31st Annual International Conference of the IEEE EMBS
Minneapolis, Minnesota, USA, September 2-6, 2009

Fast and Automatic LV Mass Calculation from Echocardiographic Images via B-spline Snake Model and Markov Random Fields

Mahdi Marsousi, Armin Eftekhari, *Student Member, IEEE*, Javad Alirezaie, *Senior Member, IEEE*
, Armen Kocharian, Ershad Sharifahmadian, *Student Member, IEEE*

***Abstract*— Left ventricular (LV) mass has several important diagnostic and indicative implications. In this paper, a fast and accurate technique for detection of inner and outer boundaries of LV and, consequently, calculation of LV mass from apical 4-chamber echocardiographic images is presented. For detection of the inner boundary, a modified B-spline snake is proposed, which relies merely on image intensity and obviates the need for computationally-demanding image forces. The outer boundary is then obtained using a Markov random fields model in the neighborhood of the estimated inner border. Experimental validation of the proposed technique demonstrates remarkable improvement over conventional algorithms.**

I. Introduction

CALCULATION of LV mass from echocardiographic images has several important diagnostic implications [1]. Automation of this task, however, requires accurate detection of inner (epicardial) and outer (endocardial) boundaries of LV, which is mainly impeded by inherent low-contrast and speckle noise in echocardiographic images, and thereby has given rise to many algorithms [2]. For example, using active contour algorithms, with B-spline snake as a member, involves finding the epicardial boundary at first and using the fitted contour as the initialization to find the endocardial border [3, 4]. The basic idea of the B-spline snake model is to define an inherently smooth, energy minimizing curve that is driven towards desired image features by external forces. This technique is prone to errors on low-quality data [3]. Another successful approach to address boundary detection is Markov random field (MRF) model (possibly in combination with active contours) [2]. A MRF is a probabilistic model of the elements of a multidimensional random variable where the components have only local interactions [5]. In practice, this family of algorithms suffers from high computational complexity.

In this paper, with focus on apical 4-chamber echocardiographic images, a fast and accurate algorithm for detection of endocardial and epicardial boundaries and, calculation of LV mass is presented. Endocardial boundary is detected using a modified B-spline snake algorithm, which avoids computationally expensive optimization. Epicardial border is then found using MRF based labeling on a small neighborhood of the estimated endocardium. Experimental validation demonstrates improvement in speed over conventional algorithms, while detecting the boundaries with high accuracy. The rest of this paper is organized as follows. Sections II and III describe a framework for detection of inner and outer boundaries of LV, respectively. Section IV focuses on LV mass calculation, and Section V is dedicated to experimental validation and this paper concludes in Section VI.

This work was supported by K.N. Toosi University of Technology, Tehran, Iran, and by RCSTIM, Tehran, Iran. M. Marsousi and A. Eftekhari, are with the Department of Biomedical Engineering at K.N. Toosi University (a.eftekhari@ee.kntu.ac.ir). J. Alirezaie is with the Department of Electrical and Computer Engineering at Ryerson University, Toronto, ON, Canada. He is also with RCSTIM. A. Kocharian is with the Tehran University of Medical Sciences. E. Sharifahmadian is with Research Centre for Science and Technology in Medicine, Imam Hospital, Tehran, Iran. He is also with Tehran University of Medical Sciences (TUMS), Tehran, Iran.

II. Endocardial Boundary Detection

In this section, an algorithm for detection of endocardial boundary in echocardiographic images is developed.

A. Cubic B-spline Snake Model

In cubic B-spline snake, the commonly used B-spline snake algorithm, the contour is represented by cubic B-spline basis functions and few control points govern the deformation of the contour [4, 6]. Cubic B-spline snake is characterized by N control points $\mathrm{Q}_n = [x_n, y_n],\ n = 1{:}N$ and N connected curve segments $\mathrm{g}_n(\theta) = [u_n(\theta), v_n(\theta)],\ n = 1{:}N$, where $0 \le \theta < 1$. Each curve segment is a linear combination of four cubic polynomials in θ, as indicated in (1).

$$\mathrm{g}_n(\theta) = [\theta^3 \quad \theta^2 \quad \theta \quad 1] \begin{bmatrix} \frac{-1}{6} & \frac{1}{2} & \frac{-1}{2} & \frac{1}{6} \\ \frac{1}{2} & -1 & \frac{1}{2} & 0 \\ \frac{-1}{2} & -1 & \frac{1}{2} & 0 \\ \frac{-1}{6} & \frac{2}{3} & \frac{1}{6} & 0 \end{bmatrix} \begin{bmatrix} \mathrm{Q}_{(n-1) \bmod N} \\ \mathrm{Q}_{n \bmod N} \\ \mathrm{Q}_{(n+1) \bmod N} \\ \mathrm{Q}_{(n+2) \bmod N} \end{bmatrix} \quad (1)$$

Cubic B-spline snake is now defined as $\mathrm{r}(\theta) = \sum_{n=1}^{N} \mathrm{g}_n(\theta)$, where $0 \le \theta < 1$. Setting $\theta = 0$ in (1), yields the so-called node points $\mathrm{P}_n,\ n = 1{:}N$, which are located on the contour. Further, denoting the collection of control points and node points by $N \times 2$ matrices Q and P, respectively, we have $\mathrm{P} = \mathrm{AQ}$, where $\mathrm{A} \in \mathrm{R}^{N \times N}$ is shown [6] to be:

$$\mathrm{A} = \begin{bmatrix} 1/6 & 2/3 & 1/6 & 0 & \dots & 0 & 0 & 0 \\ 0 & 1/6 & 2/3 & 1/6 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & 1/6 & 2/3 & 1/6 \\ 1/6 & 0 & 0 & 0 & \dots & 0 & 1/6 & 2/3 \\ 2/3 & 1/6 & 0 & 0 & \dots & 0 & 0 & 1/6 \end{bmatrix} \quad (2)$$

B. Estimating Parameters from Image Data

Due to the inherent smoothness and continuity [4, 6], deformation of the contour in B-spline snake algorithm is completely determined in interaction with external forces. Generally speaking, conventional external forces, either lack adequate capture range and ability to progress into boundary concavities, or suffer from excessive computational cost, as in the gradient vector flow (GVF) and similar forces [2]. In the following, an efficient adaptive balloon force is proposed, which relies on the so-called stopping factors for accurate fitting into the endocardial boundary.

In order to calculate the displacement of each node point P_n in cubic B-spline snake algorithm, one should consider the contribution of external forces on the four adjacent curve segments [7]. Generally, this necessitates computationally expensive optimization-based approach that requires inversion of high-dimensional matrices [6, 7]. Following the reasoning presented in [7], however, a geometrical point of view is pursued here, which only considers a weighted contribution of two adjacent curve segments. More specifically, suppose each curve segment $\mathrm{g}_n^t(\theta)$ is sampled at M points $\mathrm{g}_n^t(m/M),\ m = 0{:}\,M-1$, where t denotes the dependence on time. Suppose, also, that $MN \times 2$ matrix F_{bal}^t contains the balloon forces (with unit norm) at all sampled points along the contour. To calculate the displacement of P_n, we first transmit the balloon forces on g_n and g_{n-1} to P_n. This is performed using a weighted sum of balloon forces, which aims to weigh the contribution of sampled points close to P_n. In short, the proposed scheme transmits weighted external forces associated with adjacent curve segments to their common node point to determine the displacement of the corresponding node. As demonstrated in Section V, this simplified model, summarized in (3), proves to be fast and reliable in practice.

$$\tilde{\mathrm{F}}_{bal}^t = \mathrm{D}\mathrm{F}_{bal}^t$$

$$\mathrm{D} = \begin{bmatrix} \mathrm{d}_1^{\mathrm{T}} & 0 & \dots & 0 & \mathrm{d}_2^{\mathrm{T}} \\ 0 & \mathrm{d}^{\mathrm{T}} & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & \dots & \mathrm{d}^{\mathrm{T}} & 0 \end{bmatrix}$$

$$\mathrm{d} = [\mathrm{d}_2^{\mathrm{T}}, \mathrm{d}_1^{\mathrm{T}}]^{\mathrm{T}}$$

$$\mathrm{d}_1(m) = 1 - \tfrac{m-1}{M},\ m = 1{:}\,M$$

$$\mathrm{d}_2(m) = \tfrac{m-1}{M},\ m = 1{:}\,M \qquad (3)$$

In (3), $N \times 2$ matrix $\tilde{\mathrm{F}}_{bal}^t$ stores the balloon forces on all node points. Finally, displacement of nodes is determined in proportion to the balloon force on the corresponding nodes:

$$\Delta\mathrm{P}^t = \mathrm{diag}(\mathrm{b}^t)\tilde{\mathrm{F}}_{bal}^t$$
$$\Delta\mathrm{Q}^t = \mathrm{A}^{-1}\Delta\mathrm{P}^t \qquad (4)$$

where $\mathrm{diag}(\mathrm{a})$, for a vector a, denotes the diagonal matrix A with $\mathrm{A}(i,i) = \mathrm{a}(i)$. The proportionality factors, stored in $\mathrm{b}^t \in \mathbb{R}^N$, are iteratively reduced from an initial scale-dependent value $(\beta_0/2^r)1_N$ to guarantee convergence to the boundary:

$$\mathrm{b}^t = \mathrm{b}^{t-1} - \tilde{\mathrm{s}}\,,\ \text{where } \mathrm{b}^0 = (\beta_0/2^r)1_N \qquad (5)$$

in which, $1_N = [1, \dots, 1]^{\mathrm{T}}$ and r denotes the scale at which the algorithm is applied (see Section II.B.4). Moreover, $\tilde{\mathrm{s}} \in \mathbb{R}^N$ stores the associated reduction rates, dubbed "stopping factors". Calculation of $\tilde{\mathrm{s}}$ is described in the next section.

1) Stopping Factors

As mentioned above, stopping factors determine the rate at which balloon force vanish, thus guaranteeing appropriate fitting into the boundaries. Here, $\tilde{\mathrm{s}}$ is defined as $\tilde{\mathrm{s}} = \mathrm{D}\mathrm{s}$, where $\mathrm{s} = [\mathrm{s}_1^{\mathrm{T}}, \dots, \mathrm{s}_N^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^{MN}$ is obtained by concatenating the vectors $\mathrm{s}_n \in \mathbb{R}^M$. Therefore, $\tilde{\mathrm{s}}(n)$ is calculated as a weighted sum of the entries of s_n and $\mathrm{s}_{(n-1) \bmod N}$, where $\mathrm{s}_n \in \mathbb{R}^M$ merely estimates the proximity of the sampled points along the nth curve segment g_n to the boundary:

$$\mathrm{s}_n = \begin{bmatrix} I[f(\mathrm{g}_n(0)) \geq T_n] \\ \vdots \\ I[f(\mathrm{g}_n(\frac{M-1}{M})) \geq T_n] \end{bmatrix} \qquad (6)$$

in which $I[\cdot]$ is the indicator function and $f(\mathrm{o})$ gives the image intensity at point o. Further, $T_n = 0.7 f_{Max,n}$ is the "intensity threshold" associated with nth curve segment, and $f_{Max,n}$ is the maximum intensity along the normal line at P_n within an appropriate range. To improve the computational efficiency, stopping factors are calculated once at each scale (except for new nodes, where corresponding intensity thresholds are calculated upon insertion). This is justified because deformation of g_n only slightly changes T_n.

2) Node Insertion Strategy

Sharp intensity variations, noise and artifacts necessitate more than primitive distance-based node insertion strategy to avoid progressing into gaps. In this paper, each curve segment is checked for violation of the distance threshold T_d at each iteration. A new node is inserted at the midpoint of every "long" segment g_n, if the corresponding intensity threshold (defined in previous section) does no differ largely from T_n and T_{n-1}. This simple modification effectively precludes progressing into gaps.

3) Automatic Contour Initialization

To obtain the initial seed point, a template formed by few training images is matched with the image at the coarsest resolution to reduce the effect of noise. Finally, initial node points are inserted, equally spaced, on the perimeter of a small circle centered at this seed point.

(1) Determine the initial contour (Section II.B.3).
(2) For resolutions $r = 1{:}\,3$
  a. Consider the $r-1$th level of wavelet decomposition of the original image.
  b. (Re)calculate the stopping factors for all nodes (see Section II.B.1).
  c. Store the balloon forces along the contour in F_{bal}^t.
  d. Use (3) to calculate the normal components of balloon forces on nodes, which is then stored in $\tilde{\mathrm{F}}_{bal,\perp}^t$.
  e. Calculate the displacement of nodes (and thereby evolution of the contour) using (4).
  f. Using the node insertion criterion, add enough number of nodes to the contour (see Section II.B.2).
  g. If convergence is not achieved, go to step (c).
(3) Final contour is the estimated endocardial boundary.

Fig. 1. Final multiscale algorithm for endocardial boundary detection.

4) Multi-Resolution Strategy

Employing a multi-resolution strategy generally increases the robustness against noise and local minima and improves the convergence speed of active contours [4]. Here, Daubechies wavelet decomposition has been used to construct a three layer image pyramid. Our algorithm is then applied at the coarsest level, here $r = 3$. Upon convergence, the solution is propagated to the next finer level ($r = 2$) as an initial snake. This coarse-to-fine iterative strategy continues until the finest level of the pyramid (the original image) is reached. Our final multiscale algorithm for endocardial boundary detection is summarized in Fig. 1.

III. Epicardial Boundary Detection

Detection of the outer boundary of LV is described in this section. A Markov random field (MRF) based labeling is developed as follows [5]. Consider two random fields X and Y, where $X = \{X_s, s \in S\}$ is the label field, and $Y = \{Y_s, s \in S\}$ is the field of observations. Here, each $Y_s \in \{0:255\}$, and each $X_s \in \{e_0 = \text{blood}, e_1 = \text{myocardium}\}$. Furthermore, the lattice S with $\#S$ sites s is defined as follows. Starting at the estimated inner boundary, sites are placed on R evenly-distributed rays eradiated from the seed point obtained in Section II.B.3 (see Fig. 2.a). Now, denoting the discrete probability $P(X = x)$ by $P(x)$ for simplicity, the maximum *a posteriori* estimation of x is obtained as:

$$\hat{x} = \arg\max_x P(y|x)P(x) \quad (7)$$

Markovian assumption on $P(x)$ implies that the conditional probability of a particular site s depends only on its immediate neighbors $N(s)$:

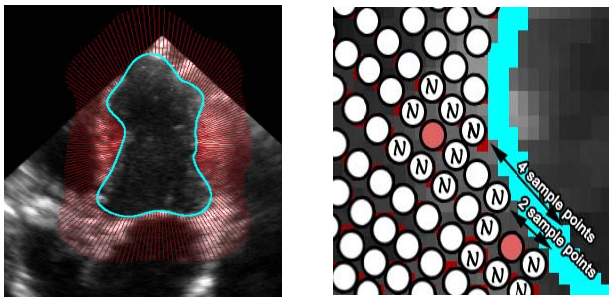
$$P(x_s|x_{\acute{s}}, \acute{s} \neq s) = P\big(x_s|x_{\acute{s}}, \acute{s} \in N(s)\big) \quad (8)$$

Corresponding neighborhood system is schematized in Fig. 2.b. Similar to 2D MRF model in Cartesian coordinates, the second-order neighborhood system is used for each site, i.e. all doubleton sets of eight immediate neighbors are considered. Then, noting that an MRF is completely described by a Gibbs distribution [2], we may write:

$$P(x_s) = \frac{1}{Z_X} e^{-U(x_s)}, \text{where } U(x_s) = \gamma \sum_{\acute{s} \in N(s)} \delta(x_s, x_{\acute{s}}) \quad (9)$$

where $\delta(\cdot,\cdot)$ and Z_X are the Kronecker delta function and normalization factor , respectively. Now, it remains to determine the conditional distribution $P(y|x)$, which is usually assumed to be Gaussian: $P(y_s|x_s = e_0) = \mathcal{N}(\mu_{e_0,i}, \sigma_i^2)$ and $P(y_s|x_s = e_1) = \mathcal{N}(\mu_{e_1,i}, \sigma_i^2)$. Here, $\mathcal{N}(\mu, \sigma^2)$ denotes the normal density with mean μ and variance σ^2. In addition, $\mu_{e_0,i}$ and $\mu_{e_1,i}$ denote the average intensity of a site on the ith ray, located in the blood pool and myocardium, respectively. These parameters are experimentally set to $\mu_{e_0,i} = 0.7 f_{Max,i}$, and $\mu_{e_1,i} = 1.3 f_{Min,i}$, where $f_{Max,i}$ and $f_{Max,i}$ are the maximum and minimum intensities along the range of the ith ray, respectively. Finally, the maximum *a posteriori* estimation of x is is obtained as:

$$\hat{x} = \arg\max_x P(y|x)P(x) = \arg\min_x (y - \mu_x)^2 + U(x) \quad (10)$$



(a) (b)

Fig. 2. (a) Sites in our problem. (b) Corresponding neighborhood system, in which two sample sites and their neighbors are depicted by red and N-labeled circles, respectively.

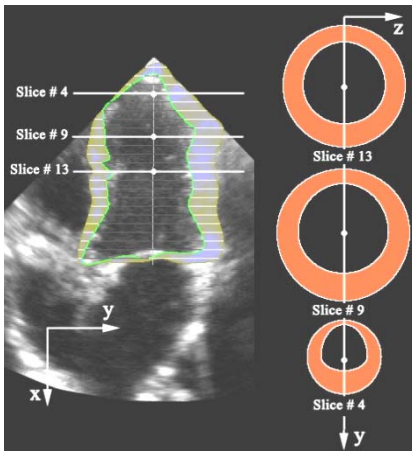


Fig. 3. LV mass calculation based on inner and outer boundaries.

where the latter equality is obtained by taking natural logarithm and ignoring the constants. Note that γ, defined in (9), determines the contribution of our prior knowledge in the labeling process. To solve (10), instead of conventional EM algorithm [5], a simpler and faster approach is pursued which builds upon the fact that most of the sites belong to myocardium. Starting at the sites with locally maximum intensity as the initial set, at each iteration only immediate neighbors of the set are processed and are added to the set provided the posterior probability of belonging to myocardium outweighs that of belonging to the blood pool. This procedure continues until convergence.

IV. LV Mass Calculation

For LV mass calculation, first, the long axis in LV is found by the algorithm presented in [4]. Then, LV is divided into K slices along the long axis, as depicted in Fig. 3. Clearly, intersection of y axis with the inner boundary in kth slice generates two distances from the axis, namely r_{k,i_1} and r_{k,i_2}. Similarly, r_{k,o_1} and r_{k,o_2} are associated with the outer boundary. Assuming that radius of each boundary varies linearly in the yz plane, the area of kth slice is found via elementary calculus:

$$A_k = \frac{\pi}{6}\left[{r_{k,o_2}}^2 + r_{k,o_2} r_{k,o_1} + r_{k,o_1}^2 - {r_{k,i_2}}^2 - r_{k,i_2} r_{k,i_1} - r_{k,i_1}^2\right] \quad (11)$$

Then, LV volume is calculated as $V = h(\sum_{k=1}^{K} A_k)$ and LV mass is obtained as $\rho_d V$, where h is the length of the long axis and $\rho_d = 1.05$ (gram/cm^3) is the average density

of the myocardium at end of diastole (maximum expansion of heart).

V. EXPERIMENTS

In this section, the proposed framework for LV mass calculation is validated through comprehensive experiments on a database of 45 apical 4-chamber echocardiographic images. Acquisition of images (as well as extraction of boundaries and mass calculation for comparison) was performed by an expert using a VIVID3 echocardiography instrument manufactured by GE. Original images had a depth of 14 Centimeters (equivalent to 400 pixels) scanning over a 90° arc. Our software was written in MATLAB7 and implemented on a computer with an Intel Core 2 Doe, 2.2 GHz processor. In addition, CPU time was used as a rough measure of complexity for our algorithm. Furthermore, accuracy of a detected boundary was defined as the summation of all closed areas formed as a result of the mismatch of the contour with reference boundary, divided by the area of the reference boundary. Recall that the reference boundary is provided by the expert.

As our first experiment, the proposed B-spline snake algorithm for inner boundary detection was evaluated by calculating the average accuracy of the obtained boundaries in our database. Fig. 4 illustrates the output of the proposed algorithm compared to manually extracted boundaries. Using the experimentally set parameters, LV mass was calculated for all images in the database and compared with other conventional algorithms. As the reference for this comparison, we used the LV mass calculated by the expert (using the modified Simpson's formula [4]). Results, as reported in Table 1, demonstrate clear improvement in computational complexity over conventional algorithms, while producing excellent average accuracy. We note that, in contrast to conventional MRF based techniques, proposed framework is robust against high intensity speckle noise in the blood pool. In addition, for successful boundary detection, conventional active contour algorithms require calculation of the computationally-demanding GVF (or similar) forces. Since our algorithm is tailored for echocardiographic images, this requirement is obviated in the proposed framework, which is based merely on image the proposed framework, which is based merely on image intensities for contour evolution. LV mass was calculated for all images in the database and compared with other conventional algorithms. As the reference for this comparison, we used the LV mass calculated by the expert (using the modified Simpson's formula [4]). Results, as reported in Table 1, demonstrate clear improvement in computational complexity over conventional algorithms, while producing excellent average accuracy. We note that, in contrast to conventional MRF based techniques, proposed framework is robust against high intensity speckle noise in the blood pool. In addition, for successful boundary detection, conventional active contour algorithms require calculation of the computationally-demanding GVF (or similar) forces. Since our algorithm is tailored for echocardiographic images, this requirement is obviated in the proposed framework, which is based merely on image intensities for contour evolution.

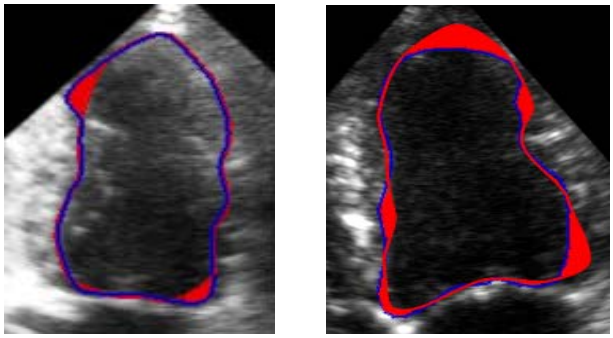


Fig. 4. Manually and automatically detected boundaries and the closed areas formed as the result of the mismatch between two contours, for two echo images. Images are cropped for better visualization.

Table 1. Average accuracy and computational cost of LV mass calculation for several algorithms.

Method	Elapsed Time (Sec.)		Accuracy (%)	
	Mean	Variance	Mean	Variance
MRF [5]	8.1	2.2	94.22	4.38
Snake & GVF [6]	5.3	3.5	90.56	6.01
B-snake & GVF [6]	3.7	1.2	95.43	3.23
Proposed Method	1.1	0.5	93.75	2.12

VI. CONCLUSIONS

In this paper, an accurate framework for LV boundary detection and mass calculation was presented. Proposed technique relies on a modified B-spline snake algorithm for extraction of the inner boundary and employs an MRF model for detecting the outer border. Experimental results demonstrated that our method, while computationally efficient and fast, achieves remarkable accuracy in boundary detection and mass calculation.

REFERENCES

[1] E.G Caiani, C. Corsi, L. Sugeng, P. MacEneaney, L. Weinert, V. Mor-Avi, R.M. Lang, "Improved quantification of left ventricular mass based on endo- and epicardial surface detection using real-time three-dimensional echocardiography," British Med. Jr., 2005.

[2] J.A. Noble, D. Boukerroui, "Ultrasound Image Segmentation: A Survey," IEEE Trans. Med. Imag., vol. 25, pp. 987-1010, 2006.

[3] G. Jacob, J.A. Noble, C. Behrenbruch, A.D. Kelion, A.P. Banning, "A Shape-Space-Based Approach to Tracking Myocardial Borders and Quantifying Regional Left-Ventricular Function Applied in Echocardiography," IEEE Trans. Medical Imag., vol. 21, pp. 226-238, 2002.

[4] M. Marsousi, A. Eftekhari, J. Alirezaie, A. Kocharian, " Extraction of left ventricular endocardial boundaries in echocardiography images using fast and adaptive B-spline snake," Int. Symp. Computer Assisted Radiology, 2009, Accepted.

[5] J. Cheng, S. Foo, "Boundary detection in echocardiographic images using Markovian level set method," IEICE Trans. Inf. & Syst., vol. E90–D, pp. 1292-1300, 2007.

[6] Y. Wang, E.K. Teoch, D. Shen, "Structure Adaptive B-Snake for Segmenting Complex Objects," IEEE Int. Conf. Image Processing, vol. 2, Issue 7, pp. 769-772, 2001.

[7] M. Marsousi, A. Eftekhari, J. Alirezaie, "Object contour extraction in medical images by fast adaptive B-Snake," 30th annual Int. Conf. of the IEEE Eng. in Medicine and Biology Society, pp. 3068-3071, 2008.