3D prostate surface detection from ultrasound images

Shao, Fan

2006

Shao, F.(2006). 3D prostate surface detection from ultrasound images. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/3221>

<https://doi.org/10.32657/10356/3221>

Nanyang Technological University

Downloaded on 26 Apr 2023 20:48:01 SGT

3D Prostate Surface Detection from Ultrasound Images

Shao Fan

School of Electrical and Electronic Engineering

A thesis submitted to the Nanyang Technological University
in fulfillment of the requirement for the degree of
Doctor of Philosophy

2006

Acknowledgements

First and foremost, I would like to express my deepest gratitude and sincere appreciation to my supervisor, Assoc. Prof. Ling Keck Voon, who gave continuous advice and guidance that enabled me to complete this thesis successfully. I am especially grateful to Assoc. Prof. Ng Wan Sing, the founder and head of CIMIL (Computer Integrated Medical Intervention Laboratory) for his knowledgeable comments and suggestions, which helped me greatly in the process of research.

I am deeply thankful to Dr. H. Wijkstra, Dr. H. M. Ladak, Dr. M. Alcaniz, Dr. C. K. Kwoh and Prof. Y. Kim for their permissions to reprint the figures and tables in this thesis.

Financial support for this work was provided by the Research Scholarship from Nanyang Technological University. Thanks to Dr. Tan Hung Hoe at the Gleneagles Hospital and Singapore General Hospital (SGH) for providing test images.

I am very grateful to Dr. Tan Puay Hoon (pathologist) and Dr. Thng Choon Hua (radiologist), who are from Singapore General Hospital, for their instruction of how to outline the prostate boundaries in ultrasound and MRI images. Thanks to Dr. John Yuen and Dr. Christopher Cheng (surgeons) from SGH for their knowledgeable suggestions and the valuable discussion. Special thanks go to the well-trained members, Yu Wen, Lin Jing, Ma Hongyun and Shao Wei, for their outlining the prostate boundaries for references. And I thank all the past and present members of CIMIL for their help in many ways.

Finally, I dedicate this thesis to my wife Liu Chunyan and my daughter Shao

Yuxuan for their self-giving support and encouragement. Their love is the constant source of comfort and happiness.

Summary

Benign prostatic hyperplasia (BPH) and prostate cancer are two common diseases in adult and elderly men. With the increasing number of men seeking medical care, prostate boundary detection from ultrasound images plays a key role in prostate disease diagnoses and treatments. Currently, prostate boundary detection are performed manually, which is arduous and heavily user dependent. A possible solution to improve the efficiency is to automate the boundary detection process with minimal manual involvement. While numerous methods have been proposed to (partially or fully) automatically detect prostate boundary from ultrasound images, there is no one method which has gained wide acceptance in practice. The poor contrast between the prostate and surrounding tissues, speckle noise, shadowing and refraction artifacts in ultrasound images still make it a challenging task to automatically detect prostate boundaries from ultrasound images. Manual contouring is currently the only robust, reliable segmentation procedure available for ultrasound images of the prostate.

As a part of research on computer-assisted surgery, this thesis describes the research and development of a 3-dimensional prostate surface detection method for transrectal ultrasound (TRUS) images, aiming to achieve high validity (or accuracy), reliability (or consistency) and good speed (fast computation) of the algorithm. The research is mainly concerned with the use of the level set method to detect 3-dimensional object surfaces. We adopted the level set method as the main surface detection mechanism to develop our approach. This is because the level set method is able to handle complex geometry and topological changes, with

no significant differences in tracking fronts in 2D and 3D. In addition, it is computationally efficient in 3D with the use of Narrow-Band techniques, and its property of relatively insensitive to the initial hypersurface selection can potentially relieve user involvement. The key task of level set application is to design an appropriate speed function F which can drive the evolving level-set to the desired object surface. Conventional level set methods driven by image gradient usually result in a so-called “boundary leaking” problem due to the poor quality of ultrasound images. Although gradient flow, which had been proposed elsewhere to alleviate this problem, can be added to improve the algorithm performance, it is not satisfactory for our intended application. In this work, image gradient is replaced by the region information in the proposed method to remedy the “boundary leaking” problem. Meanwhile, knowledge-based features, such as the expected shape (kidney-like) and the ultrasound appearance of the prostate (looking from within the gland, the intensities are transitions from dark to light), are also included to improve the algorithm performance. The kidney-like shape is used to optimize the initialization procedure where users can define the initial multiple level-sets according to the image property. A Gaussian mixture model (GMM) is employed to describe the intensity distribution of the prostate, and to modify the speed function F . Thus a new practical method, “Evolving Bubbles”, is achieved. To speed up the algorithm, a 5-layer narrow band is built and applied.

The proposed algorithm was evaluated for its validity (accuracy) and reliability (consistency) by comparing the detected results with the outlines of experts. The validity is evaluated by measuring the volume difference and the surface distance while the reliability is measured by the use of intra-class correlation coefficients

(ICC). The experimental results showed that the algorithm can obtain reproducible results in 4 and half minutes on average and give an accuracy of near 87% when applied to 3D TRUS images, and an accuracy of near 77% when applied to 2D image stacks. The decrease in accuracy of the algorithm when applied to 2D image stacks gives a justification for using 3D transrectal ultrasound in diagnoses and treatments of prostate diseases.

Table of Contents

Acknowledgements	i
Summary	iii
Table of Contents	viii
List of Figures	xviii
List of Tables	xx
1 Introduction	1
1.1 Motivation	1
1.2 Challenges	7
1.3 Objectives	9
1.4 Contributions	10
1.5 Organization of the Thesis	13
2 Related Works	16
2.1 2D prostate boundary detection	16
2.1.1 Edge-based detection methods	16
2.1.2 Texture-based detection methods	24
2.1.3 Model-based detection methods	25
2.2 3D prostate surface detection	31
2.3 Discussion	35
3 3D Prostate Surface Detection Using The Level Set Method	37
3.1 Level set method	37
3.1.1 Preliminary knowledge	38
3.1.2 Mathematical background	41
3.1.3 Numerical algorithm	48
3.2 3D prostate surface detection with level set method	50
3.2.1 Initialization	50
3.2.2 Narrow band solutions	51
3.2.3 Big challenge: the “Boundary Leaking” problem	57
3.2.4 Region-based level set method	59
3.2.5 Stopping criterion	62
3.2.6 Algorithm implementation	63
3.3 Discussion	67
3.4 Summary	68

4	Evolving Bubbles for 3D Prostate Surface Detection	71
4.1	Designing the speed function F with respect to the statistical distribution model	71
4.1.1	Gaussian mixture model (GMM) for intensity distribution of prostate	74
4.1.2	Parameter estimation of GMM with EM algorithm	76
4.1.3	Redesign the speed function F with respect to GMM	80
4.2	Adding gradient flow	88
4.3	“Evolving Bubbles” for 3D prostate surface detection	91
4.4	A faster algorithm	99
4.5	Summary	104
5	Experimental Results and Evaluation	107
5.1	Validity evaluation	107
5.1.1	Previous works	108
5.1.2	Validation metrics	113
5.1.3	Implementation	117
5.2	Reliability evaluation	125
5.3	Experimental results	127
5.3.1	Prostate surface detection from 3D TRUS	127
5.3.2	Prostate surface detection from a stack of 2D TRUS images	141
5.4	Analysis of the sensitivities of the method to the parameter choices	152
5.5	Discussion	163
5.6	Summary	164
6	Conclusion and Recommendations	165
6.1	Conclusion	165
6.2	Recommendations for further research	167
	Author’s Publications	172
	Bibliography	173
	Glossary and Acronym List	182
A	Bubble 3D: A Java application for handling surfaces in volumetric medical images	185
A.1	Smart curve outlining	185
A.2	Surface construction	186
A.3	Surface — Image registration	186
A.4	Needling simulation	187
A.5	Portable project data	188

B	The STL Format	190
B.1	Format specifications	190
B.1.1	Facet orientation	190
B.1.2	Vertex-to-vertex rule	191
B.2	STL file format	192
B.2.1	STL ASCII format	192
B.2.2	STL binary format	193

List of Figures

1.1	Prostate anatomy and BPH. (a) The prostate and nearby organs, (b) benign prostatic hyperplasia.	1
1.2	Transrectal ultrasound of the prostate. (a) Transrectal ultrasound imaging of the prostate, (b) a typical prostate scan of TRUS. . . .	5
1.3	Illustration of “boundary leaking” problem of image gradient based deformable models for prostate boundary detection. (a) When the active contour model is applied (<i>Reprinted with permission from M. Alcaniz (Universidad Politecnica de Valencia, Valencia, Spain).</i>), (b) when the level set method is applied.	8
1.4	3D view of the prostate reconstructed from the sequence of manually outlined cross-sectional 2D boundaries. (a) Parallel 2D boundaries drawn by experts, (b) 3D reconstruction. <i>Reprinted with permission from R. Y. Wu (Nanyang Technological University, Singapore).</i>	10
1.5	Summary of contributions and algorithm development.	13
2.1	Example of the contour detection process using minimum/maximum filtering. (a) Original ultrasound image, (b) blurred and compressed image, (c) local minimum of the image, (d) local maximum, (e) gradient in the direction from the minimum, (f) gradient in the direction of the maximum, (g) second-derivative in gradient direction, (h) zero crossings in the second-derivative, (i) edge strength, (j) edge strength masked by the zero crossings, (k) selected contour, (l) ultrasound image with the projected contour. <i>Reprinted with permission from H. Wijkstra (University Medical Center Nijmegen, Nijmegen, the Netherlands).</i>	19
2.2	Edge-guided boundary detection. (a) Detected edges before <i>a priori</i> knowledge-based filtering, (b) edge image after removal of false edges using knowledge-based filtering, (c) edges overlaid on top of the prostate image, (d) final boundary delineation on the prostate image. <i>Reprinted with permission from Y. Kim.</i>	21

2.3	Result of RBR and its binarized image. (a) Original image, (b) inverted image, (c) RBR-processed image, (d) binarized image. <i>Reprinted with permission from W. S. Ng (Nanyang Technological University, Singapore).</i>	22
2.4	Prostate boundary detected by RBR. (a) Inverted binary image, (b) labeled boundary areas, (c) skeleton of boundary areas, (d) boundary complementation. <i>Reprinted with permission from W. S. Ng.</i>	23
2.5	Outline of prostate boundary using harmonics fitting. (a) After second iteration, (b) after third iteration, (c) after fourth iteration, (d) final image. <i>Reprinted with permission from C. K. Kwok (Nanyang Technological University, Singapore).</i>	24
2.6	Prostate boundary detection by DDC. (a) Initial DDC, (b) after initial deformation, (c) three points (indicated by squares) were edited and clamped, (d) after second time deformation. <i>Reprinted with permission from H. M. Ladak (University of Western Ontario, London, Ontario, Canada).</i>	27
2.7	Comparison of detected results between conventional unrestricted snakes (top row) and the proposed shape-restricted snakes method (bottom row). <i>Reprinted with permission from M. Alcaniz (Universidad Politecnica de Valencia, Valencia, Spain).</i>	28
2.8	Labeling scheme for feature modeling. <i>Reprinted with permission from R. Y. Wu.</i>	30
2.9	Prostate boundary detection by feature modeling (top row: boundary recognition from a comparatively clear TRUS image; bottom row: boundary recognition from a blurring image). (a) Original image, (b) outlining of expert observer 1, (c) outlining of expert observer 2, (d) outlining of computer. <i>Reprinted with permission from R. Y. Wu.</i>	31
2.10	Simple example of the 3D deformable model structure. $v(i, j)$ stands for the j -th neighboring vertex of V_i ; $\mathbf{r}_i$, unit radial (normal) vector at vertex V_i ; and $\mathbf{d}_{i,0}$, unit vector along the edge between V_i and $v(i, 0)$	32

2.11	3D prostate surface detected by 3DView . (a) Selection of two contour points on the rotation axis in reference image B, (b) result of the surface detection.	33
2.12	3D prostate surface detected by 3DView from a different reference image. (a) Selection of two contour points on the rotation axis in reference image A, (b) final result of the surface detection.	34
3.1	An interface separating the image apart.	38
3.2	Interface moving with the speed $\vec{F}$ which has a magnitude of F . . .	39
3.3	Level-set function and zero level-set. (a) The original front, which lies in xy plane, (b) The level-set function, where the front is intersection of surface and xy plane.	40
3.4	Front expanding under the speed function $F = 1 - 0.5K$. (a) The initial level set, (b) front expanding after 1000 iterations, (c) front expanding after 2000 iterations.	44
3.5	Illustration of the image gradient magnitude. (a) Raw image, (b) image of the image gradient magnitude.	45
3.6	Illustration of the object boundary detection of the level set method constrained by the image gradient magnitude. (a) Raw image, (b) image of the image gradient magnitude, (c) magnitude value of the image gradient, (d) the speed function constraint, (e) initial circle for the level set method, (f) detection result.	46
3.7	Initialization of the level set method by automatically putting a small sphere ($r = 20$ voxels) at the image center.	51
3.8	An extra dimension added in solving the front propagation problem.	52
3.9	Narrow band illustration. (a) A 2D narrow band with the width of $2d$, (b) A 3D narrow band with the width of $2d$	53
3.10	2D projection of a five layer narrow band construction.	55

3.11	Ultrasound image of the prostate and its gradient magnitude image. (a) Ultrasound image of the prostate, (b) its gradient magnitude image.	57
3.12	Ultrasound image of the prostate and its gradient magnitude image. (a) Ultrasound image of the prostate, (b) its gradient magnitude image.	58
3.13	An example of “boundary leaking” problem of the level set method. <i>Ragged white curve</i> : the detected boundary of the level set method, <i>smooth green curve</i> : the manually outlined boundary, <i>yellow arrow</i> : the boundary leaking caused by shadowing, <i>red arrow</i> : the bound- ary leaking caused by low contrast.	58
3.14	Demonstration of a $3 \times 3 \times 3 (n = 3)$ sliding cubic window, where the grey cube stands for the intensity of voxel at (x, y, z)	60
3.15	Preset Region for calculation of R_T	60
3.16	Example of the binary image of R . (a) Ultrasound image of the prostate, (b) the binary image of R	61
3.17	Diagram of the algorithm implementation.	63
3.18	3D prostate surface and its cross-sectional contours detected by our proposed algorithm. (a) Transverse view, (b) sagittal view, (c) coronal view, (d) initial surface, (e) final surface, (f) final surface with cutting planes; (1) slice of the original image, (2) slice with the initial contour overlaid, (3) slice with the final contour overlaid.	64
3.19	The comparison between detected contours and manual contours in 2D cross-sectional slices. (a) Cross-sectional slices, (b) detected contours, (c) manual contours, (d) contour comparison.	65
3.20	Detected Results of algorithm test on another typical 3D prostate scan. (a) Transverse view, (b) sagittal view, (c) coronal view, (d) initial surface, (e) final surface, (f) final surface with cutting planes.	66
3.21	Results obtained by gradient based level set method. (a) Same slice as in Fig. 3.20(a), (b) same slice as in Fig. 3.20(b), (c) same slice as in Fig. 3.20(c).	67

4.1	Histogram of the whole image. (a) Histogram of image A ($256 \times 256 \times 256$), (b) histogram of image B ($250 \times 174 \times 236$).	73
4.2	Diagram of the steps of parameter estimation of GMM using EM algorithm.	81
4.3	2D images collected from different frequency with different zoom rate at the same slice, where the red line box represents the ROI for the calculation of the intensity distribution within the prostate. (a) Image collected with the frequency of 32Hz and the zoom rate of R08, (b) image collected with the frequency of 27Hz and the zoom rate of R10 at the same slice.	82
4.4	Corresponding histogram of ROI. (a) The histogram calculated from ROI shown in Fig. 4.3(a), (b) the histogram calculated from ROI shown in Fig. 4.3(b).	82
4.5	Direct comparison of the intensity distributions shown in Fig. 4.4. .	83
4.6	Gaussian mixture model for the intensity distribution of prostate. (a) Data sampling in ROI, (b) Gaussian mixture model (amplified to match the histogram of sample data.	84
4.7	Three more GMMs with typical histogram shapes over image V1 (a), image V2 (b), and image V4 (c)	86
4.8	Combination effect of $R + P$. (a) Ultrasound image of the prostate, (b) binary image of R , (c) binary image of P , (d) binary image of $0.5R + 0.5P$	87
4.9	Boundary leaking (marked by arrows) in the low contrast areas when the smoothing strength increases before adding the gradient flow.	88
4.10	The speed function constraint k_I ($p = 2$) and its gradient . (a) Raw image, (b) the speed function constraint k_I ($p = 2$), (c) magnitude image of ∇k_I , (c) vector flow of ∇k_I	89
4.11	Vector flow of $\nabla_D(R + P)$. (a) Image of (R+P), (b) vector flow of $\nabla_D(R + P)$	91

4.12	A single level set and its final detected result.	92
4.13	Multiple level sets and their final detected result.	93
4.14	Limitation of the fully automatic method — failure to detect the whole desired prostate surface. (a) Original image ($252 \times 194 \times 256$), (b) detected surface, (c) 3 sets of contours in 2D cross-sectional images.	94
4.15	5 evenly spaced slices in transverse view used for level-sets generation.	95
4.16	User defined initial level sets in transverse view for the image shown in Fig. 4.14. (a) 2 level sets at slice 42, (b) 3 level sets at slice 84, (c) 3 level sets at slice 126, (d) 1 level set at slice 168.	96
4.17	Another example of user specification of initial level sets. (a) 3 level sets at slice 63, (b) 3 level sets at slice 84, (c) 3 level sets at slice 126, (d) 2 level sets at slice 168.	96
4.18	Three views of a level set in level set generation. (a) Transverse view, (b) sagittal view related to the vertical red dotted line, (c) coronal view related to the horizontal red dotted line.	97
4.19	Merging of three bubbles when evolving with constant speed along normal direction.	97
4.20	Greatly improved result obtained by the user-defined bubbles shown in Fig. 4.16. (a) Original image ($252 \times 194 \times 256$), (b) detected surface, (c) 3 sets of contours in 2D cross-sectional images.	98
4.21	An example of the experimental results obtained by our algorithm with the narrow band solution. Image size: $256 \times 256 \times 256$, iteration number: 2538, elapsed time: 68 min.	99
4.22	Illustration of image data sampling. (a) Voxels concerned with the computation in the original image, (b) image data sampling by every other slice along x, y, z direction respectively.	100
4.23	Experimental results before and after image down sampling. (a) The detected result before image down sampling, iteration number: 2538, elapsed time: 68'04'', (b) the detected result after image down sampling, iteration number: 927, elapsed time: 4'34''.	101

4.24	Comparison of the experimental results before and after image down sampling. <i>Yellow line</i> : detected result before image down sampling, <i>red line</i> : detected result after image down sampling. . . .	101
4.25	Two other examples of detected results. (a) Image size: $252 \times 194 \times 256$, iteration number: 1243, elapsed time: 5'01", (b) image size: $250 \times 174 \times 236$, iteration number: 1001, elapsed time: 3'51". . . .	102
4.26	Accuracy comparison between the narrow band solution and the faster algorithm.	104
5.1	Fractional area A_{XOR} — The grey area between two contours. . . .	110
5.2	Assessment scheme — D_1, D_2, D_3 represent the comparison results (i.e., disagreement between the two corresponding contours being compared).	110
5.3	Metrics for comparing semi-automatically outlined boundaries (dotted) against manually outlined boundaries (solid). (a) Local difference, $d_j(\theta_i)$, (b) area difference.	111
5.4	One-sided error. In this case, $d(S, S')$ will remain much smaller than $d(S', S)$, since here $d(A, S') \ll d(B, S)$. Thus a small one-sided distance does not imply a small distortion.	116
5.5	Boundary outlining using Bubble 3D : the red curve in the right figure is the outline, whose cross points on the 2 orthogonal planes along the lines are shown in the left figures.	118
5.6	An example of 3D prostate surface reconstruction using Bubble 3D . (a) Parallel manually outlined NURBS curves, (b) reconstructed 3D surface.	118
5.7	Calculation of the mean NURBS curve. (a) One slice of image V1, (b) 3 superposed curves obtained from the reference surfaces, (c) linearly spaced points on the curves, (d) the center of gravity (CG) determined from the yellow line, (e) averaging the radii around the specified angles to get the control points, (d) the mean NURBS curve (red solid line) obtained from the control points (red circle). .	120

5.8	Area generation by scanning the contour line by line. (a) Scan the contour to get the pairs which record the coordinates of the pixel on the contour, (b) set the intensity value of all pixels between the pair to 255 to generate the area.	122
5.9	Relationship between a given point, V , and a given triangle. (a) Point outside triangle, (b) point inside triangle.	123
5.10	A cube defined by the coordinates of the triangle vertices.	124
5.11	Detected result of image $V0$. (a) Raw image and 3 orthogonal 2D slices, (b) detected surface and corresponding contours on the 3 slices.	128
5.12	Volume difference of image $V0$. (a) Reference surface and detected surface, (b) volume difference in middle slices, (c) volume difference in one end, (d) volume difference in the other end.	129
5.13	Surface distance of image $V0$, from detected surface to reference. .	130
5.14	Detected result of image $V1$. (a) Raw image and 3 orthogonal 2D slices, (b) detected surface and corresponding contours on the 3 slices.	131
5.15	Volume difference of image $V1$. (a) Reference surface and detected surface, (b) volume difference in middle slices, (c) volume difference in one end, (d) volume difference in the other end.	132
5.16	Surface distance of image $V1$, from detected surface to reference. .	133
5.17	Detected result of image $V2$. (a) Raw image and 3 orthogonal 2D slices, (b) detected surface and corresponding contours on the 3 slices.	133
5.18	Volume difference of image $V2$. (a) Reference surface and detected surface, (b) volume difference in middle slices, (c) volume difference in one end, (d) volume difference in the other end.	134
5.19	Surface distance of image $V2$, from detected surface to reference. .	135
5.20	Three different sets of initial level-sets (column by column) presented by three users over image $V1$ at slice 84, slice 126, slice 168 and slice 210.	138

5.21	Similar Results (superposed each other on three orthogonal slices) obtained from the three different sets of initial level-sets shown in Fig. 5.20. <i>Yellow line</i> : the first result, <i>red line</i> : the second result, <i>green line</i> : the third result.	139
5.22	Diagram of the oscillator system.	142
5.23	Interpolation for creating 3D images from image stacks.	143
5.24	Detected result from image stack <i>S0764782D</i> . (a) 3D image constructed from the image stack, (b) detected surface and corresponding contours on the 3 orthogonal slices.	145
5.25	Surface distance for image stack <i>S0764782D</i> , from detected surface to reference.	145
5.26	Volume difference for image stack <i>S0764782D</i> . (a) Reference surface and detected surface, (b) volume difference in middle slices, (c) volume difference in one end, (d) volume difference in the other end.	146
5.27	Detected result from image stack <i>S2532520F</i> . (a) 3D image constructed from the image stack, (b) detected surface and corresponding contours on the 3 orthogonal slices.	148
5.28	Surface distance for image stack <i>S2532520F</i> , from detected surface to reference.	148
5.29	Volume difference for image stack <i>S2532520F</i> . (a) Reference surface and detected surface, (b) volume difference in middle slices, (c) volume difference in one end, (d) volume difference in the other end.	149
5.30	Analysis of the sensitivity of the method to threshold value for stopping criteria. (a) Variation of the average accuracy with respect to the threshold value, (b) variation of the average computation time with respect to the threshold value.	154
5.31	GMMs with different components ($n = 2, 3, 4$ from top to bottom). (a) GMMs for image V0, (b) GMMs for image V4.	155
5.32	Comparison of the accuracy with respect to GMM components . .	156

5.33	Comparison of the computation time with respect to GMM components	157
5.34	Average accuracy with respect to side length of ROI	159
5.35	Average computation time with respect to side length of ROI . . .	160
5.36	Comparison of the average accuracy with respect to parameter a .	162
A.1	Open curve outlining.	186
A.2	Closed curve outlining.	186
A.3	Baby surface: an open surface construction.	187
A.4	Prostate surface: a closed surface construction.	187
A.5	Register a prostate surface with a 3D ultrasound image.	188
A.6	Needling simulation on a prostate model.	188
B.1	Orientation of a facet is determined by the direction of the unit normal and the order in which the vertices are listed.	191
B.2	The vertex-to-vertex rule. The left figure shows a violation of the rule. A correct configuration is shown on the right.	191

List of Tables

1.1	Major applications for which prostate boundaries and volumes play key roles in clinical decisions	6
2.1	Categorization of methods for 2D prostate boundary detection from ultrasound images	17
4.1	Comparison of the run time between narrow band solution and the faster algorithm	103
5.1	Summary of the results of algorithm validation on 3D images . . .	136
5.2	Summary of the accuracy in test-retest	137
5.3	Summary of the accuracy in inter-rater test	137
5.4	Disagreements between the algorithm results and segmentations by experts	140
5.5	Disagreements (D) between the manual segmentations by experts .	141
5.6	Image resolution (mm/pixel) of different zoom rate for ultrasound imaging	144
5.7	Summary of the results of algorithm validation on 2D image stack .	150
5.8	Summary of the accuracy in test-retest on 2D image stacks	151
5.9	Summary of the accuracy in inter-rater test on 2D image stacks . .	151
5.10	Disagreements between the algorithm results and segmentations by experts	152
5.11	Disagreements between the manual segmentations by experts . . .	152
5.12	Accuracy of the algorithm with respect to the side length of ROI (I)	158

5.13	Accuracy of the algorithm with respect to the side length of ROI (II)	158
5.14	Accuracy of the results with respect to a ($b = 1.5$)	161
5.15	Accuracy of the algorithm with respect to a	161

CHAPTER 1

Introduction

1.1 Motivation

The prostate is a walnut-sized gland of the male reproductive system. It is partly muscular and partly glandular, with ducts opening into the prostatic portion of the urethra. As shown in Fig. 1.1(a), the prostate is located in front of the rectum and just below the bladder, which stores urine. The main purpose of the prostate is to produce fluid for semen, which transports sperm during the male orgasm. However, as men age, the prostate can be a source of problems.

Figure 1.1: Prostate anatomy and BPH. (a) The prostate and nearby organs, (b) benign prostatic hyperplasia.

Benign prostatic hyperplasia (BPH) and prostate cancer are two common diseases in adult and elderly men. Benign prostatic hyperplasia usually leads to urinary symptoms of varying severity in almost 50 percent of men over 60 years

old [1]. As the prostate enlarges, the layer of tissue surrounding it stops it from expanding, causing the gland to press against the urethra like a clamp on a garden hose (see Fig. 1.1(b)). The bladder wall becomes thicker and irritable. The bladder begins to contract even when it contains small amounts of urine, causing more frequent urination. Eventually, the bladder weakens and loses the ability to empty itself. Urine remains in the bladder. The narrowing of the urethra and partial emptying of the bladder cause many of the problems associated with BPH.

The dreadful problem, however, is prostate cancer. It has the second highest mortality rate, after lung cancer, among all cancers in men in North America [2]. It is also the sixth most common cancer in Singapore male [3]. The American Cancer Society estimates that during 2003 about 220,900 new cases of prostate cancer will be diagnosed in the United States. One man in six will be diagnosed with prostate cancer during his lifetime [4]. However, many of the patients remain asymptomatic until extensive local growth or metastasis of the tumor has occurred. Once the tumor is no longer encapsulated within the prostate, there is a dramatic increase in the risk of metastasis, and a corresponding decrease in the prospect of a cure. When diagnosed at an early stage, prostate cancer is curable. Nevertheless, even in the later stages, treatment can still be effective. Thus, the early diagnosis and accurate staging of prostate cancer are of primary importance.

When BPH or prostate cancer is suspected, several tests can help the doctor identify the problem and decide whether surgery is needed. The tests vary from patient to patient, but the following are the most common.

- **Digital rectal examination (DRE)**

This is usually the first test. The doctor inserts a gloved finger into the rec-

tum and feels the part of the prostate next to the rectum. This examination gives the doctor a general idea of the size and the condition of the gland and, therefore, can be used to detect many cases of prostate cancer but not all, since it is not possible to feel the entire prostate gland through the wall of the rectum. Nevertheless, the examination is easy and quick to do, and does not require an unreasonable effort for men over 40 to have this included with an annual physical examination.

- **The PSA**

Prostate specific antigen (PSA), a protein produced by prostate cells, is frequently present at elevated levels in the blood of men who have prostate cancer. This testing technique, particularly in combination with a rectal examination, is effective in detecting prostate cancers that are small in size and localized to the prostate. After PSA testing was introduced, nearly two-thirds of the cancers that are detected in screening programs are confined to the prostate and are potentially curable.

However, high levels of PSA also can be found in other conditions that are noncancerous, including prostatitis and BPH. If the PSA test result is outside the normal range, most urologists will recommend a test called a “transrectal ultrasound”. Depending on the results of the transrectal ultrasound, a second test called a “needle biopsy of the prostate gland” may be recommended.

- **Prostate ultrasound**

An ultrasound test uses very high frequency sound waves that are passed through the body to detect potential problems with the prostate. The pattern

of reflected sound waves, or “echoes,” shows the outline of the prostate. This test can show whether the prostate is enlarged, and whether an abnormal growth indicating the presence of cancer. If prostate cancer is present and the doctor decides to treat it with a surgical procedure, ultrasound is used as an aid in this procedure.

As reported in [5], the increase of incidence of prostate diseases is believed to be due to longer life expectancy, increased prostate awareness, prostate-specific antigen (PSA) screening, and imaging techniques such as prostate ultrasound.

Prostate ultrasound is generally done using a technique called the transrectal method—transrectal ultrasound (TRUS) [6, 7]. Transrectal ultrasound of the prostate is a procedure in which visual images of the prostate are produced with the use of an ultrasound probe placed into the rectum. This is usually done with the patient lying on his left side or on his back. A well-lubricated probe, about twice the thickness of a finger, is placed through the anus into the rectum. Fig. 1.2(a) illustrates the TRUS imaging mechanism. If a cancer is present it may appear as a less dense, darker area than the surrounding tissues. A typical prostate scan of TRUS is shown in Fig. 1.2(b).

Compared to other medical imaging modalities, ultrasound imaging has several favorable properties [8]. It does not require the elaborate (and huge) equipment that X-ray CT and magnetic resonance imaging (MRI) require and portable ultrasound imaging instruments are available. Unlike other tomographic techniques, ultrasound offers interactive visualization of the underlying anatomy in real-time and has the ability to image dynamic structures within the body. Needles and catheters can also be deployed under ultrasonic guidance. For this reason, TRUS

Figure 1.2: Transrectal ultrasound of the prostate. (a) Transrectal ultrasound imaging of the prostate, (b) a typical prostate scan of TRUS.

is widely used for diagnosis of prostatism, and for real-time image guidance of minimally invasive therapeutic procedures

One of the important aspects of TRUS imaging is to identify the prostate boundaries; boundaries provide the anatomical description and volume of the prostate needed for disease treatment, for planning treatment procedures and for monitoring the progression of these treatments. Table 1.1 lists the major applications for which prostate boundaries and volumes obtained from TRUS images play key roles in clinical decisions [9].¹ The gland volume may be derived from the boundaries using planimetric volumetry [10]. With the number of men seeking medical care for prostate disease rising steadily, the need for a fast and accurate prostate boundary detection and volume estimation tool increases correspondingly.

Due to the poor contrast between the prostate and surrounding tissues, speckle noise, shadowing and refraction artifacts [11], prostate boundary detection from

¹Table 1.1 is Adapted from [9] with permission from Prof Y. Kim (University of Washington, Seattle, WA).

Table 1.1: Major applications for which prostate boundaries and volumes play key roles in clinical decisions

Application	Parameter of Interest
<u>Diagnosis</u>	
PSA(prostate-specific antigen) density	Volume
Assessment of BPH(benign prostatic hyperplasia)	Volume
<u>Treatment</u>	
Prostate brachytherapy	Boundary
High-intensity focused ultrasound	Boundary
Cryotherapy	Boundary
Transurethral microwave therapy	Volume
Resection of benign tissue	Volume
<u>Follow-up</u>	
Hormonal treatment	Volume

ultrasound images still remains as a challenging task. Manual contouring is currently the only robust, reliable segmentation procedure available for TRUS of the prostate. This method relies on expert observers, such as experienced surgeons or radiologists to directly outline the desired boundaries in the raw images. Unfortunately, manual contouring is too time-consuming and arduous. The results are heavily dependent on observers' experience and consequently variable between observers (inter-observability) and even within an observer when performing the same job at different time (intra-observability). A possible solution to improve

the efficiency is to automate the boundary detection process with minimal manual involvement, especially for robotic (or computer-assisted) surgery [12].

Most prostate diseases are currently diagnosed with conventional 2D ultrasound equipment. However, conventional 2D TRUS has some serious limitations [13], because only one thin slice of the prostate can be viewed at any time, and the location of this image plane is controlled by physically manipulating the transducer orientation. As a result, the diagnostician or the therapist/surgeon must mentally integrate many 2D images to form an impression of the 3D anatomy and pathology. This process is time-consuming and inefficient, but more importantly, variable and subjective, possibly leading to incorrect decisions in diagnosis, planning, and delivery of the therapy. Moreover, it is difficult to place the 2D image plane at a particular location within an organ, and even more difficult to find the same location again later. To overcome these limitations, 3D transrectal ultrasound of the prostate has been developed [13, 14, 15]. Consequently, 3D prostate surface detection techniques are attracting more and more attentions from the image processing community. However, to develop a 3D prostate surface detection method is even more difficult. Complexity increases greatly in 3D space.

1.2 Challenges

While numerous methods have been proposed to (partially or fully) automatically detect prostate boundary from ultrasound images [16, 9, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32], there is no one method which has been adopted for practical use. Nevertheless, model-based methods seem to be more promising and recently becoming the main-stream in this area [24, 25, 26, 27, 28, 29, 30, 31, 32].

The reason for adopting model-based techniques is the ability to incorporate prior knowledge about the application into these methods to improve their performance.

However, “boundary leaking” problem is a big challenge in image gradient based deformable models for prostate boundary detection from ultrasound images, including the level set method [33] which is the basis of our algorithm. Fig. 1.3(a) demonstrates the leakage when the active contour model is used [26, 27] and Fig. 1.3(b) illustrates the leakage when the level set method is applied.

(a)

(b)

Figure 1.3: Illustration of “boundary leaking” problem of image gradient based deformable models for prostate boundary detection. (a) When the active contour model is applied (*Reprinted with permission from M. Alcaniz (Universidad Politecnica de Valencia, Valencia, Spain).*), (b) when the level set method is applied.

Since our mission is to detect the prostate surface in 3D, most of the existing

methods cannot be used due to the difficulty in extension to 3D version. Although the active contour model can be extended to 3D with adding extra machinery, it needs the user to carefully choose the initial contour, that is, the initial contour should be very close to the desired boundary in order to achieve satisfactory results. This too much user involvement in initialization is against our goal. For these reasons we chose the level set method [33] as the basis to develop our algorithm.

1.3 Objectives

The increasing need for efficient and reliable prostate boundary (or surface) detection motivates us to continue the research work in this area. We hope to achieve a practical method with good reliability and validity to efficiently detect the 3D prostate surface from ultrasound images.

There are two main classes of methodologies to obtain 3D prostate surface. One is reconstructing the 3D prostate surface from a sequence of 2D contours detected in the parallel cross-sectional images [19, 20, 23, 24]; we call this the pseudo-3D approach. An example of this mechanism is shown in Fig. 1.4, where the 2D contours are manually outlined by an expert [34].

The disadvantages of this approach are (1) broken boundary in one slice usually leads to poor result; (2) a segmentation of a slice along different axes may lead to different results; and (3) the reconstruction of the surface and its properties from 2D contours may lead to inaccurate results.

The alternative, which is believed to be more robust and accurate, is to carry out the computation in 3D space and detect the 3D prostate surface directly [31]; we call this a volume approach. Accordingly, we will focus our work on the latter

Figure 1.4: 3D view of the prostate reconstructed from the sequence of manually outlined cross-sectional 2D boundaries. (a) Parallel 2D boundaries drawn by experts, (b) 3D reconstruction. *Reprinted with permission from R. Y. Wu (Nanyang Technological University, Singapore).*

one: volume approach techniques.

Therefore, the goal of this thesis is to develop methods which require minimal manual involvement for 3D prostate surface detection from transrectal ultrasound images.

1.4 Contributions

Consistent with the aforementioned goal, a 3D prostate surface detection technique based on the level set method [33] is proposed. Unlike the standard level set methods, domain specific knowledge about the prostate has been embedded in the proposed algorithm to automatically detect the 3D prostate surface from TRUS images. Region information and Gaussian mixture model (GMM) are both incorporated with the level set method to improve the algorithm performance. The major contributions of the thesis include the following:

- Using the prostate region information to replace the image gradient in design-

ing the evolving speed term for the level set method. The level set method is very powerful in automatic segmentation for CT, MRI images [33], and the key task of its application is to develop an appropriate speed function that can drive the evolving front to the desired boundaries. The standard level set methods generally use the image gradient to define this speed. However, they suffer from the weak image gradient information due to the bad quality of ultrasound images and cause a “boundary leaking” problem consequently. Motivated by region-based segmentation techniques [35, 36], first we design a region classification function R to quickly discriminate the prostate region, then reformulate the level set method with it instead of the image gradient to improve the algorithm performance.

- Knowledge-based features, such as expected shape (kidney-like) and ultrasound appearance of the prostate (looking from within the gland, the intensities are transitions from dark to light) are also integrated to increase the accuracy and robustness. The intensity function inside the prostate gland is modeled as a mixture of three Gaussian distributions whereas the expected shape of the prostate is used to optimize the initialization of evolving surfaces, and the parameters for GMM are automatically estimated using the Expectation-Maximization (EM) algorithm. The incorporation with both knowledge-based information and region information makes our algorithm more accurate and robust.
- Both semi-automatic and fully-automatic surface detection processes are investigated. In this work, the EM algorithm is used to automatically estimate the parameters that describe the mixture of Gaussian distributions. Then, a

threshold is automatically determined from the distributions and it is used in the region classification function for prostate region discrimination. For fully automatic surface detection, the initial surface for deformation is automatically given by putting the centroid of initial spheres close to the image center. In contrast, for semi-automatic surface detection, the initial level sets are provided by user according to the image property.

- A novel and practical “Evolving Bubbles” model for semi-automatic prostate surface detection has been developed. The investigation of semi- and fully-automatic surface detection shows that it is extremely hard to detect the prostate surface from ultrasound images fully automatically. The reliability and validity of the fully-automatic method are difficult to be guaranteed due to the poor quality of ultrasound images. It is much more productive to provide some degree of manual intervention in the algorithm. The “Evolving Bubbles” model allows the user to define the initial level sets (bubbles) in several particular slices according to the prostate shape. When the bubbles evolve, they expand, shrink, merge and split, and finally produce the desired prostate surface.

Besides the main contributions in remedying the boundary leaking problem, the thesis also makes great efforts to speed up the algorithm. According to the idea of the narrow band solution proposed by Sethian in his book [33], a 5-layer narrow band is introduced to implement the algorithm. Moreover, down-sampling technique is employed to further reduce the number of voxels concerned with computation as well as the number of necessary iterations before applying the narrow band solution.

Fig. 1.5 summarizes the contributions of the thesis. Please note the notation ∇_D denotes the finite difference operator applicable for discrete functions only, whereas the notation ∇ denotes the gradient operator applicable for both continuous and discrete functions throughout the thesis. Details of the algorithm development are elaborated in Chapter 3 and Chapter 4.

Figure 1.5: Summary of contributions and algorithm development.

1.5 Organization of the Thesis

The thesis is organized as follows.

Chapter 2 gives an overview of existing methods used for prostate boundary detection from ultrasound images. It includes a survey of different techniques for identifying the prostate boundaries, and a discussion of their advantages and disadvantages.

Chapter 3 describes the 3D prostate surface detection techniques using the level

set method. The advantages and drawbacks of the level set method are carefully analyzed on the basis of its practical application to ultrasound images. Region information instead of image gradient is used to improve the standard level set method to solve the “boundary leaking” problem. To speed up the algorithm performance, a 5-layer narrow band solution is applied to the proposed 3D level set method.

Chapter 4 analyzes the impact of incorporating prior knowledge in the proposed algorithm. A Gaussian mixture model (GMM) is introduced to model the signals inside the prostate and the kidney-like shape of the prostate is used to optimize the level sets initialization. The gradient flow is also included to improve the performance of algorithm . Both fully-automatic and semi-automatic detection processes are investigated. A novel and practical “Evolving Bubbles” model for semi-automatic prostate surface detection is then presented.

Chapter 5 presents the experimental results on a number of clinically obtained images. Evaluation criteria used and the results of the evaluation are presented in this chapter.

Chapter 6 concludes this thesis by summarizing the progress that we have achieved so far. Future work for continuing research as well as the “ground truth” method for algorithm evaluation are also discussed.

Appendix A introduces the **Bubble 3D**, a software developed in Computer Integrated Medical Intervention Laboratory (CIMIL) for creating anatomic geometries of objects from volumetric images, with which the reference surface is created from the manually outlined boundaries.

Appendix B gives the concepts of STL (StereoLithography) and its file format.

The reference surface output from **Bubble 3D** is stored in this format, which we need to know for validating the performance of algorithm.

CHAPTER 2

Related Works

There have been a number of works so far on automatic segmentation of the prostate from ultrasound images. This chapter provides an overview of the approaches used to solve this problem. A full description of competing methods is beyond the scope of this chapter. Interested readers should refer to the references for additional details. Instead, this chapter focus on providing an introduction to the different methods and various issues that must be encountered to achieve automatic segmentation of the prostate from ultrasound images.¹

2.1 2D prostate boundary detection

This section presents a review on common methods that have appeared in recent literature for 2D prostate boundary detection. According to the image segmentation mechanism applied to these methods, we classify them into three categories: edge-based methods, texture-based methods and model-based methods. Table 2.1 summarizes algorithms and representative works within these three categories.

2.1.1 Edge-based detection methods

The common feature of this class of methods is the use of edge detectors to identify all the edges in the image, followed by edge selection and linking to outline the

¹The material presented in this chapter was published as a review paper in *Journal of Ultrasound in Medicine*, 2003.

Table 2.1: Categorization of methods for 2D prostate boundary detection from ultrasound images

Methods	Representative Works
<u>Edge-based</u>	
Minimum/Maximum filtering	Second derivative in gradient direction (SDGD) with local minimum/maximum filters [17, 18, 16]
Derivative edge detector	Match filtering in radial directions [19, 20]
Weak membrane fitting	Edge-based technique [9]
RBR	RBR, harmonics method [21, 22]
<u>Texture-based</u>	
Pixel classifier	Pixel classifying based on four texture energy measures [23]
<u>Model-based</u>	
Deformable contour models	Snakes integrated with sticks algorithm [24] Discrete dynamic contour (DDC) [25] Snakes with shape restrictions based on the wavelet transform [26, 27]
Statistical models	Feed-forward neural networks [28] Markov random field (MRF) [29] Feature Modeling [30] Statistical Shape Models [32]

prostate boundary. Typically, these methods locate edges that correspond to local peaks in the intensity gradient of an image. Such a strategy is very straightfor-

ward and works well when the boundary is clearly defined. However, because of the poor quality of ultrasound images, these methods usually lead to both spurious boundaries in highly textured areas and missed boundaries where prostate boundary is not well delineated. Therefore, both edge detectors and edge selection and linking algorithms should be well designed in order to achieve the desired prostate boundaries in ultrasound images.

In their work, Aarnink *et al.* [17] and Huynen [18] introduced a practical method for automated prostate contour detection based on an edge detection pre-processing algorithm (minimum/maximum filtering). In this algorithm, the second derivative in gradient direction (SDGD) is implemented with local minima and maxima filters, combined with gray value to assess the local gradient in the direction from the minimum to the maximum, i.e.,

$$SDGD_n(x, y) = GradMax_n(x, y) - GradMin_n(x, y), \quad (2.1)$$

$$GradMax_n(x, y) = Max_n(x, y) - Grey(x, y), \quad (2.2)$$

$$GradMin_n(x, y) = Grey(x, y) - Min_n(x, y), \quad (2.3)$$

where $GradMax$ and $GradMin$ are the gradient magnitude in the direction of the maximum and minimum respectively, Max_n and Min_n are the maximum and minimum intensity value in a neighborhood n , $Grey$ stands for the pixel gray value. With this algorithm, all edges in the images are detected (Fig. 2.1(h)). Then knowledge-based features, such as expected shape (kidney-like) and ultrasound appearance of the prostate (looking from within the gland, the edges to be detected, are transitions from dark to light), are used to select the correct edges. With adaptive interpolation, the prostate boundary is finally produced. Fig. 2.1 presents

an example of the contour detection in an ultrasound prostate image [18]. A major disadvantage of their method is that artifacts such as cysts, calcification, and shadowing could lead to erroneous edges.

Figure 2.1: Example of the contour detection process using minimum/maximum filtering. (a) Original ultrasound image, (b) blurred and compressed image, (c) local minimum of the image, (d) local maximum, (e) gradient in the direction from the minimum, (f) gradient in the direction of the maximum, (g) second-derivative in gradient direction, (h) zero crossings in the second-derivative, (i) edge strength, (j) edge strength masked by the zero crossings, (k) selected contour, (l) ultrasound image with the projected contour. *Reprinted with permission from H. Wijkstra (University Medical Center Nijmegen, Nijmegen, the Netherlands).*

To improve the edge detection and localization, Aarnink *et al.* [16] further introduced a technique by adjusting the edge detection parameter to signal information. First they investigated the influence of parameter settings associated with the filtering (the size of smoothing filter and the size of minimum/maximum filter). Then local standard deviation of gray value is used to identify more or fewer homogeneous regions that are filtered with coarse resolution, while areas with larger deviation indicate that gray level transitions occur, which should be preserved using smaller filter sizes to improve edge localization. With the improved result, more interpolation to find a closed contour is needed, and the definition of the prior knowledge becomes more important.

In their proposed method, Lee *et al.* [19] and Chen *et al.* [20] employed low-pass filters to smooth the image; after that, the center of the prostate is estimated. A search process in radial directions from the estimated center is then performed to locate the prostate boundary by using derivative edge detectors. The 2D contour is obtained by properly linking the detected edges. Unfortunately, unwanted structures, such as bladder neck and pelvic bones, are usually included [9].

Pathak *et al.* [9] used an edge-based technique for outlining the prostate boundary. Their algorithm comprises of three stages. First an algorithm called sticks [37] is used to enhance the contrast and at the same time reduce speckle in the TRUS. Then the sticks-enhanced image is further smoothed using an algorithm called weak membrane fitting, which is similar to the anisotropic diffusion filter [38]. Finally, some basic prior knowledge of the prostate, such as shape and ultrasound appearance, as addressed in [17, 18], is used to detect the most probable edges describing the prostate. This detected prostate boundary is presented as a

visual guide to the observer, followed by manual editing. Fig. 2.2(a) and Fig. 2.2(b) show the edge-detection result of a midgland image before and after the knowledge-based filtering. The detected edges are overlaid on top of the image (Fig. 2.2(c)) and then presented as a visual guide to the observers to manually delineate the prostate boundary (Fig. 2.2(d)).

Figure 2.2: Edge-guided boundary detection. (a) Detected edges before *a priori* knowledge-based filtering, (b) edge image after removal of false edges using knowledge-based filtering, (c) edges overlaid on top of the prostate image, (d) final boundary delineation on the prostate image. *Reprinted with permission from Y. Kim.*

Liu *et al.* [21] developed a Radial bas-relief (RBR) method, which is adopted and extended from a darkroom technique used in conventional photography (i.e.,

bas-relief), to segment the prostate boundary from TRUS. The proposed method consists of three steps: (1) image inversion, (2) image panning (to provide a position offset register by moving every pixel radically from the center of the image by a certain distance), and (3) dual-images addition (superimposing positive and negative images). The above process can be expressed as

$$\hat{I}(x, y) = I(x, y) + (255 - I(kx, ky)), \quad (2.4)$$

where $I(kx, ky)$ is the enlarged image with $k < 1$. During the addition process, those image pixels with intensity values greater than 255 (overflow) are set to zero, that is

$$I_d(x, y) = \begin{cases} 0 & \text{if } \hat{I}(x, y) > 255 \\ \hat{I}(x, y) & \text{otherwise} \end{cases}. \quad (2.5)$$

Figure 2.3: Result of RBR and its binarized image. (a) Original image, (b) inverted image, (c) RBR-processed image, (d) binarized image. *Reprinted with permission from W. S. Ng (Nanyang Technological University, Singapore).*

Fig. 2.3(c) shows the result obtained by RBR. It can be seen that the boundary of the prostate is clearly highlighted, and the image looks binary-like. The

resulting image is then binarized by thresholding (Fig. 2.3(d)) and dilation and erosion operators are employed to smooth the boundary area. The smoothed, binarized image is then inverted again to switch the boundary areas to the foreground (Fig. 2.4(a)). Foreground areas with boundary length greater than a given number of pixels are then labeled as boundary areas (Fig. 2.4(b)). To thin these boundary areas, a skeletonization process is introduced. The prostate boundary (Fig. 2.4(d)) is finally produced by complementing the skeleton image (Fig. 2.4(c)) through a one-dimensional polynomial fitting process. Alternatively, the contour complementation can be fulfilled with harmonics [22]. Fig. 2.5 illustrates the process of harmonics curve fitting.

Figure 2.4: Prostate boundary detected by RBR. (a) Inverted binary image, (b) labeled boundary areas, (c) skeleton of boundary areas, (d) boundary complementation. *Reprinted with permission from W. S. Ng.*

A major problem of RBR method is that the RBR center should be close to the boundary centroid in order to produce a good result. If the image center and the object boundary centroid have a big deviation, RBR method would fail to detect the desired boundary.

Figure 2.5: Outline of prostate boundary using harmonics fitting. (a) After second iteration, (b) after third iteration, (c) after fourth iteration, (d) final image. *Reprinted with permission from C. K. Kwoh (Nanyang Technological University, Singapore).*

2.1.2 Texture-based detection methods

Another image segmentation technique for boundary detection is texture discrimination. Rather than trying to find the edges in an image, texture discrimination attempts to characterize regions of an image on the basis of measures of texture. It can also produce an edge map of an image, as with edge detection, by creating a border between regions determined to have different texture. In other words, the edge map created in terms of texture discrimination is by examining regions rather than edges.

In their work, Richard and Keen [23] presented a texture-based segmentation method to extract the prostate boundary by segmenting 2D images of the prostate gland into prostate and non-prostate regions. The texture-based method is in fact a pixel classifier based on four texture energy measures associated with each pixel in the image. A clustering procedure is then used to label each pixel in the image with the label of its most probable class. Although some progress has been made on the ability of their algorithms to delineate the prostate boundaries, Richard

and Keen [23] have acknowledged that the effect of using texture information is marginal.

2.1.3 Model-based detection methods

Most of recent research has shown that model-based segmentation methods are more efficient and powerful in delineating object boundaries [39]. These model-driven image analysis techniques integrate some prior knowledge, such as anatomical information, physical characteristics of the object and radiological features of imaging, into the model, thus improving the boundary detection algorithms. This section reviews two different classes of model-driven techniques that have been applied to prostate boundary detection: (1) methods based on deformable contour models and (2) methods based on statistical models.

Methods based on deformable contour models

Deformable contour models are physically motivated, model-based techniques for delineating object boundaries by using closed curves or surfaces that deform under the influence of internal and external forces. Deformable contour models were first introduced as a tool for image segmentation by Kass *et al.* [40]. In their formulation, known as active contour models or snakes, image segmentation is considered as an energy minimization problem. When an active contour model is placed on an image, solving corresponding equations yields positions along a curve that minimize the energy function. Many sections of the curve will follow edges in the image, but the internal energy constraints will also keep the curve relatively smooth.

In their work, Pathak *et al.* [24] presented an algorithm based on snakes to detect the prostate boundary from TRUS. First the sticks algorithm [37] is used to selectively enhance the contrast along the edges, and afterwards a snake model is applied. This integrated algorithm requires an initial curve input by the user for each ultrasound image to initiate the boundary detection process, and the results show their sensitivity to the curve initialization. With an initial contour reasonably close to the prostate boundaries, the algorithm successfully delineates the prostate boundaries in an image.

Ladak *et al.* [25], however, developed a method based on a deformable contour model, named discrete dynamic contour (DDC) [41]. In this method, initialization requires the user to select only four points (Fig. 2.6(a)) from which the outline of the prostate is estimated using cubic interpolation functions and shape information. To improve the model's performance, gradient direction information is used during deformation to push the model toward the boundaries. However, the success of their approach is dependent on the careful initialization of the contour (Fig. 2.6(b)), which requires the user to select points on the prostate boundary. To overcome this drawback, the authors added as an editing tool to edit the detected boundary and then re-deform it (Fig. 2.6, (c) and (d)).

Knoll *et al.* [26, 27] attempted to address the problem of initialization for deformable contours and modeling for segmentation of the prostate in ultrasound images. They proposed a technique to restrict elastic deformation according to particular prostate shapes of closed planar curve using localized multiscale contour parameterization based on one-dimensional dyadic wavelet transform (WT) [42]. For this purpose, they employed the multiscale parametrical contour analysis to

Figure 2.6: Prostate boundary detection by DDC. (a) Initial DDC, (b) after initial deformation, (c) three points (indicated by squares) were edited and clamped, (d) after second time deformation. *Reprinted with permission from H. M. Ladak (University of Western Ontario, London, Ontario, Canada).*

determine the WT-coefficients and then reconstruct the contour. The difference of the original deformed contour and the reconstructed contour was further used to define inner snake forces. This restricted contour deformation and its initialization by template matching are performed in a coarse to fine segmentation process based on multiscale image edge representation which contains the important edges of the image at various scales. The comparison of detected results between conventional unrestricted snakes and the proposed shape-restricted snake method is shown in Fig. 2.7. It is very clear that this method provides much better results compared with conventional snakes.

Figure 2.7: Comparison of detected results between conventional unrestricted snakes (top row) and the proposed shape-restricted snakes method (bottom row). *Reprinted with permission from M. Alcaniz (Universidad Politecnica de Valencia, Valencia, Spain).*

Methods based on statistical models

The idea behind the statistical models for image segmentation is to code the variations of the selected parameters, which are used to describe the detected object, in an observed population (the training sets) and characterize this in a compact manner. The parameters are estimated from the available segmented images and are used for segmentation of new images. Furthermore, the results from new segmentations can be incorporated into the model, thus updating *a priori* information available to the system.

In their work, Prater and Richard [28] described a method for segmenting transrectal ultrasound images of the prostate using feed-forward neural networks. They presented three neural network architectures for this purpose. Each of these networks was trained using a small portion of a training image segmented by an expert sonographer. This method could provide a good result of segmentation;

however, it requires extensive teaching sets and this greatly complicates the detection procedure. Besides, incorporating user specified boundary information into the neural networks is also a difficult task.

Lorenz *et al.* [29] used the Markov random field (MRF) to extract the prostate boundaries. They adapted and extended a probabilistic method which was first described by Dias and Leitão [43] in 1996 for the estimation of heart wall motion and thickness. They assumed the contour sequence to be a 2D first-order Markov random process and prior knowledge about the contour shape is incorporated by clique potentials defined on a neighborhood system. The contour is estimated iteratively on the basis of the *maximum a posteriori* principle and produced finally in this way. A difficulty associated with MRF method is the proper selection of the parameters controlling the strength of spatial interactions. The other drawback is its intensive computational load.

To automatically detect the prostate boundary, Wu *et al.* [30] presented a “Feature Modeling” method derived from the conventional point distribution modeling (PDM) [44], based on which powerful active shape models were developed [45]. In their models, some specific information of the prostate, such as the shape, actual size, relative position to the ultrasound probe and orientation of its boundary, was successfully integrated. As shown in Fig. 2.8, the authors proposed a simple automatic labeling scheme to sample the boundary and model it. The black round circle in the lower part of the image shows the position of the ultrasound probe and the area centroid of each prostate boundary sample was calculated first. Then the boundaries were labeled clockwise at equal angular increments along the shape. After that, a prostate boundary model is then built up from the sample

vectors. The modeling on 27 sample boundaries captures 9 principle variations of the boundary. New possible prostate boundary can be produced from a weighted sum of the 9 principal variations and the mean boundary. Because 9 weights need to be decided to reconstruct a boundary, genetic algorithm (GA) is employed to optimize the decision process. To improve the GA searching process, both image gradient magnitude and direction are used. Fig. 2.9 shows the results detected by the “Feature Modeling” method. According to the evaluation of Wu *et al.* [30], this method has achieved good accuracy and robustness. A similar strategy of using statistical shape model to segment the prostate from ultrasound images was reported in [32].

Figure 2.8: Labeling scheme for feature modeling. *Reprinted with permission from R. Y. Wu.*

However, the “Feature Modeling” method is only suitable for some particular shape-based prostate images. The specification in the model limits its application to general prostate boundary detection. Moreover, the proposed method requires that the center point (i.e., the symbol ‘+’ in Fig. 2.8) of the ultrasound probe be visible in the image, unfortunately, in most cases, this is unavailable.

Figure 2.9: Prostate boundary detection by feature modeling (top row: boundary recognition from a comparatively clear TRUS image; bottom row: boundary recognition from a blurring image). (a) Original image, (b) outlining of expert observer 1, (c) outlining of expert observer 2, (d) outlining of computer. *Reprinted with permission from R. Y. Wu.*

2.2 3D prostate surface detection

Two kinds of methods for detecting the 3D prostate surface have been reported in the literature. One is pseudo-3D approach, which reconstructs the 3D prostate surface from a sequence of 2D contours detected in parallel cross-sectional images [19, 20, 23, 24]. As addressed in Section 1.3, this kind of method has some limitations in practice, and we will not discuss them here. Only the volume approach techniques are described in this section.

Ghanei *et al.* [31] proposed a 3D deformable surface model for automatic segmentation of prostate. In their method, a model is initialized from a few initial contours that are drawn on different slices. These initial contours, which are outlined as polygons, are similar to the discrete dynamic contours [41]. The model

creates a closed initial surface from the drawn contours (Fig. 2.10), and then it converges from this initial shape to its equilibrium iteratively by movement of its vertices under a weighted sum of internal and external forces. To improve the robustness and speed, the entire model is applied in a multiscale scheme. As shown in their work, the agreement between the model contour and the manual contour in several slices is quite good. However, in order to produce good results, typically 40%-70% of the slices are needed to define the initial surface. This initialization process is not easy to deal with, especially when the number of slices is quite large. Moreover, this method is still subject to the initialization sensitivity problem.

Figure 2.10: Simple example of the 3D deformable model structure. $v(i, j)$ stands for the j -th neighboring vertex of V_i ; $\mathbf{r}_i$, unit radial (normal) vector at vertex V_i ; and $\mathbf{d}_{i,0}$, unit vector along the edge between V_i and $v(i, 0)$.

Fig. 2.11 illustrates a good example of detecting the 3D prostate surface based on deformable models created in **3DView** (GE Medical Systems, Waukesha, WI, originally from Kretz, Austria), a personal computer software for diagnosis and 3D rendering of examination data produced on a VOLUSON 530D MT system.² The

²**3DView** used to be a free software that can be downloaded from Kretz website.

(a)

(b)

Figure 2.11: 3D prostate surface detected by **3DView**. (a) Selection of two contour points on the rotation axis in reference image B, (b) result of the surface detection.

surface is constructed based on the so-called virtual organ computer aided analysis (VOCAL) technology by 3D triangularization of the 2D contours detected in each plane [46], which rotates around a fixed axis (main contour axis; Fig. 2.11(a), red dotted line). In fact, all the 2D contours are produced from the deformation of a

prior prostate shape integrated in VOCAL. Fig. 2.11(a) illustrates the selection of two contour points (selected by the user with a mouse) on the rotation axis, and Fig. 2.11(b) shows the detected result.

(a)

(b)

Figure 2.12: 3D prostate surface detected by **3DView** from a different reference image. (a) Selection of two contour points on the rotation axis in reference image A, (b) final result of the surface detection.

3D View provides an efficient tool for 3D prostate surface detection; however, its limitations are also obvious. The detected result is quite sensitive to the selection of the two contour points on the rotation axis and the selection of the reference slices (Fig. 2.11(a), A, B, C). Fig. 2.12 shows the result obtained from different reference slice as shown in A. It is very clear that the results vary considerably when different reference image is selected.

2.3 Discussion

The previous review has shown that edge-based methods (edge detectors) or region-based methods (texture segmentation) alone are not accurate enough to extract the prostate boundaries from the fuzzy ultrasound images such that researchers currently are mainly focusing on the model-based method development for prostate boundary and surface detection [24, 25, 26, 27, 28, 29, 30, 31, 32]. This does not mean that the edge information or region information is of no use in the development of model-based methods. On the contrary, edge information (image gradient) and region information (via region growing, clustering, or texture discrimination) are widely used in the model-driven methods. In particular, edge information (image gradient) is used in all of the model-based methods referenced in this chapter. Compared with the edge-based detection methods, although more complicated and still needing improvement, model-driven methods elide the edge selection and linking process, which needs to be carefully designed and requires more prior knowledge to consequently improve efficiency. Likewise, in solving the challenging problem of 3D prostate surface detection, we expect that 3D models which make use of region information and edge information, as well as incorporate

prior knowledge, are required. As described in detail in the following chapters, the work in this thesis makes an initial effort in integrating the prior information and region information into a level set based approach.

CHAPTER 3

3D Prostate Surface Detection Using The Level Set Method

3D Prostate surface detection is essentially a volume segmentation problem. Although many approaches for volume segmentation are proposed in the literature, they vary widely depending on the specific application and imaging modality (CT, MRI, ultrasound, etc.). The same algorithm which gives excellent results for one application, might not even work for another. General imaging artifacts like noise, motion and partial volume effects can significantly affect the outcome of a segmentation algorithm. These variabilities make volume segmentation in medical images a very challenging problem.

In general, a volume segmentation approach could consist of more than one segmentation algorithms applied one after the other. Excellent reviews of recent volume segmentation techniques in medical images can be found in [39, 47, 48]. In this thesis we select the level set method [33] as the basic algorithm upon which to develop our new approach for 3D prostate surface detection.¹

3.1 Level set method

We chose the level set method [33] as the surface detection mechanism of our proposed approach for a number of reasons: its ability to handle complex geometry

¹The sections on initialization and speed function design were published in *Proceedings of Medical Image Computing and Computer-Assisted Intervention—MICCAI 2002*.

and topological changes, no significant differences in tracking fronts in 2D and 3D, its computational efficiency in 3D by use of *Narrow-Band* techniques [49, 50], and its free initial hypersurface selection which can potentially relieve the user involvement.

3.1.1 Preliminary knowledge

Suppose we are given an interface separating one region from another, and a speed $\vec{F}$ that tells us how to move each point of the interface. In the figure below, a black curve separates a dark gray inside from a light gray outside, and at each point of the black curve the speed $\vec{F}$ is given. This speed can depend on a variety of physical effects. For example,

Figure 3.1: An interface separating the image apart.

- Imagine that the dark gray is ice and the light gray is water. Then the boundary can shrink as the ice melts, or grow as the ice freezes; the speed

then depends on the temperature change between the two.

- Imagine that the dark gray is honey and the light gray is tea. Then the boundary moves as the heavy fluid falls into to the light one, and the speed depends on gravity, the ratio of the fluid densities, and the surface tension between the two.

Most numerical techniques, such as snakes [40], rely on markers, which try to track the motion of the boundary by breaking it up into buoys that are connected by pieces of rope. The idea is to move each buoy under the speed $\vec{F}$, and rely on the connecting ropes to keep things straight(see Fig. 3.2). The hope is that more buoys will make the answer more accurate.

Figure 3.2: Interface moving with the speed $\vec{F}$ which has a magnitude of F .

Unfortunately, things get pretty bad if the buoys try to cross over themselves, or if the shape tries to break into two. In these cases, it is very hard to keep

the connecting ropes organized. In three dimensions, following a surface like a breaking ocean wave is particularly tough.

Rather than follow the interface itself, the level set method [33] instead takes the original curve (Fig. 3.3(a)), and builds it into a surface. That cone-shaped surface, which is shown in blue (Fig. 3.3(b)), has a special property; it intersects the xy plane exactly where the curve sits. The blue surface is called the *level-set function*, because it accepts any point in the plane as input and feedbacks its height as output. The red front is called the *zero level-set*, because it is the collection of all points that are at height zero.

Figure 3.3: Level-set function and zero level-set. (a) The original front, which lies in xy plane, (b) The level-set function, where the front is intersection of surface and xy plane.

Another way to see why this is called a “level set surface” is to imagine a saw that can cut a slice of the surface and then drop it onto the xy plane. However, the slice has to be perfectly **level**.

- If the saw cuts the blue level set surface at height zero above the xy plane, the ring that will drop to the xy plane will be the original red front.
- If the saw cuts at some other height, a different ring will drop down, producing one of the blue curves instead.

3.1.2 Mathematical background

Level set methods are numerical techniques describing the motion of a closed interface $\gamma(t)$ propagating along its normal direction. This closed interface $\gamma(t) : [0, \infty) \longrightarrow \mathbb{R}^n$, can either be a curve in the 2D space ($n = 2$) or a surface in the 3D space ($n = 3$). The essential idea is to represent the front $\gamma(t)$ as the *zero* level set of a higher dimensional function ψ . Let $\psi(p, t = 0)$, where $p \in \mathbb{R}^n$, be defined by

$$\psi(p, t = 0) = \pm d, \quad (3.1)$$

where d is the distance from point p to $\gamma(0)$, and *plus(minus)* sign is chosen if the point p is outside(inside) the initial front $\gamma(0)$. Thus we have an initial function $\psi(p, t = 0) : \mathbb{R}^n \longrightarrow \mathbb{R}$ with the property that

$$\gamma(t = 0) = \{p \mid \psi(p, t = 0) = 0\}. \quad (3.2)$$

The goal now is to produce an equation for the evolving function $\psi(p, t)$ so that ψ always remains zero on the propagating interface. Let $\mathbf{x}(t), t \in [0, \infty)$ be the path

of a point on the propagation front, that is, $\mathbf{x}(t = 0)$ is a point on the initial front $\gamma(t = 0)$, and $\vec{\mathbf{x}}_t$ be the speed normal to the front at $\mathbf{x}(t)$. Supposed that $F(\mathbf{x}(t))$ is the magnitude of speed $\vec{\mathbf{x}}_t$, then $\vec{\mathbf{x}}_t$ can be written as $\vec{\mathbf{x}}_t = F(\mathbf{x}(t))\vec{\mathbf{N}}(\mathbf{x}(t))$, where $\vec{\mathbf{N}}$ is the outward unit normal. Since the evolving function ψ is always zero on the propagating front, we have

$$\psi(\mathbf{x}(t), t) = 0. \quad (3.3)$$

By the chain rule [51], we get

$$\psi_t + \sum_{i=1}^n \psi_{x_i} x_{it} = 0, \quad (3.4)$$

where x_i is the i -th component of $\mathbf{x}$. Since

$$\sum_{i=1}^n \psi_{x_i} x_{it} = (\psi_{x_1}, \psi_{x_2}, \dots, \psi_{x_n}) \cdot (x_{1t}, x_{2t}, \dots, x_{nt}) = \nabla \psi(\mathbf{x}, t) \cdot \vec{\mathbf{x}}_t, \quad (3.5)$$

by applying the fact that $\nabla \psi = |\nabla \psi| \vec{\mathbf{N}}$ and $\vec{\mathbf{N}} \cdot \vec{\mathbf{N}} = 1$, we then have the evolution equation for ψ , namely

$$\psi_t + F|\nabla \psi| = 0. \quad (3.6)$$

As discussed in [33], F may depend on many factors and can be written as

$$F = F(L, G, I), \quad (3.7)$$

where

- L =*Local properties* are those determined by local geometrical information, such as curvature and normal direction.

- *G=Global properties of the front* are those that depend on the shape and position of the front. For example, the speed might depend on integrals along the front and/or associated differential equations.
- *I=Independent properties* are those that are independent of the shape of the front, such as an underlying fluid velocity that passively transports the front.

Since F is a scalar function as shown in Eq. 3.7, for highlight we call it the speed function rather than the speed magnitude hereafter. Much of the challenge in evolving front problems comes from producing an adequate model for the speed function F ; this is a separate issue independent of the goal of an accurate scheme for advancing the front based on the model for F . The following gives an example of producing a model for the speed function F which determines how the front evolves. That is

$$F = F_0 - \epsilon K, \quad (3.8)$$

where F_0 is a constant term (independent property) which makes the front contract or expand, K is the mean curvature of the evolving front (local property) which regulates the smoothness of the front, and ϵ is the entropy condition expressing the importance of the smoothness regularization (note that ϵ must be positive [33]). The mean curvature K of the evolving front is easily obtained from the divergence of the unit normal vector to the front, for example in 2D,

$$K = \nabla \cdot \vec{\mathbf{N}} = \nabla \cdot \frac{\nabla \psi}{|\nabla \psi|} = \frac{\psi_{xx}\psi_y^2 - 2\psi_x\psi_y\psi_{xy} + \psi_{yy}\psi_x^2}{(\psi_x^2 + \psi_y^2)^{3/2}}. \quad (3.9)$$

As a consequence, Fig. 3.4 illustrates how the initial circle and square expand under the speed function $F = 1 - 0.5K$. It is clear that both initial circle and

square finally become a circle due to the smoothness regularization by the local property term $-\epsilon K$.

Figure 3.4: Front expanding under the speed function $F = 1 - 0.5K$. (a) The initial level set, (b) front expanding after 1000 iterations, (c) front expanding after 2000 iterations.

Fig. 3.4 shows that as the number of iterations increases, the evolving front expanded beyond the object boundary. How can the level set method be used to detect the object boundaries? One of the solutions is using the image gradient magnitude to constrain the evolving speed function which drives the front to the object boundary:

$$F_I = \frac{1}{1 + |\nabla I|^p} (F_0 - \epsilon K), \quad p = 1, 2. \quad (3.10)$$

where $I = I(x, y, z)$ denotes the image intensity. Since the image gradient ∇I has

its peak values only at the object boundary as shown in the Fig. 3.5, the speed function constrained by the image gradient magnitude could be very small in the vicinity of the object boundaries, i.e.,

$$F_I = \frac{1}{1 + |\nabla I|^p} (F_0 - \epsilon K) \rightarrow 0 \quad \text{if } |\nabla I| \rightarrow \infty, \quad p = 1, 2. \quad (3.11)$$

Figure 3.5: Illustration of the image gradient magnitude. (a) Raw image, (b) image of the image gradient magnitude.

This means the evolving front could stop in the vicinity of the object boundaries thus the task of boundary detection is fulfilled. Fig. 3.6 clearly shows how the image gradient magnitude can constrain the speed function to drive the evolving front to the object boundaries.

Unlike other deformable models for object boundary detection, the extension of the level set method to 3D domain is straightforward; no other extra machinery is needed [33]. This is one of the key reasons why we employ it to develop our algorithm. In the level set formulation, the volume segmentation problem can be

Figure 3.6: Illustration of the object boundary detection of the level set method constrained by the image gradient magnitude. (a) Raw image, (b) image of the image gradient magnitude, (c) magnitude value of the image gradient, (d) the speed function constraint, (e) initial circle for the level set method, (f) detection result.

expressed as the computation of a 3D surface $S(t)$ propagation in time along its normal direction, and the propagating front $S(t)$ is embedded as the *zero* level set of a time-varying 4D function $\psi(p, t)$:

$$S(t) = \{p \in \mathbb{R}^3 \mid \psi(p, t) = 0\}. \quad (3.12)$$

The level set method then evolves the 4D function $\psi(p, t)$ that contains the embedded motion of $S(t)$ instead of the original 3D surface. Apparently, the evolution of the 4D function ψ has the same formula as in Eq. 3.6. The mean curvature K

in 3D can be obtained through the following equation,

$$K = \frac{\psi_{xx}(\psi_y^2 + \psi_z^2) + \psi_{yy}(\psi_x^2 + \psi_z^2) + \psi_{zz}(\psi_x^2 + \psi_y^2) - 2(\psi_{xy}\psi_x\psi_y + \psi_{xz}\psi_x\psi_z + \psi_{yz}\psi_y\psi_z)}{(\psi_x^2 + \psi_y^2 + \psi_z^2)^{3/2}}. \quad (3.13)$$

If we write

$$k_I = \frac{1}{1 + |\nabla I|^p}, \quad p = 1, 2, \quad (3.14)$$

then Eq. 3.10 can be rewritten as

$$F_I = k_I(F_0 - \epsilon K). \quad (3.15)$$

Compared with the classical energy based snakes model, the term $(F_0 - \epsilon K)$ acts as the internal force and the external force is given by k_I , which is supposed to prevent the evolving front from penetrating the object boundaries into the image background. However, unlike the traditional deformable models, the level set method does not depend on the parameterization of the fronts, but only on the velocity the fronts traveled. This makes it very attractive and flexible in shape modeling and image segmentation.

The most attractive advantage of the level set method is that the whole segmentation procedure could be fully automatic provided that the initial hypersurface (zero level set) is given. Moreover, the initial hypersurface can be chosen freely, that means, it does not need to be close to the desired boundaries as does in traditional parameterization models. This can relieve the user interaction and consequently improve the algorithm efficiency.

3.1.3 Numerical algorithm

For numerical solution of Eq. 3.6, it is necessary to perform discretization in both space and time. For this purpose we can discretize space coordinates using a uniform mesh of spacing h , with grid nodes denoted by indices ijk .² Let ψ_{ijk}^n be the approximation to the solution $\psi(ih, jh, kh, n\Delta t)$, where Δt is the time step. The expression for ψ_{ijk}^{n+1} can be derived using the forward definite difference method:

$$\psi_{ijk}^{n+1} = \psi_{ijk}^n - \Delta t F |\nabla \psi_{ijk}^n|. \quad (3.16)$$

From Eq. 3.15 and Eq. 3.16, we can see that the level set formulation includes two kinds of terms: the first-order derivative term $|\nabla \psi|$ and the second-order derivative term $K |\nabla \psi|$. The first-order term describes a moving front with a space-varying or non-varying velocity. Equations of this form cannot be solved with a simple finite forward difference scheme. Such schemes tend to overshoot, and they are unstable. Osher and Sethian [33] have proposed an *upwind scheme*. The upwind scheme relies on a one-sided derivative that looks in the upwind direction of the moving wave front, and thereby avoids the overshooting associated with finite forward differences. The idea of upwind scheme in [33] can be expressed as: Consider a spatial derivative of u , define the *forward difference operator* as

$$D^{+x}u = \frac{u(x+h, t) - u(x, t)}{h} \quad (3.17)$$

²In 2D space, the grid nodes are denoted by indices ij . However, because our objective is 3D surface detection, in this thesis, we express the algorithms all in 3 dimensions.

and the *backward difference operator* as

$$D^{-x}u = \frac{u(x, t) - u(x - h, t)}{h}. \quad (3.18)$$

Then the 3D discrete form of level set can be rewritten as

$$\psi_{ijk}^{n+1} = \psi_{ijk}^n - \Delta t [\max(F_{ijk}, 0) \nabla^+ + \min(F_{ijk}, 0) \nabla^-], \quad (3.19)$$

where

$$\nabla^+ = \left[\begin{array}{ccc} \max(D_{ijk}^{-x}, 0)^2 & + & \min(D_{ijk}^{+x}, 0)^2 & + \\ \max(D_{ijk}^{-y}, 0)^2 & + & \min(D_{ijk}^{+y}, 0)^2 & + \\ \max(D_{ijk}^{-z}, 0)^2 & + & \min(D_{ijk}^{+z}, 0)^2 & \end{array} \right]^{1/2} \quad (3.20)$$

and

$$\nabla^- = \left[\begin{array}{ccc} \max(D_{ijk}^{+x}, 0)^2 & + & \min(D_{ijk}^{-x}, 0)^2 & + \\ \max(D_{ijk}^{+y}, 0)^2 & + & \min(D_{ijk}^{-y}, 0)^2 & + \\ \max(D_{ijk}^{+z}, 0)^2 & + & \min(D_{ijk}^{-z}, 0)^2 & \end{array} \right]^{1/2}. \quad (3.21)$$

Here, the D_{ijk}^{+x} stands for $D^{+x}\psi_{ijk}^n$ and so on.

The second-order term generally do not require the upwind scheme and can be solved with a finite forward difference schemes using centralized differences for the derivatives. For example,

$$u_x = \frac{u(x + h, y, z) - u(x - h, y, z)}{2h}, \quad (3.22)$$

$$u_{yy} = \frac{u(x, y + h, z) + u(x, y - h, z) - 2u(x, y, z)}{h^2}, \quad (3.23)$$

$$u_{xy} = \frac{1}{4h^2} [u(x + h, y + h, z) + u(x - h, y - h, z) - u(x + h, y - h, z) - u(x - h, y + h, z)]. \quad (3.24)$$

3.2 3D prostate surface detection with level set method

Now the level set method described in the previous section can be used to fulfill our task: 3D prostate surface detection from ultrasound images. As mentioned before, the key point of the level set method is to design an appropriate speed function F which can drive the evolving front to the desired object surface. Appropriate designs of F are discussed here and in the next chapter as well. Besides, some other issues, such as initialization of the level set and algorithm using narrow band solutions to reduce computation, are also discussed in detail.

3.2.1 Initialization

The ease of initialization has always been a desirable feature of any segmentation algorithm. Since the level set method is able to handle the change of topology, our initializing surfaces do not need to be placed close to the boundaries of interest and in similar topological form. We thus automatically put a small sphere at the image center as the initial zero level set to start the surface detection (The optimization of initialization will be discussed in Chapter 4). Fig. 3.7 illustrates the initial surface and its corresponding 2D curves in three orthogonal cross-sectional images.

Now the signed distance function ψ at $t = 0$ can be expressed as

$$\psi(i, j, k, t = 0) = (i - x_c)^2 + (j - y_c)^2 + (k - z_c)^2 - r^2, \quad (3.25)$$

where (x_c, y_c, z_c) denotes the coordinates of the image center, and r the radius of the sphere. Obviously, ψ has the positive value outside the initial surface, and negative inside. This makes the evolving front deform from inside the prostate to the desired prostate surface as we expected. The coordinates of the image center,

Figure 3.7: Initialization of the level set method by automatically putting a small sphere ($r = 20$ voxels) at the image center.

x_c, y_c and z_c , are determined by

$$x_c = D_x/2, \quad y_c = D_y/2, \quad z_c = D_z/2, \quad (3.26)$$

where D_x, D_y and D_z stand for the image dimension along x, y , and z axis respectively.

3.2.2 Narrow band solutions

For a 2D interface evolving in three space dimensions, the level set algorithm is at least an $O(N^3)$ method per time step, where N is the number of points in the spatial direction. One drawback of this technique stems from the computation expense; by embedding the interface as the zero level-set of a higher dimensional function, a

two-dimensional interface problem has been transformed into a three-dimensional problem (see Fig. 3.8). Clearly, the disadvantage of heavy computational load is even worse for a 3D level set based approach.

Figure 3.8: An extra dimension added in solving the front propagation problem.

In order to overcome this drawback, Adalsteinsson and Sethian proposed a fast level set method for propagating interfaces, named *Narrow-Band Method* [49]. The main idea of the narrow band method is to modify the level set method so it only affects points close to the region of interest, namely the cells where the front is located.

In Fig. 3.9 the bold curve depicts the level set and the shaded region around it is the narrow band. The narrow band is bounded on either side by two curves (in 2D) or surfaces (in 3D) which are a distance $2d$ apart. The value of d determines the number of grid points that fall within the narrow band. During a given time step the value of ψ outside the narrow band is stationary and propagating front cannot move past the narrow band. As a consequence of this update strategy, the front can be moved through a maximum distance of d , either inward or outward, at which point we must rebuild an appropriate (a new) narrow band. We reini-

Figure 3.9: Narrow band illustration. (a) A 2D narrow band with the width of $2d$, (b) A 3D narrow band with the width of $2d$.

tialize the level set function by treating the current zero level-set configuration, i.e., $\psi = 0$, as the initial hypersurface $\gamma(0)$. The grid points in the narrow band can be accessed through a linked-list structure, and in this way the number of computations increases with the size of the model rather than the resolution of the embedding. This technique reduces the computation time significantly when compared with the full sized method, especially for 3D level set based approaches.

The narrow band method relies on the fact that when solving for only a single level-set, $\psi(p, t) = 0$ (p is the point on the front), the evolution of ψ is important only in the vicinity of that level-set. The evolution equations for ψ are such that the level-sets of ψ evolve independently (to within the error introduced by the discrete grid). Thus, one should perform calculations for the evolution of ψ only in a neighborhood of the set $S = \{p \mid \psi(p) = 0\}$; the rest can be ignored.

To demonstrate the efficiency of the narrow band method [49], we can estimate an operation count for it as follows. Assume a three-dimensional calculation, and thus with N grid points in each direction, we have N^3 points in the mesh. To

advance a front throughout the entire domain requires roughly N steps, leading to an $O(N^3 * N) = O(N^4)$ calculation in the full sized level set method. With the narrow band approach, the operation count is reduced to $O((N^2 * k) * N) = O(kN^3)$, given that k is the width of this narrow band. Neglecting the overhead associated with computing the narrow band at each iteration, the gain in computation of narrow banding versus the full sized method is thus proportional to N/k , which is substantial for large images and small band widths.

Basically, the width of the narrow band is predefined and should be as small as possible. However, since it is necessary to rebuild a new narrow band whenever the front hits the boundary of the current narrow band and the narrow band rebuilding costs plenty of time, in actual implementations it needs to find the balance between the width of the narrow band and the frequency of reinitializations. Unfortunately, the choice of the balance value depends on some experimentation [49], and even worse it usually varies from one application to another.

To simplify the algorithm implementation, the so-called sparse-field method [50] takes the narrow band strategy to the extreme: It computes updates on a band of grid points that is only one point wide at each side. It uses an approximation to the distance function that makes it feasible to recompute the neighborhood of the zero level set at each time step. This approach has two main advantages: 1) the width of the neighborhood is such set that derivatives for the next time step can be easily calculated, and 2) it is much easier to keep track of the neighbors using a linked-list structure as the *zero* level-set of ψ moves from one set of voxels to another, which remains a tough work when the band size is greater than 2 voxels. Because at least $2d + 1$ layers are needed to keep track of the neighbors within the

narrow band, where d is half of the narrow-band width as shown in Fig.3.9. The larger d , the more number of layers needed. Following this idea (i.e., d is set to the width of one voxel), in this thesis we built a five layer narrow band to implement the level set method. Its 2D projection is shown in Fig. 3.10, where *active set* contains the set of grid points which lie adjacent to the zero level set, *immediate neighbors* construct the actual boundaries of the narrow band, and *next neighbors* are used to compute the second order derivatives of ψ for calculating the mean curvature K .

Figure 3.10: 2D projection of a five layer narrow band construction.

The number of layers should coincide with the size of the neighborhood used to calculate derivatives as reported in [50] is the reason why a five layer narrow band is applied. Since the second order derivatives of ψ need to be calculated

according to Eq. 3.22, Eq. 3.23, and Eq. 3.24, when we update the level set value at those *immediate neighbors*, two more layers (*next neighbors*) that are two points far away should be included. The grid points on each layer are accessed through a linked-list structure as mentioned before. When the surface evolves, we need to update the value of the level set function ψ at each point which is initially on the zero level-set, as well as the narrow band. The strategy for rebuilding the narrow band and reinitializing ψ at each time step is as follows.

```

for every point  $a$  on the current front
{
    update  $\psi(a)$ ;
    if  $\psi(a)$  is greater than  $1/2$ 
    {
        surface expands and then add the nearest outside neighbor  $b$  to the
        active set;
        remove  $a$  from the active set and set it as the inside neighbor;
    }
    else if  $\psi(a)$  is less than  $-1/2$ 
    {
        surface shrinks and then add the nearest inside neighbor  $b$  to the
        active set;
        remove  $a$  from the active set and set it as the outside neighbor;
    }
}

```

Similarly, update the outside and inside layers and thus build a new narrow band.

3.2.3 Big challenge: the “Boundary Leaking” problem

Due to the poor quality of ultrasound images as addressed in Section 1.1, the boundary feature of the object is usually not salient enough and the image gradient information is weak as shown in Fig. 3.11 for the real ultrasound image of the prostate. Accordingly, the speed function F_I constrained by the image gradient magnitude could have a quite big value (compared to zero in ideal situation) in some areas where the prostate boundaries are not well defined. That means the evolving front is hardly to stop in these areas and thus the “Boundary Leaking” problem occurs. Fig. 3.12 illustrates the magnitude value of the image gradient as well as the speed function constraint for the transverse view in Fig. 3.11(b), and the white arrows marked in Fig. 3.12(b) designate the areas where the “boundary leaking” problem is easy to happen.

Figure 3.11: Ultrasound image of the prostate and its gradient magnitude image. (a) Ultrasound image of the prostate, (b) its gradient magnitude image.

Figure 3.12: Ultrasound image of the prostate and its gradient magnitude image. (a) Ultrasound image of the prostate, (b) its gradient magnitude image.

As a result, the “boundary leaking” problem of the the level set method for 3D prostate surface detection is demonstrated in Fig. 3.13.

Figure 3.13: An example of “boundary leaking” problem of the level set method. *Ragged white curve*: the detected boundary of the level set method, *smooth green curve*: the manually outlined boundary, *yellow arrow*: the boundary leaking caused by shadowing, *red arrow*: the boundary leaking caused by low contrast.

3.2.4 Region-based level set method

What can we do to remedy the “boundary leaking” problem? Can we use other information instead of the image gradient to constrain the evolving speed? If we can redesign the speed function to let it drive the evolving front expand quickly inside the prostate but move slowly in the vicinity of the prostate boundary or even shrink back when it passes the prostate boundary, then the “boundary leaking” problem could be solved. As we know, the ultrasound appearance of the prostate scan is a transition from dark to light if looking from within the gland [16, 17, 18]. Motivated by the region-based strategy for active contour models to improve their performance [35, 36], we try to integrate the region information instead of the image gradient into the level set method to remedy the leaking problem. For this purpose, we first develop a fast discrimination method to roughly extract the prostate region according to the intensity homogeneity as

$$R(x, y, z) = \begin{cases} 0 & \text{if } 2 < [\max(I_n) - \min(I_n)] < \text{givenThreshold} \\ 1 & \text{otherwise} \end{cases}, \quad (3.27)$$

where $\max(I_n)$ and $\min(I_n)$ stand for the maximal intensity and the minimal intensity respectively inside a sliding cubic (n^3) window (Fig. 3.14). We set the lower bound as 2 to exclude the black background in the images. In our algorithms, we take n as 5 empirically and the *givenThreshold*, denoted as R_T for simplicity, is given by

$$R_T = \frac{1}{N} \sum_{i=1}^N [\max(I_n) - \min(I_n)]_i, \quad (3.28)$$

where N is the number of sliding cubes fitted inside the region that is preset by

Figure 3.14: Demonstration of a $3 \times 3 \times 3 (n = 3)$ sliding cubic window, where the grey cube stands for the intensity of voxel at (x, y, z) .

the user as shown in Fig. 3.15.³

Figure 3.15: Preset Region for calculation of R_T .

³In our algorithm tests, the accuracy of results most often reaches the maximum at $n = 5$ or 7. Here we set n to 5 for tradeoff between speed and accuracy.

As a result, Fig. 3.16 illustrates an example of what the region extraction function R looks like in image. Compared with the image gradient magnitude as shown in Fig. 3.11(b), the image boundary information is clearly enhanced even though the boundary gaps are a little bit worse. The compensation for the boundary gaps worsened by the region extraction function R will be addressed later in Chapter 4.

Figure 3.16: Example of the binary image of R . (a) Ultrasound image of the prostate, (b) the binary image of R .

The function R is then incorporated into the level set method and forms a new speed function as

$$F_{new} = (1 - R) F_0 - \epsilon K. \quad (3.29)$$

Now F_{new} is not related to image gradient any more, it only depends on the prostate region and evolving front curvature. Conceptually, F_{new} has the following proper-

ties:

(I) $R = 0$, then

$$F_{new} = F_0 - \epsilon K . \quad (3.30)$$

This means inside the prostate region, the evolving front will deform according to its mean curvature only, and no image constraints are included.

(II) $R = 1$, then

$$F_{new} = -\epsilon K . \quad (3.31)$$

This means in the region immediately outside the prostate, the evolving front will shrink back.

Therefore, the interaction of these two properties may make the evolving front eventually attracted to the desired boundaries.

3.2.5 Stopping criterion

When the evolving front arrives at the desired surface, we need to stop the iterative computation process of our algorithm. To satisfy this requirement, a stopping criterion is introduced in this thesis: if the ratio of new voxels to the total number of voxels on the evolving front in a certain number of consecutive iterations is below a threshold, the surface is thought to reach the desired boundaries already and the iteration stops.

The choice of this threshold value expresses a tradeoff between the accuracy of the detection and the speed of convergence. It has been empirically determined from the experiments. In our algorithm, 0.0035 is finally set as the threshold value

for stopping criterion. However, the quality of the results does not appear to be sensitive to this choice (sensitivity analysis can be found in Section 5.4).

3.2.6 Algorithm implementation

Full steps of the algorithm implementation can be diagrammatized in Fig. 3.17.

Figure 3.17: Diagram of the algorithm implementation.

Here we apply this algorithm to eight typical 3D prostate scans to test if it can work correctly. All the input images to the algorithm are collected from Singapore Gleneagles Hospital by using Voluson 530D. In this algorithm, $F_0 = 1$ and $\epsilon = 1$ are selected empirically from the experiments and the median filter is used for preprocessing the image for its good property of removing speckle noise efficiently and at the same time preserving boundary information [52]. Fig. 3.18 shows one of the results detected by our proposed algorithm.

Figure 3.18: 3D prostate surface and its cross-sectional contours detected by our proposed algorithm. (a) Transverse view, (b) sagittal view, (c) coronal view, (d) initial surface, (e) final surface, (f) final surface with cutting planes; (1) slice of the original image, (2) slice with the initial contour overlaid, (3) slice with the final contour overlaid.

The 3D image size in Fig. 3.18 is $256 \times 256 \times 256$. The initial and final 3D

shapes are shown in the most right column, and the slices shown as transverse view, sagittal view and coronal view respectively are selected by the cutting planes as shown in (f). For this image, the region size is set to $150 \times 150 \times 150$ to calculate the R_T . To validate the effectiveness of our algorithm, the above results are compared with the manually outlined contours (drawn by an expert from a local hospital) in cross-sectional images as shown in Fig. 3.19.

Figure 3.19: The comparison between detected contours and manual contours in 2D cross-sectional slices. (a) Cross-sectional slices, (b) detected contours, (c) manual contours, (d) contour comparison.

The evaluation carried out at this stage is purely subjective, that means the disagreements between the detected results and the outlines of the expert are evaluated by the experts' naked eyes.⁴ From their experience, the radiologist and pathologist from local hospitals deem that there is a good agreement between the

⁴Details of the quantitative evaluation methods and the evaluation results are given in Chapter 5.

detected contours and the manual contours except some divergence in the right sides of sagittal view and coronal view. Without other specification, the green lines presented in the figures throughout this thesis are the outlines drawn by the experts, and the white lines represent the detected results. The new method is then applied to other patients' images. One of the tests is shown in Fig. 3.20, where the image size is $250 \times 174 \times 236$ and the region size for R_T is set to $180 \times 100 \times 150$. Note that the legend (1), (2) and (3) in Fig. 3.20 has the same meaning as in Fig. 3.18.

Figure 3.20: Detected Results of algorithm test on another typical 3D prostate scan. (a) Transverse view, (b) sagittal view, (c) coronal view, (d) initial surface, (e) final surface, (f) final surface with cutting planes.

Although the detected result is not good enough to be accepted by the clinicians, compared with the result obtained by the image gradient based level set

method as shown in Fig. 3.21, our new approach performs much better and successfully detected most of the desired boundaries.

Figure 3.21: Results obtained by gradient based level set method. (a) Same slice as in Fig. 3.20(a), (b) same slice as in Fig. 3.20(b), (c) same slice as in Fig. 3.20(c).

3.3 Discussion

In our algorithm, parameters such as F_0 , ϵ and the sliding window size n for region discrimination are predetermined experimentally and kept unchanged in all the various implementations over different images. In these tests, we take a small sphere as the initial surface put at the image center to start the detection procedure, thus only one parameter R_T is determined according to the region preset by the user. Consequently, the human involvement is minimized.

Although great improvements have been made by the substitution of region information for image gradient to guide the level set evolution, the final detected results still vary with the choice of the R_T value. How to preset a proper region for the calculation of R_T is still an open problem. The second problem is the high computation cost. We have used narrow band solutions to speed up the 3D level set algorithm, however, the computation time increases rapidly with the number

of grid points which reside in the narrow band and the narrow band rebuilding at each iteration takes up most of the processing time. As reported in the algorithm tests, every run of our algorithm to completely detect the 3D prostate surface takes more than 40 minutes. This obviously restrains its practical use in clinical activities.

The first problem might be solved by including the statistical information of the image intensity distribution or some prior knowledge of the prostate shape to improve the algorithm performance [53]. As for speeding up the algorithm further, the straightforward idea is to reduce the number of voxels in narrow band or the number of iterations or both of them. Techniques for solving these problems are described in detail in chapter 4.

3.4 Summary

This chapter presents a level set based approach to detect the 3D prostate surface from ultrasound images. The major advantages of using this method over other deformable models include the following:

- First, the embedded propagating front $\gamma(t)$ may change topology, break, merge and form sharp corners as ψ evolves. This ability of handling flexible topology change and complex geometry is taken care of without additional machinery as in traditional parameterization models such as the splines. As a result, initializing curves or surfaces for segmentation in an image do not need to be placed close to the boundaries of interest and in similar topological form, thus relieving user-interaction in the stage of initialization.

- The second advantage of this method is that its extension to 3D is straightforward. There are no significant differences in tracking fronts in three dimensions. By simply extending the array structures and gradient operators, propagating surfaces are easily handled.

Moreover, the computational efficiency of the level set method in 3D space can be greatly improved by the use of narrow band solutions.

In level set formulation, 3D prostate surface detection is expressed as calculating the deformation of evolving function ψ which contains the embedded motion of $\gamma(t)$ according to Eq. 3.6, namely

$$\psi_t + F|\nabla\psi| = 0.$$

The key task in this application is to design an appropriate speed function F which can drive the evolving front to the desired object surface. Due to the poor contrast between the prostate and surrounding tissues, speckle noise, shadowing and refraction artifacts, the commonly used gradient-based speed function F usually leads to “boundary leaking” problem. To overcome this drawback, we first develop a fast discrimination method to extract the prostate region, then this region information, instead of the spatial image gradient, is integrated into the level set method to design the new speed function F_{new} . F_{new} has two good properties:

- (I) when the evolving front is inside the prostate, F_{new} will drive it to expand to the desired prostate surface;
- (II) when the evolving front penetrates the desired surface to the outside of the prostate, F_{new} will drive it to shrink back to the surface.

The interaction of these two properties may make the evolving front eventually attracted to the desired prostate surface.

To start the detection procedure, in this work, a small sphere is automatically put at the image center as the initial surface, and the algorithm is sped up by applying a 5-layer narrow band solution. The region-based speed function F_{new} then drives the small sphere to deform to the desired prostate surface according to the mean curvature of the front. The accuracy of the results depends on the threshold R_T , the only one parameter that is determined from the region pre-defined by the user. However, how to select a proper region for the calculation of R_T still remains as a big problem. Though it needs to be improved, the proposed method has shown its effectiveness from the experimental results.

CHAPTER 4

Evolving Bubbles for 3D Prostate Surface Detection

We have developed an algorithm based on the level set method with region information integrated to detect the 3D prostate surface from ultrasound images. The user involvement is reduced greatly and the experiments have shown its effectiveness. However, the difficulty in presetting a proper region for the calculation of R_T and its high computation expense limit the algorithm in practical use. This chapter focuses on solving these problems by developing some new techniques. “Evolving Bubbles”, a novel and practical model for 3D prostate surface detection, is then presented.¹

4.1 Designing the speed function F with respect to the statistical distribution model

As reported in the literature, knowledge-based features, such as the expected shape (kidney-like) and the ultrasound appearance of the prostate (looking from within

¹The content of this chapter is the colligation of the following 4 papers: (1) Automatic 3D prostate surface detection from TRUS with level sets. In *International Journal of Image and Graphics*, 2004. (2) A fast algorithm of level set method for 3D prostate surface detection. In *Proceedings of the International Conference on Scientific and Engineering Computation (IC-SEC) 2002*. (3) Evolving bubbles for prostate surface detection from TRUS images. In *Proceedings of Third IEEE Symposium on BioInformatics and BioEngineering—BIBE 2003*. (4) A practical method for prostate surface detection from 3D TRUS (Abstract). In *Proceedings of World Congress on Medical Physics and Biomedical Engineering*, 2003.

the gland, the intensities are transitions from dark to light) are used to select the most probable edges describing the prostate from all edges detected by the edge detectors [9, 17, 18, 16]. Since the incorporation of prior information of object shape and appearance into the level set methods has shown its advantage in image segmentation [53, 54, 55], the aforementioned features can also be integrated into our method to improve the algorithm's accuracy and robustness though the strategy is different from others. The kidney-like shape is used to optimize the initialization procedure whereas the ultrasound appearance of the prostate is used to modify the speed function F . This section addresses the latter, modification of the speed function F .

Since the ultrasound appearance of the prostate is a transition from dark to light, theoretically, the intensity distribution of the prostate should be different from that of the background. If we can get the intensity distribution of the prostate together with the intensity distribution of the background, then we can easily distinguish the intensity of the prostate from background. However, how to get the intensity distribution of the prostate together with the intensity distribution of the background still remains unsolved. The difficulty arises from two perspectives.

- Trying to separate the prostate from background by use of intensity distribution of the whole image seems impossible. The intensity distribution of the whole image is only of one-modal shape, except for the black background (image intensity $\cong 0$), as shown in Fig. 4.1.
- Due to the data availability, some images are too difficult to take the sample data which represent the voxels outside the prostate in the background, even with the user careful intervention.

(a)

(b)

Figure 4.1: Histogram of the whole image. (a) Histogram of image A ($256 \times 256 \times 256$), (b) histogram of image B ($250 \times 174 \times 236$).

Therefore, in this thesis we would only model the intensity distribution inside the prostate to classify the intensity main contribution.

4.1.1 Gaussian mixture model (GMM) for intensity distribution of prostate

In this work, we assume that the intensity probability density function (PDF) inside the prostate can be modeled as a mixture of n Gaussian distributions. The idea behind using Gaussian mixture model (GMM) is that a linear combination of Gaussian basis functions can form smooth approximations of arbitrarily shaped densities.

In fact, Gaussian mixture models are among the most statistically mature methods for density estimation and clustering [56]. One of the advantages is that the size of the model only increases with the complexity of the problem being solved, rather than the size of the obtained data set. Therefore, it supplies a simple framework for modeling population heterogeneity. A general form of GMM with n distributions can be given by

$$p(X | \Theta) = \sum_{i=1}^n \alpha_i p_i(X | \Theta_i), \quad (4.1)$$

where X is the observation variable and Θ is the parameter vector for the mixture of Gaussian distributions, the mixing parameter, α_i expresses the prior probability of the corresponding component function $p_i(X | \Theta_i)$ and conforms to the following constraint

$$\sum_{i=1}^n \alpha_i = 1. \quad (4.2)$$

For a multivariable (for example, d -dimensional) Gaussian mixture model, the i -th component distribution is expressed as

$$p_i(X | m_i, \Sigma_i) = \frac{1}{(2\pi)^{d/2} |\Sigma_i|^{1/2}} \exp \left\{ -\frac{1}{2} (X - m_i)^T \Sigma_i^{-1} (X - m_i) \right\}, \quad (4.3)$$

where X is a d -dimensional observation vector, m_i is the mean vector and Σ_i the covariance matrix. In our work, the observation data are only from the intensities of the prostate. Clearly this is a one variable statistical problem. Hence the Gaussian mixture model for intensity distribution of prostate can be simply written as

$$p(I | \Phi) = \sum_{i=1}^n \alpha_i p_i(I | \mu_i, \sigma_i), \quad (4.4)$$

where $\Phi = \{\alpha_1, \alpha_2, \dots, \alpha_n, \mu_1, \mu_2, \dots, \mu_n, \sigma_1, \sigma_2, \dots, \sigma_n\}$ is the vector of parameters, I is the vector of intensity values, (μ_i, σ_i) is the mean and standard deviation pair for the i -th Gaussian component, and α_i the mixing weight.

The GMM is built on the sample data taken from the region of interest (ROI), a cube whose centroid is put at the image center with the side length of 1/4 the minimal image dimension.² Although this cube may not be totally inside the prostate, practically it is sufficient to represent the intensity distribution of the prostate as long as the prostate is close to the image center.

Besides, we set up the GMM with three Gaussian distributions in our algorithm. Choosing three Gaussian components for the GMM is based on two considerations: one is that too many components will greatly increase the computation complexity for parameter estimation of the GMM; the other is that from our experience

²Like other parameter settings in our algorithm, the selection of side length is also empirical. The algorithm tests have shown the side length is not a sensitive parameter. We set it to 1/4 the minimal image dimension, so tradeoff between accuracy and speed.

3 Gaussian components are sufficient to approximate the intensity distribution within the prostate. The number of components n set to 3 is for tradeoff between accuracy and speed. More details can be found in the analysis of the sensitivity of the algorithm to the choice of Gaussian components in Section 5.4. Therefore, the GMM for the intensity distribution of prostate is given by

$$p(I | \Phi) = \alpha_1 p_1(I | \mu_1, \sigma_1) + \alpha_2 p_2(I | \mu_2, \sigma_2) + \alpha_3 p_3(I | \mu_3, \sigma_3), \quad (4.5)$$

with the constraint of

$$\sum_{i=1}^3 \alpha_i = 1. \quad (4.6)$$

The issue arises here is how to determine the parameters $\alpha_i, \mu_i, \sigma_i (i = 1, 2, 3)$ which can accurately describe the distribution of the sample data.

4.1.2 Parameter estimation of GMM with EM algorithm

The Expectation-Maximization (EM) algorithm is a general method of finding the maximum-likelihood estimate of the parameters of an underlying distribution from a given data set when the data are incomplete or have missing values [57, 58]. The “missing values” here has two meanings: 1) the collected data indeed have missing values, due to problems with or limitations of the observation process; 2) optimizing the likelihood function is analytically intractable but the likelihood function can be simplified by assuming the existence of additional but *missing* or *hidden* parameters. The EM algorithm is essentially an iterative optimization algorithm which, at least under certain conditions, will converge to parameter values at a local maximum of the likelihood function. Consequently, the density parameter estimation problem is probably one of the most widely used applications of EM algorithm in the computational pattern recognition community [58].

Bilmes presented a very good tutorial of the EM algorithm and its application to parameter estimation in his report [59]. Before we derive our own equations from his work to calculate the GMM parameters for modeling the intensity distribution within the prostate, it is necessary to understand how the EM algorithm works.

Consider a density function $p(x | \Theta)$ that is governed by a set of parameters Θ , we have a data set of size N , supposedly drawn from this distribution, i.e., $X = \{x_1, x_2, \dots, x_N\}$. If we assume that the data vectors are independent and identically distributed with the distribution p , the resulting probability density of the sample is given by:

$$p(X | \Theta) = \prod_{i=1}^N p(x_i | \Theta) = L(\Theta | X). \quad (4.7)$$

This function $L(\Theta | X)$ is called the likelihood function, which gives the likelihood of the parameters given the data. In maximum likelihood (ML) problem, the goal is to find the parameters that maximize the likelihood. That is, we need to find Θ^* where

$$\Theta^* = \operatorname{argmax}_{\Theta} L(\Theta | X). \quad (4.8)$$

It is generally hard to find the parameters analytically that satisfies this requirement. To estimate the parameters, we assume that the observed data X is generated by some distribution. X is called the incomplete data. It is assumed that the complete data $Z = (X, Y)$ exists and they are jointly distributed as:

$$p(z | \Theta) = p(x, y | \Theta) = p(y | x, \Theta)p(x | \Theta). \quad (4.9)$$

Now we can define two likelihood functions — the complete data likelihood and the incomplete data likelihood given by $L(\Theta | Z)$ and $L(\Theta | X)$ respectively. Given

the distribution, the complete data likelihood function is given by:

$$L(\Theta | Z) = L(\Theta | X, Y) = p(X, Y | \Theta). \quad (4.10)$$

Since Y is random and assumed to be governed by some underlying distribution, the likelihood function itself is a random variable. EM algorithm finds the expected value of the complete-data log-likelihood $\log(p(X, Y | \Theta))$ with respect to the unknown data Y given the incomplete data X and current parameter estimates $\Theta^{(i-1)}$ as

$$Q(\Theta, \Theta^{(i-1)}) = E [\log(p(X, Y | \Theta)) | X, \Theta^{(i-1)}]. \quad (4.11)$$

We need to evaluate the expectation and Θ are the new parameters that we optimize to increase Q . The evaluation of this expectation is called the E-step of the EM algorithm.

The second step (the M-step) of the EM algorithm is to maximize the expectation we computed in the first step. That is, we find:

$$\Theta^{(i)} = \operatorname{argmax}_{\Theta} Q(\Theta, \Theta^{(i-1)}). \quad (4.12)$$

These two steps are repeated as necessary and each iteration is guaranteed to increase the log-likelihood and the algorithm is guaranteed to converge to a local maximum of the log-likelihood function.

Recall that the GMM for the intensity distribution of prostate on a given data set I_g of size N (i.e., $I_g = \{I_1, I_2, \dots, I_N\}$) is given by

$$p(I | \Phi) = \sum_{i=1}^3 \alpha_i p_i(I | \mu_i, \sigma_i) = \prod_{j=1}^N \left(\sum_{i=1}^3 \alpha_i p_i(I_j | \mu_i, \sigma_i) \right), \quad (4.13)$$

where $I_j, j = 1, 2, \dots, N$ is assumed to be independent from each other. If we

consider I_g as incomplete, we can think of the full description of each instance I_j as

$$\hat{I}_j = (I_j, y_j) = (I_j, y_{j1}, y_{j2}, y_{j3}), \quad (4.14)$$

where y_j is a class indicator vector (hidden variable):

$$y_{jk} = \begin{cases} 1 & \text{if } I_j \text{ was generated by } k^{th} \text{ component, } k = 1, 2, 3 \\ 0 & \text{otherwise} \end{cases}. \quad (4.15)$$

Then we can write $p(\hat{I}_j | \Phi)$ as

$$p(\hat{I}_j | \Phi) = p(I_j, y_j | \Phi) = p(I_j | y_j, \Phi) p(y_j | \Phi) = \alpha_k p(I_j | \mu_k, \sigma_k) \quad (4.16)$$

if we assume that $y_{jk} = 1$ and $y_{ji} = 0$ for $i \neq k$. Furthermore, we can rewrite Eq. 4.16 as:

$$p(\hat{I}_j | \Phi) = \prod_{i=1}^3 [\alpha_i p(I_j | \mu_i, \sigma_i)]^{y_{ji}}. \quad (4.17)$$

Thus, $p(\hat{I} | \Phi)$ can be expressed as:

$$p(\hat{I} | \Phi) = \prod_{j=1}^N \prod_{i=1}^3 [\alpha_i p(I_j | \mu_i, \sigma_i)]^{y_{ji}}. \quad (4.18)$$

Now we can compute the complete-data log-likelihood function by

$$\begin{aligned} \log p(\hat{I} | \Phi) &= \sum_{j=1}^N \sum_{i=1}^3 y_{ji} \log(\alpha_i p(I_j | \mu_i, \sigma_i)) \\ &= \sum_{j=1}^N \sum_{i=1}^3 y_{ji} \log(\alpha_i) + \sum_{j=1}^N \sum_{i=1}^3 y_{ji} \log(p(I_j | \mu_i, \sigma_i)). \end{aligned} \quad (4.19)$$

Consequently, the expected value of the complete-data log-likelihood $\log p(\hat{I} | \Phi)$ can be taken as

$$\begin{aligned} E \left[\log p(\hat{I} | \Phi) | I, \Phi^t \right] &= \sum_{j=1}^N \sum_{i=1}^3 E(y_{ji}) \log(\alpha_i) + \\ &\quad \sum_{j=1}^N \sum_{i=1}^3 E(y_{ji}) \log(p(I_j | \mu_i, \sigma_i)), \end{aligned} \quad (4.20)$$

where $E(y_{ji})$ is the probability that instance I_j was generated by the i -th component (i.e., $E(y_{ji}) = \sum_k y_{jk} p(y_{jk}) = p(y_{ji}) = p(i | I_j)$):

$$E(y_{ji}) = \frac{\alpha_i^t p(I_j | \mu_i^t, \sigma_i^t)}{\sum_{k=1}^3 \alpha_k^t p(I_j | \mu_k^t, \sigma_k^t)}. \quad (4.21)$$

After the expected value $Q(\Phi, \Phi^t)$ is obtained, we can maximize it subject to the constraint $\sum_{k=1}^3 \alpha_k = 1$ by partially differentiating the following equation.

$$\begin{aligned} Q'(\Phi, \Phi^t) = & \sum_{j=1}^N \sum_{i=1}^3 E(y_{ji}) \log(\alpha_i) + \\ & \sum_{j=1}^N \sum_{i=1}^3 E(y_{ji}) \log(p(I_j | \mu_i, \sigma_i)) + \lambda(1 - \sum_{i=1}^3 \alpha_i), \end{aligned} \quad (4.22)$$

where λ is the Langrange multiplier.

$$\frac{\partial Q'}{\partial \alpha_i} = 0 \Rightarrow \sum_{j=1}^N E(y_{ji}) \frac{1}{\alpha_i} - \lambda = 0 \quad \text{or} \quad \alpha_i^{t+1} = \frac{1}{N} \sum_{j=1}^N E(y_{ji}) \quad (4.23)$$

$$\frac{\partial Q'}{\partial \mu_i} = 0 \Rightarrow \mu_i^{t+1} = \frac{1}{N \alpha_i^{t+1}} \sum_{j=1}^N E(y_{ji}) I_j \quad (4.24)$$

$$\frac{\partial Q'}{\partial \sigma_i} = 0 \Rightarrow \sigma_i^{t+1} = \frac{1}{N \alpha_i^{t+1}} \sum_{j=1}^N E(y_{ji}) (I_j - \mu_i^{t+1})^2 \quad (4.25)$$

Fig. 4.2 summarizes the steps of parameter estimation of GMM (number of components $i = 3$) using EM algorithm in this thesis.

4.1.3 Redesign the speed function F with respect to GMM

Since the calibration or placement of the ultrasound instrument affects the image intensities hence the distribution will be affected accordingly. To illustrate this,

Figure 4.2: Diagram of the steps of parameter estimation of GMM using EM algorithm.

Fig. 4.3 presents two 2D images collected from different frequency (32Hz vs 27Hz) with different zoom rate (R08 vs R10) at the same slice, and the corresponding histogram of the ROI labeled by the red line box within the prostate is given in Fig. 4.4. A direct comparison of the intensity distributions on ROI is shown in Fig. 4.5.

(a)

(b)

Figure 4.3: 2D images collected from different frequency with different zoom rate at the same slice, where the red line box represents the ROI for the calculation of the intensity distribution within the prostate. (a) Image collected with the frequency of 32Hz and the zoom rate of R08, (b) image collected with the frequency of 27Hz and the zoom rate of R10 at the same slice.

(a)

(b)

Figure 4.4: Corresponding histogram of ROI. (a) The histogram calculated from ROI shown in Fig. 4.3(a), (b) the histogram calculated from ROI shown in Fig. 4.3(b).

Clearly, a different configuration of the ultrasound instrument will lead to a different intensity distribution within the prostate. As a consequence, it is too

Figure 4.5: Direct comparison of the intensity distributions shown in Fig. 4.4.

difficult to determine how the intensity distribution depends on the configuration then use a unified statistical model to describe the intensity distribution based on the limited images collected so far. However, the effects of the configuration of the ultrasound instrument on the intensities of the same image are consistent. In other words, the effects of the calibration or placement of the ultrasound instrument on intensities do not change the ultrasound appearance of the prostate (looking from within the prostate gland, the intensities are transitions from dark to light) in every image. Therefore, in this thesis, we analyze the intensity distribution within the prostate for every input image based on the sample data taken from the image center as shown in Fig. 4.6(a), where the prostate is always located in practice. Thus the difficulty in determining how the distribution depends on the calibration or placement of the ultrasound instrument can be avoided.

Fig. 4.6(b) illustrates the Gaussian mixture model (amplified) that describes the intensity PDF inside the prostate shown in Fig. 4.6(a). Note that the probabil-

Figure 4.6: Gaussian mixture model for the intensity distribution of prostate. (a) Data sampling in ROI, (b) Gaussian mixture model (amplified to match the histogram of sample data).

ity density function is amplified so that it can match the histogram of the sample data. Even though the sample data are totally taken from inside the prostate, obviously they contain plenty of noise contributions. In other words, the intensity distribution of the prostate built on the sample data covers a large scale of intensities that might mainly contribute to the background. According to the ultrasound appearance of the prostate, the distribution we calculated must contain most of the intensities that mainly contribute to the prostate. So we can use a threshold to distinguish these intensities from those mainly contributed to the background. Thus another indicator function is designed to determine whether or not the intensity mainly contributes to the prostate, i.e.,

$$P(x, y, z) = \begin{cases} 0 & \text{if } p(I | \Phi) > P_T \\ 1 & \text{otherwise} \end{cases}, \quad (4.26)$$

where P_T is a threshold value defined as follows.

In our previous work [60], the parameter P_T was determined in combination with the GMM parameter μ and σ by

$$P_T = \begin{cases} a \bar{p}(I | \Phi) & \text{if } |\mu_2 - \mu_1| < b \sigma_2 \\ (1 - a) \bar{p}(I | \Phi) & \text{otherwise} \end{cases} \quad (4.27)$$

based on the observation that the third component of GMM is always much lower than the first two components in our algorithm tests, where $\bar{p}(I | \Phi)$ denotes the maximum probability of GMM given in Eq. 4.5, a and b are two constant terms that control the value of P_T . To illustrate this fact, three more GMMs with typical histogram shapes are shown in Fig. 4.7.

Clearly, the value of P_T depends on a and b . And $a = 0.4$, $b = 1.5$ were fixed based on trial and error in [60]. However, from the quantitative evaluation of the algorithm (more details of the evaluation are given in Chapter 5), the average accuracy reaches the maximum at $a = 0.5$. If we substitute $a = 0.5$ in Eq. 4.27, we get $P_T = 0.5 \bar{p}(I | \Phi)$ no matter what value b is set. Thus we can rewrite Eq. 4.27 in a simpler form as

$$P_T = a \bar{p}(I | \Phi), \quad (4.28)$$

where P_T is independent of μ and σ so that the complexity of the algorithm is reduced, and $a = 0.5$ is finally applied.

It is easy to see from Eq. 4.26 that $P(x, y, z) = 0$ means intensity $I(x, y, z)$ mainly contributes to the prostate while $P(x, y, z) = 1$ means the contrary. Thus we can reformulate the speed function F with respect to GMM as

$$F = (1 - 0.5R - 0.5P) F_0 - \epsilon K, \quad (4.29)$$

where functions R and P act as the two constraints which restrict the surface deformation and finally attract it to the desired prostate boundary (surface). Moreover,

(a)

(b)

(c)

Figure 4.7: Three more GMMs with typical histogram shapes over image V1 (a), image V2 (b), and image V4 (c)

from the experiments we found that the following equation can well approximate the optimal R_T selections, that is

$$R_T = \sqrt{6} \sum_{i=1}^3 \alpha_i \sigma_i. \quad (4.30)$$

Because R and P are combined in designing the speed function F , the algorithm sensitivity to R_T selection is reduced. The combination effect of $R + P$ is shown in Fig. 4.8.

Figure 4.8: Combination effect of $R + P$. (a) Ultrasound image of the prostate, (b) binary image of R , (c) binary image of P , (d) binary image of $0.5R + 0.5P$.

From Fig. 4.8 it is easy to see that R and P can compensate each other for boundary gaps, hence the combination effect of $R + P$ not only enhances the prostate boundary information but also greatly reduces the boundary gaps. Consequently, the evolving front driven by the speed function F defined in Eq. 4.29 has the power to remedy the “boundary leaking” problem.

4.2 Adding gradient flow

In the experiments, when the entropy condition ϵ was increased to smoothen the surface further, boundary leaking was hardly avoided in the low contrast areas. This effect is shown in Fig. 4.9 for illustration.

Figure 4.9: Boundary leaking (marked by arrows) in the low contrast areas when the smoothing strength increases before adding the gradient flow.

In fact, to remedy the “boundary leaking” problem, Caselles *et al.*, Malladi

et al. and Yezzi *et al.* have reformulated the level set method in their works by adding another gradient-based term, which is called the gradient vector or gradient flow [61, 62, 63, 64]. Their formulation can be expressed as:

$$\frac{\partial \psi}{\partial t} + k_I (F_0 - \epsilon K) |\nabla \psi| + \nabla k_I \cdot \nabla \psi = 0. \quad (4.31)$$

To understand why the extra gradient-based term $\nabla k_I \cdot \nabla \psi$ can be employed to remedy the “boundary leaking” problem, let us take a look at the following figure, Fig. 4.10.

Figure 4.10: The speed function constraint k_I ($p = 2$) and its gradient . (a) Raw image, (b) the speed function constraint k_I ($p = 2$), (c) magnitude image of ∇k_I , (d) vector flow of ∇k_I .

It is clear to see from Fig. 4.10 that the gradient flow only affects the front evolving in the vicinity of the object boundary. Let us consider the front evolution over a time step Δt in the vicinity of the object boundary, we get this equation

$$\psi_{t+\Delta t} - \psi_t = -\Delta t(k_I (F_0 - \epsilon K)|\nabla\psi| + \nabla k_I \cdot \nabla\psi). \quad (4.32)$$

When the evolving front is trying to penetrate the object boundary to the background, the gradient of the speed function constraint (i.e., ∇k_I) is directed inwards whereas the gradient of the evolving front (i.e., $\nabla\psi$) is directed outwards, thus the extra term $\nabla k_I \cdot \nabla\psi$ has a minus value. Let α denote its absolute value as

$$\alpha = ABS(\nabla k_I \cdot \nabla\psi), \quad (4.33)$$

we get

$$\psi_{t+\Delta t} - \psi_t = -\Delta t(k_I (F_0 - \epsilon K)|\nabla\psi| - \alpha). \quad (4.34)$$

Obviously, the evolution of the front in the vicinity of the object boundary during Δt is getting smaller because of the gradient flow. In other words, the evolving front moves more slowly in the vicinity of the object boundary so that the “boundary leaking” problem could be overcome to some extent in case the object boundary is slightly blurred.

However, this purely gradient-based level set method does not perform well in our project either. Nevertheless, the extra gradient-based term can be added to our algorithm to improve its performance. Certainly, this new term should also be expressed with respect to R and P other than the image gradient. Thus our new method can be formulated as

$$\frac{\partial\psi}{\partial t} + [(1 - 0.5R - 0.5P) F_0 - \epsilon K]|\nabla\psi| - \nabla_D(R + P) \cdot \nabla\psi = 0. \quad (4.35)$$

Because both R and P have only two values (0 and 1) over whole image domain, theoretically $R + P$ is not differentiable and $\nabla(R + P)$ does not exist. However, in practice we can use finite difference to get the discrete analog of the derivative and the operation of finite difference is denoted as ∇_D . In addition, the vector flow of $\nabla_D(R + P)$ is directed outwards as shown in Fig. 4.11, $\nabla_D(R + P) \cdot \nabla\psi$ has a positive value. Therefore we change the sign before the extra term from *plus* to *minus* to make it negative.

Figure 4.11: Vector flow of $\nabla_D(R+P)$. (a) Image of $(R+P)$, (b) vector flow of $\nabla_D(R+P)$.

In the enhanced algorithm, $F_0 = 1$ and $\epsilon = 2$ are selected empirically. From experience, F_0 and ϵ can be selected from a large range of values while keeping the algorithm's results similar, as long as the values are not set too large making the change of level set function overflow the narrow band in every time step.

4.3 “Evolving Bubbles” for 3D prostate surface detection

In earlier attempts on initialization, the procedure was constructed by putting a small sphere as the initial surface at the image center. However, a single level set

may fail to detect all the desired boundaries due to the intensity inhomogeneity inside the prostate. As shown in Fig. 4.12, even though the iteration number is increased, the contours in YZ_Plane and XY_Plane cannot reach the upper-left corners while they have already penetrated the desired boundaries into lower-right corners.

Figure 4.12: A single level set and its final detected result.

Considering that the prostate shape is kidney-like, we can use multiple level sets to optimize the initialization procedure. As shown in Fig. 4.13, we use five small spheres as the initial surfaces to start the detection process. We lower the middle sphere by several grids and move the 4 surrounding spheres a bit up at the same time to keep the centroid of the five spheres close to the image center. As a result, a more accurate surface is obtained.

Note that although one of the initial spheres crosses over the prostate boundaries to the black background (designated by the arrows in Fig. 4.13), as expected,

Figure 4.13: Multiple level sets and their final detected result.

that part shrinks and finally is attracted to the desired boundaries according to the properties of the new speed function F .

So far a fully automatic method have been developed for 3D prostate surface detection. In this algorithm, we assume that the prostate gland is located closely at the image center. Based on this basic assumption, we automate the initialization of zero level sets (spheres) by putting the centroid of initial zero level sets close to the image center and model the intensity PDF of the prostate by sampling data from the image central area. Therefore, if the prostate gland appeared in the 3D image would deviate greatly from the image center or the shape of the prostate is rather asymmetrical, the algorithm might fail to detect the result. The example of failure is illustrated in Fig. 4.14.

The main reason that causes the failure in detection is that the ratio of new voxels (for instance, those farther away from the initial level sets) to the total voxels (most of them consist of the desired surface) on the evolving front is below

Figure 4.14: Limitation of the fully automatic method — failure to detect the whole desired prostate surface. (a) Original image ($252 \times 194 \times 256$), (b) detected surface, (c) 3 sets of contours in 2D cross-sectional images.

the threshold set in the stop criterion so that the deformation stops accordingly. This drawback cannot be overcome by setting the threshold in stop criterion to a very small value, because in that case the deformation might never stop due to the large image noise. However, the performance of the algorithm can be improved via initializing the level sets by user according to the image property.

To do this, a friendly interface is provided to let the user specify the level sets in 5 particular fixed slices, which are spaced evenly in transverse view. Fig. 4.15 demonstrates the 5 fixed slices for level-sets generation.

Figure 4.15: 5 evenly spaced slices in transverse view used for level-sets generation.

However, the first slice and the last slice usually do not clearly contain the prostate content. In that event, they could be discarded directly (see Fig. 4.16) or replaced by other slices (see Fig. 4.17). Concretely, in Fig. 4.16, 4 of 5 slices are used to initialize the level sets whereas the fifth fixed slice does not contain any prostate content thus is discarded directly; in Fig. 4.17, the first fixed slice (slice 42) for level-sets generation is replaced by the middle slice (slice 63) between the first and the second fixed slices (slice 84) due to the lack of prostate information, and the last slice is also discarded directly for the same reason as in Fig. 4.16.³ Observing that the prostate shape is kidney-like, we can place at most 3 level-sets in such kind of slices as shown in Fig. 4.16(b) and Fig. 4.16(c).

³Likewise, the last fixed slice for level set generation can be replaced by the middle slice between the fourth and the fifth fixed slices.

Figure 4.16: User defined initial level sets in transverse view for the image shown in Fig. 4.14. (a) 2 level sets at slice 42, (b) 3 level sets at slice 84, (c) 3 level sets at slice 126, (d) 1 level set at slice 168.

Figure 4.17: Another example of user specification of initial level sets. (a) 3 level sets at slice 63, (b) 3 level sets at slice 84, (c) 3 level sets at slice 126, (d) 2 level sets at slice 168.

To ensure that the initial level sets are totally inside the prostate gland, when initializing each level set, the corresponding contours in both sagittal view and coronal view are displayed as well for the references as shown in Fig. 4.18. Hence the user can modify the level set position and radius in case it is placed somewhat outside the prostate. Once the level sets are well defined, they expand, shrink, merge and split like bubbles during the front propagation and finally are driven to the desired prostate surface by the new speed function designed in Eq. 4.29. Fig. 4.19 demonstrates the merging effect of the evolving bubbles.

Figure 4.18: Three views of a level set in level set generation. (a) Transverse view, (b) sagittal view related to the vertical red dotted line, (c) coronal view related to the horizontal red dotted line.

Figure 4.19: Merging of three bubbles when evolving with constant speed along normal direction.

To indicate the behavior of this deformable model, we call it “Evolving Bubbles”. For comparison, the final result detected from the image in Fig. 4.14 by “Evolving Bubbles” is shown here in Fig. 4.20. It is easy to see that the result is significantly improved with the evolving bubbles (see Fig. 4.16) specified by the user according to the image property: the bubbles capture the whole prostate surface and the contours in cross-sectional images match the references much better.

Figure 4.20: Greatly improved result obtained by the user-defined bubbles shown in Fig. 4.16. (a) Original image ($252 \times 194 \times 256$), (b) detected surface, (c) 3 sets of contours in 2D cross-sectional images.

4.4 A faster algorithm

As described in Section 3.2.2, the narrow band solution improves the speed of the algorithm dramatically when compared with the full size method. However, the computation time increases rapidly with the number of grid points which reside in the narrow band and the narrow band rebuilding at each iteration takes up most of the processing time. As highlighted in Fig. 4.21, every run of the algorithm to completely detect the 3D prostate surface takes more than 60 minutes (note that the algorithm is programmed using c++ and runs on a PC with 1.8GH CPU and 512MB memory).

Figure 4.21: An example of the experimental results obtained by our algorithm with the narrow band solution. Image size: $256 \times 256 \times 256$, **iteration number: 2538**, **elapsed time: 68 min**.

This obviously restrains its practical use in clinical activities. One way to speed up the algorithm further is to reduce the number of voxels in the narrow band or the

number of iterations or both of them. In this work, the narrow band size is reduced simply by compressing the original image into 1/8 of it, that is, rearranging the image data by sampling them every other slice along x, y, z direction respectively.

Fig. 4.22 illustrates the data sampling mechanism.

Figure 4.22: Illustration of image data sampling. (a) Voxels concerned with the computation in the original image, (b) image data sampling by every other slice along x, y, z direction respectively.

After image down sampling, we apply the narrow band solution to start the surface detection procedure. Through this way, the voxels concerned with the computation are reduced significantly. Moreover, the necessary iteration number for deforming the evolving front to the desired prostate surface is reduced accordingly. As expected, this fast algorithm of level set method reduces the computation time to at least 1/8 of that needed by the narrow band solution alone. Fig. 4.23 demonstrates the experimental results before and after image down sampling. A direct comparison of these two sets of results is shown in Fig. 4.24 by putting them together.

It can be seen from Fig. 4.24 that these two sets of detected results are quite similar to each other. And they are both in a good agreement (from the clinicians'

Figure 4.23: Experimental results before and after image down sampling. (a) The detected result before image down sampling, iteration number: 2538, elapsed time: 68'04'', (b) the detected result after image down sampling, iteration number: 927, elapsed time: 4'34''.

Figure 4.24: Comparison of the experimental results before and after image down sampling. *Yellow line*: detected result before image down sampling, *red line*: detected result after image down sampling.

point of view) with the reference as shown in Fig. 4.23, while the run time is reduced to 4.5 minutes from 68 minutes. In this thesis, we applied this fast algorithm to

seven other 3D TRUS images to test its effectiveness. Among them, two of the detected results are shown in Fig. 4.25. Evaluated by the experts in vision, the detected results are similarly satisfactory.

Figure 4.25: Two other examples of detected results. (a) Image size: $252 \times 194 \times 256$, iteration number: 1243, elapsed time: 5'01'', (b) image size: $250 \times 174 \times 236$, iteration number: 1001, elapsed time: 3'51''.

The run time of each algorithm test is summarized in Table 4.1.

As we know, the prostate boundary is *de facto* a transition from dark to light (looking from within the prostate gland). Therefore, after rearranging the image data by sampling them every other slice along x , y , z direction respectively, the compressed image still contains enough boundary information so that it will not affect the final detected result too much. To measure the effects on the accuracy of results caused by such reduction in image size, the two sets of results obtained by the narrow band solution and the faster algorithm are compared with the reference

Table 4.1: Comparison of the run time between narrow band solution and the faster algorithm

3D TRUS image	Run Time	
	narrow band solution	fast algorithm
Image V0 ($250 \times 174 \times 236$)	49'23"	3'51"
Image V1 ($256 \times 256 \times 256$)	68'04"	4'34"
Image V2 ($252 \times 194 \times 256$)	72'48"	5'01"
Image V3 ($252 \times 194 \times 256$)	75'09"	5'03"
Image V4 ($252 \times 194 \times 256$)	70'31"	4'53"
Image V5 ($256 \times 256 \times 256$)	65'15"	4'32"
Image V6 ($212 \times 192 \times 248$)	41'03"	3'38"
Image V7 ($256 \times 256 \times 256$)	60'27"	4'25"

respectively by using the validation method proposed in Chapter 5, and the results of comparison are shown in Fig. 4.26.

Fig. 4.26 has clearly shown us that there is only a slight difference in the accuracy between the narrow band solution and the faster algorithm. However, if we increase the image down-sampling ratio, for instance, 1:3 along each direction, some of the prostate boundary information might be lost in the reduced image hence large boundary gaps could be caused since this down-sampling scheme inevitably results in the loss of information. This will lead to a serious boundary-leaking problem that we have made great efforts to overcome. Consequently, the down-sampling rate is fixed to 1:2 along each direction in this thesis.

The experimental evidence has shown that after down sampling, the detected results do not tend to leak from boundaries as much as the results do before down sampling. An example can be seen from the bottom-left of the corresponding

Figure 4.26: Accuracy comparison between the narrow band solution and the faster algorithm.

contours in the first sub-figure in Fig. 4.24. Therefore, the proposed fast algorithm not only reduces the run time dramatically, but also reduces the risk of “boundary leaking” problem.

4.5 Summary

This chapter describes the development of the novel and practical model, “Evolving Bubbles”, for 3D prostate surface detection.

To address the problem that the result accuracy is sensitive to the value of R_T , which depends on the user selection, a Gaussian mixture model (GMM) is introduced to model the intensity distribution inside the prostate gland. The

Gaussian mixture model is made up of three components to tradeoff between accuracy and speed, and built on the sample data taken from the region of interest, a cube whose centroid is put at the image center with the side length of 1/4 the minimal image dimension. Parameters for GMM are automatically determined by Expectation-Maximization (EM) algorithm. Since GMM models the intensity distribution inside the prostate, another indicator function $P(I)$ is designed to determine the intensity main contribution. Meanwhile, the optimal R_T selection is approximated by an equation derived from the parameters of GMM. By combining the statistical information of the intensity distribution and region information, the speed function F is reformulated as

$$F = (1 - 0.5R - 0.5P) F_0 - \epsilon K,$$

where R and P constrains the front deformation to the desired prostate surface.

To further remedy the “boundary leaking” problem caused by increasing the entropy condition ϵ for smoothening the prostate surface, gradient flow based on $R + P$ is added to the level set method. The extra gradient-based term acts as a pulling back force to prevent the propagating front from penetrating the prostate surface.

Once the evolving bubbles are specified by the user according to the image property, they expand, shrink, merge and split during the front propagation and finally are driven to the desired prostate surface by the new speed function. While the statistical distribution model successfully prevents the evolving front from penetrating into the dark areas caused by artifacts, the gradient flow attracts the front back to the vicinity of the desired prostate surface in the low contrast areas.

To improve the speed of the algorithm, a down-sampling scheme is applied. In

this work, the image data are sampled every other slice along x , y , z direction respectively. Obviously, the image data involved in computation are reduced to $1/8$ of the original image. Thus the narrow band size and the number of iterations are both reduced. As a result, the speed of the algorithm is improved dramatically.

CHAPTER 5

Experimental Results and Evaluation

Key issues concerned with the segmentation of medical images are validity and reliability. Any large disagreement between the detected results and the real targets might result in severe damages in clinical activities. For this reason, a very important task for us is to evaluate the algorithm performance before the detected results are used in clinical decision for the diagnoses and treatments of prostate diseases. This chapter sets up the evaluation criteria and carries out the algorithm verification with experimental results.

5.1 Validity evaluation

Validity refers to the degree to which a study accurately reflects the specific concept that the researcher is attempting to measure. Simply speaking, validity means accuracy. To evaluate the validity, usually a comparison of a new measure against an existing gold standard is conducted.

The accuracy of computer-assisted segmentation of medical images has been difficult to quantify in the absence of a “ground truth” segmentation for clinical data. Traditionally, there are two mechanisms for validation if no ground truth data are provided.

1. Create synthetic images with known size and shapes. Segment them with the proposed method and contrast the values to those used when generating

the image.

2. Recruit human operators with strong background in anatomy. Train them in manual segmentation techniques and convince them to pass hours in front of the screen segmenting a set of defined volumes (most of the time they need to repeat the segmentation several times, in order to evaluate their variability). Then compare the segmentations done with the proposed method against their manual segmentations.

Although physical or digital phantoms can provide a level of known “ground truth” [65, 66], they have so far been unable to reproduce the full range of imaging characteristics (partial volume effect, intensity inhomogeneity, artifact, etc.), normal and abnormal anatomical variability observed in clinical data. Interactive drawing of the desired segmentation by domain experts has often been the only acceptable approach, though it suffers from intra-expert and inter-expert variability.

5.1.1 Previous works

To evaluate the efficiency of the prostate boundary detection algorithms, various methods have been proposed in the literature [9, 24, 25, 26, 27, 30, 31].

Pathak *et al.* [9] used two error metrics to measure the average disagreement between multiple observers’ segmentation (inter-observer disagreement [67]) when edge guidance is used compared with when no such guidance is present, and the average disagreement between computer-aided outlining and fully manual outlining compared with the average inter-observer disagreement during fully manual outlining. In general, a metric $e(A, B)$ can be used to measure the difference between the two contours, specified as sets of points, $A = \{a_1, a_2, \dots, a_m\}$ and

$B = \{b_1, b_2, \dots, b_n\}$, where each a_i or b_i is a point on the corresponding contour. The distance to the closest point (DCP) for a_i to the contour B is defined as $d(a_i, B) = \min_j \|b_j - a_i\|$, where $\|\cdot\|$ is the 2D Euclidean distance between any two points. The distance between the contour A and a point on B is similarly defined. In their method, the two error metrics used to determine the inter-observer variability are the Hausdorff distance [67, 68] and the mean absolute distance [69]. The Hausdorff distance (e_H) is defined as the maximum DCP over all points on the two contours whereas the mean absolute distance (e_M) is the average of the DCPs over the two contours. While the Hausdorff distance measures the worst possible disagreement between the two observers' outlines, the mean absolute distance estimates the disagreement averaged over the two contours. Their assessment showed that the automatic and semi-automatic segmentation of the prostate indeed leads to a better consistency.

On the other hand, R.Y. Wu *et al.* [30] used fractional area [70] to estimate the disagreement between two contours as shown in Fig. 5.1 by

$$D = \frac{2A_{XOR}}{A_1 + A_2}, \quad (5.1)$$

where D stands for the disagreement level, A_{XOR} is the fractional area, A_1 and A_2 are the areas closed by the two contours respectively. To estimate the accuracy of the algorithm, they set up an assessment scheme as shown in Fig. 5.2. The average disagreement between the outline of computer and the outlines of expert observers is $D_0 = (D_2 + D_3)/2$, where D_2 and D_3 denote the outline disagreement between the computer and expert 1, expert 2 respectively. Then D_0 is compared with D_1 , the disagreement between two experts, to determine if the computer aided outlining agrees with the results provided by the experts. Apparently, this

assessment scheme is a special case of the joint agreement study in [67].

Figure 5.1: Fractional area A_{XOR} — The grey area between two contours.

Figure 5.2: Assessment scheme — D_1 , D_2 , D_3 represent the comparison results (i.e., disagreement between the two corresponding contours being compared).

As illustrated in Fig. 5.3, Ladak *et al.* [25] used both distance-based [67] and area-based [71] metrics to compare semi-automatically outlined boundaries against manually outlined boundaries. The distance-based metrics are computed by first choosing the center of gravity (CG) of the manually outlined boundary as a reference point, and then finding corresponding vertices on the two boundaries, as described in [72]. To obtain the difference between any two corresponding vertices,

the difference in the distances of the vertices from the reference point is calculated. The difference, $d_j(\theta_i)$, between the two boundaries extracted from image j was computed as a function of angle θ_i (Fig. 5.3(a)) where $i = 1, 2, \dots, N$ and N is the total number of angles. Based on $d_j(\theta_i)$, three quantities (mean difference, mean absolute difference and maximum difference) describing the difference between the two sets of boundaries are calculated to determine the agreement between the algorithm and the experts. Meanwhile, three different areas (Fig. 5.3(b)) defined by the two corresponding outlines, i.e., true positive (TP) area: the common region between the manual outline and the algorithm outline, false positive (FP) area: the region enclosed by the algorithm outline but outside of the manual outline, and false negative (FN) area: the region enclosed by the manual outline that is missed by the algorithm, are also used to estimate the accuracy further.

Figure 5.3: Metrics for comparing semi-automatically outlined boundaries (dotted) against manually outlined boundaries (solid). (a) Local difference, $d_j(\theta_i)$, (b) area difference.

Similarly, besides measuring the average distance of the model and manual contours, Ghanei *et al.* [31] employed a similarity measure, which is based on

Kappa statistics to compare the model results with the manual segmentation. Suppose A_1 and A_2 are the areas (in pixels) obtained by the manual and automatic methods, respectively, the relative amount of agreement (similarity value) and disagreement between the model and manual segmentation can be computed for each slice by R_c and R_n as

$$R_c = \frac{A_1 \text{ AND } A_2}{A_1 \text{ OR } A_2}, \quad (5.2)$$

$$R_n = \frac{(A_1 \text{ OR } A_2) - (A_1 \text{ AND } A_2)}{A_1 \text{ OR } A_2} = 1 - R_c, \quad (5.3)$$

where R_c and R_n are the ratio of the common pixels (marked as prostate by both methods) and non-common pixels (marked only by one method) to the total area, respectively.

In [24], Pathak *et al.* used planimetric volumetry [10] to calculate the prostate volume after the boundaries were identified. These volumes are then compared against those manually measured by 3 experts. As addressed in [9], comparing volumes is not a stringent evaluation criterion since it is a cumulative function of the delineations made on each image over the entire prostate volume. Even very different outlines made by two observers could result in the same volume estimates. Therefore, contour comparison is still indispensable for 3D segmentation assessment. In their work, Knoll *et al.* [26, 27] used three complementary error measures for validation.

- The volume difference (VD), which is defined as the number of different voxels between the manual segmented reference volume (RV) and the automatic

segmented volume (AV) of the whole scan-set of a patient respect to RV :

$$VD = \frac{(RV \cup AV) - (RV \cap AV)}{2RV}. \quad (5.4)$$

- The contour difference (CD), which is calculated as the mean distance of the manually drawn contour pixel to its nearest automatically detected contour pixel.
- The absolute volume difference (AVD), which is simply the relative difference between the two volumes

$$AVD = \frac{|RV - AV|}{RV}. \quad (5.5)$$

Obviously, all these evaluation methods need the experts' manually outlined boundaries as references. However, as a large number of clinical data sets are involved and each data set will be outlined by one or several observers, the collection of such manually outlined boundaries become tedious and time consuming. Aiming to provide a convenient channel to collect such kind of data, Wu *et al.* [73] developed a web-based evaluation system for boundary detection algorithms. Through this system, registered experts can manually outline the boundaries from anywhere at any time as long as they have access to the internet.

5.1.2 Validation metrics

However, the most appropriate way to carry out the comparison of an automated segmentation to a group of expert segmentations is so far unclear [74]. To validate our algorithm, both volume disagreement and contour disagreement are used in this thesis. The following lists the metrics used for validation of our algorithm.

Volume difference

As described in Knoll *et al.*'s work [26, 27], we use the volume difference as one of the metrics to measure the disagreement between our algorithm and the experts' opinion. However, the calculation for the volume difference (VD) and the absolute volume difference (AVD) is slightly different from Eq. 5.4 and Eq. 5.5, and is expressed as follows.

$$VD = \frac{(RV \cup AV) - (RV \cap AV)}{RV \cup AV}, \quad (5.6)$$

$$AVD = \frac{|RV - AV|}{RV \cup AV}. \quad (5.7)$$

Volumetric overlap

In [25], Ladak *et al.* used overlap areas defined by two corresponding boundaries to tell the disagreement in detail. Inspired by this, the overlap volumes can be used to describe the details of volume difference. Similarly, we define the overlap volumes as follows.

- True positive (TP) volume: the common voxels contained by both the manually outlined surface and the algorithm detected surface.
- False positive (FP) volume: the volume enclosed by the algorithm detected surface but outside of the manually outlined surface.
- False negative (FN) volume: the volume enclosed by the manually outlined surface that is missed by the algorithm.

Refer to Eq. 5.6, clearly we can get

$$RV \cup AV = TP + FP + FN \quad \text{and} \quad RV \cap AV = TP. \quad (5.8)$$

Thus we can rewrite Eq. 5.6 and Eq. 5.7 as:

$$VD = \frac{FP + FN}{TP + FP + FN}, \quad (5.9)$$

and

$$AVD = \frac{|RV - AV|}{TP + FP + FN}. \quad (5.10)$$

Maximum surface distance (Hausdorff distance)

The Hausdorff-Chebyshev metric defines the largest difference between two contours and is well explained in Section 5.1.1. Likewise, we can define the Hausdorff distance between two surfaces, which describes the largest difference between them.

Let us first define the distance $d(p, S')$ between a point p belonging to a surface S and a surface S' as:

$$d(p, S') = \min_{p' \in S'} \|p - p'\|_3, \quad (5.11)$$

where $\|\cdot\|_3$ denotes the usual Euclidean norm in 3D space. From this definition, the Hausdorff distance between S and S' , denoted by $d(S, S')$ is given by

$$d(S, S') = \max_{p \in S} d(p, S'). \quad (5.12)$$

It is important to note that distance is in general not symmetrical, i.e., $d(S, S') \neq d(S', S)$. However, we can define a *symmetrical Hausdorff distance*, denoted by $d_s(S, S')$, by maximizing these two “one-sided” errors. That is,

$$d_s(S, S') = \max(d(S, S'), d(S', S)). \quad (5.13)$$

The symmetrical distance, also the maximum surface distance (MSD), provides a more accurate measurement of the difference between two surfaces, since the “one-sided” error can lead to significantly underestimated distance values, as illustrated in Fig. 5.4.

Figure 5.4: One-sided error. In this case, $d(S, S')$ will remain much smaller than $d(S', S)$, since here $d(A, S') \ll d(B, S)$. Thus a small one-sided distance does not imply a small distortion.

Mean absolute surface distance

The mean absolute surface distance tells us how much on average the two surfaces differ. From the point-to-surface distance defined in Eq. 5.11, we can define a mean error d_m between two surfaces S and S' :

$$d_m(S, S') = \frac{1}{|S|} \int \int_{p \in S} d(p, S') dS, \quad (5.14)$$

where $|S|$ denotes the area of S . For the same reason of asymmetry as addressed before, we need to define a symmetrical version of the mean error as the mean

absolute surface distance (MASD). From Eq. 5.13 we get

$$MASD = d_sm(S, S') = \max(d_m(S, S'), d_m(S', S)). \quad (5.15)$$

5.1.3 Implementation

Reference construction

The reference surface is produced by **Bubble 3D**, a software developed by R.Y. Wu [34] to reconstruct the surface from 2D contours using NURBS [75] (for more information about **Bubble 3D**, please refer to Appendix A). Although **Bubble 3D** provides a very powerful tool to produce the reference surfaces, outlining the boundaries slice by slice for constructing them is still time-consuming and arduous. It is very hard to ask the experts to draw all the references we needed. In this thesis, we first train 3 people under the instruction of experts, then the 3 people separately draw the references 3 times with the interval of one week. Once the references are produced, they are shown to the experts and manipulated by the experts if in need. This process may bring about some errors to the accuracy. However, because the other 2 orthogonal planes (see Fig. 5.5) are concerned to determine whether the outline is correct or not, the control points of the outline can be adjusted to more accurate positions. Therefore, we have the confidence that the references drawn by the trained people and further manipulated by the experts are reasonable.

After adequate parallel NURBS curves are manually outlined, a NURBS surface of the prostate is then reconstructed correspondingly as show in Fig. 5.6. Since more or less there are variations between expert opinions as mentioned before, to reduce the inter-observer variability and intra-observer variability in these surfaces,

Figure 5.5: Boundary outlining using **Bubble 3D**: the red curve in the right figure is the outline, whose cross points on the 2 orthogonal planes along the lines are shown in the left figures.

Figure 5.6: An example of 3D prostate surface reconstruction using **Bubble 3D**. (a) Parallel manually outlined NURBS curves, (b) reconstructed 3D surface.

the mean surface constructed from these 9 reconstructed NURBS surfaces is used as the final reference in this thesis.

The strategy to produce a mean surface by averaging the 9 reconstructed NURBS surfaces is straightforward. Since the coordinates of every point on the NURBS surface can be obtained when the surface is reconstructed, we can get 9 NURBS curves superposed each other on one slice if we put these surfaces together. Thus a mean NURBS curve can be obtained on this slice by averaging the 9 superposed NURBS curves. To give a demonstration in simplicity, Fig. 5.7 uses 3 superposed NURBS curves to calculate the mean NURBS curve based on the following steps.

1. Sample the 3 superposed NURBS curves with the same rule. For example, in this particular case, 100 points are linearly spaced on each curve as shown in Fig. 5.7(c).
2. Determine the center of gravity (CG) from one of the curve samples. As illustrated in Fig. 5.7(d), CG is determined from the yellow line.
3. Determine how many control points needed to obtain the mean NURBS curve. After which average the radii around the specified angles to get the control points as shown in Fig. 5.7(e).
4. Produce the mean NURBS curve from the calculated control points.

Consequently, after adequate parallel mean NURBS curves are produced, the mean NURBS surface of the prostate is similarly reconstructed as the final reference.

Figure 5.7: Calculation of the mean NURBS curve. (a) One slice of image V1, (b) 3 superposed curves obtained from the reference surfaces, (c) linearly spaced points on the curves, (d) the center of gravity (CG) determined from the yellow line, (e) averaging the radii around the specified angles to get the control points, (d) the mean NURBS curve (red solid line) obtained from the control points (red circle).

Measure of surface distance

The surface distances, i.e., maximum surface distance and mean absolute surface distance, are calculated by a software called **MESH** [76], which can be freely downloaded from <http://mesh.epfl.ch/>. MESH is an efficient tool that measures distortion between two discrete surfaces (triangular meshes) using the Hausdorff distance. However, it does not provide any approach that measures the error of volume between the two surfaces. The calculation of volume difference and volume overlap is then integrated into our algorithm.

Measure of volume error

It seems that the voxels inside the zero level set are easy to determine if we use Eq. 3.1, since those voxels that have negative level set function are obviously inside the final detected surface. Unfortunately we cannot directly label the voxels enclosed by the detected surface, because we implement the level set algorithm using a narrow band solution (see Section 3.2.2). Nevertheless, once the surface is obtained, we can generate the volume by the following scheme.

- *Algorithm detected volume generation*

Suppose we know the intensity value (for example, 255¹) of the voxels lying on the surface, we can scan the contour in the 2D slice line by line as demonstrated in Fig. 5.8. Then we can get the pairs in each line which record the coordinates of the voxels on the contour. After that, we set the intensity value of all voxels

¹In fact, the intensity value of voxels lying on the surface is set to 255 in our algorithm. From our experience, the value 255 is not used by the raw ultrasound images. Hence this setting can exclude the error caused by those voxels that has the same value as the setting when we scan the contours in the 2D slices.

between the pair to 255 and thus the area enclosed by the contour is generated. By repeating the above scanning in every slice, we can generate the volume as a result.

Figure 5.8: Area generation by scanning the contour line by line. (a) Scan the contour to get the pairs which record the coordinates of the pixel on the contour, (b) set the intensity value of all pixels between the pair to 255 to generate the area.

From Fig. 5.8(a), we can see that there is one pair in line m and there are 2 pairs in line n . No matter what shape the object has, this scanning scheme can obtain the needed pairs to generate the volume.

- *Reference volume generation*

The output of the reference is a triangularized mesh stored in STL binary format (details of STL format are addressed in Appendix B). It is not trivial to estimate the volume enclosed by the triangularized mesh. In this algorithm, we first find out all the voxels that lie on the surface defined by the triangles, then use

the same scanning scheme as described in the previous section to label the voxels inside of the surface.

To determine if a given point V lies in the interior of a given triangle, let us consider an individual vertex, denoted V_0 , and let $\vec{v}_1$ and $\vec{v}_2$ be the vectors from V_0 to the other two vertices as shown in Fig. 5.9. Expressing the vector from V_0 to V in terms of $\vec{v}_1$ and $\vec{v}_2$ gives

Figure 5.9: Relationship between a given point, V , and a given triangle. (a) Point outside triangle, (b) point inside triangle.

$$V - V_0 = a\vec{v}_1 + b\vec{v}_2, \quad (5.16)$$

where a and b are constants. Solving for a and b gives

$$a = \frac{\det((V - V_0), \vec{v}_2)}{\det(\vec{v}_1, \vec{v}_2)}, \quad (5.17)$$

and

$$b = -\frac{\det((V - V_0), \vec{v}_1)}{\det(\vec{v}_1, \vec{v}_2)}, \quad (5.18)$$

where

$$\det(\vec{u}, \vec{v}) \triangleq \mathbf{k} \cdot (\vec{u} \times \vec{v}) = u_x v_y - u_y v_x. \quad (5.19)$$

The point V will be “inside” the angle formed at V_0 if $a, b > 0$, and so will be in the interior of the triangle if the corresponding $a, b > 0$ for each of the three vertices.

For each given triangle from the triangularized mesh, it is not necessary to determine all the relationships between voxels of the image and it. Instead, we just concern about those voxels contained by the cube defined by the triangle vertices V_1 , V_2 and V_3 as illustrated in Fig. 5.10, where $\min X$, $\min Y$, $\min Z$, $\max X$, $\max Y$, $\max Z$ are obtained from the following definitions.

$$\min X = \min(V_{1x}, V_{2x}, V_{3x}), \quad \max X = \max(V_{1x}, V_{2x}, V_{3x}),$$

$$\min Y = \min(V_{1y}, V_{2y}, V_{3y}), \quad \max Y = \max(V_{1y}, V_{2y}, V_{3y}),$$

$$\min Z = \min(V_{1z}, V_{2z}, V_{3z}), \quad \max Z = \max(V_{1z}, V_{2z}, V_{3z}).$$

Figure 5.10: A cube defined by the coordinates of the triangle vertices.

After both algorithm detected volume and reference volume are generated, the comparison between them can be carried out by calculating the volume difference and volume overlap as addressed in previous section.

5.2 Reliability evaluation

Reliability refers to the amount of agreement between the results of repeated measurements, or assessments, of the same sample. In other words, reliability means the consistency of results. For algorithm evaluation, the following two types of reliability are commonly applied.

- **Test-retest reliability**, which measures *stability over time*, is used to assess the consistency of a measure from one time to another. The appropriate length of the interval depends on the stability of the variables which causally determine what is measured. A year might be too long for an opinion item but appropriate for a physiological measure.
- **Inter-rater reliability**, which measures *homogeneity*, is used to assess the degree to which different raters/observers give consistent estimates of the same phenomenon.

To evaluate the reliability of the algorithm, *intraclass correlation coefficient* (ICC) is used to measure the above two types of reliability [77]. Although Pearson correlation coefficient may be used to assess test-retest reliability, ICC is preferred when sample size is small (< 15) or when there are more than two tests (one test, one retest) to be correlated [78].

Conceptionally, ICC can be defined as the ratio of between-groups variance to total variance. For illustration: Let A be the true variance in subjects' ratings due to the normal expectation that different subjects will have true different scores on the rating variable. Let B be the error variance in subjects' ratings attributable to rater unreliability. The intent of ICC is to form the ratio, $ICC = A/(A + B)$. That is, intraclass correlation is to be true inter-subject variance as a percent of total variance, where total variance is true variance plus variance attributable to rater error in classification. Both A and B are estimated based on mean squares obtained by applying analysis of variance (ANOVA) models [79].

In their important paper, Shrout and Fleiss [77] described six forms of ICC for reliability according to the study design. In order to select an appropriate ICC, the first decision that must be made is whether the data are to be treated via a one way or a two way ANOVA model. In this thesis, to measure the test-retest reliability, 4 images are randomly selected to test the algorithm and the test is repeated 3 times by one person with the interval of one week. On the other hand, to measure the inter-rater reliability, 3 people are selected to test the algorithm with the given rule: cover the areas as largely as possible while ensure all the bubbles are inside the prostate. Referring to Nichols' suggestion [80], we use $ICC(1, k)$ and $ICC(3, k)$ to measure the inter-rater reliability and the test-retest reliability, respectively [81]. In formulation,

$$ICC(1, k) = \frac{bMS - wMS}{bMS}, \quad (5.20)$$

and

$$ICC(3, k) = \frac{bMS - eMS}{bMS}, \quad (5.21)$$

where

bMS : between target mean square, wMS : within target mean square,

eMS : error mean square, k : number of raters/tests.

Given that X is the variable of test, by applying ANOVA [79] we can get

$$bMS = \frac{1}{N-1} \left[k \cdot \sum_{i=1}^N (\bar{x}_i - \bar{x})^2 \right], \quad (5.22)$$

$$wMS = \frac{1}{N(k-1)} \sum_{i=1}^N \sum_{j=1}^k (x_{ij} - \bar{x}_i)^2, \quad (5.23)$$

and

$$eMS = \frac{1}{(N-1)(k-1)} \left[\sum_{i=1}^N \sum_{j=1}^k (x_{ij} - \bar{x}_i)^2 - N \cdot \sum_{j=1}^k (\bar{x}_j - \bar{x})^2 \right], \quad (5.24)$$

where

N = number of images being tested,

$\bar{x}_i$ = mean score for image i ,

$\bar{x}$ = overall mean of scores for whole sample,

x_{ij} = score of test j for image i ,

$\bar{x}_j$ = mean score for test j .

5.3 Experimental results

5.3.1 Prostate surface detection from 3D TRUS

The proposed model, “Evolving Bubbles”, is now applied to eight 3D prostate scans to test its validity and reliability. Detected surfaces of 3 typical prostate scans are illustrated below one by one, followed by the error measurement. To highlight the

difference between detected surface and reference, the detected surface is shown in dark red whereas the reference is shown in dark green. In the three orthogonal 2D slices, the grey area denotes the common voxels (TP); the white area and the black area around it stand for the voxels enclosed by the algorithm detected surface but outside the manually outlined one (i.e., FP), and the voxels enclosed by the manually outlined surface that is missed by the algorithm (i.e., FN) respectively.

Fig. 5.11 demonstrates one of the results detected by the algorithm, where the image size is $250 \times 174 \times 236$. The volume difference is illustrated in Fig. 5.12, and quantified as follows.

Figure 5.11: Detected result of image $V0$. (a) Raw image and 3 orthogonal 2D slices, (b) detected surface and corresponding contours on the 3 slices.

RV	AV	TP	FP	FN	$RV \cup AV$
2230737	2043363	1954354	89009	276383	2319746

(a)

(b)

(c)

(d)

Figure 5.12: Volume difference of image V_0 . (a) Reference surface and detected surface, (b) volume difference in middle slices, (c) volume difference in one end, (d) volume difference in the other end.

$$VD = \frac{FP + FN}{TP + FP + FN} = \frac{89009 + 276383}{2319746} = 15.75\%,$$

$$AVD = \frac{|RV - AV|}{TP + FP + FN} = \frac{|2230737 - 2043363|}{2319746} = 8.08\%.$$

The surface distance errors calculated by **MESH** are illustrated in Fig. 5.13, with $MSD = 4.044mm$ and $MASD = 0.688mm$. Note that in **MESH**, the color value (from dark blue to bright red) maps the scale of surface distance (from small to large), and the most left bar chart displays the distance distribution on the surface.

Figure 5.13: Surface distance of image $V0$, from detected surface to reference.

Fig. 5.14 gives another example of the results detected by our algorithm, where the image size is $256 \times 256 \times 256$, and the prostate organ resides very close to

the image center and its shape is quite symmetrical. The volume difference is illustrated in Fig. 5.15 and quantified as follows.

Figure 5.14: Detected result of image V1. (a) Raw image and 3 orthogonal 2D slices, (b) detected surface and corresponding contours on the 3 slices.

RV	AV	TP	FP	FN	$RV \cup AV$
2456637	2463015	2313754	149261	142883	2605898

$$VD = \frac{FP + FN}{TP + FP + FN} = \frac{149261 + 142883}{2605898} = 11.21\%,$$

$$AVD = \frac{|RV - AV|}{TP + FP + FN} = \frac{|2456637 - 2463015|}{2605898} = 0.24\%.$$

The surface distance errors calculated by **MESH** are shown in Fig. 5.16 with $MSD = 2.757mm$ and $MASD = 0.524mm$.

(a)

(b)

(c)

(d)

Figure 5.15: Volume difference of image V1. (a) Reference surface and detected surface, (b) volume difference in middle slices, (c) volume difference in one end, (d) volume difference in the other end.

Figure 5.16: Surface distance of image V1, from detected surface to reference.

Figure 5.17: Detected result of image V2. (a) Raw image and 3 orthogonal 2D slices, (b) detected surface and corresponding contours on the 3 slices.

The third example of detected results is shown in Fig. 5.17, where the image size is $252 \times 194 \times 256$, and the prostate locates a certain distance away from

the image center and its shape is rather asymmetrical. The volume difference is illustrated in Fig. 5.18 and the quantities are given below.

(a)

(b)

(c)

(d)

Figure 5.18: Volume difference of image V2. (a) Reference surface and detected surface, (b) volume difference in middle slices, (c) volume difference in one end, (d) volume difference in the other end.

RV	AV	TP	FP	FN	$RV \cup AV$
2783408	2857049	2660810	196239	122598	2979647

$$VD = \frac{FP + FN}{TP + FP + FN} = \frac{196239 + 122598}{2979647} = 10.70\%,$$

$$AVD = \frac{|RV - AV|}{TP + FP + FN} = \frac{|2783408 - 2857049|}{2979647} = 2.47\%.$$

Its surface distance errors calculated by **MESH** are shown in Fig. 5.19 with $MSD = 3.27mm$ and $MASD = 0.505mm$.

Figure 5.19: Surface distance of image V2, from detected surface to reference.

Table 5.1 summarizes the results of validity evaluation for the algorithm on the eight 3D TRUS images.

It can be seen from Table 5.1 that the average volume difference (VD) is 12.58%, in other words, the average similarity between the reference and algorithm detected

Table 5.1: Summary of the results of algorithm validation on 3D images

3D TRUS image	VD(%)	AVD(%)	MSD		MASD	
			(mm)	voxel	(mm)	voxel
V0 ($250 \times 174 \times 236$)	15.75	8.08	4.044	23.33	0.688	3.97
V1 ($256 \times 256 \times 256$)	11.21	0.24	2.757	12.21	0.524	2.32
V2 ($252 \times 194 \times 256$)	10.70	2.47	3.270	17.08	0.505	2.64
V3 ($252 \times 194 \times 256$)	13.76	3.50	4.462	21.05	0.705	3.33
V4 ($252 \times 194 \times 256$)	13.55	9.34	4.268	20.21	0.891	4.22
V5 ($256 \times 256 \times 256$)	10.14	0.36	2.676	11.85	0.478	2.12
V6 ($212 \times 192 \times 248$)	13.37	3.78	4.374	15.94	0.772	2.81
V7 ($256 \times 256 \times 256$)	12.13	1.54	3.046	13.49	0.585	2.59
Average	12.58	3.66	3.612	16.90	0.644	3.00

VD: volume difference, AVD: absolute volume difference, MSD: maximum surface distance, MASD: mean absolute surface distance

volume is 87% - 88%. Besides, the average distance of the two surfaces is about 3 voxels. According to the standard set up in [31] (a similarity value between 0.7 and 1.0 represents an excellent agreement while a similarity between 0.4 and 0.7 represents a good agreement), there exists an excellent agreement between detected results and references.

Nevertheless, as reported in Table 5.1, the average maximum surface distance (AMSD: 16.90 voxels) seems a bit large. This disagreement mainly occurs in the two ends of the prostate (see sub-figure (c), (d) in Fig. 5.12, Fig. 5.15 and Fig. 5.18), where even the manual outlines often vary a lot because the image quality is usually much worse in these areas.

Now let us evaluate the reliability of the algorithm. To measure the test-retest reliability, 4 images are randomly selected to test the algorithm and the test is

repeated 3 times by one person with the interval of one week. Without loss of generality, the first 4 images in Table 5.1) from the eight 3D prostate scans are applied. Detected results are summarized in Table 5.2.

Table 5.2: Summary of the accuracy in test-retest

image	Accuracy (100-VD) (%)			
	test 1	test 2	test3	Average
V0	81.549	81.902	82.322	81.924
V1	90.403	87.217	89.122	88.914
V2	88.659	89.561	88.607	88.943
V3	86.324	89.129	84.488	86.647
Average of average				86.607

According to Eq. 5.22, Eq. 5.24 and Eq. 5.21, we get $bMS = 32.7065$, $eMS = 2.584$, and $ICC(3, 3) = 0.921$.

To measure the inter-rater reliability, 3 people are chosen to test the algorithm on previous selected 4 images with the given rule: cover the areas as largely as possible while ensure all the bubbles are inside the prostate. Detected results are tabularized in Table 5.3. Similarly, we can get $bMS = 26.7199$, $wMS = 2.6887$, and $ICC(1, 3) = 0.8994$.

Table 5.3: Summary of the accuracy in inter-rater test

image	Accuracy (100-VD) (%)			
	rater 1	rater 2	rater 3	Average
V0	86.967	86.295	81.383	84.881
V1	90.014	88.447	88.045	88.836
V2	92.023	91.321	91.692	91.679
V3	86.829	86.352	85.814	86.332
Average of average				87.932

Figure 5.20: Three different sets of initial level-sets (column by column) presented by three users over image V1 at slice 84, slice 126, slice 168 and slice 210.

The intraclass correlation coefficients show that both test-retest reliability and inter-rater reliability of the algorithm are quite high. Recall that the initial level sets are defined by the user according to the image property in this algorithm, the test-retest reliability and inter-rater reliability also reflect the sensitivity of the result to the definition of initial bubbles. The high reliability makes it sure that the user interaction in the initialization process is of little concern in this algorithm. For example, Fig. 5.20 gives three different sets of initial level-sets (different radii at different positions) presented by three users on image V1, and the similar results are superposed each other on three orthogonal slices in Fig. 5.21.

Figure 5.21: Similar Results (superposed each other on three orthogonal slices) obtained from the three different sets of initial level-sets shown in Fig. 5.20. *Yellow line*: the first result, *red line*: the second result, *green line*: the third result.

Another issue concerned with the evaluation is to compare the disagreements between the algorithm results and segmentations by experts with the disagreements between experts. To do this, the algorithm result is firstly compared with one reference surface which is randomly selected from the three segmentations obtained at different time by each expert. And the disagreements between the algorithm results and segmentations by experts are reported in Table 5.4.

Table 5.4: Disagreements between the algorithm results and segmentations by experts

Disagreement (%)	Manual segmentation			Average
	expert 1	expert 2	expert 3	
Algorithm result (V0)	14.7763	16.3267	16.9202	16.0077
Algorithm result (V1)	10.8926	10.5411	12.8883	11.4407
Algorithm result (V2)	9.8865	10.7540	9.9521	10.1975
Algorithm result (V3)	13.7673	13.6252	14.0431	13.8119
Algorithm result (V4)	12.9698	12.9600	14.5821	13.5040
Algorithm result (V5)	10.6819	9.1621	9.2611	9.7017
Algorithm result (V6)	12.7554	14.0308	12.8653	13.2172
Algorithm result (V7)	12.3710	11.4280	13.4094	12.4028
Average	12.2626	12.3535	12.9902	12.5354

After that, the manual segmentations are compared with each other. And Table 5.5 summarizes the disagreements between experts.

It can be seen from Table 5.4 And Table 5.5 that the average disagreement between the algorithm results and segmentations by experts is at the same level of the average disagreement between experts though the former is a little bit smaller than the latter. Overall, the proposed algorithm generates reproducible results with an accuracy around 87% when applied to 3D ultrasound images for 3D prostate sur-

Table 5.5: Disagreements (D) between the manual segmentations by experts

D (%)	Expert 1								Average
	V0	V1	V2	V3	V4	V5	V6	V7	
Expert 2	16.16	11.09	10.77	13.64	13.88	11.22	14.25	12.23	12.91
D (%)	Expert 1								Average
	V0	V1	V2	V3	V4	V5	V6	V7	
Expert 3	15.46	10.90	10.57	13.62	13.57	9.88	13.52	11.76	12.41
D (%)	Expert 2								Average
	V0	V1	V2	V3	V4	V5	V6	V7	
Expert 3	15.76	11.41	11.96	14.60	13.69	10.16	13.22	12.79	12.95
Average									12.75

face detection, and it greatly relieves the user involvement as well.

5.3.2 Prostate surface detection from a stack of 2D TRUS images

As mentioned before, currently the most widely used equipment for diagnoses and treatments of prostate diseases is conventional 2D ultrasound machine. This section tries to apply our algorithm to detect the prostate surface from a stack of 2D TRUS images.

Image acquisition

The 2D images are collected by a system called “OSCILLATOR” as shown in Fig. 5.22, which was developed by E. Seah and F. Liu [82] in CIMIL. The oscillator

system can be viewed as a combination of 2 parts — the **image acquisition unit** and the **motor control unit**.

Figure 5.22: Diagram of the oscillator system.

- The **image acquisition unit** is responsible for the capturing the ultrasound images and storing as an image file in standard picture format (BMP). The Matrox Meteor frame grabber card is used to interface the PC with the analog video signal from the ultrasound machine.
- The **motor control unit** is responsible for communication between the PC and the DC Motor. The Galil Motion Controller card controls the movement of the DC motor and gathers the encoder's feedback data. This data was used to represent the distance move of the probe.

With “OSCILLATOR”, we can collect a stack of parallel 2D TRUS images with the interval of 1mm.

2D into 3D interpolation

Because our algorithm is designed for volumetrical data, we need to generate a 3D image from this image stack before we start the surface detection procedure. This involves some sophisticated interpolation techniques[83] for obtaining high-quality 3D images. However, for simplicity, we just use the linear interpolation in our algorithm. Fig. 5.23 illustrates this interpolation scheme, where the dark grey slices stands for the collected data, and the light gray slices mean the interpolation between the 2 input neighboring slices while the current interpolating one is highlighted in orange.

Figure 5.23: Interpolation for creating 3D images from image stacks.

By using linear interpolation, the intensity value at location n between the 2 neighboring slices (e.g., i and $i + 1$) is defined as

$$I_n = (1 - \frac{n}{N})I_i + \frac{n}{N}I_{i+1}, \quad (5.25)$$

where N is the number of total slices between the 2 input images. Note that N may differ from one pair of neighboring slices to another, because the error occurs when we convert the float data to integer for locating the input slices according to the image resolution.

The following table gives the image resolution of different zoom rate for ultrasound imaging. Clearly, for every pair of input neighboring slices with the zoom rate, for example, R12, there exists $1/0.275493 = 3.63$ pixels. However, we have to arrange the image slices in integer locations, this results in slice 1 at new location $1 * 3.63 \hat{=} 4$, slice 2 at new location $2 * 3.63 \hat{=} 7$, and so forth². In this arrangement, the displacement of the input slice in the 3D image is at most $0.5/0.275492 \hat{=} 2$ pixels for R12.

Table 5.6: Image resolution (mm/pixel) of different zoom rate for ultrasound imaging

Zoom Rate	R03	R04	R05	R06
Image Resolution	0.066667	0.089928	0.111535	0.133785
Zoom Rate	R08	R10	R12	R15
Image Resolution	0.179212	0.247719	0.275493	0.336951

Experimental results

Fig. 5.24 gives an example of the prostate surface detection from a stack of 2D images (*S0764782D*: 46 R12 slices). Its surface distance and volume difference compared with the reference are shown in Fig. 5.25 and Fig. 5.26 respectively. The surface distance errors calculated by **MESH** are

²Here we round the float value to its nearest integer.

Figure 5.24: Detected result from image stack *S0764782D*. (a) 3D image constructed from the image stack, (b) detected surface and corresponding contours on the 3 orthogonal slices.

Figure 5.25: Surface distance for image stack *S0764782D*, from detected surface to reference.

(a)

(b)

(c)

(d)

Figure 5.26: Volume difference for image stack *S0764782D*. (a) Reference surface and detected surface, (b) volume difference in middle slices, (c) volume difference in one end, (d) volume difference in the other end.

MSD		MASD	
mm	voxel	mm	voxel
5.835	21.18	1.929	7.00

where the image resolution is 0.275493mm/voxel.

The volume difference between the detected result and reference for image stack *S0764782D* is calculated as follows.

RV	AV	TP	FP	FN	$RV \cup AV$
1677957	1666579	1470301	196278	207656	1874235

$$VD = \frac{FP + FN}{TP + FP + FN} = \frac{196278 + 207656}{1874235} = 21.55\%,$$

$$AVD = \frac{|RV - AV|}{TP + FP + FN} = \frac{|1677957 - 1666579|}{1874235} = 0.61\%.$$

Another example of the prostate surface detection from a stack of 2D images (*S2532520F*: 41 R10 slices) are shown in Fig. 5.27, where Slice 0 – Slice 3 are missing. To make sure that the front can evolve in Slice 4, we add 4 black slices to the front. The surface distance and the volume difference compared with the reference are shown in Fig. 5.28 and Fig. 5.29 respectively.

The quantities of volume difference and the surface distance between the detected result and reference for image stack *S2532520F* are calculated as follows.

RV	AV	TP	FP	FN	$RV \cup AV$
1530260	1321083	1279872	41211	250388	1571471

$$VD = \frac{FP + FN}{TP + FP + FN} = \frac{41211 + 250388}{1571471} = 18.56\%,$$

Figure 5.27: Detected result from image stack $S2532520F$. (a) 3D image constructed from the image stack, (b) detected surface and corresponding contours on the 3 orthogonal slices.

Figure 5.28: Surface distance for image stack $S2532520F$, from detected surface to reference.

(a)

(b)

(c)

(d)

Figure 5.29: Volume difference for image stack $S2532520F$. (a) Reference surface and detected surface, (b) volume difference in middle slices, (c) volume difference in one end, (d) volume difference in the other end.

$$AVD = \frac{|RV - AV|}{TP + FP + FN} = \frac{|1530260 - 1321083|}{1571471} = 13.31\%.$$

MSD		MASD	
mm	voxel	mm	voxel
6.039	24.38	1.328	5.36

where the image resolution is 0.247719mm/voxel.

The validity evaluation of the algorithm applied to four 2D image stacks is summarized in Table 5.7.

Table 5.7: Summary of the results of algorithm validation on 2D image stack

2D image stack	VD(%)	AVD(%)	MSD		MASD	
			(mm)	voxel	(mm)	voxel
R12 (46 slices)	21.55	0.61	5.835	21.18	1.929	7.00
R10 (41 slices)	18.56	13.31	6.039	24.38	1.328	5.36
R12 (53 slices)	17.81	6.20	5.877	21.33	1.852	6.72
R08 (41 slices)	25.12	5.81	5.068	28.28	1.491	8.32
Average	20.76	6.48	5.705	23.79	1.65	6.85

VD: volume difference, AVD: absolute volume difference, MSD: maximum surface distance, MASD: mean absolute surface distance

The four image stacks are then used to evaluate the reliability of the algorithm. Results obtained from test-retest and inter-rater observation are reported in Table 5.8 and Table 5.9 respectively.

Test-retest reliability and inter-rater reliability can be calculated from Table 5.8 and Table 5.9 as: $ICC(3, 3) = 0.788$, $ICC(1, 3) = 0.7819$.

It is easy to see that the average similarity between references and results detected from 2D image stacks is about 79%, around 8% less than that obtained

Table 5.8: Summary of the accuracy in test-retest on 2D image stacks

image	Accuracy (100-VD) (%)			
	test 1	test 2	test3	Average
R12 (46 slices)	75.799	76.152	76.572	76.174
R10 (41 slices)	81.613	78.427	80.332	80.124
R12 (53 slices)	80.359	81.261	80.307	80.643
R08 (41 slices)	78.084	80.889	76.248	78.407
Average of average				78.837

Table 5.9: Summary of the accuracy in inter-rater test on 2D image stacks

image	Accuracy (100-VD) (%)			
	rater 1	rater 2	rater 3	Average
R12 (46 slices)	79.251	76.971	79.318	78.515
R10 (41 slices)	79.010	78.640	77.491	78.380
R12 (53 slices)	81.730	80.293	81.601	81.208
R08 (41 slices)	77.495	79.622	76.111	77.743
Average of average				78.962

from 3D TRUS images. The reliability of algorithm also downgrades when applied to 2D image stacks: test-retest reliability downgrades from 0.921 to 0.788; inter-rater reliability downgrades from 0.8994 to 0.7819.

Similarly, to determine if the automatic method agrees with the experts, as much as the experts agree with one another, the disagreements between the algorithm results and the segmentations by experts are firstly compared and reported in Table 5.10. Afterwards the disagreements between experts are calculated and summarized in Table 5.11. Clearly, the disagreements between the algorithm and experts are compared to the disagreements between experts.

Table 5.10: Disagreements between the algorithm results and segmentations by experts

Disagreement (%)	Manual segmentation			Average
	expert 1	expert 2	expert 3	
Algorithm result (R12.1)	22.3873	20.5237	21.3129	21.4080
Algorithm result (R10)	18.1443	17.6382	18.9865	18.2564
Algorithm result (R12.2)	17.3955	16.8425	20.5710	18.2697
Algorithm result (R08)	24.7844	24.7493	27.3590	25.6309
Average	20.6779	19.9384	22.0574	20.8912

Table 5.11: Disagreements between the manual segmentations by experts

Disagreement (%)	Expert 1				Average
	R12.1	R10	R12.2	R08	
Expert 2	23.0796	19.3785	19.4205	25.7896	21.9171
Disagreement (%)	Expert 1				Average
	R12.1	R10	R12.2	R08	
Expert 3	21.7876	19.6715	18.6976	25.9081	21.5162
Disagreement (%)	Expert 2				Average
	R12.1	R10	R12.2	R08	
Expert 3	21.2930	17.7154	19.3658	25.5598	20.9835
Average					21.4723

5.4 Analysis of the sensitivities of the method to the parameter choices

The parameters in the algorithm include the narrow band width, narrow band layers, sliding window size, threshold R_T , threshold value for stopping criteria, the number of Gaussian components for GMM, side length of ROI for data sampling,

and threshold P_T . In this thesis, to simplify the algorithm implementation, the narrow band width is fixed to two voxel width, i.e., one voxel width at each side, and thus the narrow band layers are accordingly fixed to five as addressed in Section 3.2.2. The sliding window size ($n \in \{3, 5, 7, \dots\}$) is easy to determine because in our algorithm tests, the accuracy of results most often reaches the maximum at $n = 5$ or 7 when other parameters are fixed. Therefore n is set to 5 for the tradeoff between speed and accuracy. As to the threshold R_T , since it is approximated by a linear combination of the standard deviations of GMM as expressed in Eq.4.29 and the sensitivity of the result to its value is reduced (from trial and error) due to the inclusion of another function P , it is also regarded as a fixed parameter and its sensitivity test is left to GMM.

To analyze the sensitivities of the method to the parameter choices, 3D prostate images are preferentially chosen to test the algorithm and the same set of initial level-sets is applied throughout the algorithm tests.

Fig. 5.30 illustrates the sensitivity of the algorithm to the choice of threshold value for stopping criteria as well as the corresponding variation in computation time. It can be seen from Fig. 5.30(a) that the value chosen between 0.002 to 0.005 just results in a very little variation in the average accuracy. A smaller threshold value could obtain a slight higher average accuracy, however, the computation time would increase greatly as shown in Fig. 5.30(b). Therefore, it is reasonable that the threshold value for stopping criteria is set to 0.0035 in our method.

Examples of the GMM with different components are shown in Fig. 5.31. To test the sensitivity of the algorithm to the choice of GMM components (for example, $n = 2, 3, 4$), P_T is fixed at half the maximum probability and Eq. 4.30 is

(a)

(b)

Figure 5.30: Analysis of the sensitivity of the method to threshold value for stopping criteria. (a) Variation of the average accuracy with respect to the threshold value, (b) variation of the average computation time with respect to the threshold value.

Figure 5.31: GMMs with different components ($n = 2, 3, 4$ from top to bottom). (a) GMMs for image V0, (b) GMMs for image V4.

adopted to approximate R_T as follows.

$$R_T = \sqrt{6} \sum_{i=1}^n \alpha_i \cdot \sigma_i. \quad (5.26)$$

where n is the number of GMM components. As a result, the accuracy and the computation time are compared in Fig. 5.32 and Fig. 5.33 respectively.

Figure 5.32: Comparison of the accuracy with respect to GMM components

Compared with the algorithm which has 3 GMM components, it is very clear that the accuracy of the algorithm obviously decreases (about 4.5 in percentage on average) when GMM components are set to 2, though its computation time accordingly reduces (about 4% on average) as well. On the other hand, the small

Figure 5.33: Comparison of the computation time with respect to GMM components

improvement in accuracy (less than 1 in percentage on average) when GMM components are set to 4 is unworthy of the increasing cost (about 5% on average) in computation time. Therefore, 3 components of GMM best fit the algorithm in tradeoff between accuracy and speed. This result is consistent with what we observed from trial and error.

The sensitivity of the algorithm to the side length of ROI for data sampling is similarly analyzed. Noting that the minimal image dimension may greatly differ from one another among the 3D images while the maximal image dimension keeps quite similar, the maximal image dimension is also used for the sensitivity analysis.

Table 5.12: Accuracy of the algorithm with respect to the side length of ROI (I)

Min D	V0	V1	V2	V3	V4	V5	V6	V7	Ave.
2/32	76.84	82.52	82.50	79.73	79.02	83.44	80.91	81.36	80.79
3/32	79.19	83.32	83.70	82.24	81.19	84.51	81.68	83.01	82.36
4/32	80.56	85.06	84.55	82.12	83.10	85.36	82.68	83.98	83.43
5/32	81.68	86.62	86.57	83.92	84.11	87.91	84.20	86.07	85.13
6/32	82.59	87.49	87.79	84.95	84.44	88.11	85.66	86.08	85.89
7/32	83.51	87.78	88.25	85.60	85.83	88.74	85.86	87.02	86.57
8/32	83.87	88.50	88.75	85.94	86.01	89.49	86.23	87.57	87.04
9/32	84.17	88.42	88.84	86.09	86.83	89.67	86.12	87.88	87.25
10/32	84.28	88.44	89.27	86.29	86.23	89.46	86.57	87.29	87.23
11/32	84.42	87.75	88.60	86.49	85.60	88.74	86.16	86.94	86.84

Table 5.13: Accuracy of the algorithm with respect to the side length of ROI (II)

Max D	V0	V1	V2	V3	V4	V5	V6	V7	Ave.
2/32	78.56	82.52	83.14	81.41	80.74	83.44	81.36	81.36	81.57
3/32	80.98	83.32	84.21	82.42	82.89	84.51	82.49	83.01	82.98
4/32	82.36	85.06	86.87	83.99	84.10	85.36	82.91	83.98	84.33
5/32	83.68	86.62	88.01	85.11	85.11	87.91	85.72	86.07	86.03
6/32	84.03	87.49	88.59	85.73	85.92	88.11	85.97	86.08	86.49
7/32	84.31	87.78	88.88	86.17	86.91	88.74	86.12	87.02	86.99
8/32	84.59	88.50	89.25	86.31	86.02	89.49	86.63	87.57	87.30
9/32	84.65	88.42	89.07	86.33	85.45	89.54	86.04	87.75	87.16
10/32	84.38	88.44	88.65	85.80	84.06	87.96	85.23	86.09	86.34
11/32	83.97	87.75	87.93	84.94	81.85	86.78	84.12	84.94	85.29

Table 5.12 and Table 5.13 Summarize the accuracy of the algorithm with respect to the side length of ROI, where Min D and Max D refer to the minimal image

dimension and the maximal image dimension respectively. The average accuracy is then compared in Fig. 5.34, and the corresponding computation time is illustrated in Fig. 5.35.

Figure 5.34: Average accuracy with respect to side length of ROI

It is clear to see that the average accuracy reaches the maximum value at the side length of $9/32$ the minimal image dimension or $1/4$ the maximal image dimension. Although the average accuracy varies only a little when the side length of ROI ranges from $6/32$ to $10/32$ the image dimension, the average computation time increases greatly. In other words, a small improvement in average accuracy will result in a large increase in average computation time. For this reason, the side

Figure 5.35: Average computation time with respect to side length of ROI

length can be set to $1/4$ the minimal image dimension or $7/32$ the maximal image dimension for the tradeoff between accuracy and speed. Therefore, the choice of this parameter in our algorithm is reasonable.

Recall that P_T value expressed in Eq. 4.27 is regulated by two parameters a and b . Instead of directly analyzing the sensitivity of the algorithm to the choice of P_T , the sensitivity to value a is firstly analyzed. It is accomplished by summarizing the accuracy of the results with respect to value a as shown in Table 5.14.

The analysis result has shown that the average accuracy reaches the maximum at $a = 0.5$. This results in a new and simpler formulation of P_T as expressed in

Table 5.14: Accuracy of the results with respect to a ($b = 1.5$)

3D image	a								
	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
V0	75.05	79.86	83.33	83.87	84.23	82.47	82.07	79.50	77.86
V1	81.68	85.77	87.45	88.50	88.90	87.19	85.90	85.32	82.24
V2	79.94	85.34	87.62	88.75	88.45	87.34	86.27	84.77	81.84
V3	78.55	82.73	84.87	85.94	86.30	84.85	85.16	82.56	79.50
V4	77.65	81.79	85.36	86.01	86.30	84.95	84.47	81.54	80.21
V5	83.26	86.28	88.05	89.49	89.33	88.15	87.07	85.51	82.55
V6	77.89	83.62	85.09	86.23	86.46	85.41	84.28	83.33	79.77
V7	81.50	84.26	86.81	87.57	87.62	86.18	85.51	83.04	80.60
Average	79.44	83.71	86.07	87.04	87.20	85.82	85.09	83.20	80.57

Table 5.15: Accuracy of the algorithm with respect to a

3D image	b								
	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
V0	75.09	79.84	83.39	83.83	84.19	82.52	82.19	79.56	77.83
V1	82.27	85.24	85.88	87.18	88.85	88.54	87.30	85.76	81.74
V2	80.12	85.36	87.57	88.75	88.40	87.40	86.30	84.71	81.77
V3	78.64	82.70	84.98	86.04	86.33	84.88	85.14	82.70	79.55
V4	77.67	81.69	85.49	85.99	86.16	84.86	84.48	81.57	80.37
V5	82.49	85.45	87.07	88.13	89.31	89.48	88.00	86.45	83.19
V6	77.88	83.56	85.03	86.26	86.41	85.33	84.27	83.37	79.71
V7	80.66	83.20	85.43	86.18	87.58	87.54	86.83	84.35	81.53
Average	79.35	83.38	85.60	86.54	87.16	86.32	85.56	83.56	80.71

Eq. 4.28, i.e., $P_T = a \cdot \bar{p}(I | \Phi)$. As a comparison, the accuracy of the algorithm with respect to value a in this new formulation is analyzed again, and the analysis results are reported in Table 5.15.

Fig. 5.36 gives a clear illustration of the difference in average accuracy between previous work and current algorithm. The average accuracy reaches the maximum at $a = 0.5$ in both settings of P_T . And a relative large variance in input value a (for example, ranging between 0.3 and 0.7) will only lead to a small variance (about 2 percentage) in the average accuracy. In other words, the sensitivity of the algorithm to the choice of P_T is exactly minute.

Figure 5.36: Comparison of the average accuracy with respect to parameter a

The analysis of the sensitivities of the method to the parameter choices has shown that the parameters set in our algorithm based on trial and error are reasonable, however, the threshold value P_T can be better expressed in a simpler form to reduce the complexity of the algorithm.

5.5 Discussion

The experimental results have shown that the proposed algorithm can obtain reproducible results and give an accuracy of near 87% over 3D TRUS images as well as an accuracy of near 77% over 2D image stacks. The detected surface and reference fit well in the middle slices but usually differ in a large degree in the two ends of prostate as illustrated in Fig. 5.12, Fig. 5.15, Fig. 5.18, Fig. 5.26, and Fig. 5.29. About 10% better accuracy that we obtained from 3D TRUS images over that we obtained from 2D image stacks is because that the 3D TRUS image quality is better than that we constructed from the 2D image stacks. Many factors will influence the quality of the reconstructed 3D images, such as the slice interval, the interpolation algorithm, and so on. The worse image quality results in the evolving front more easily subjected to the boundary leaking, especially in the two ends of the prostate (see Fig. 5.26). On the other hand, we observed that the radiologists tend to draw the contours rather large in areas that do not have a clear boundary. Part of the disagreement between the detected result and the reference can be attributed to this fact. In those more blurred 3D images constructed from 2D image stacks, the disagreement caused by this fact may get worse.

To evaluate the algorithm, the reference used in this thesis is constructed by averaging 9 surfaces manually outlined by 3 trained people at 3 different times. Though both inter-observer variability and intra-observer variability can be reduced through this mechanism, the expert opinion is still subjective after all. Therefore, it is necessary to develop some real “ground truth” methods to validate the algorithm. In this particular case, we have made a preliminary investigation on a plausible solution, which is addressed in the next chapter.

5.6 Summary

This chapter evaluates the proposed algorithm, “Evolving Bubbles”, by applying it to both 3D TRUS images and 2D TRUS image stacks for 3D prostate surface detection. The reference is constructed based on experts’ observation. 3 trained people are selected to outline the surface of each image using **Bubble 3D** at 3 different times. The 9 manually outlined surfaces are then averaged to form the final reference in order to reduce the inter-observer variability and intra-observer variability. To evaluate the validity, volume error measurement and surface distance measurement are both carried out in this thesis. The reliability is measured by the use of intraclass correlation coefficients (ICC) based on the following design:

- measure of test-retest reliability: 4 images are randomly selected to test the algorithm and the test is repeated 3 times by one person with the interval of one week.
- measure of inter-rater reliability: 3 people are chosen to test the algorithm over the previous selected 4 images with the given rule: cover the areas as largely as possible while ensure all the bubbles are inside the prostate.

The experimental results have shown that the proposed algorithm can obtain reproducible results and give an accuracy of near 87% when 3D TRUS images were used and an accuracy of near 77% when 2D image stacks were used. This gives a justification for using 3D transrectal ultrasound in diagnoses and treatments of prostate diseases currently.

CHAPTER 6

Conclusion and Recommendations

6.1 Conclusion

With the number of men seeking medical care for prostate diseases rising steadily, the need for a fast and accurate prostate boundary (surface) detection tool increases accordingly. This thesis presents a novel and practical 3D approach, “Evolving Bubbles”, based on the level set method to solve the problem of automatic (or semi-automatic with minimal user-involvement) prostate surface detection from ultrasound images. Unlike the conventional deformable models for segmentation, the level set method can reduce the amount of user intervention and extend to three dimensional space without any extra machinery. The key task of the level set based method is to provide an appropriate speed function F , which can drive the evolving front to the desired boundaries or surface.

In most applications of the level set method, researchers commonly use the image gradient to design the speed function F . However, this image gradient based method is subjected to “boundary leaking” problem when applied to our project due to the weak gradient information caused by the poor contrast between the prostate and surrounding tissues, speckle noise, shadowing and refraction artifacts in ultrasound images. Although the so-called gradient flow can be added to improve the algorithm performance to a certain degree, it cannot solve this problem quite well.

Motivated by the region-based active contour for boundary finding in 2D medical images, we replaced the image gradient with region information in the level set method to remedy the leaking problem. To compensate the possible large gaps in the dark areas caused by artifacts when the region information is applied only, a Gaussian mixture model (GMM) that is used to describe the intensity distribution of the prostate is included. With the combination of region information and GMM, the speed function F is appropriately designed and expressed in Eq. 4.29.

Considering the prostate location in the ultrasound images and its shape and size are all different from one another, the user is brought in to define the initial multiple level-sets according to the image property. With this, a final semi-automatic (with user interaction in the initialization) approach, “Evolving Bubbles”, for the 3D prostate surface detection is achieved. The image-derived information (i.e., region information and GMM) based on gray-level rather than image gradient, gives our algorithm the ability to overcome the main drawback, “boundary leaking” problem, of the level set based applications to ultrasound images. While the statistical distribution model successfully prevents the evolving front from penetrating into the dark areas caused by artifacts, the gradient flow attracts the front to the vicinity of the desired boundaries in the low contrast areas.

To reduce the computation time, a narrow band solution is applied after the image is sampled every other slice along x , y , z direction respectively. The experimental results have shown that the proposed fast algorithm not only reduces the run time dramatically, but also reduces the risk of “boundary leaking” problem.

The proposed algorithm is then evaluated for its validity and reliability. The validity is evaluated by measuring the volume difference and the surface distance

while the reliability is measured by the use of intraclass correlation coefficients (ICC). The experimental results have shown us that the algorithm can obtain reproducible results in a few minutes and give an accuracy of near 87% when applied to 3D TRUS images, and an accuracy of near 77% when applied to 2D image stacks.

Certainly, automatic (semi or fully) prostate boundary detection methods can provide robust, consistent, and reproducible results with a certain degree of accuracy, however, it is unlikely that automatic prostate boundary detection methods will ever replace physicians. They will, however, likely become one of the crucial elements in prostate disease treatments, particularly in computer-assisted surgery. The continued development and refinement of these methods remain as an important area of research in the foreseeable future.

6.2 Recommendations for further research

As a starting point, the proposed level set based algorithm, “Evolving Bubbles” has shown its good prospect in 3D ultrasound image segmentation. However, due to the poor contrast between the prostate and surrounding tissues, speckle noise, shadowing and refraction artifacts in transrectal ultrasound images, the present method cannot accurately attract the evolving bubbles to the desired prostate surface either, especially at the two ends of the prostate. The image derived information based on the intensity is not enough to constrain the front propagation. The following listed future work may benefit this research.

Allowing for manual editing

Interactive (semi-automatic) algorithms and fully automatic algorithms represent two approaches to detect the prostate boundary (surface). There is no doubt that the automatic detection can potentially increase the speed, consistency, and reproducibility of the process. However, its accuracy is difficult to be guaranteed due to the poor quality of ultrasound images. For computer-assisted surgery, this may result in injuries to the patient. Consequently, the most immediately successful methods will likely be those that drastically decrease the labor intensiveness of boundary detection tasks through partial automation, while still allowing for interactive guidance or editing by the medical expert. Ladak *et al.* [25] and Pathak *et al.* [9] have recognized this issue and permit the manual editing to refine the final detected boundaries.

It is easy to combine such a manual editing tool with our algorithm. Once the points on the surface are detected by our algorithm, we can sample the points in an explicit order to form the control point list. With this we can use NURBS [75] to re-generate the surface. Thus the users (surgeons or radiologists) can refine the detected contours or surfaces by adjusting the control points freely to wherever they think it is more accurate.

Taking advantage of 3D ultrasound image

In the last few years, technology has progressed to make 3D ultrasound imaging an economically viable tool. Reports in [15, 14, 84] have shown that 3D ultrasound has advantages over conventional 2D techniques, and that it will play an increasingly important role in prostate disease treatments. Although from sequential cross-

sectional 2D boundaries we can reconstruct a 3D prostate surface [19, 20, 23, 24], as reported in [31, 85], algorithms that could take advantage of 3D information would be more robust and accurate.

In fact, our experimental results presented in Chapter 5 show that the accuracy obtained from the 3D TRUS images is about 10% better than that obtained from the 2D image stacks. Therefore, developing algorithms to detect the prostate surface from 3D ultrasound images is preferred.

Incorporating statistics in prostate shape and appearance

Due to the poor quality of ultrasound images, edge-based methods or region-based methods alone are not accurate enough to extract the prostate boundaries. To improve the accuracy and robustness, the information of general shape, location and orientation as well as appearance (for example, textures) of the prostate could be incorporated into the models. These might be the form of initial conditions [25, 60], data constraints [9, 18, 86, 53], constraints on the model shape and appearance parameters [26, 27, 87, 88, 54, 55], or the model fitting procedure [28, 29, 45, 89, 30, 32]. When the thesis is completed, up-to-date methods based on the statistical shape and appearance information have been proposed [90, 91]. These shape based or appearance based statistical models have shown their superiority in image segmentation, though most of them are applied to 2D image only. Our experience in using prostate shape knowledge in the initialization has also shown that it helped to guide the prostate boundary detection.

However, more works should be done in order to incorporate prior knowledge into the proposed algorithm. Among them a key work is to efficiently describe

the prostate shape using shape description methods like point distribution models [44], spherical harmonics [92], or M-Reps [93]. Another critical issue concerned with the statistical model building is to collect adequate clinical image data for training the model. Therefore, collaborative hospitals should be deeply involved in the continuous development of the method.

“Ground Truth” method for evaluation

An important aspect before advancing the diagnoses and treatments of prostate diseases is evaluating the 3D prostate surface detection methods. However, due to practical difficulties, most of the evaluation is carried out by comparing detected results with those manual outlines by experts. As mentioned before, manual contouring is variable between different observers and also within the same observer over a period of time. Although a mean contour of those manual outlines produced by multiple observers [9] can be used to reduce the variation, this evaluation criterion is still subjective.

An objective alternative is to evaluate the algorithm over simulated ultrasound images by advanced ultrasound simulators such as Field II [94] and Wave3000 Pro, the successor of Wave2500 [95]. Though all of the simulators have the limitations in reproducing the full range of imaging characteristics (partial volume effect, intensity inhomogeneity, artifact, etc.), normal and abnormal anatomical variability observed in clinical data, they can at least provide an upper bound on the effectiveness of the proposed methods.

A more objective way is to compare the detected results with those obtained from the real target, what we call “ground truth”. In this particular case, the

prostate gland has the possibility to be taken out from the patient body by radical prostatectomy for prostate cancer treatment, so we have the opportunity to scan the prostate surface as the “ground truth” reference if the prostate shape and size do not change in handling. Following this idea, some tasks have been set up to investigate the feasibility of the development of “ground truth” method.

Author's Publications

Journal articles:

1. F. Shao, K.V. Ling, W.S. Ng, and R.Y. Wu. Prostate boundary detection from ultrasonographic images. *Journal of Ultrasound in Medicine*, **22**:605–623, 2003.
2. F. Shao, K.V. Ling, and W.S. Ng. Automatic 3D prostate surface detection from TRUS with level sets. To be published in *International Journal of Image and Graphics*, **4**:385–403, 2004.

Conference papers:

1. F. Shao, K.V. Ling, and W.S. Ng. 3D prostate surface detection from ultrasound images based on level set method. In *Proceedings of Medical Image Computing and Computer-Assisted Intervention—MICCAI 2002*, pages 389–396, Tokyo Japan, September 2002. Dohi and R. Kikinis (Eds): LNCS 2489, Springer.
2. F. Shao, K.V. Ling, and W.S. Ng. A fast algorithm of level set method for 3D prostate surface detection. In *Proceedings of the International Conference on Scientific and Engineering Computation (IC-SEC) 2002*. pages 855–858, Singapore, December 2002. H.P. Lee and K. Kumar (Eds): Imperial College Press.
3. F. Shao, K.V. Ling, and W.S. Ng. Evolving bubbles for prostate surface detection from TRUS images. In *Proceedings of Third IEEE Symposium on BioInformatics and BioEngineering—BIBE 2003*. pages 115–121, Maryland, USA, March 2003. IEEE Computer Society.
4. F. Shao, K.V. Ling, and W.S. Ng. A practical method for prostate surface detection from 3D TRUS (Abstract). In *Proceedings of World Congress on Medical Physics and Biomedical Engineering*, Sydney Australia, August 2003.

Bibliography

- [1] S. J. Berry, D. S. Coffey, P. C. Walsh, and L. L. Ewing. The development of human benign prostatic hyperplasia with age. Journal of Urology, 132:474–479, 1984.
- [2] A. Von Eschenbach, R. Ho, G. P. Murphy, M. Cunningham, and N. Lins. American Cancer Society guideline for the early detection of prostate cancer: update 1997. CA: a Cancer Journal for Clinicians, 47:261–264, 1997.
- [3] K. S. Chua, H. P. Lee, A. Seow, and K. Shanmugaratnam. Trends in cancer incidence in Singapore. Singapore Cancer Registry Report No.4, 1996.
- [4] American Cancer Society. What are the key statistics about prostate cancer? World Wide Web, http://www.cancer.org/docroot/CRI/content/CRI_2_4_1X_What_are_the_key_statistics_for_prostate_cancer_36.asp?sitearea=&level=, 2003.
- [5] P. J. Littrup, R. A. Kane, C. J. Mettlin, G. P. Murphy, F. Lee, A. Toi, R. Badalament, and R. L. Babaian. Cost-effective prostate cancer detection. Cancer, 74:3146–3158, 1994.
- [6] R. G. Aarnink, H. P. Beerlage, J. J. M. C. H. de la Rosette, F. M. J. Debruyne, and H. Wijkstra. Transrectal ultrasound of the prostate: innovations and future applications. Journal of Urology, 159:1568–1579, 1998.
- [7] F. Lee, D. K. Bahn, D. B. Siders, and C. Greene. The role of TRUS-guided biopsies for determination of internal and external spread of prostate cancer. Seminars in Urologic Oncology, 16:129–136, 1998.
- [8] W. N. McDicken. Diagnostic Ultrasonics: Principles and Use of Instruments. Churchill Livingstone, third edition, 1991.
- [9] S. D. Pathak, V. Chalana, D. R. Haynor, and Y. Kim. Edge-guided boundary delineation in prostate ultrasound images. IEEE Transactions on Medical Imaging, 19:1211–1219, 2000.
- [10] M. K. Terris and T. A. Stamey. Determination of prostate volume by transrectal ultrasound. Journal of Urology, 145:984–987, 1991.
- [11] G. Sakas, L. Schreyer, and M. Grimm. Preprocessing and volume rendering of 3d ultrasonic data. IEEE Computer Graphics and Applications, 15:47–54, 1995.

- [12] F. A. Cosio and B. L. Davies. Automated prostate recognition: a key process for clinically effective robotic prostatectomy. Medical & Biological Engineering & Computing, 37:236–243, 1999.
- [13] S. Tong, D. B. Downey, H. N. Cardinal, and A. Fenster. A three-dimensional ultrasound prostate imaging system. Ultrasound in Medicine and Biology, 22:735–746, 1996.
- [14] S. Tong, H. N. Cardinal, D. B. Downey, B. Donal, and A. Fenster. Three-dimensional ultrasound imaging of the prostate. In Proceedings of SPIE: Medical Imaging 1998, volume 3336, pages 113–123, San Diego, CA, 1998.
- [15] A. Fenster, S. Tong, H. N. Cardinal, C. Blake, and D. B. Downey. Three-dimensional ultrasound imaging system for prostate cancer diagnosis and treatment. IEEE Transactions on Instrumentation and Measurement, 47:1439–1447, 1998.
- [16] R. G. Aarnink, S. D. Pathak, J. J. M. C. H. de la Rosette, F. M. J. Debruyne, Y. Kim, and H. Wijkstra. Edge detection in prostatic ultrasound images using integrated edge maps. Ultrasonics, 36:637–644, 1998.
- [17] R. G. Aarnink, R. J. B. Giesen, A. L. Huynen, J. J. M. C. H. de la Rosette, F. M. J. Debruyne, and H. Wijkstra. A practical clinical method for contour determination in ultrasonographic prostate images. Ultrasound in Medicine and Biology, 20:705–717, 1994.
- [18] A. L. Huynen. Automated interpretation of ultrasonographic prostate images. PhD thesis, The University of Nijmegen, 1996.
- [19] J. Y. Lee, C. H. Chen, H. B. Hsien, D. L. Yang, and Y. N. Sun. 3d reconstruction of prostate from transrectal ultrasound images. In Proceedings of the International Conference and Exhibition on Electronic Measurement and Instrumentation, pages 19–26, Shanghai, China, 1995.
- [20] C. H. Chen, J. Y. Lee, W. H. Yang, C. M. Chang, and Y. N. Sun. Segmentation and reconstruction of prostate from transrectal ultrasound images. Biomedical Engineering - Application, Basis Communications, 8:39–44, 1996.
- [21] Y. J. Liu, W. S. Ng, M. Y. Teo, and H. C. Lim. Computerised prostate boundary estimation in ultrasound images using the radial bas-relief method. Medical & Biological Engineering & Computing, 35:445–454, 1997.
- [22] C. K. Kwoh, M. Y. Teo, W. S. Ng, S. N. Tan, and L. M. Jones. Outline of prostate boundary using harmonics method. Medical & Biological Engineering & Computing, 36:768–771, 1998.

- [23] W. D. Richard and C. G. Keen. Automated texture-based segmentation of ultrasound images of the prostate. Computerized Medical Imaging and Graphics, 20:131–140, 1996.
- [24] S. D. Pathak, R. G. Aarnink, J.J. de la Rosette, V. Chalana, H. Wijkstra, F. M. J. Debruyne, and Y. Kim. Quantitative three-dimensional transrectal ultrasound for prostate imaging. In Proceedings of SPIE: Medical Imaging 1998, volume 3335, pages 83–92, San Diego, CA, 1998.
- [25] H. M. Ladak, F. Mao, Y. Wang, D. B. Downey, D. A. Steinman, and A. Fenster. Prostate segmentation from 2d ultrasound images. Medical Physics, 27:1777–1788, 2000.
- [26] C. Knoll, M. Alcaniz, V. Grau, C. Monserrat, and M. C. Juan. Outlining of the prostate using snakes with shape restrictions based on the wavelet transform. Pattern Recognition, 32:1767–1781, 1999.
- [27] C. Knoll, M. Alcaniz, C. Monserrat, V. Grau, and M. C. Juan. Multiresolution segmentation of medical images using shape-restricted snakes. In Proceedings of SPIE: Medical Imaging 1999, volume 3661, pages 222–233, San Diego, CA, 1999.
- [28] J. S. Prater and W. D. Richard. Segmenting ultrasound images of the prostate using neural networks. Ultrasonic Imaging, 14:159–185, 1992.
- [29] A. Lorenz, C. Haas, and H. Ermert. Segmentation of ultrasonic prostate images using a probabilistic model based on Markov random processes. Ultrasonic Imaging, 19:44–45, 1997.
- [30] R. Y. Wu, K. V. Ling, and W. S. Ng. Automatic prostate boundary recognition in sonographic images using feature model and genetic algorithm. Journal of Ultrasound in Medicine, 19:771–782, 2000.
- [31] A. Ghanei, H. Soltanian-Zadeh, A. Ratkewicz, and F. F. Yin. A three-dimensional deformable model for segmentation of human prostate from ultrasound images. Medical Physics, 28:2147–2153, 2001.
- [32] D. Shen, Y. Zhan, and C. Davatzikos. Segmentation of prostate boundaries from ultrasound images using statistical shape model. IEEE Transactions on Medical Imaging, 22:539–551, 2003.
- [33] J. A. Sethian. Level Set Methods and Fast Marching Methods : Evolving Interfaces in Geometry, Fluid Mechanics, Computer Vision, and Materials Science. Cambridge University Press, 1996.

- [34] R. Y. Wu. Bubble 3D: a Java application for handling surfaces in volumetric medical images. Technical report, Computer Integrated Medical Intervention Laboratory (CIMIL), Nanyang Technological University, Singapore, 2003.
- [35] R. Ronfard. Region-based strategies for active contour models. International Journal of Computer Vision, 13:229–251, 1994.
- [36] C. S. Poon and M. Braun. Image segmentation by a deformable contour model incorporating region analysis. Physics in Medicine and Biology, 42:1833–1841, 1997.
- [37] R. N. Czerwinski, D. L. Jones, and Jr. W. D. O’Brien. An approach to boundary detection in ultrasound imaging. In IEEE Ultrasonics Symposium Proceedings, pages 951–955, Baltimore, MD, 1993.
- [38] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. IEEE Transactions on Pattern Analysis and Machine Intelligence, 12:629–639, 1990.
- [39] T. McInerney and D. Terzopoulos. Deformable models in medical image analysis: a survey. Medical Image Analysis, 1:91–108, 1996.
- [40] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: active contour models. International Journal of Computer Vision, 1:321–331, 1988.
- [41] S. Lobregt and M. A. Viergever. A discrete dynamic contour model. IEEE Transactions on Medical Imaging, 14:12–24, 1995.
- [42] S. Mallat. Wavelets for a vision. Proceedings of the IEEE, 84:604–614, 1996.
- [43] J. Dias and J. Leit ao. Wall position and thickness estimation from sequences of echocardiograms images. IEEE Transactions on Medical Imaging, 15:25–38, 1996.
- [44] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. Training models of shape from sets of examples. In Proceedings of British Machine Vision Conference, pages 9–18, Berlin, 1992. Springer.
- [45] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. Active shape models - their training and application. Computer Vision and Image Understanding, 61:38–59, 1995.
- [46] A. Bordes, A. M. Bory, M. Benchaïb, R. C. Rudigoz, and B. Salle. Reproducibility of transvaginal three-dimensional endometrial volume measurements with virtual organ computer-aided analysis (VOCAL) during ovarian stimulation. Ultrasound in Obstetrics and Gynecology, 19:76–80, 2002.

- [47] D. L. Pham, C. Xu, and J. L. Prince. Current methods in medical image segmentation. Annual Review of Biomedical Engineering, 2:315–338, 2000.
- [48] S. Lakare. 3D segmentation techniques for medical volumes. World Wide Web, <http://www.cs.sunysb.edu/~mueller/teaching/cse616/sarangRPE.pdf>, 2000.
- [49] D. Adalsteinsson and J. A. Sethian. A fast level set method for propagating interfaces. Journal of Computational Physics, 118:269–277, 1995.
- [50] D. E. Breen and R. T. Whitaker. A level set approach for the metamorphosis of solid models. IEEE Transactions on Visualization and Computer Graphics, 7:173–192, 2001.
- [51] H. Anton. Calculus: A New Horizon. John Wiley & Sons, New York, sixth edition, 1999.
- [52] R. C. Gonzalez and R. E. Woods. Digital Image Processing, chapter 4 Image Enhancement. Prentice Hall, second edition, 2002.
- [53] M. Leventon, O. Faugeras, W. Grimson, and W. Wells. Level set based segmentation with intensity and curvature priors. In Proceedings of IEEE Workshop on Mathematical Methods in Biomedical Image Analysis, pages 4–11, Hilton Head, South Carolina, 2000.
- [54] M. Rousson and N. Paragios. Shape priors for level set representations. In Proceedings of the European Conference on Computer Vision, volume II, pages 78–92, Copenhagen, Denmark, 2002.
- [55] J. Yang, L. H. Staib, and J. S. Duncan. Neighbor-constrained segmentation with 3d deformable models. In Proceedings of Information Processing in Medical Imaging, pages 198–209. Springer, 2003.
- [56] G. J. McLachlan and K. E. Basford. Mixture models: inference and applications to clustering. Marcel Dekker, New York, 1988.
- [57] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society, 39:1–38, 1977.
- [58] R. A. Redner and H. F. Walker. Mixture densities, maximum likelihood and the EM algorithm. Society for Industrial and Applied Mathematics Review, 26:195–239, 1984.

- [59] J. A. Bilmes. A gentle tutorial on the EM algorithm and its application to parameter estimation for gaussian mixture and hidden markov models. Technical Report TR-97-021, Computer Science Division, Department of Electrical Engineering and Computer Science, U. C. Berkeley, 1998.
- [60] F. Shao, K. V. Ling, and W. S. Ng. Automatic 3d prostate surface detection from TRUS with level sets. International Journal of Image and Graphics, 4:385–403, 2004.
- [61] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. International Journal of Computer Vision, 22:61–79, 1997.
- [62] V. Caselles, R. Kimmel, G. Sapiro, and C. Sbert. Minimal surfaces: A geometric three dimensional segmentation approach. Numerische Mathematik, 77:423–451, 1997.
- [63] R. Malladi, R. Kimmel, D. Adalsteinsson, G. Sapiro, V. Caselles, and J. A. Sethian. A geometric approach to segmentation and analysis of 3d medical images. In Proceedings on IEEE/SIAM Workshop on Mathematical Methods in Biomedical Image Analysis, pages 244–252, San Francisco, California, 1996.
- [64] A. Yezzi, S. Kichenassamy, A. Kumar, P. Olver, and A. Tannenbaum. A geometric snake model for segmentation of medical imagery. IEEE Transactions on Medical Imaging, 16:199–209, 1997.
- [65] D. Collins, A. Zijdenbos, V. Kollokian, J. Sled, N. Kabani, C. Holmes, , and A. Evans. Design and construction of a realistic digital brain phantom. IEEE Transactions on Medical Imaging, 17:463–468, 1998.
- [66] M. Styner, C. Brechbühler, G. Székely, and G. Gerig. Parametric estimate of intensity inhomogeneities applied to mri. IEEE Transactions On Medical Imaging, 19:153–165, 2000.
- [67] V. Chalana and Y. Kim. A methodology for evaluation of boundary detection algorithms on medical images. IEEE Transactions on Medical Imaging, 16:642–652, 1997.
- [68] D. P. Huttenlocher, G. A. Klanderman, and W. J. Rucklidge. Comparing images using the hausdorff distance. IEEE Transactions on Pattern Analysis and Machine Intelligence, 15:850–863, 1993.
- [69] V. Chalana, D. T. Linker, D. R. Haynor, and Y. Kim. A multiple active contour model for cardiac boundary detection in echocardiographic sequences. IEEE Transactions on Medical Imaging, 15:290–298, 1996.

- [70] S. L. Breen. The evaluation of automated boundary detection algorithms for ultrasound images of prostate: Development of a gold standard and statistical methods for comparison. Unpublished work of Computer Integrated Medical Intervention Laboratory, Nanyang Technological University, Singapore, 1999.
- [71] F. M. Carrascal, J. M. Carreira, M. Souto, P. G. Tahoces, L. Gomez, and J. J. Vidal. Automatic calculation of total lung capacity from automatically traced lung boundaries in postero-anterior and lateral digital chest radiographs. Medical Physics, 25:1118–1131, 1998.
- [72] F. Mao, J. Gill, and A. Fenster. Technique for evaluation of semiautomatic segmentation methods. In Proceedings of SPIE: Medical Imaging 1999, volume 3661, pages 1027–1036, San Diego, CA, 1999.
- [73] R. Y. Wu, S. L. Breen, K. V. Ling, B. H. Sim, and W. S. Ng. A web-based evaluation system for boundary detection algorithms on medical images. Unpublished work of Computer Integrated Medical Intervention Laboratory, Nanyang Technological University, Singapore, 2000.
- [74] S. K. Warfield, K. H. Zou, and W. M. Wells. Validation of image segmentation and expert quality with an Expectation-Maximization algorithm. In Proceedings of Medical Image Computing and Computer-Assisted Intervention—MICCAI 2002, pages 298–306, Tokyo, Japan, 2002.
- [75] L. A. Piegl and W. Tiller. The NURBS Book (Monographs in Visual Communications). Springer-Verlag, New York, second edition, 1997.
- [76] N. Aspert, D. Santa-Cruz, and T. Ebrahimi. MESH: Measuring Errors between Surfaces using the Hausdorff Distance. In Proceedings of the IEEE International Conference on Multimedia and Expo, volume I, pages 705–708, Lausanne, Switzerland, 2002. <http://mesh.epfl.ch>.
- [77] P. E. Shrout and J. L. Fleiss. Intraclass correlations: uses in assessing rater reliability. Psychological Bulletin, 86:420–427, 1979.
- [78] W. G. Hopkins. Measures of reliability in sports medicine and science. Sports Medicine, 30:1–15, 2000.
- [79] O. J. Dunn and V. A. Clark. Applied Statistics: Analysis of Variance and Regression. John Wiley & Sons, New York, second edition, 1987.
- [80] D. P. Nichols. Choosing an intraclass correlation coefficient. World Wide Web, <http://www.ats.ucla.edu/stat/spss/library/whichicc.htm>, 1998.
- [81] R. A. Yaffee. Enhancement of reliability analysis: application of intraclass correlations with SPSS/Windows v.8. World Wide Web, <http://www.nyu.edu/its/socsci/Docs/intracls.html>, 1998.

- [82] E. Seah and F. Liu. Ultrasound probe oscillator for acquisition of ultrasound images. Technical report, Computer Integrated Medical Intervention Laboratory (CIMIL), Nanyang Technological University, Singapore, 2000.
- [83] J. M. Sanches and J.S. Marques. A Rayleigh reconstruction/interpolation algorithm for 3D ultrasound. Pattern Recognition Letters, 21:917–926, 2000.
- [84] T. R. Nelson and D. H. Pretorius. Three-dimensional ultrasound imaging. Ultrasound in Medicine and Biology, 24:1243–1270, 1998.
- [85] T. McInerney and D. Terzopoulos. Topologically adaptable snakes. In Proceedings of The Fifth International Conference on Computer Vision (ICCV’95), pages 840–845, Cambridge, MA, 1995.
- [86] G. Székely, A. Kelemen, C. Brechbüler, and G. Gerig. Segmentation of 2D and 3D objects from MRI volume data using constrained elastic deformations of flexible Fourier surface models. Medical Image Analysis, 1:19–34, 1996.
- [87] L. H. Staib and J. S. Duncan. Boundary finding with parametrically deformable models. IEEE Transactions on Pattern Analysis and Machine Intelligence, 14:1061–1075, 1992.
- [88] Y. Chen, S. Thiruvenkadam, H. Tagare, F. Huang, D. Wilson, and E. Geiser. On the incorporation of shape priors into geometric active contours. In IEEE Workshop on Variational and Level Set Methods, pages 145–152, Vancouver, CA, 2001.
- [89] T. F. Cootes and C. J. Taylor. Statistical models of appearance for medical image analysis and computer vision. In Proceedings of SPIE: Medical Imaging, volume 4322, pages 236–248, San Diego, CA, 2001.
- [90] L. Gong, S. D. Pathak, D. R. Haynor, P. S. Cho, and Y. Kim. Parametric shape modeling using deformable superellipses for prostate segmentation. IEEE Transactions on Medical Imaging, 23:340–349, 2004.
- [91] D. Freedman, R. J. Radke, T. Zhang, Y. Jeong, and G.T.Y. Chen. Model-based multi-object segmentation via distribution matching. In Third IEEE Workshop on Articulated and Nonrigid Motion (in conjunction with IEEE CVPR 2004), Baltimore, MD, 2004.
- [92] C. Brechbuhler, G. Gerig, and O. Kubler. Parametrization of closed surfaces for 3-d shape description. Computer Vision and Image Understanding, 61:154–170, 1995.

- [93] S. M. Pizer, P. T. Fletcher, S. Joshi, A. Thall, J. Z. Chen, Y. Fridman, D. S. Fritsch, A. G. Gash, J. M. Glotzer, M. R. Jiroutek, C. Lu, K. E. Muller, G. Tracton, P. Yushkevich, and E. L. Chaney. Deformable M-Reps for 3d medical image segmentation. International Journal of Computer Vision, 55:85–106, 2003.
- [94] J. A. Jensen. Field: A program for simulating ultrasound systems. Medical and Biological Engineering and Computing, 10:351–353, 1996.
- [95] J. J. Kaufman, G. M. Luo, and R. S. Siffert. Ultrasound simulation for 3d-axisymmetric models. In Proceedings of 2003 IEEE International Ultrasonics Symposium, pages 2065–2068, 2003.

Glossary and Acronym List

As a supplement, this part contains definitions of terms and descriptions of acronyms used in this thesis.

AV : the automatic segmented volume

AVD : absolute volume difference, the relative difference between **RV** and **AV** respect to the union of **RV** and **AV**

bMS : between target mean square

BPH : benign prostatic hyperplasia

CG : center of gravity

CT : computer tomography

DCP : the distance to the closest point

DDC : discrete dynamic contour

DRE : digital rectal examination

DWT : dyadic wavelet transform

EM : Expectation-Maximization algorithm

eMS : error mean square

FN : false negative volume, the volume enclosed by the manually outlined surface that is missed by the algorithm

FP : false positive volume, the volume enclosed by the algorithm detected surface but outside of the manually outlined surface

GA : genetic algorithm

GMM : Gaussian mixture model

ICC : intraclass correlation coefficient

inter-observer variability : disagreements between observations by multiple observers

intra-observer variability : disagreements between observations by an observer at different time

Level Set Method : a state-of-the-art method that tracks the motion of an interface by embedding the interface as the zero level set of a time-varying function in higher dimension

MASD : mean absolute surface distance

ML : maximum likelihood

MRF : Markov random field

MRI : magnetic resonance imaging

MSD : maximum surface distance

NURBS : nonuniform rational B-splines

PDF : probability density function

PDM : point distribution model

PSA : prostate specific antigen

RBR : radial bas-relief method

reliability : consistency, the amount of agreement between the results of repeated measurements, or assessments, of the same sample

ROI : region of interest

RV : the manual segmented reference volume

SDGD : second derivative in gradient direction

TP : true positive volume, the common voxels contained by both the manually outlined surface and the algorithm detected surface

TRUS : transrectal ultrasound

validity : accuracy, the degree to which a study accurately reflects the specific concept that the researcher is attempting to measure

VD : volume difference, the number of different voxels between **RV** and **AV** of the whole scan-set of a patient respect to the union of **RV** and **AV**

VOCAL : virtual organ computer aided analysis

wMS : within target mean square

APPENDIX A

Bubble 3D: A Java application for handling surfaces in volumetric medical images

Bubble 3D [34] is a Java application for handling a variety of medical images and creating anatomic geometries in the 3D domain, plus some value-added tools such as surface — image registration and virtual surgery simulation. Bubble 3D aims to be a cross-platform tool for off-line medical image analysis and off-line surgical planning in both 2D and 3D. Seven popular 2D image formats (TIFF, FlashPix, JPEG, GIF, BMP, PNG, PNM) and one ultrasound 3D image format (Kretz Voluson 530D) are currently supported in Bubble 3D.

A.1 Smart curve outlining

Bubble 3D offers a sophisticated curve-outlining tool based on non-uniform rational B-splines (NURBS) [75], supporting both interactive open and closed curves outlining. A few mouse clicks can define a smooth NURBS curve to describe boundaries of objects in medical images. An open curve is initialized by 2 points and may be deformed or extended by the new added points. A closed curve is initialized by 3 points and may be deformed to the desired shape by adding more points. Every user-defined point can be removed or deleted later on. Fig. A.1 and Fig. A.2 illustrate how the curves are defined.

Figure A.1: Open curve outlining.

Figure A.2: Closed curve outlining.

A.2 Surface construction

Given a series of parallel 2D curves, a high-quality parametrical surface can be easily constructed by skinning over them with the NURBS interpolation. Fig. A.3 demonstrates an open surface (baby face) construction and Fig. A.4 shows that how the prostate surface is constructed. The user can freely manipulate the surfaces by adjusting the curves afterwards.

A.3 Surface — Image registration

By using a Genetic Algorithm (GA), a previously constructed NURBS surface can be robustly registered to match the surface within corresponding 3D ultrasound volume data in dozens of seconds, assuming that these two surfaces have similar shape and size. This feature is useful for tracking a surface with known shape during volumetric imaging and for merging sets of 3D volume data of the same organ

Figure A.3: Baby surface: an open surface construction.

Figure A.4: Prostate surface: a closed surface construction.

(taken using different imaging modalities) using the organ surface as a common marker. The registration progress is illustrated in Fig. A.5.

A.4 Needling simulation

As animation is supported in Bubble 3D, virtual surgery simulation can be achieved by extending the software. Fig. A.6 illustrates an example of needling simulation

Figure A.5: Register a prostate surface with a 3D ultrasound image.

on a prostate model using Bubble 3D.

Figure A.6: Needling simulation on a prostate model.

A.5 Portable project data

Except the input images, all of the project data are stored in an Extensible Markup Language (XML) file. XML format is a text-based markup language that is fast becoming the standard for data interchange between different applications and OS platforms. Standard XML parsers written in different languages are available for free. It makes the results of Bubble 3D, especially the geometries, much easier to be

parsed by other applications. Besides, the constructed surface can be represented as a triangular mesh and output as a STL format file for follow-up use.

APPENDIX B

The STL Format

An STL (“StereoLithography”) file is a triangular representation of a 3-dimensional surface geometry. The surface is tessellated or broken down logically into a series of small triangles (facets). This very simple format has become an industry standard for the Rapid Prototyping sector. Virtually all modern CAD systems now include STL or Rapid Prototyping output as a standard feature.

B.1 Format specifications

An STL file consists of a list of facet data. Each facet is uniquely identified by a unit normal (a line perpendicular to the triangle and with a length of 1.0) and by three vertices (corners). The normal and each vertex are specified by three coordinates each, so there is a total of 12 numbers stored for each facet.

B.1.1 Facet orientation

The facets define the surface of a 3-dimensional object. As such, each facet is part of the boundary between the interior and the exterior of the object. The orientation of the facets (which way is “out” and which way is “in”) is specified redundantly in two ways which must be consistent. First, the direction of the normal is outward. Second, the vertices are listed in counterclockwise order when looking at the object from the outside (right-hand rule). These rules are illustrated

in Fig. B.1.

Figure B.1: Orientation of a facet is determined by the direction of the unit normal and the order in which the vertices are listed.

B.1.2 Vertex-to-vertex rule

Each triangle must share two vertices with each of its adjacent triangles. In other words, a vertex of one triangle cannot lie on the side of another. This is illustrated in Fig. B.2.

Figure B.2: The vertex-to-vertex rule. The left figure shows a violation of the rule. A correct configuration is shown on the right.

The figure on the left shows a top triangle containing a total of 4 vertex points. The outer vertices of the top triangle are not shared with and the lower 2 triangles each contain one of the points as well as the fourth invalid vertex point that lies on the side of the top triangle. To make this valid under the vertex-to-vertex rule the top triangle must be subdivided as in the example on the right.

B.2 STL file format

The STL standard includes two data formats, ASCII and binary. These two formats are described separately as follows.

B.2.1 STL ASCII format

For illustration, the following gives a sample ASCII description of a single triangle within an STL file.

```

solid name

...

facet normal   $n_x$    $n_y$    $n_z$ 

  outer loop

    vertex   $v1_x$    $v1_y$    $v1_z$ 

    vertex   $v2_x$    $v2_y$    $v2_z$ 

    vertex   $v3_x$    $v3_y$    $v3_z$ 

  endloop

endfacet

...

endsolid

```

The ASCII STL file usually has a filename extension of “.stla”. It contains a description of the surface of a solid that has been decomposed into triangles. The file begins with a **solid** record (which can include a name for the object), and ends with an **endsolid** record. Each triangle begins with a **facet** record and ends with an **endfacet** record. The normal vector, if given, is included as part of the facet record, and is identified by the **normal** keyword. The normal vector should have unit length. The three vertices of the triangle are delimited by **outer loop** and **endloop** records. Each vertex is described on a **vertex** record that lists its (x, y, z) coordinates. If the normal information is not included then the three values should be set to zero.

B.2.2 STL binary format

The binary format uses the IEEE integer and floating point numerical representation. Binary (.STL) files are organized as an 84 byte header followed by 50-byte records each of which describes one triangle facet as shown in next page.

Notice that each facet entry is 50 bytes. So adding the 84 bytes in the header space, a binary file should have a size in bytes = $84 + (\text{number of facets}) * 50$. Notice the 2 extra bytes thrown in at the end of each entry to make it a nice even 50.

The recent introduction of colour RP introduced the need for a 'colour STL' extension. One of these proposed extension make use of the 2 padding bytes to store the RGB data. However, this is beyond the scope of this appendix. Readers can refer to this web page (http://rpdrc.ic.polyu.edu.hk/old_files/stl_colour_binary_format.htm) to see the definition of the extension.

Bytes	Data Type	Description
80	ASCII	Any text such as the creator's name
4	unsigned long integer	Number of facets in file
⋮	⋮	⋮ } 50 bytes
4	float	x for normal
4	float	y
4	float	z
4	float	x for vertex 1
4	float	y
4	float	z
4	float	x for vertex 2
4	float	y
4	float	z
4	float	x for vertex 3
4	float	y
4	float	z
2	float	Padding bytes
⋮	⋮	⋮ } 50 bytes