

An Efficient Method for Deformable Segmentation of 3D US Prostate Images

Yiqiang Zhan^{1,2,3}, Dinggang Shen^{1,2}

¹ Sect. of Biomedical Image Analysis, Dept. of Radiology, University of Pennsylvania, Philadelphia, PA

² Center for Computer-Integrated Surgical Systems and Technology, Johns Hopkins University, Baltimore, MD

³ Dept. of Computer Science, Johns Hopkins University, Baltimore, MD
yzhan@cs.jhu.edu, dinggang.shen@uphs.upenn.edu

Abstract We previously proposed a deformable model for automatic and accurate segmentation of prostate boundary from 3D ultrasound (US) images by matching both prostate shapes and tissue textures in US images[6]. Textures were characterized by a Gabor filter bank and further classified by support vector machines (SVM), in order to discriminate the prostate boundary from the US images. However, the step of tissue texture characterization and classification is very slow, which impedes the future applications of the proposed approach in clinic applications. To overcome this limitation, we firstly implement it in a 3-level multi-resolution framework, and then replace the step of SVM-based tissue classification and boundary identification by a Zernike moment-based edge detector in both low and middle resolutions, for fast capturing boundary information. In the high resolution, the step of SVM-based tissue classification and boundary identification is still kept for more accurate segmentation. However, SVM is extremely slow for tissue classification as it usually needs a large number of support vectors to construct a complicated separation hypersurface, due to the high overlay of texture features of prostate and non-prostate tissues in US images. To increase the efficiency of SVM, a new SVM training method is designed by effectively reducing the number of support vectors. Experimental results show that the proposed method is 10 times faster than the previous one, yet without losing any segmentation accuracy.

1 Introduction

Prostate cancer continues to be the second-leading cause of cancer death in American men [1]. As transrectal ultrasound (TRUS) images have been widely used for the diagnosis and treatment of prostate cancer, the accurate segmentation of the prostate from TRUS images plays an important role in many clinical applications [1]. Accordingly, a number of automatic or semi-automatic segmentation methods have been proposed. Ghanei *et.al.* [2] and Hu *et.al.* [3] designed 3D discrete deformable models to semi-automatically outline the prostate boundaries. Shao *et al* [4] proposed a level set method to detect the prostate in the 3D TRUS images. Gong *et al* [5] provided a Bayesian segmentation algorithm, based on deformable superellipses model, to segment 2D prostate contours.

We previously proposed a statistical shape model to segment the prostate from 3D TRUS images by matching both prostate shapes and tissue textures in TRUS images [6]. The effectiveness of our previous method is mainly resulted from the joint use of two novel techniques, (i) a Gabor filter bank used for 3D texture features extraction and (ii) support vector machines (SVM) used for texture-based tissue classification. However, both of these two techniques are computationally very expensive, thereby impeding the fast segmentation of prostates from 3D TRUS images. To overcome this limitation, this paper presents an efficient segmentation approach, which is implemented in a 3-level multi-resolution framework and is further speeded up by two techniques respectively designed for different resolutions. In both low and middle resolutions, a Zernike moment-based edge detector is used to replace the step of SVM-based tissue classification and boundary identification, for fast boundary detection. In the high resolution, a new SVM training method is designed to improve the

efficiency of SVMs by reducing the number of support vectors, which are initially required to construct a very complicated separation hypersurface for the classification of the highly confounding prostate and non-prostate tissues in TRUS images. By using these techniques, our approach for prostate segmentation is highly speeded up, yet without losing any segmentation accuracy.

2 Methods

Our previous deformable shape model [6] uses both statistical shape information and image texture information to segment the prostate boundary. Its success in prostate segmentation results from the texture analysis, which distinguishes prostate and non-prostate tissues from noisy TRUS images. However, the two techniques employed in texture analysis, i.e., a Gabor filter bank for texture characterization and SVMs for texture-based tissue classification, are both computationally very expensive. For example, it takes about 40 minutes to segment a prostate from a 256x256x176 TRUS image, using SGI workstation with a 500MHz processor. Therefore, it's necessary to speed up the segmentation approach.

For fast segmentation, we *firstly* formulate the segmentation approach in a 3-level multi-resolution framework, which has been widely used to increase the speed of the algorithms in the literature [7,8]. For example, the original TRUS image is decomposed into three multi-resolution images, i.e. the image of original size and the images down-sampled by factors 2 and 4. The surface model is initialized at the lowest resolution, and subsequently deforms to the prostate boundary. The segmentation result in the lower resolution is up-sampled to the next higher resolution, and used as initialization of the deformable model in the higher resolution. These steps are iterated until the deformable model converges to the prostate boundary in the highest resolution.

Besides the multi-resolution framework designed above, two effective methods, i.e., Zernike moment-based edge detector and a new training method for generating efficient SVMs, are particularly designed to speed up the segmentation approach in low resolutions and high resolution, respectively. The details of these two techniques are described next.

2.1 Zernike Moment Based Edge Detection

As discussed above, although a Gabor filter bank [9] is capable of extracting robust and rich texture features, it is computationally very expensive due to the use of Gabor filters at multiple scales and orientations. Additionally, as texture features are region-based features, prostate tissues in the down-sampled TRUS images usually have less distinguished texture features, compared to those in the original images. Therefore, boundary information, directly computed from the intensities, is better than texture information for guiding the deformable model in both low and middle resolutions.

Zernike moment-based edge detector has been proposed in [10]. It has three advantages in edge detection. First, as Zernike moments are integral-based operators, it is noise tolerant, which is especially important for detecting prostate boundaries in the noisy TRUS images. Second, as detailed next, this edge detection method provides a more complete description of the detected edges than the traditional edge detector, e.g. Canny edge detector. Third, as only three masks, i.e., two real masks and one complex mask, are required to get the edge features of each voxel, it is computationally more efficient than the Gabor filter bank which used 10 masks [6].

Zernike moment operator projects the image data onto a set of complex polynomials, which form a complete orthogonal set over the interior of a unit circle. For an image $f(x, y)$, its Zernike moment of order n and repetition m can be defined as:

$$Z_{nm} = \frac{n+1}{\pi} \iint_{x^2+y^2 \leq 1} f(x,y) V_{nm}^*(\rho, \theta) dx dy \quad (1)$$

where $V_{nm}(\rho, \theta) = R_{nm}(\rho)e^{jm\theta}$, $R_{nm}(\rho) = \sum_{s=0}^{(n-|m|)/2} [(-1)^s (n-s)! \rho^{n-2s}] / [s! (\frac{n+|m|}{2}-s)! (\frac{n-|m|}{2}-s)!]$, and (ρ, θ) are the polar coordinates of (x,y) .

Considering an ideal step edge (c.f. Fig 1), its important features include the step height k , the background gray level h , the perpendicular distance from the center of the circular kernel l , and the angle of edge with respect to the x -axis φ . All these features can be mathematically represented by three low order Zernike moments (Z_{00}, Z_{11}, Z_{20}) as:

$$\varphi = \tan^{-1}[\text{Im}(Z_{11})/\text{Re}(Z_{11})] \quad (2a)$$

$$l = Z_{20}/Z'_{11} \quad (2b)$$

$$k = 3Z'_{11}/2(1-l^2)^{3/2} \quad (2c)$$

$$h = \left(Z_{00} - k\pi/2 + k \sin^{-1}(l) + kl\sqrt{1-l^2} \right) / \pi \quad (2d)$$

where $Z'_{11} = Z_{11}e^{j\varphi}$, and $\text{Re}(\cdot)$ and $\text{Im}(\cdot)$ represent the real part and the imaginary part of a complex value, respectively. Similarly, Zernike moments can be used to measure the general edges in the 2D image by using eq. (2).

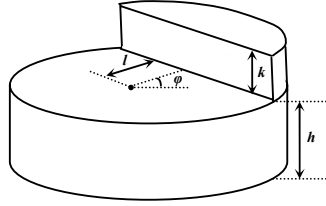


Fig. 1. A 2D ideal step edge model.

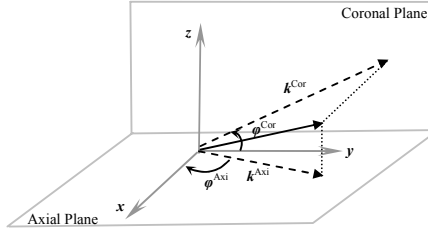


Fig. 2. Schematic explanation of using two 2D edge vectors to roughly reconstruct a 3D edge vector.

Notably, rather than extending the 2D Zernike moment to 3D, we simply apply two orthogonal 2D Zernike moment operators, which respectively lie on the axial plane and the coronal plane (c.f. Fig 2), to get two sets of edge features for each voxel \bar{v} , i.e., $\{h^{\text{Axi}}(\bar{v}), k^{\text{Axi}}(\bar{v}), l^{\text{Axi}}(\bar{v}), \varphi^{\text{Axi}}(\bar{v})\}$ and $\{h^{\text{Cor}}(\bar{v}), k^{\text{Cor}}(\bar{v}), l^{\text{Cor}}(\bar{v}), \varphi^{\text{Cor}}(\bar{v})\}$. As shown in Fig 2, two 2D edge vectors can be roughly considered as two projections (black dashed arrows) of a 3D edge vector (black solid arrow) in the axial and the coronal planes, respectively. (Edge vector is a vector whose magnitude and direction represent the edge strength and the normal direction of the edge, respectively.) Thus, the 3D edge vector, i.e., $\bar{e}(\bar{v})$, can be represented by two 2D edge vectors as follows:

$$\bar{e}(\bar{v}) = k^{\text{Axi}}(\bar{v}) \cdot [\cos(\varphi^{\text{Axi}}(\bar{v})), \sin(\varphi^{\text{Axi}}(\bar{v})), \sin(\varphi^{\text{Axi}}(\bar{v})) \tan(\varphi^{\text{Cor}}(\bar{v}))]^T \quad (3)$$

In our previous segmentation approach [6], an energy function is defined on each vertex \bar{p}_i of the deformable surface model, and it is used to evaluate the matching degree of the deformable model with the prostate boundaries in the TRUS images. The energy function consists of two energy terms, i.e., the external

boundaries in the TRUS images. The energy function consists of two energy terms, i.e., the external energy, which drives the deformable model to the prostate boundary, and the internal energy, which preserves the geometric regulation of the model during deformation. By jointly minimizing these two terms, the deformable model is able to converge to the prostate boundaries. In this study, the external energy is re-formulated such that the edge features captured by Zernike moments are employed to guide the deformable segmentation, while the internal energy remains the same. Accordingly, for each vertex \bar{P}_i , its external energy is defined as:

$$E^{\text{Ext}}(\bar{P}_i) = w_{\text{Str}} E_{\text{Str}}(\bar{P}_i) + w_{\text{Dist}} E_{\text{Dist}}(\bar{P}_i) + w_{\text{Int}} E_{\text{Int}}(\bar{P}_i) \quad (4)$$

where $E_{\text{Str}}(\bar{P}_i) = -\left(\frac{\sum_{\bar{v} \in N(\bar{P}_i)} \langle \bar{n}(\bar{P}_i), \bar{e}(\bar{v}) \rangle}{\sum_{\bar{v} \in N(\bar{P}_i)} 1} \right)$, $E_{\text{Dist}}(\bar{P}_i) = \min(I^{\text{Axi}}(\bar{P}_i), I^{\text{Cor}}(\bar{P}_i))$ and

$$E_{\text{Int}}(\bar{P}_i) = \left| \frac{\sum_{\bar{v} \in N(\bar{P}_i)} (h^{\text{Axi}}(\bar{v}) + h^{\text{Cor}}(\bar{v}))}{\sum_{\bar{v} \in N(\bar{P}_i)} 2 - H(\bar{P}_i)} \right|.$$

There are three items in Eq. (4). The first item denotes the integrated edge strength in the spherical neighborhood of \bar{P}_i , i.e., $N(\bar{P}_i)$. Notably, the edge strength is projected to the normal direction of the deformable surface at \bar{P}_i , $\bar{n}(\bar{P}_i)$, by the inner product $\langle \cdot, \cdot \rangle$. The second item denotes the distance from \bar{P}_i to the boundary. The third item requires that the deformable model converges only to the boundary with the intensity similar to the learned average intensity, $H(\bar{P}_i)$, which is captured for vertex \bar{P}_i from a set of training samples. By jointly minimizing these three items, the deformable model is thus driven to the prostate boundaries in both low and middle resolutions of TRUS images.

2.2 A Training Method for Increasing the Efficiency of SVM

Zernike moment-based edge detector is able to detect prostate boundaries in the low and the middle resolutions, however, it is not effective in accurately delineating prostate boundaries in the high resolution as the prostate boundaries are usually blurred by speckle noise in the original TRUS images. Accordingly, we still use the statistical texture matching method [6], which consists of texture characterization by a Gabor filter bank and texture-based tissue classification by SVMs, for prostate segmentation in the high resolution stage of our multi-resolution framework.

In our method, a set of SVMs are employed for texture-based tissue classification [6]. Each of them is attached to a sub-surface of the model surface and trained by the manually-labeled prostate and non-prostate samples around that sub-surface. In the testing stage, the input of the SVM is a feature vector, which consists of Gabor features extracted from the neighborhood of a voxel, and the output denotes the likelihood of the voxel belonging to the prostate. In this way, the prostate tissues are differentiated from the surrounding ones. However, since the Gabor features of TRUS prostate images vary greatly across the individuals and their distribution is highly overlapped between prostate and non-prostate regions, the trained SVM usually has a huge number of support vectors. This is because (i) a large number of the support vectors, locating at the margins, are required to construct a highly convoluted hypersurface, in order to separate two classes; (ii) even the highly convoluted separation hypersurface has been constructed, quite a lot of confounding samples are still misclassified and thus selected as other support vectors, locating beyond the margins. Notably, this huge number of support vectors will dramatically increase the computational cost of the SVM. Therefore, it is necessary to design a training method to decrease the number of support vectors of the finally trained SVM, by simplifying the shape of the separation hypersurface.

The basic idea of this training method is to selectively exclude some training samples, thereby the remaining samples are possible to be separated by a less convoluted hypersurface. Since the support vectors determine the shape of the separation hypersurface, they are the best candidates to be excluded from the training set, in order to simplify the shape of the separation hypersurface.

However, excluding different sets of support vectors from the training set will lead to different simplifications of the separation hypersurface. Fig 3 presents a schematic example in the 2-dimensional feature space, where we assume support vectors exactly locating on the margins. As shown in Fig 3(a), SVM trained by all the samples has 10 support vectors, and the separation hypersurface is convoluted. Respective exclusion of two support vectors, SV_1 and SV_2 , denoted as red and blue crosses in Fig 3(a), will lead to different separation hypersurfaces as shown in Figs 3(b) and 3(c), respectively. SVM in Fig 3(b) has only 7 support vectors, and its hypersurface is less convoluted, after re-training SVM with all samples except SV_1 . Importantly, two additional samples, denoted as green circle/cross, were previously selected as support vectors in Fig 3(a), but they are no longer selected as support vectors in Fig 3(b). In contrast, SVM in Fig 3(c) still has 9 support vectors, and the hypersurface is very similar to that in Fig 3(a), even SV_2 has been excluded from the training set.

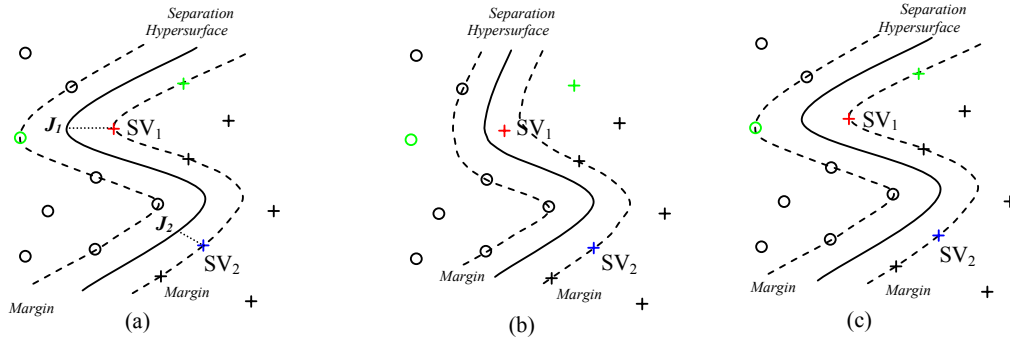


Fig.3. Schematic explanation of how to selectively exclude the support vectors from the training set, in order to effectively simplify the separation hypersurface. The solid and dashed curves denote the separation hypersurfaces and their margins, respectively. The circles and the crosses denote the positive and the negative training samples, which are identical in (a), (b) and (c). The training samples locating on the margins are the support vectors.

The reason of SVM in Fig 3(b) being more efficient than that in Fig 3(c) is that the excluded support vectors SV_1 contributes more to the convolution of the hypersurface. For each support vector, its contribution to the convolution of hypersurface can be approximately defined as the generalized curvature of its projection point on the hypersurface. For example, for SV_1 and SV_2 in Fig 3(a), their projection points on the hypersurface are J_1 and J_2 . The curvature of the hypersurface at point J_1 is much larger than that at point J_2 , which means the support vector SV_1 has more contribution to make the hypersurface convoluted. Therefore, it is more effective to “flatten” the separation hypersurface by excluding the support vectors, like SV_1 , with their projection points having the larger curvatures on the hypersurface.

Accordingly, the new training method is designed to have the following four steps.

Step 1. Use all the training samples to train an initial SVM, resulting in l_1 initial support vectors $\{SV_i^{in}, i=1,2,\dots,l_1\}$ and the corresponding decision function d_1 .

Step 2. Exclude the support vectors, whose projections on the hypersurface have the largest curvatures, from the training set:

- 2a. For each support vector SV_i^{in} , find its projection on the hypersurface, $p(SV_i^{in})$, along the gradient of distance function d_1 .
- 2b. For each support vector SV_i^{in} , calculate the generalized curvature of $p(SV_i^{in})$ on the hypersurface, $c(SV_i^{in})$.
- 2c. Sort SV_i^{in} in the decrease order of $c(SV_i^{in})$, and exclude the top n percentage of support vectors from the training set.

Step 3. Use the remaining samples to re-train the SVM, resulting in l_2 support vectors $\{SV_i^{\text{Re}}, i=1,2,\dots,l_2\}$ and the corresponding decision function d_2 . Notably, l_2 is usually less than l_1 .

Step 4. Use the l_2 pairs of data points $\{SV_i^{\text{Re}}, d_2(SV_i^{\text{Re}})\}$ to finally train the SVRM (Support Vector Regression Machine) [12], resulting in l_3 final support vectors $\{SV_i^{\text{Fl}}, i=1,2,\dots,l_3\}$ and the corresponding decision function d_3 . Notably, l_3 is usually less than l_2 .

Using this four-step training algorithm, the efficiency of the trained SVMs will be highly enhanced with very limited loss of classification rate, which will be shown in the first experiment. Notably, as in the statistical texture matching method, the matching degree of the deformable model with the prostate boundaries is defined in a noise tolerant fashion [6], a little loss of classification, i.e., a little number of mis-classified voxels, will not influence the segmentation accuracy, while the segmentation speed is greatly increased.

3 Experimental Results

The first experiment is presented to test the performance of the proposed training method in increasing the efficiency of SVMs. We firstly select prostate and non-prostate samples from six manually labeled TRUS images. 3621 samples from one image are used as testing samples, while 18105 samples from other five images are used as training samples. Each sample has 10 texture features, extracted by a Gabor filter bank [9]. We use our method to train a series of SVMs by excluding different percentages of support vectors in Step 2c of our training method. The performances of these SVMs are measured by the number of support vectors finally used and the number of correct classifications among 3621 testing samples. As shown in Fig 4(a), after excluding 50% of initially selected support vectors, the finally-trained SVM has 1330 support vectors, which is only 48% of the support vectors (2748) initially selected in the original SVM; but its classification rate still reaches 95.39%. Compared to 96.02% classification rate achieved by original SVM with 2748 support vectors, the loss of classification rate is relatively trivial. If we want to further reduce the computational cost, we can exclude 90% of initially selected support vectors from the training set. Our finally-trained SVM has only 825 support vectors, which means the speed is triple, and it still has 93.62% classification rate. To further validate the effect of our trained SVM in prostate segmentation, the SVM with 825 support vectors (denoted by the red triangle in Fig 4(a)) is applied to a real TRUS image for tissue classification. As shown in Figs 4(b1) and 4(b2), the result of our trained SVM is not inferior to that of the original SVM with 2748 support vectors (denoted by the red square in Fig 4(a)), in terms of differentiating prostate tissues from the surrounding ones.

In the second experiment, the proposed segmentation approach is applied to segment prostates from six real 3D TRUS images. A leave-one-out validation method is used, i.e., each time five images are used for training, and the remaining one is used for testing. The size of 3D images is 256x256x176, with the spatial resolution 0.306mm. Fig 5(a) shows the multi-resolution deformation procedure on one of the TRUS images. The red contours, labeled as “LF”, “MF” and “HF”, denote the finally deformed models in the low, middle and high images, respectively. Notably, the models in both low and middle resolutions are guided by the Zernike moment-based edge detector, while the model in the high resolution is guided by the statistical texture matching method. The algorithm-based segmentation result is compared to the hand-labeled result in Fig 5(b). Moreover, Table 1 gives a quantitative evaluation of this comparison to all the six TRUS images. From both visual results and quantitative analysis, we can conclude that our automated segmentation method is able to segment the prostate from noisy TRUS images. Importantly, using a SGI workstation with 500MHz processor, the average running time for segmenting a prostate is 4 minutes, which is 10 times faster than our previous method [6].

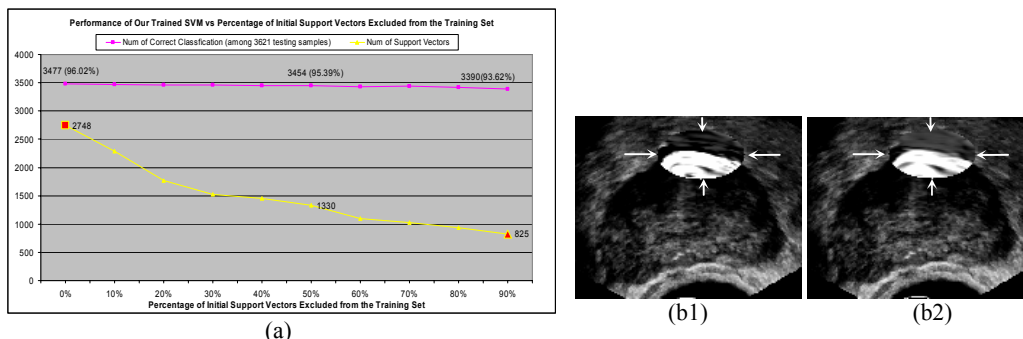


Fig. 4. (a) The performance of the finally-trained SVM changes with the percentages of initial support vectors excluded from the training set. (b) Comparisons of tissue classification results using (b1) the original SVM with 2748 support vectors and (b2) our trained SVM with 825 support vectors. The tissue classification results are shown only in an ellipsoidal region and mapped to 0~255 for the display purpose.

4 Conclusion

We have proposed an efficient segmentation approach for fast segmentation of prostates from 3D TRUS images. Our segmentation approach was formulated as a multi-resolution framework, and it was speeded up by two techniques, respectively designed for different resolutions. In both low and middle resolutions, Zernike moment-based edge detector is used to replace the step of SVM-based tissue classification and boundary identification, for fast capturing boundary information for deformable segmentation. In the high resolution, a new training method has been designed to increase the efficiency of the finally trained SVM for texture-based tissue classification, thereby equally increasing the efficiency of texture matching step in deformable segmentation procedure. Compared to our previous segmentation method [6], the proposed one is 10 times faster in segmenting 3D prostate from TRUS images, yet without losing any segmentation accuracy.

Table 1. Comparison of the algorithm-based segmentation and the hand-labeled segmentation on six real TRUS images.

Subjects	Average Distance (Voxels)	Overlap Volume Error (%)	Volume Error (%)
Image1	1.01	3.90	2.06
Image2	0.96	4.04	2.15
Image3	0.95	3.32	1.12
Image4	1.22	4.63	4.47
Image5	1.34	4.87	1.30
Image6	1.13	4.03	2.25
Mean	1.10	4.13	2.23
Stand. Deviation	0.16	0.55	1.19

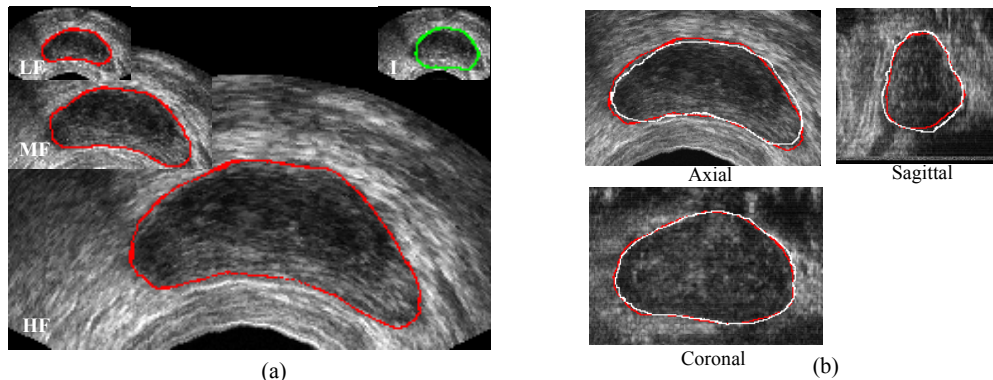


Fig. 5. (a) A typical multi-resolution deformation procedure. The color contour denotes the model on a selected slice of the TRUS image. The green one is the initialized model in the low resolution. The red ones denote the finally deformed models in the low, middle and high resolution images. (b) Visual comparisons between algorithm-based and hand-labeled segmentation results. The white contours are the hand-labeled results, while the red ones are the algorithm-based segmentation results.

References

1. "Overview: Prostate Cancer", <http://www.cancer.org>, 2004.
2. A. Ghanei, H. Soltanian-Zadeh, A. Ratkesicz and F. Yin, "A three-dimensional deformable model for segmentation of human prostate from ultrasound image", *Med. Phy.*, Vol. 28, pp. 2147-2153, 2001.
3. N. Hu, D. Downey, A. Fenster, and H. Ladak, "Prostate surface segmentation from 3D ultrasound images", *ISBI*, pp. 613-616, Washington, D.C., 2002.
4. F. Shao, K.V. Ling and W.S. Ng, "3D Prostate Surface Detection from Ultrasound Images Based on Level Set Method", *MICCAI 2002*, pp. 389-396, 2002.
5. L. Gong, S.D. Pathak, D.R. Haynor, P.S. Cho and Y. Kim, "Parametric Shape Modeling Using Deformable Superellipses for Prostate Segmentation", *TMI*, pp. 340-349, Vol. 23, 2004.
6. Y. Zhan and D. Shen, "Automated Segmentation of 3D US Prostate Images Using Statistical Texture-Based Matching Method", *MICCAI, 2003*, Nov 16-18, Canada.
7. P. Chalermwat and T. El-Ghazaw, "Multi-resolution Image Registration Using Genetics", *ICIP*, Japan, Oct. 1999.
8. D. Shen and C. Davatzikos, "HAMMER: Hierarchical Attribute Matching Mechanism for Elastic Registration", *IEEE Trans. on Medical Imaging*, 21(11):1421-1439, Nov 2002.
9. B.S. Manjunath and W.Y. Ma, "Texture Features for Browsing and Retrieval of Image Data", *IEEE Trans. on Pattern Anal. Mach. Intell.*, Vol. 18, pp. 837-842, 1996.
10. S. Ghosal and R. Mehrotra, "Orthogonal Moment Operators for Subpixel Edge Detection", *Pattern Recognition*, Vol 26, pp. 295-305, 1993.
11. C.J.C. Burges, "A Tutorial on Support Vector Machines for Pattern Recognition", *Data Mining and Knowledge Discovery*, Vol. 2, pp. 121-167, 1998.
12. E. Osuna, F. Girosi, "Reducing the run-time complexity of Support Vector Machines", *ICPR*, Brisbane, Australia, 1998.