ADJUSTMENT OF DEFORMABLE MODELS TO ORGAN SURFACES IN 3D IMAGES

Carlos Santiago[1], Jorge S. Marques[1] and Jacinto Nascimento[1]
[1]*Institute for Systems and Robotics, Instituto Superior Tecnico, Lisbon, Portugal*
carlos.santiago@ist.utl.pt

Keywords: 3D Echocardiography, Left Ventricle, Segmentation, Deformable Models, Feature Extraction, Robust Estimation.

Abstract: The segmentation of the left ventricle (LV) in echocardiographic data has proven itself a useful methodology to assess cardiac function and to detect abnormalities. Traditionally, cardiologists segment the LV border at the end-systolic and end-diastolic phases to determine the ejection fraction. However, the manual segmentation of the LV is a tedious and time demanding task, which means automated segmentation systems can provide a powerful tool to improve workflow in a clinical setup. This paper proposes a robust 3D segmentation system consisting of a deformable model that uses a probabilistic data association filter (PDAF) to robustly detect the LV border. Results show that the algorithm performs well in both synthetic and real data, without significantly compromising its performance. The obtained LV segmentations are compared with the manual segmentations performed by an expert, yielding an average distance of 4 pixel between points from both segmentations.

1 INTRODUCTION

Echocardiography has arguably become amongst the most preferred medical imaging modality to visualize the left ventricle (LV). This is mainly due to several reasons, such as, its low cost and portability [1]. The diagnosis usually comprises the intervention of an expert who manually segments the LV boundary at the end-systole and end-diastole phases. This is a necessary step for further quantitative analysis of the heart in order to detect possible cardiopathies present in the LV. Such procedure is generally (i) tedious and time consuming, (ii) prone to errors and (iii) has a significant inter-variability of the segmentation among specialists. Besides, this image modality presents several challenges among which we point out (i) the poor images quality (low SNR ratio), (ii) the edge dropout specially in the diastole phase, (iii) the presence of outliers, and (iv) the presence of multiplicative noise (*i.e*, Rayleigh), see Fig. 1 for an illustration. Consequently, only experts are able to correctly locate the LV boundary.

The above mentioned problems can be alleviated with the use of an automatic LV segmentation system. Initially, automatic LV segmentation systems were developed for 2D echocardiography. As soon as the 3D echocardiography became available, methods to perform the segmentation also became available in the literature. One approach to perform 3D segmentation is to consecutively applying 2D segmentations to each image plane and assembling them into a 3D structure [3, 4] - as cardiologists manually do in such cases. However, such approaches require additional methods to prevent inconsistencies in the surface. Other approaches have performed the 3D segmentation using the level-set method, such as in [1, 5, 6], and even 3D tracking systems [7, 8].

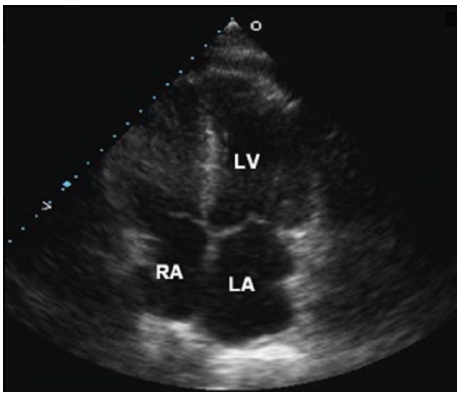

Figure 1: Echocardiography - apical four-chamber view [2].

One of the above mentioned difficulties, is the presence of outliers, *i.e.* the invalid features that do not belong to the boundary of the object of interest,

in this case, the LV surface. The presence of outliers should be avoided as far as possible, since it often leads to meaningless shape estimation results. To overpass this difficulty, we propose a robust 3D segmentation algorithm capable of discerning between valid and invalid image features. To accomplish this, the algorithm is based on a *probabilistic data association filter* [9]. Two main underlying ideas of the algorithm are as follows. First *middle level features* are considered. More specifically, patches are used. Second, a labeling process (valid-invalid) is assigned to each patch. Since we do not know beforehand, the reliability of the patches, all possible labeling sequences of valid/invalid patch labels are considered. Each patch sequence is called here as *patch interpretation*. Finally, a probability (*association probability*) is assigned to each patch interpretation. Thus, in the adopted strategy, all the patches contribute to the evolution of the deformable model with different weights.

The paper is organized as follows: Section 2 presents an overview of the proposed segmentation system; Section 3 describes the deformable model used; Section 4 addresses the feature extraction algorithm and the middle-level features assemblage; and Section 5 presents the robust model estimation technique inspired in the S-PDAF algorithm. Section 6 shows results of segmentation system applied to synthetic data and to the segmentation of the LV in echocardiographic images. Finally, Section 7 concludes the paper with final remarks about the developed system and future research areas.

2 SYSTEM OVERVIEW

The idea behind of the present approach is to tackle the difficulties of classic deformable contour methods associated with noisy images (such as ultrasound images) by introducing a robust estimation scheme. The robust framework is inspired in the S-PDAF [10], developed for shape tracking in cluttered environments. Here we extend it to the context of 3D shape estimation.

The proposed segmentation system uses a 3D deformable model to characterize the surface of the segmentation. This deformable surface requires an initialization procedure that ensures it is initialized in the vicinity of the LV boundary.

The adaptation procedure is an iterative process that consists of the following steps: after initialization of the model, an adaptation cycle begins with the detection of low-level features, searched in the vicinity of the model. Then, these are grouped into middle-level features (patches). Based on the assembled patches, the S-PDAF algorithm determines all possible interpretations of considering a patch valid or invalid and assigns to each patch interpretation a confidence degree that is used to define the estimate of the boundary location. The model estimate is then used to fit the surface to the LV boundary, ending an iteration of the adaptation cycle. The process repeats until the surface is considered close to the LV boundary. The following figure shows a diagram of the adaptation cycle.

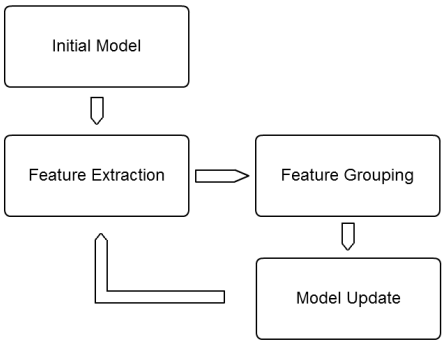

Figure 2: Diagram of the proposed segmentation system.

3 SURFACE MODEL

The proposed segmentation system uses a simplex mesh [11] as the deformable model. A 3D simplex mesh is a meshed surface composed of vertices and edges, where each vertex has three neighboring vertices (i.e., belongs to three edges) (see Fig. 4). This particular structure allows to define geometric relations between vertices that are used in the adaptation procedure to ensure a smooth surface and good vertices distribution.

3.1 Law of Motion

Each vertex adapts in an iterative process under the influence of external and internal forces and its final position is determined by the equilibrium of forces of the following equation [11]:

$$P_i(k+1) = P_i(k) + (1-\gamma)(P_i(k) - P_i(k-1)) + \alpha_i \mathbf{F}_i^{int}(k) + \beta_i \mathbf{F}_i^{ext}(k) \quad (1)$$

where the parameters γ, α and β are constants.

The internal force, F^{int}, is responsible for maintaining the smoothness of the surface, making use of the geometric relations between vertices. On the other hand, the external force, F^{ext}, is responsible to attract each vertex towards the object boundary.

3.2 Model Initialization

The initialization procedure has to meet the following conditions: 1) the initial model should be initialized in the vicinity of the LV boundary; and 2) it should be a simplex mesh. These two conditions are met using the following three step procedure.

First, to ensure that the model is initialized in the vicinity of the LV boundary, the user manually defines a coarse outline of the LV in three orthogonal planes. A 3D region is then obtained by space carving [12] (see Fig. 3).

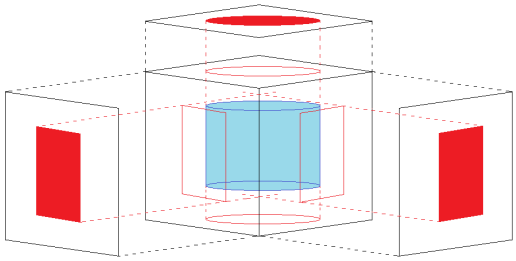

Figure 3: Schematic example of the computation of the carved volume (cylinder) (in light blue) by intersection of the segmented projections (in red).

Second, the simplex mesh is initialized as a sphere in the center of the carved volume. After uniformly sampling sphere points, the convex hull algorithm [13] is applied, resulting in a triangular mesh on the sphere surface. Then, taking into account the duality between simplex meshes and triangulations [11], an associated simplex mesh can be formed by considering the center of each triangle as vertices and linking each vertex with the center of the three neighboring triangles, resulting in the simplex mesh shown in Fig. 4.

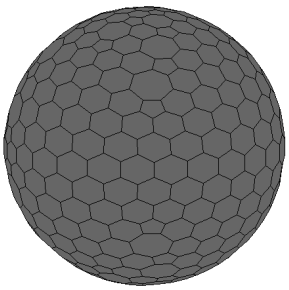

Figure 4: Simplex mesh initialized as a sphere.

Finally we let the spherical simplex mesh deform until it fits the carved region. This region corresponds to the silhouette of the LV boundary and simplifies an initial adaptation to the dataset since it is a noiseless binary volume.

4 FEATURE EXTRACTION

The detection of the LV border is performed by the feature extraction algorithm, using a directional feature search in the vicinity of the surface model, as explained next.

4.1 Feature Detection

At each vertex, P_i, of the simplex mesh, we compute the normal to the surface and define a search line parallel to the normal vector that passes through P_i. Then, the intensity signal along the search line is analyzed. The LV border features are detected using an edge detector filter, as described in [14]. The filter output's maxima are extracted using a threshold and a non-maximum suppression technique.

Although this methodology has good results, many undesired features are still detected, depending on the threshold used. In our experimental setup we obtained, for each vertex, an average number of features that varies from one up to four . Recall that, only one of these features corresponds to the LV boundary.

4.2 Middle-level Features

To increase the robustness of the feature detection, these are grouped into middle-level features (patches). To assemble these patches, we use a labeling algorithm that assumes that features should belong to the same patch if: 1) the corresponding vertices are neighbors in some level (*i.e.*, the patch is a connected graph of its features); 2) all features associated with the same vertex have different labels; and 3) the distance between neighboring features in a patch must not exceed a chosen threshold. Fig. 5 shows an example of the labeling result.

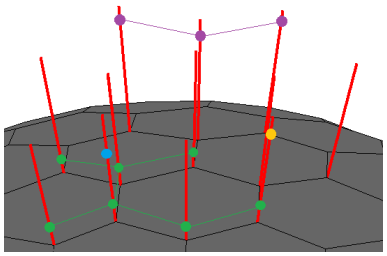

Figure 5: Example of middle-level features. Each color represents a different patch.

In order to achieve the desired label configuration, L, an energy function is used composed of three terms:

$$E(L) = E_1(L) + E_2(L) + E_3(L) \qquad (2)$$

This energy is computed as a sum of the energy of each individual label.

The first term, $E_1(L)$, is minimum when features with the same label are the closest features associated with the neighboring vertices. The second term, $E_2(L)$, prevents patches from having features too far apart. This is done by assigning an energy of ∞ to labels where the distance between neighboring features exceeds the labeling threshold. If the distance is lower than the threshold, the energy yields the value 0. Finally, the third term, $E_3(L)$, prevents repeated labels in features associated with the same vertex, again assigning an energy value of ∞ if this occurs and 0 otherwise.

The label configuration L that minimizes the total energy function (2) corresponds to the configuration that obeys all the conditions.

The energy minimization algorithm uses a region growing scheme where a label is seeded in a random feature and it propagates to the surrounding features whenever an energy decrease is possible. This process repeats until all features have been labeled. The pseudocode in Table 1 describes the developed labeling algorithm:

```
Q = {} % labeling queue
C = {} % labeled features
repeat
If Q is empty
    seed a new label l in a random feature y^i ∉ C
    add y^i to C
    for each feature y^k neighbor of y^i
        if y^k ∉ C & labeling y^k with l lowers E(L)
            add y^k to Q
Else
    repeat
    y^i = Q(1)
    label y^i with l
    add y^i to C
    remove y^i from Q
    for each feature y^k neighbor of y^i
        if y^k ∉ C & labeling y^k with l lowers E(L)
            add y^k to Q
    until Q is empty
until all features have been labeled
```

Table 1: Labeling algorithm.

The size of the resulting patches and their distance to the surface provides good differentiation measures to assess if the features in that patch belong to the LV boundary or if they were produced by the background.

5 ROBUST MODEL ESTIMATION

The robust model estimation used is an extension of the S-PDAF algorithm described in [10] to the 3D case. In each iteration k, this estimation technique considers all possible combinations of considering each patch as valid or invalid. Assuming M_k patches were detected, there are $m_k = 2^{M_k}$ possible interpretations. Each combination is defined as a patch interpretation $I_i(k) = \{I_i^1(k), \ldots, I_i^n(k), \ldots, I_i^{M_k}(k)\}$, where $I_i^n(k) = 0$ if the nth patch in the ith interpretation is considered invalid and $I_i^n(k) = 1$ otherwise.

The model assumes that the LV boundary position is described by

$$x(k) = x(k-1) + w(k) \tag{3}$$

where $w(k) \sim N(0, Q)$ is white Gaussian noise with normal distribution.

For each interpretation $I_i(k)$, the observations $y_i(k)$ are generated by a different model. If an observation $y_i(k)$ is considered invalid (outlier), the model assumes it is generated by uniform distribution. Otherwise, the model assumes it relates to the boundary points $x(k)$ by:

$$y_i(k) = x(k) + v_i(k) \tag{4}$$

where $v_i(k) \sim N(0, R_i)$ is a white Gaussian noise with normal distribution associated with the valid features $y_i(k)$ of the interpretation $I_i(k)$.

The state estimate is then defined by:

$$\hat{x}(k) = \sum_{i=0}^{m_k} \hat{x}_i(k)\alpha_i(k) \tag{5}$$

where $\hat{x}_i(k)$ is the updated state conditioned on the hypothesis that $I_i(k)$ is correct (which is the same as the update state equation of a traditional Kalman filter), and $\alpha_i(k)$ is the association probability of the interpretation $I_i(k)$. A similar analysis is done to predict and update the covariance matrix [10].

5.1 Association Probabilities

The association probabilities, $\alpha_i(k)$, define the strength of the corresponding interpretation $I_i(k)$ in each iteration k of the adaptation procedure (from this point on we omit the dependence of k for the sake of simplicity). It is defined as

$$\alpha_i = P(I_i|Y, L, \hat{x}) \tag{6}$$

where Y and L are the set of the detected features and patches, respectively. Using a Bayesian approach, this probability can be decomposed into:

$$\alpha_i = \frac{P(Y|I_i, L, \hat{x}) \times P(I_i|L, \hat{x})}{\beta} \tag{7}$$

where $\beta = P(Y|L,\hat{x})$ is a normalization constant that does not depend on I_i, $P(Y|I_i,L,\hat{x})$ is the likelihood of the set of features Y and $P(I_i|L,\hat{x})$ is the prior probability of the interpretation I_i conditioned on the patches (i.e., based on its valid and invalid patches). Note that interpretations with overlapping patches will be assigned an association probability of 0, and patches considerably smaller then the larger ones will be promptly discarded to avoid an exponential growth of possible interpretations.

Assuming that the patches are independently generated, the likehood $P(Y|I_i,L,\hat{x})$ is expressed as the product of each individual probability of having a patch l_n at an average distance d to the surface. However, this probability is dependent on the hypothesis that l_n is considered valid or invalid: if $I_i^n = 0$ it is assumed that the probability distribution is uniform along the search line, whereas if $I_i^n = 1$ the probability distribution is assumed Gaussian with mean 0 and covariance proportional to the length of the search line, V. Formally:

$$P(l_n|I_i,L,\hat{x}) \sim \begin{cases} V^{-1} & \text{if } I_i^n = 0 \\ \rho^{-1}N(d;0,\sigma) & \text{otherwise} \end{cases} \quad (8)$$

where ρ is the normalization constant.

As to the prior probability of each interpretation I_i, it is related to the size of its valid and invalid patches. The probability $P(I_i|L,\hat{x})$ is also decomposed as the product of each individual probability of the patches, $P(I_i^n|L,\hat{x})$. It is assumed that larger patches are more likely to belong to the LV boundary. Therefore, these should receive a higher probability. On the other hand, when considered invalid, these should be assigned a small probability.

The resulting prior probability yields:

$$\begin{aligned} P(I_i|L,\hat{x}) \quad = \quad & \prod_{l_n:I_i^n=1} [a\log(A_{l_n}+1)+b] \times \\ & \prod_{l_n:I_i^n=0} 1-[a\log(A_{l_n}+1)+b)] \end{aligned} \quad (9)$$

where:

$$\begin{aligned} a &= \tfrac{P_A-P_B}{1-\log(A_{max}+1)} \\ b &= P_B - a\log(A_{max}+1). \end{aligned} \quad (10)$$

and P_A, P_B and A_{max} are constants, and A_{l_n} is the area of the patch l_n (number of features it comprises). These assumptions assure that patches above a certain area A are preferably considered.

6 RESULTS

The proposed system was tested using several different datasets, both real and synthetic. The purpose of the synthetic data was to assess the functionality of the model, for which we will present more insight of the framework. The real data corresponds to echocardiographic volumes (courtesy of Dr. Jacinto Nascimento). The algorithm was applied to four different echocardigraphic volumes. A quantitative assessment of the system's performance will be provided using error metrics between the obtained segmentation and the manual segmentation performed by an expert - the ground truth (GT).

6.1 Evaluation Metrics

We use five similarity metrics to compare the output of the algorithm with the reference contours, namely: the Hammoude metric [15], d_{HMD}, the average metric, d_{AV}, the Hausdorff metric [16], d_{HDF}, mean sum of squared distance metric, d_{MSSD}, and the mean absolute distance metric, d_{MAD}. These are defined as follows: consider R_Ψ as the region delimited by model segmentation and R_Ω as the region delimited by the GT. The Hammoude metric is defined by:

$$d_{HMD}(\Psi,\Omega) = \frac{\#((R_\Psi \cup R_\Omega) - (R_\Psi \cap R_\Omega)}{\#((R_\Psi \cup R_\Omega))} \quad (11)$$

This error metric corresponds to the fraction of area between the two contours using a XOR operator. Low values of d_{HMD} indicate high similarity between both regions.

Now consider the border of the model segmentation defined by the points $\Psi = \{\psi_1,\ldots,\psi_{N_\psi}\}$ and the border of the GT $\Omega = \{\omega_1,\ldots,\omega_{N_\omega}\}$. The average metric between two contours is defined as the average distance between each point ψ_i to the closest point in Ω, $d(\psi_i,\Omega) = \min_j ||\omega_j - \psi_i||$,

$$d_{AV} = \frac{1}{N_\Psi}\sum_{i=1}^{N_\Psi} d(\psi_i,\Omega) \quad (12)$$

where N_Ψ is the length of Ψ. The Hausdorff metric is defined as the maximum value of $d(\psi_i,\Omega)$ between the two contours:

$$d_{HDF}(\Psi,\Omega) = \max\left(\max_i\{d(\psi_i,\Omega)\}, \max_j\{d(\omega,\Psi)\}\right) \quad (13)$$

Finally, the MSSD and MAD metric is defined by:

$$d_{MSSD}(\Psi,\Omega) = \frac{1}{N}\sum_{i=1}^{N} ||\psi_i - \omega_i||^2 \quad (14)$$

and

$$d_{MAD}(\Psi,\Omega) = \frac{1}{N}\sum_{i=1}^{N} ||\psi_i - \omega_i|| \quad (15)$$

which correspond, respectively, to the squared distance's average and to the absolute distance's average between corresponding points in the boundaries.

6.2 Parameter Definition

All the presented results were obtained using the parameters that achieve better overall results:

- In (1), we used $\alpha = 0.7$, $\beta = 0.05$ and $\gamma = 0.9$; the stopping criterion was the average displacement of the vertices $k_{stop} < 0.005$;
- In the feature extraction algorithm, the threshold using the maxima detection was $t_f = 0.5c_{max}$, where c_{max} is the highest peak of the filter output; as to the labeling threshold (i.e., the maximum distance allowed between neighboring features with the same label) used was $t_l = 8$;
- Finally, in (9) we used $P_A = 0.05$, $P_B = 0.95$ and $A_{max} = 700$ (the number of vertices in the surface).

6.3 Synthetic Data

We will present one particular test using a synthetic volume. This synthetic volume contains a sphere corrupted by white Gaussian noise with zero mean (see Fig. 7). Although many features are detected (an average of approximately 3 features associated with a single vertex), only one large patch is extracted (see Fig. 6) and all the other smaller noise-originated patches are discarded. The association probabilities of the existing interpretations are the following:

$$\alpha_1 = P(I_1 = \{I^1 = 0\}) = 0.04$$
$$\alpha_2 = P(I_2 = \{I^1 = 1\}) = 0.96$$

which means a high confidence degree is assigned to the large (correct) patch. Fig. 7 shows that the model is able to correctly adapt to the desired sphere.

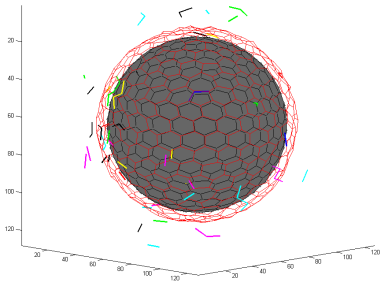

Figure 6: Patch detection in a synthetic noisy sphere.

6.4 Echocardiographic Data

As mentioned before, the segmentations of the LV in four different echocardiographic volumes are presented in Fig. 8. For each volume, a single slice is shown twice: one containing the obtained segmentation and one containing the GT. The final three-dimensional surface is also presented.

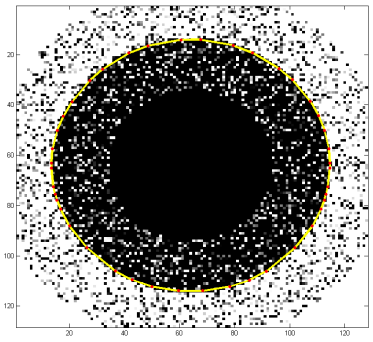

Figure 7: Slice view of the final configuration of the surface (yellow contour).

6.5 Quantitative Assessment

Although the previous results show that the developed segmentation system performs reasonably well, quantification measurements are required to compare its performance with other similar methods. Table 2 presents the values of the similarity metrics.

	Volume 1	Volume 2
$\bar{d}_{HMD}$	0.17 ± 0.04	0.25 ± 0.07
$\bar{d}_{AV}$	4.1 ± 0.8	4.1 ± 1.0
$\bar{d}_{HDF}$	9.9 ± 1.8	12.6 ± 4.5
$\bar{d}_{MSSD}$	51 ± 26	83 ± 44
$\bar{d}_{MAD}$	6.3 ± 1.7	7.7 ± 2.3
	Volume 3	Volume 4
$\bar{d}_{HMD}$	0.20 ± 0.04	0.26 ± 0.13
$\bar{d}_{AV}$	4.1 ± 0.9	4.2 ± 1.1
$\bar{d}_{HDF}$	10.3 ± 2.0	12.8 ± 6.3
$\bar{d}_{MSSD}$	37 ± 17	65 ± 93
$\bar{d}_{MAD}$	5.2 ± 1.4	6.0 ± 3.5

Table 2: Results of the evaluation methods for each volume (the distance values are expressed in pixels).

The table shows high similarity between the estimated contour and the GT, indicating a good match.

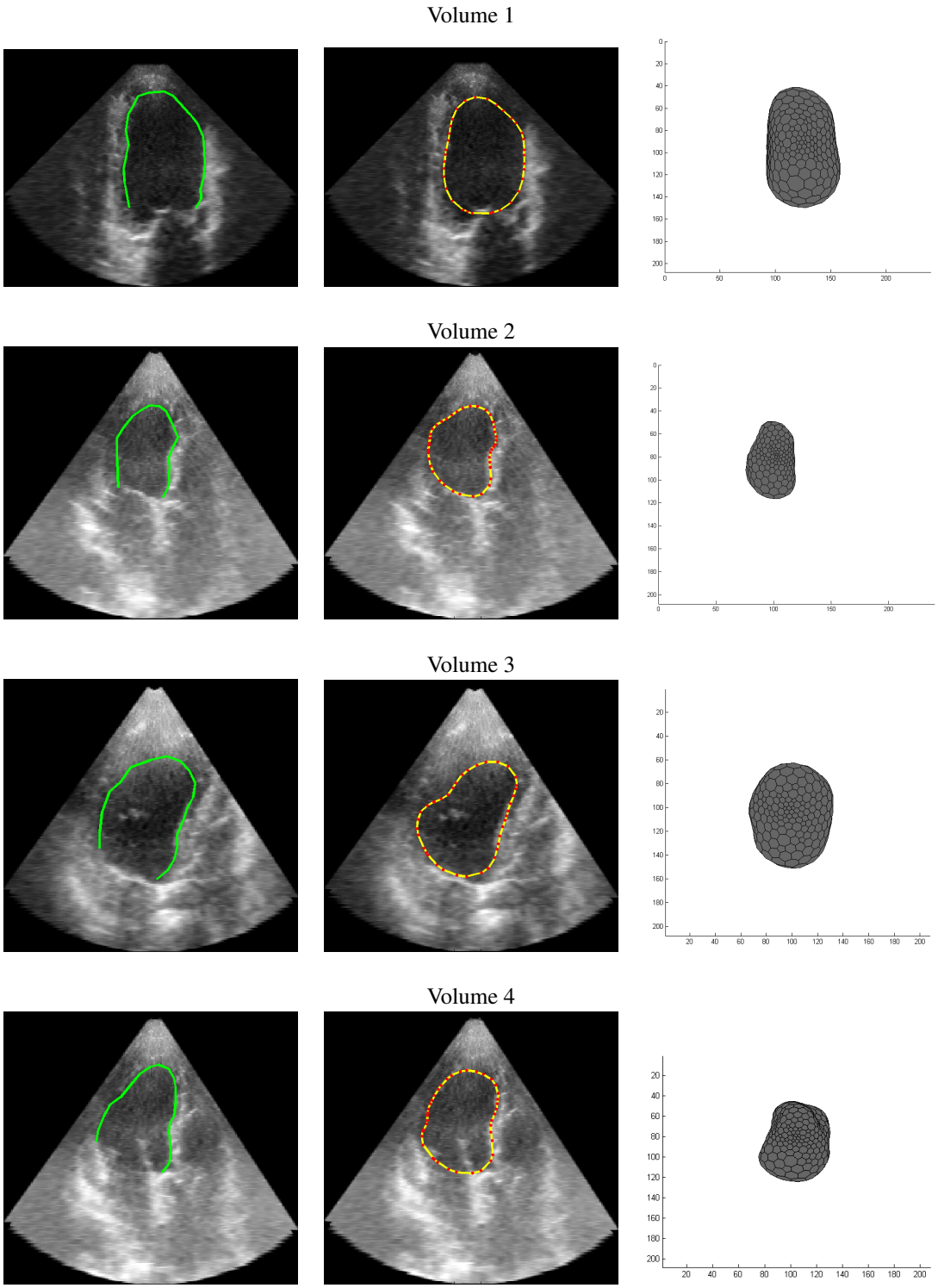

Figure 8: Slice from the echocardiographic showing (on the left) the GT and (on the center) the obtained segmentation. Final configuration of the surface (on the right).

7 CONCLUSIONS

This paper addresses the automatic LV segmentation problem in 3D echocardiographic data. Due to the nature of the volumes, many of the detected features usually do not belong to the LV boundary. The proposed system uses a robust estimation technique based on PDAF that prevents the segmentation to be misguided by those outliers.

The results shown demonstrate that the proposed system performs a good segmentation of the LV, with potential application to accurately compute cardiac measurements such the systemic and diastolic volumes and the corresponding ejection fraction.

Further tests shows that the developed system is slightly over-dependent on the initialization procedure, which does not help improving the repeatability of LV segmentations. This could be avoided using an automatic initialization scheme, such as in [7].

ACKNOWLEDGEMENTS

This work was supported by FCT under contract PTDC/EEA-CRO/103462/2008 (project HEARTRACK).

REFERENCES

[1] R. Juang, E. McVeigh, B. Hoffmann, D. Yuh, and P. Burlina, "Automatic segmentation of the left-ventricular cavity and atrium in 3d ultrasound using graph cuts and the radial symmetry transform," in *2011 IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pp. 606–609, 2011.

[2] K.-L. Chan and J. P. Veinot, *Anatomic Basis of Echocardiographic Diagnosis*. Oxford Univesity Press, 1st ed., 2011.

[3] M. Nillesen, R. Lopata, and et al., "3d segmentation of the heart muscle in real-time 3d echocardiographic sequences using image statistics," *IEEE Ultrasonics Symposium, 2006*, pp. 1987–1990, 2006.

[4] B. Scowen, S. Smith, and M. Vannan, "Quantitative 3d modelling of the left ventrical from ultrasound images," in *Euromicro Conference*, vol. 2, pp. 432–439, 2000.

[5] X. Hang, N. Greenberg, and J. Thomas, "Left ventricle quantification in 3d echocardiography using a geometric deformable model," *Computers in Cardiology*, pp. 649–652, 2005.

[6] H. Yu, M. Pattichis, and M. Goens, "Robust segmentation and volumetric registration in a multi-view 3d freehand ultrasound reconstruction system," in *ACSSC '06*, pp. 1058–6393, 2006.

[7] L. Yang, B. Georgescu, Y. Zheng, P. Meer, and D. Comaniciu, "3d ultrasound tracking of the left ventricle using one-step forward prediction and data fusion of collaborative trackers," in *IEEE CVPRW'08*, pp. 1–8, 2008.

[8] F. Orderud, *Real-time segmentation of 3D echocardiograms using a state estimation approach with deformable models*. PhD thesis, Norwegian University of Science and Technology, Department of Computer and Information Science, 2010.

[9] Y. Bar-Shalom and T. Fortmann, *Tracking and Data Associaton*. New York: Academic, 1988.

[10] J. C. Nascimento and J. S. Marques, "Robust shape tracking in the presence of cluttered background," *IEEE Trans. Multimedia*, vol. 6, December 2004.

[11] H. Delingette, "General object reconstruction based on simplex mesh," *International Journal of Computer Vision*, no. 32, pp. 111–142, 1999.

[12] K. N. Kutulakos and S. M. Seitz, "A theory of shape by space carving," *International Journal of Computer Vision*, no. 38, pp. 199–218, 2000.

[13] C. B. Barber, D. P. Dobkin, and H. Huhdanpaa, "The quickhull algorithm for convex hulls," *ACM Transactions on Mathematical Software*, vol. 22, pp. 469–483, December 1996.

[14] A. Blake and M. Isard, *Active Contour*. Springer, 1998.

[15] A. Hammoude, *Computer-assisted Endocardial Border Identification from a Sequence of Two-dimensional Echocardiographic Images*. PhD, University Washington, 1988.

[16] D. P. Huttenlocher, G. A. Klanderman, and W. J. Rucklidge, "Comparing images using hausdorff distance," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 15, no. 9, pp. 850–863, 1993.