourier Descriptors are well suited for reducing and constraining the dimensionality of our problem. We assume a smooth and continuous heart boundary, so we should be able to encode the contour as a sum of a few Fourier Coefficients, 4 or 5 maximum. In order to project our curve into the Fourier space we first find the polar coordinates  $(r_i, \theta_i)$  of the points  $(x_i, y_i)$  obtaining a vector of 72 radius. Thereby, we find the first  $N_c$  FFT coefficients of the curve radius:

$$R_k = \sum_{n=0}^{N-1} r_n e^{-\frac{2\pi i}{N} nk} \quad k = 0, \dots, N_c - 1 \quad (4)$$

Given that  $r_n$  are real we find the conjugate pairs by  $R_{N-k} = R_k^*$ , therefore we only need to compute the first  $N_c$  Fourier Coefficients. So in total we have the DC component plus  $2 * (N_c - 1)$  coefficients because we want amplitude and phase. Once we have the first  $N_c$  harmonics we recover the low-pass curve in polar coordinates  $(r, \theta)$  via *IFFT*, and finally we go back to  $(x, y)$ . The whole filtering method to smooth the contour is the following:

$$(x, y) \xrightarrow{xy2polar} (r, \theta) \xrightarrow{FFT} (R_k) \xrightarrow{IFFT} (r, \theta) \xrightarrow{polar2xy} (x, y) \quad (5)$$

With this procedure we find the first 5 FFT coefficients that best fit the heart border.

## V. TIME AND SPACE CONSISTENCY

For each frame we have obtained a set of planar contours with cylindrical topology. The rendered surfaces are not smooth and present jitter in time. In order to ensure time and space consistency we constrain our segmentation by fitting an m-tensor smoothing spline to all time surfaces in a single optimization step. Although global optimization methods like truncated 3D Fourier descriptors are optimal in general, local optimization methods like smoothing splines are best suited for our problem because they lead to more satisfying reconstructions. Therefore, we construct a tensor  $\Theta$  of order  $N = 3$  with the radius of the time-space curves. Columns of the tensor are formed by the radial basis functions of the planar contours in a given time and slice location. With this arrangement the topology of the data is preserved, i.e. the physical neighbors are also neighbors within the tensor. An m-tensor product smoothing spline approximates the multivariate data taking the geometric average of univariate smoothing splines. Thereby, each element of the tensor  $\Theta$  is approximated by:

$$R(\theta, z, t) = f(\theta)g(z)h(t) \quad (6)$$

where  $R(\theta, z, t)$  is an element of the tensor  $\Theta$  and  $f(\theta), g(z), h(t)$  are 1-variate splines that enforce consistency in each of the 3 dimensions involved: within the same contour, across the slices and in time respectively. An intuition of what we are doing is the following: For each of the 3 dimensions we have a set of data points  $y = g(x) + n_i$  that follow a smooth function  $g(x)$  and are corrupted by some random unknown noise product of our algorithm  $n$ . We are interested in recovering  $g(x)$ , so the data is projected to a subspace of polynomial functions. The new basis will be able to represent  $g(x)$  but won't be able to follow the fast noise. We choose piece-

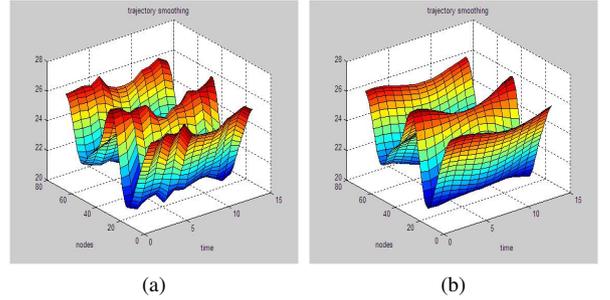


Fig. 4: Cross-section of the 3D tensor  $\Theta$  for a constant slice (a) Radius before smoothing; (b) Radius after smoothing;

wise cubic polynomial functions with constraint equations to approximate our points. Given a set of points  $(x_i, g(x_i) + n_i)$  we construct the following function:

$$f(x) = P_i(x) \quad x_i \leq x < x_{i+1} \quad i = 0, \dots, n \quad P_i \in \mathbb{P}_3 \quad (7)$$

The function  $f(x)$  minimizes the following expression:

$$p \sum_{j=1}^N w_j |y(j) - f(j)|^2 + (1-p) \int |f''(x)|^2 dx \quad (8)$$

The first term of the cost function controls the accuracy of  $f(x)$  i.e. how close to the points we are, and the second term forces  $f(x)$  to be smooth. We have a trade off between accuracy and smoothness. The choice of  $p$  depends on which of this two conflict goals we accord the greater importance. Besides this constraint we impose that  $f(x)$  is twice differentiable,  $f(x) \in C^2$ . This gives rise to three additional equations for each data site:

$$P_{i-1}^{(n)}(x_i) = P_i^{(n)}(x_i) \quad n = 0, 1, 2 \quad (9)$$

We can generate a linear system  $Ax = b$  from equations (8),(9) and solve in a Least Squares fashion to find the unknown vector of polynomial coefficients. For further details on smoothing splines we refer to [12]. The effect of the smoothing can be seen in Fig. 4 where we show the radius of one contour over time: the undesired jitter in time, Fig. 4a, is removed after the smoothing process Fig. 4b.

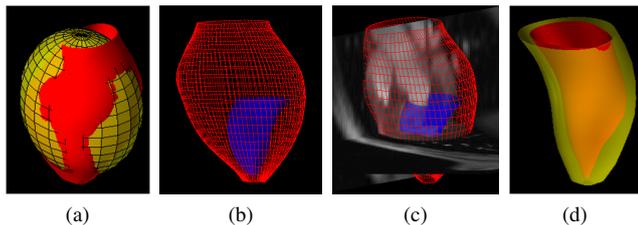


Fig. 5: (a) Initial ellipsoid localization in yellow overlapped with the final segmentation; (b) Epi volume with the LV inside; (c) Segmentation with the long axis cross-section; (d) LV during end-systole and end-dyastole

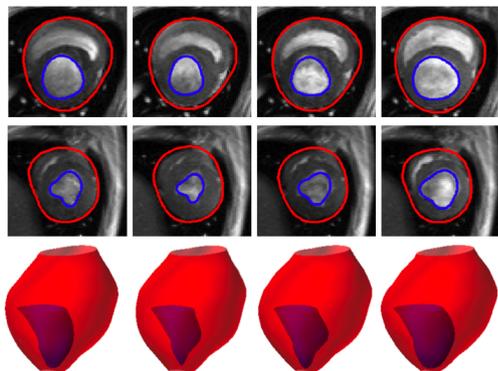


Fig. 6: Epi and LV surfaces and contours-Result of the segmentation for different slices 9 and 15 in the vertical direction. The cardiac phases correspond to mid-systole, end-systole, mid-dyastole, end-dyastole

## VI. RESULTS

We have conducted experiments on two 4D canine cardiac datasets. Each dataset consists of 12 cardiac phases with 20 Short Axis (SA) cross-sections per phase. The resolution of the images is  $1\text{mm} \times 1\text{mm}$  and the separation between the slices is 6 mm. To test the generality of our algorithm we used it to also track the Left Ventricle and we obtained acceptable results by just a small adaptation of the parameters. Quantitative validation was performed by comparing the automatic segmentation results with manual segmentation. As in [10], we define the following error measures: true positive fraction (TPF) as the fraction of tissue in the gold standard overlapped with our segmentation; False Negative Fraction (FNF) as the amount of tissue missed by our segmentation and False Positive Fraction (FPF) as the fraction of tissue falsely detected. We obtained encouraging results on the validation datasets, TPF, FNF and FPF are shown in table 1 as a percentage of the total true amount of tissue. In addition, the resulting segmented surfaces (Epi and LV) are smooth with no jitter in time Fig. 6,5. In Fig. 3c we show how the (Epi) surface, in red, and the (LV) surface in blue, perfectly fit the cardiac volume and we can see how the rendered surfaces can provide a very useful 3-dimensional reference for computer assisted

cardiac surgery.

	TPF	FNF	FPF
Before TSC	94%	6%	5.22%
After TSC	96.46%	3.54%	4.5%

Table 1: Quantitative results of our segmentation method before Time and Space Consistency (TSC) and after

## VII. CONCLUSION

We have presented a method for segmenting the epicardium in 4D cardiac MRI datasets that exploits the spatial and time dependencies of the heart and does not require any previous knowledge learned from a database. Our method can track the epicardium with an error of less than 2 mm with automatic initialization. We have tested the generality of our method by successfully segmenting the left ventricle with only minor changes on the parameters. Future work will be further validation of the method and extensions to make it suitable for patient specific model creation for cardiac surgery interventions.

## REFERENCES

1. Cootes T. F., Taylor C. J.. Active Appearance Models in *IEEE Trans. on Pattern Analysis and Machine Intelligence*:484–498Springer 1998.
2. Huang X., Li Z., Metaxas D.. Learning coupled prior shape and appearance models for segmentation in *MICCAI Vol. 1*:60–69 2004.
3. Andreopoulos A., Tsotsos J.. Efficient and generalizable statistical models of shape and appearance for analysis of cardiac MRI *Medical Image Analysis*. 2008;12:335–357.
4. Assen H. C., Danilouchkine M. G., Frangi A. F., et al. SPASM: Segmentation of Sparse and Arbitrarily Oriented Cardiac MRI Data Using a 3D-ASM in *FIMH*;3504:33–43Springer 2005.
5. Peters T. M., Linte C. A., J. Moore D. Bainbridge, Jones D. L., Guiraudon G.. Towards a Medical Virtual Reality Environment for Minimally Invasive Cardiac Surgery. in *MIAR*;5128:1-11Springer 2008.
6. Osher S, Sethian J. Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations *Journal of Computational Physics*. 1988.
7. Kohlberger T., Cremers D., Rousson M., Ramaraj R., Funke-Lea G.. 4D Shape Priors for a Level Set Segmentation of the Left Myocardium in SPECT Sequences in *MICCAI*;4190:92–100Springer 2006.
8. Lynch M., Ghita O., Whelan P. Left-ventricle myocardium segmentation using a coupled level-set with a priori knowledge *Computerized Medical Imaging and Graphics*. 2006;30:255–262.
9. Michael K., Witkin A., Terzopoulos D.. Snakes: Active contour models *International Journal of Computer Vision*. 1988;V1:321–331.
10. H. J., Huang X., Metaxas D. N., Axel L.. Dynamic Texture Based Heart Localization and Segmentation in 4-D Cardiac Images in *ISBI*:852–855IEEE 2007.
11. Sörgel W., Vaerman V.. Automatic heart localization from a 4D MRI dataset in *In Proc. of SPIE Conf. on Medical Imaging*:333–344 1997.
12. Boor Carl. *A Practical Guide to Splines*. Applied Mathematical Sciences, New York: Springer 1978.