IMPLEMENTATION OF THREE SEGMENTATION ALGORITHMS FOR CT IMAGES OF TORSO

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

SİNAN ÖZ

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

JANUARY 2011

Approval of the thesis:

IMPLEMENTATION OF THREE SEGMENTATION ALGORITHMS FOR CT IMAGES OF TORSO

submitted by **SİNAN ÖZ** in partial fulfillment of the requirements for the degree of **Master of Science in Electrical and Electronics Engineering Department, Middle East Technical University** by,

Prof. Dr. Canan Özgen
Dean, Graduate School of **Natural and Applied Sciences** _____

Prof. Dr. İsmet Erkmen
Head of Department, **Electrical and Electronics Engineering** _____

Assist. Prof. Dr. Yeşim Serinağaoğlu Doğrusöz
Supervisor, **Electrical and Electronics Engineering Dept., METU** _____

Examining Committee Members:

Prof. Dr. B. Murat Eyüboğlu
Electrical and Electronics Engineering Dept., METU _____

Assist. Prof. Dr. Yeşim Serinağaoğlu Doğrusöz
Electrical and Electronics Engineering Dept., METU _____

Prof. Dr. Nevzat G. Gençer
Electrical and Electronics Engineering Dept., METU _____

Assist. Prof. Dr. İlkay Ulusoy
Electrical and Electronics Engineering Dept., METU _____

Zeki Bozkurt, MSc.
Image Processing Dept., MGEO, ASELSAN _____

Date: _____

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: SİNAN ÖZ

Signature :

ABSTRACT

IMPLEMENTATION OF THREE SEGMENTATION ALGORITHMS FOR CT IMAGES OF TORSO

Öz, Sinan
M.Sc., Department of Electrical and Electronics Engineering
Supervisor : Assist. Prof. Dr. Yeşim Serinağaoğlu Doğrusöz

January 2011, 173 pages

Many practical applications in the field of medical image processing require valid and reliable segmentation of images. In this dissertation, we propose three different semi-automatic segmentation frameworks for 2D-upper torso medical images to construct 3D geometric model of the torso structures. In the first framework, an extended version of the Otsu's method for three level thresholding and a recursive connected component algorithm are combined. The segmentation process is accomplished by first using Extended Otsu's method and then labeling in each consecutive slice. Since there is no information about pixel positions in the outcome of Extended Otsu's method, we perform some processing after labeling to connect pixels belonging with the same tissue. In the second framework, Chan-Vese (CV) method, which is an example of active contour models, and a recursive connected component algorithm are used together. The segmentation process is achieved using CV method without egde information as stopping criteria. In the third and last framework, the combination of watershed transformation and K-means are used as the segmentation method. After segmentation operation, the labeling is performed for the determination of the medical structures. In addition, segmentation and labeling operation is realized for

each consecutive slice in each framework. The results of each framework are compared quantitatively with manual segmentation results to evaluate their performances.

Keywords: Image Segmentation, thresholding, watershed transform, active contour models

ÖZ

GÖVDE BT GÖRÜNTÜLERİ İÇİN ÜÇ BÖLÜTLEME ALGORİTMASI UYGULAMASI

Öz, Sinan
Yüksek Lisans, Elektrik ve Elektronik Mühendisliği Bölümü
Tez Yöneticisi : Yrd. Doç. Dr. Yeşim Serinağaoğlu Doğrusöz

Ocak 2011, 173 sayfa

Medikal görüntü işlemeleri alanındaki çoğu pratik uygulamalar geçerli ve güvenilir bir şekilde görüntü bölütlemeye ihtiyaç duymaktadır. Bu dokümanda, üst gövde 2B medikal görüntüler için üç farklı 3B yarı-otomatik bölütleme yöntemlerini medikal yapıların 3B modellerini tekrar oluşturmak için önermekteyiz. Birinci yöntemde, Otsu metodunun üç seviyeli eşikleme için geliştirilmiş versiyonu ile 4-Bağlantılı Ardışık Bağlanmış öğeler algoritması bir arada kullanılmaktadır. Bölütleme işlemi ardışık gelen kesitlerde sırasıyla Geliştirilmiş Otsu metodu kullanılarak ve etiketleme yapılarak gerçekleştirilir. Geliştirilmiş Otsu metodunun sonucunda piksel konum bilgisinin etkisi olmadığı için, etiketleme işleminden sonra bazı işlemleri gerçekleştirmeye ihtiyaç duymaktayız. İkinci yöntemde, aktif dış hat modellerinin örneği olan Chan-Vese methodu ve 4-Bağlantılı Ardışık Bağlanmış öğeler algoritması birlikte kullanılmaktadır. Bölütleme işlevi durdurma kriteri olarak kenar bilgisi kullanılmayan CV metodu kullanılarak yapılmaktadır. Üçüncü ve son olan yöntemde, havza dönüşüm ve K-means kümeleme algoritmalarının birleşimi, bölütleme methodu olarak kullanılmaktadır. Bölütleme işleminden sonra, etiketleme işlemi medikal yapıların belirlenmesi için gerçekleştirilmektedir. Ayrıca, her üç yöntemde, bölütleme ve etiketleme

işlemleri her ardışık kesit için ayrı ayrı gerçekleştirilmektedir. Her üç yöntemin sonuçları, performanslarını değerlendirmek için manuel bölütleme sonuçlarıyla karşılaştırılmaktadır.

Anahtar Kelimeler: Görüntü Bölütleme, eşikleme, havza dönüşüm, aktif dış hat modelleri

To My Family

ACKNOWLEDGMENTS

I am very grateful to my supervisor Assist. Prof. Dr. Yeşim Serinağaoğlu Doğrusöz for many fruitful suggestions and discussions, as well as to my family for their support and understanding at every stage of this study. I also would like to thank to my managers and colleagues for their priceless support and understanding.

TABLE OF CONTENTS

ABSTRACT iv

ÖZ vi

ACKNOWLEDGMENTS ix

TABLE OF CONTENTS x

LIST OF TABLES xiv

LIST OF FIGURES xv

CHAPTERS

1 INTRODUCTION 1

1.1 Motivation 1

1.2 Objective 2

1.3 Scope and contribution of the thesis 3

1.4 Thesis Outline 3

2 BACKGROUND 5

2.1 Applications of Medical Image Segmentation 7

2.1.1 Cardiac Segmentation in CT and MR Images . . . 8

2.1.2 Lung Segmentation 8

2.2 Methods of Medical Image Segmentation 9

2.2.1 Thresholding 11

2.2.2 Region-Based Methods 13

2.2.2.1 Watershed Transform 13

Oversegmentation: 13

Sensitivity to noise: 13

Poor detection of significant boundaries with low contrast: . . 14
Poor detection of thin structures: . . . 14
2.2.2.2 Region Growing 15
2.2.3 Boundary Dedection Methods 15
2.2.4 Atlas-Based Methods 17
2.2.5 Hybrid Methods 19
2.2.6 Other Segmentation Concepts 19
Neural Networks: 19
Normalized Cuts: 19
Fuzzy Connectedness: 20
2.3 The Image Dataset Used In This Dissertation 20
2.4 Some Operations in Medical Image Segmentation 21
Anisotropic Filter: 21
Morphological Operations: 22
2.5 Evaluation of Segmentation Results 22
2.5.1 Manual Segmentation For Comparison Purpose . . 22
Completely Manual Segmentation: . . 23
Manual Segmentation using Watershed Transform: 23
2.5.2 Quantitative Performance Measures 28
3 MEDICAL IMAGE SEGMENTATION USING THRESHOLDING . 31
3.1 Mathematical Theory 31
3.1.1 OTSU's Bi-Level Thresholding Method For Image Segmentation 31
3.1.2 Iterative (Optimal) Threshold Selection For Bi-Level Thresholding Method 33
3.1.3 Multi-Level Thresholding Using Extended Version of OTSU's Method 34
3.1.3.1 Experiments of Multi-Level Thresholding on A Single Slice 36
3.1.4 Component Labeling 43

3.2 A Framework For Sequential Segmentation on Medical Image Slices . . . 46
3.3 Results . . . 51
3.3.1 Quantitative Evaluation of Segmentation Results . 56
4 MEDICAL IMAGE SEGMENTATION USING ACTIVE CONTOURS 70
4.1 Active Contour Model Without Edges . . . 71
4.2 A Framework For Segmentation on Consecutive Medical Image Slices . . . 71
4.3 Results . . . 75
4.3.1 Quantitative Evaluation of Segmentation Results . 82
5 MEDICAL IMAGE SEGMENTATION USING WATERSHED TRANSFORM . . . 98
5.1 Mathematical Theory . . . 99
5.1.1 Watershed Definitions for Digital Images . . . 99
Algorithmic definition by immersion: . 99
5.1.2 K-Means Clustering . . . 100
5.2 A Framework For Sequential Segmentation on Medical Image Slices . . . 101
5.3 Results . . . 104
5.3.1 Quantitative Evaluation of Segmentation Results . 111
6 DISCUSSIONS AND CONCLUSIONS . . . 127
Lung Segmentation: . . . 129
Thoraxial Segmentation: . . . 129
Bone Segmentation: . . . 130
Cardiac Segmentation: . . . 133
REFERENCES . . . 134
APPENDICES
A SEGMENTATION VALIDATION METRIC . . . 145
A.1 Negative Rate Metric . . . 145
A.2 Quantitative evaluation based on calculating missed, extra segmented and overlapped regions . . . 146

B BACKGROUND: THE ACTIVE CONTOUR MODEL WITHOUT EDGES 147

B.1 The Chan-Vese Active Contour Model For Image Segmentation 147

The Level Set Formulation of the CV Model: 149

Numerical Approximations: 153

B.2 2-Phase Piecewise Constant Active Contour Model Without Edges 155

Numerical Approximations 159

C GATEAUX DERIVATIVES OF THE CHAN-VESE ENERGY FUNCTIONAL TERMS 162

C.1 Gateaux derivative of the term $\mu \int_\Omega \int \delta_\epsilon(\phi) |\nabla\phi| \partial x \partial y$ 162

C.2 Gateaux derivative of the term $\nu \int_\Omega \int H_\epsilon(\phi) \partial x \partial y$ 164

C.3 Gateaux derivative of the term $\lambda_1 \int_\Omega \int |I - c_1|^2 H(\phi) \partial x \partial y$ . . 164

C.4 Gateaux derivative of the term $\lambda_2 \int_\Omega \int |I - c_0|^2 (1 - H(\phi)) \partial x \partial y$ 165

D SOFTWARE USER GUIDES 166

D.1 Graphical user interface for filtering operation 166

D.2 Graphical user interface to monitor the segmentation results . 168

D.3 Graphical user interface for manual segmentation using watershed transform 168

D.4 Graphical user interface for the framework based on thresholding 169

D.5 Graphical user interface for the frameworks based on active contours and watershed transform 170

E HOMOGENEOUS REPRESENTATION OF LINES AND POINTS . 172

E.1 Representation of lines 172

E.2 Representation of points 172

E.3 Line joining points 173

LIST OF TABLES

TABLES

Table 2.1 Features of the image data set 21

Table 2.2 Parameters for the anisotropic diffusion filter 22

Table 3.1 The approximate durations for segmentation of lungs, heart and bones using thresholding 51

Table 3.2 The mean values and the standard deviations of NRM and SI for segmentation of lungs, heart and bones using thresholding 69

Table 4.1 The approximated durations for the framework based on active contour model 77

Table 4.2 The mean values and the standard deviations of NRM and SI for segmentation of lungs, heart and bones using thresholding 97

Table 5.1 The approximated durations for the framework based on watershed transform 106

Table 5.2 The mean values and the standard deviations of NRM and SI for segmentation of lungs, heart and bones using thresholding 126

Table 6.1 Numerical results for segmentation of lungs 128

Table 6.2 Numerical results for segmentation of torso 129

Table 6.3 Numerical results for segmentation of bones 131

Table 6.4 Numerical results for segmentation of heart 132

LIST OF FIGURES

FIGURES

Figure 2.1 Histogram of an image with two thresholds T_1 and T_2 dividing the histogram into three seperate regions. 12

Figure 2.2 One of the images in the CT dataset with mapping to 8-bit resolution 21

Figure 2.3 The original images (right) and the manual cardiac segmentation results (left) for (a) the slice 18, (b) the slice 30, and (c) the slice 38. The segmented objects are represented the color white. 24

Figure 2.4 The original images (right) and the manual bone segmentation results (left) for (a) the slice 11, (b) the slice 29, and (c) the slice 44. The segmented objects are represented with the color white. 25

Figure 2.5 The original images (right) and the manual lung segmentation results (left) for (a) the slice 14, (b) the slice 31, and (c) the slice 45. The segmented objects are represented with the color white. 26

Figure 2.6 (a) The problem of manual segmentation only using watershed transform (b) The proposed solution for this problem. 27

Figure 2.7 The original images (right) and the manual segmentation results using the watershed transform (left) for (a) the slice 23, (b) the slice 31, and (c) the slice 39. Background, lungs, torso, bones, and heart are represented the color black, blue, green, yellow, and red, respectively. 29

Figure 2.8 The original images (right) and the manual segmentation results (left) for the slice 31. Background, lungs, torso, bones, and heart are represented the color black, blue, green, yellow, and red, respectively. . . . 30

Figure 3.1 The selection of threshold values using Three-Level Extended Version of OTSU's Method 35

Figure 3.2 Original Image 37
Figure 3.3 The histogram of the original image 37
Figure 3.4 The outcome of the bi-level thresholding ($T = 574$). The pixel whose values are greater than the threshold are represented as white. The remains are black. 38
Figure 3.5 The result of the iterative selection method ($T = 574$), which is the same as that of the bi-level thresholding. 38
Figure 3.6 The outcome of the multi-level thresholding ($T_1 = 196$, $T_2 = 682$). The pixel whose values are greater than the threshold T_2 are represented as white. Other pixels whose values are between the thresholds are gray and the remains are black. 39
Figure 3.7 The outcome of the multi-level thresholding is shown as (a), (b) and (c). 40
Figure 3.8 The outcome of the multi-level thresholding ($T_1 = 196$, $T_2 = 682$, $T_3 = 1187$). The pixel whose values are greater than the threshold T3 are represented as white. Other pixels whose values are between the thresholds T_2 and T_3 are dark gray, others between the threshold T_1 and T_2 are light gray, and the remains are black. 41
Figure 3.9 The outcome of the multi-level thresholding is shown as (a), (b), (c), and (d). 42
Figure 3.10 The binary image before and after labeling 43
Figure 3.11 After first pass over an image 45
Figure 3.12 After second pass over an image 46
Figure 3.13 Flow Chart of The Proposed Framework 47
Figure 3.14 User Interference 48
Figure 3.15 Determination of the range of intesity which designate the object of interest 49
Figure 3.16 The binary image representing object and background distinction . 49
Figure 3.17 The outcome of the segmentation operation for a single slice 50
Figure 3.18 3D representation of the object 50

Figure 3.19 The segmentation of lungs for the slices (a) 18 and (b) 31 51
Figure 3.20 The segmentation of bones for the slice 11 52
Figure 3.21 The cardiac segmentation for the slice 18 52
Figure 3.22 3D representation of 2D segmentation of lungs from different views 53
Figure 3.23 3D representation of 2D segmentation of bones from different views 54
Figure 3.24 3D representation of 2D segmentation of heart from different views 55
Figure 3.25 Negative Rate Metric results based on comparison between the automatically and manually segmented images for lungs. 56
Figure 3.26 Missed and extra segmented rates based on comparison between the automatically and manually segmented images for lungs. 58
Figure 3.27 Overlapped rate and similarity index based on comparison between the automatically and manually segmented images for lungs. 58
Figure 3.28 Two examples of bad lung segmentation results (right); the slices (a) 11 and (b) 55. The original images (left) are also shown. 59
Figure 3.29 Two examples of good lung segmentation results (right); the slices (a) 25 and (b) 37. The original images (left) are also shown. 60
Figure 3.30 Negative Rate Metric results based on comparison between the automatically and manually segmented images for bones. 61
Figure 3.31 Missed and extra segmented rates based on comparison between the automatically and manually segmented images for bones. 62
Figure 3.32 Overlapped rate and similarity index based on comparison between the automatically and manually segmented images for bones. 62
Figure 3.33 Two examples of bad bone segmentation results (right); the slices (a) 01 and (b) 57. The original images (left) are also shown. 63
Figure 3.34 Two examples of good bone segmentation results (right); the slices (a) 23 and (b) 45. The original images (left) are also shown. 64
Figure 3.35 Negative Rate Metric results based on comparison between the automatically and manually segmented images for heart. 65
Figure 3.36 Missed and extra segmented rates based on comparison between the automatically and manually segmented images for heart. 66

Figure 3.37 Overlapped rate and similarity index based on comparison between the automatically and manually segmented images for heart. 66

Figure 3.38 Two examples of bad cardiac segmentation results (right); the slices (a) 25 and (b) 41. The original images (left) are also shown. 67

Figure 3.39 Two examples of good cardiac segmentation results (right); the slices (a) 17 and (b) 45. The original images (left) are also shown. 68

Figure 4.1 Flow Chart of The Proposed Framework 72

Figure 4.2 User Interference 73

Figure 4.3 The outcome of the segmentation operation for a single slice 74

Figure 4.4 3D representations of the objects 74

Figure 4.5 The segmentation results for slice 17 75

Figure 4.6 The segmentation results for slice 35 75

Figure 4.7 The segmentation results for slice 43 76

Figure 4.8 The segmentation results for slice 54 76

Figure 4.9 3D representation of 2D segmentation of lungs from different views 78

Figure 4.10 3D representation of 2D segmentation of torso from different views 79

Figure 4.11 3D representation of 2D segmentation of bones from different views 80

Figure 4.12 3D representation of 2D segmentation of heart from different views 81

Figure 4.13 Negative Rate Metric results based on comparison between the automatically and manually segmented images for lungs. 82

Figure 4.14 Missed and extra segmented rates based on comparison between the automatically and manually segmented images for lungs. 84

Figure 4.15 Overlapped rate and similarity index based on comparison between the automatically and manually segmented images for lungs. 84

Figure 4.16 Two examples of bad lung segmentation results (right); the slices (a) 07 and (b) 55. The original images (left) are also shown. 85

Figure 4.17 Two examples of good lung segmentation results (right); the slices (a) 29 and (b) 37. The original images (left) are also shown. 86

Figure 4.18 Negative Rate Metric results based on comparison between the automatically and manually segmented images for torso. 87

Figure 4.19 Missed and extra segmented rates based on comparison between the automatically and manually segmented images for torso. 88

Figure 4.20 Overlapped rate and similarity index based on comparison between the automatically and manually segmented images for torso. 88

Figure 4.21 Negative Rate Metric results based on comparison between the automatically and manually segmented images for bones. 89

Figure 4.22 Missed and extra segmented rates based on comparison between the automatically and manually segmented images for bones. 90

Figure 4.23 Overlapped rate and similarity index based on comparison between the automatically and manually segmented images for bones. 90

Figure 4.24 Two examples of bad bone segmentation results (right); the slices (a) 27 and (b) 51. The original images (left) are also shown. 91

Figure 4.25 Two examples of good bone segmentation results (right); the slices (a) 05 and (b) 31. The original images (left) are also shown. 92

Figure 4.26 Negative Rate Metric results based on comparison between the automatically and manually segmented images for heart. 93

Figure 4.27 Missed and extra segmented rates based on comparison between the automatically and manually segmented images for heart. 94

Figure 4.28 Overlapped rate and similarity index based on comparison between the automatically and manually segmented images for heart. 94

Figure 4.29 Two examples of bad cardiac segmentation results (right); the slices (a)25 and (b) 39. The original images (left) are also shown. 95

Figure 4.30 Two examples of good cardiac segmentation results (right); the slices (a)35 and (b) 51. The original images (left) are also shown. 96

Figure 5.1 Watershed transform by immersion on the 8-connected grid. (a): Original image; (b-g): labelling steps based on 5.1. 100

Figure 5.2 Flow Chart of The Proposed Framework 101

Figure 5.3 User Interference 102

Figure 5.4 The outcome of the segmentation operation for a single slice 103
Figure 5.5 3D representations of the objects 103
Figure 5.6 The segmentation results for slice 17 104
Figure 5.7 The segmentation results for slice 35 104
Figure 5.8 The segmentation results for slice 43 105
Figure 5.9 The segmentation results for slice 54 105
Figure 5.10 3D representation of 2D segmentation of lungs from different views 107
Figure 5.11 3D representation of 2D segmentation of torso from different views 108
Figure 5.12 3D representation of 2D segmentation of bones from different views 109
Figure 5.13 3D representation of 2D segmentation of heart from different views 110
Figure 5.14 Negative Rate Metric results based on comparison between the automatically and manually segmented images for lungs. 111
Figure 5.15 Missed and extra segmented rates based on comparison between the automatically and manually segmented images for lungs. 113
Figure 5.16 Overlapped rate and similarity index based on comparison between the automatically and manually segmented images for lungs. 113
Figure 5.17 Two examples of bad lung segmentation results (right); the slices (a) 07 and (b) 55. The original images (left) are also shown. 114
Figure 5.18 Two examples of good lung segmentation results (right); the slices (a) 29 and (b) 41. The original images (left) are also shown. 115
Figure 5.19 Negative Rate Metric results based on comparison between the automatically and manually segmented images for torso. 116
Figure 5.20 Missed and extra segmented rates based on comparison between the automatically and manually segmented images for torso. 117
Figure 5.21 Overlapped rate and similarity index based on comparison between the automatically and manually segmented images for torso. 117
Figure 5.22 Negative Rate Metric results based on comparison between the automatically and manually segmented images for bones. 118
Figure 5.23 Missed and extra segmented rates based on comparison between the automatically and manually segmented images for bones. 119

Figure 5.24 Overlapped rate and similarity index based on comparison between the automatically and manually segmented images for bones. 119

Figure 5.25 Two examples of bad bone segmentation results (right); the slices (a) 13 and (b) 51. The original images (left) are also shown. 120

Figure 5.26 Two examples of good bone segmentation results (right); the slices (a) 19 and (b) 33. The original images (left) are also shown. 121

Figure 5.27 Negative Rate Metric results based on comparison between the automatically and manually segmented images for heart. 122

Figure 5.28 Missed and extra segmented rates based on comparison between the automatically and manually segmented images for heart. 123

Figure 5.29 Overlapped rate and similarity index based on comparison between the automatically and manually segmented images for heart. 123

Figure 5.30 Two examples of bad cardiac segmentation results (right); the slices (a)21 and (b) 39. The original images (left) are also shown. 124

Figure 5.31 Two examples of good cardiac segmentation results (right); the slices (a)19 and (b) 37. The original images (left) are also shown. 125

Figure 6.1 The quantitative performance metrics for lungs using each method. (a) Negative rate, (b) Overlap Index, (c) Similarity Index 128

Figure 6.2 The quantitative performance metrics for torso using each method. (a) Negative rate, (b) Overlap Index, (c) Similarity Index 130

Figure 6.3 The quantitative performance metrics for bones using each method. (a) Negative rate, (b) Overlap Index, (c) Similarity Index 131

Figure 6.4 The quantitative performance metrics for heart using each method. (a) Negative rate, (b) Overlap Index, (c) Similarity Index 132

Figure A.1 Quantitative evaluation is performed by calculating missed, extra segmented and overlapped regions. 146

Figure B.1 A curve from the zero level set of the Lipschitz function ϕ is the boundary between the regions: $\{(x,y) : \phi(x,y) > 0\}$ and $\{(x,y) : \phi(x,y) < 0\}$.149

Figure B.2 An example of partition of the image in regions with boundaries represented via two level set of the Lipschitz functions $\{\phi_1 = 0\} \cup \{\phi_2 = 0\}$. 155

Figure D.1 GUI for filtering 166

Figure D.2 GUI to monitor the segmentation results 167

Figure D.3 GUI for manual segmentation using watershed transform 168

Figure D.4 GUI for the framework based on thresholding 169

Figure D.5 GUI for the framework based on active contours 170

Figure D.6 GUI for the framework based on watershed transform 171

CHAPTER 1

INTRODUCTION

1.1 Motivation

There is a rapid progress in the field of medical imaging. Owing to this, some imaging modalities, such as Computed Tomography (CT) and Magnetic Resonance Imaging (MRI), can be effectively used for non-invasive mapping of the anatomy of a subject. Thus, scientists and physicians have an opportunity to get a virtual information about anatomical structures via these modalities, which brings about the acquirement of potentially life saving information.

There are many research areas related to medical imaging, such as the enhancement of the image segmentation techniques to facilitate a more precise interpretation of the data and to help for training in diagnosis, the automation of medical image processing for minimization of manual intervention, and the development of acceleration techniques for reducing the processing time required for addressing the large amount of data obtained from medical capture devices.

The use of medical imaging varies according to field of study. Simple visualization and inspection of anatomic structures, patient diagnosis, advanced surgical planning and simulation or computer-aided surgery, radiotherapy planning, measurement of tissue volumes, and localization of tumors and other pathologies are the most common usage. Nowadays, there is little progress in the utilization of modern volume visualization techniques to perform precise and efficient analysis even though it guarantees exceedingly precise and high quality 3D view of anatomical structures.

In this thesis, we focus on the automatization of medical image segmentation and the visualization of anatomic structures. Combining different segmentation algorithms, filtering and morphological operations, we aim to propose a semi-automatic framework for sequential segmentation of 2D thoraxial CT images and to develop a visualization tool based on the library (OpenGL) for the reconstruction of anatomic structures using a simple surface meshing method.

1.2 Objective

The main objective of this thesis is the development of three semi-automatic torso segmentation frameworks using the image segmentation techniques available in the literature. These segmentation frameworks are used to delineate anatomical structures belonging to torso, such as lung, heart, and rib cage, and to reconstruct 3D geometric models of these structures. In order to reach this objective we aim to:

1. Propose new semi-automatic frameworks with least user intervention to delineate anatomical structures belonging to torso: Automatic segmentation approaches are achieved within exactly defined bounds. Hence, these approaches require the formalization of the necessary priori knowledge about the object of interest, which is a challenging task. Manual segmentation, on the other hand, requires a long amount of time and highly relies on the performance of the expert. We explore semi-automatic algorithms, which extract the object of interest with a little user interaction. Semi-automatic approaches are more robust and preferable than automatic approaches [1] and do not suffer from disadvantages of manual segmentation.

2. Develope a medical image segmentation tool with user friendly interface: For the optimal cooperation between computer-based image analysis algorithms and human operators, a proposed medical image segmentation tool should have user friendly interface so as to improve information flow between the tool and the operator. It should be comprehensible and easy to use.

3. Develope a visualization tool for displaying 3D geometric models of the structures: 3D medical visualization provides the operator with a 3D representa-

tion of the patient's anatomy reconstructed from the set of image slices. Most widely used rendering techniques are surface rendering and direct rendering (volume rendering). In this thesis, we aim to facilitate surface rendering.

1.3 Scope and contribution of the thesis

The scope of the thesis are:

- The development of three semi-automatic frameworks for segmentation of torso and its structures; lungs, heart and rib cage. Otsu's method based on [2, 3], Chan-Vese's level set method based on [4], and watershed transform based on [5] are used as the segmentation techniques. Besides these methods, morphological operations (erosion and dilation) and labeling operation based on [6] are also applied.

- The use of segmentation results for one (initial) slice as prior information for the segmentation of remaining slices to minimize user intervention.

- The developement of a visualization tool to display 3D geometries of segmented structures using the OpenGL library based on a simple surface rendering technique [7].

The main contribution of the thesis is new semi-automatic segmentation frameworks with least user intervention to delineate anatomical structures belonging to upper torso from CT images.

1.4 Thesis Outline

This dissertation is organized into six chapters. The first two introductory chapters make mention of previous work. The next three chapters deal with 3D segmentation methods, and the corresponding results. Eventually, a concluding chapter is submitted. In more detail:

- Chapter 2: Background: The applications and methods of medical image segmentation
- Chapter 3: Medical image segmentation using thresholding
- Chapter 4: Medical image segmentation using active contours
- Chapter 5: Medical image segmentation using watershed transform
- Chapter 6: Discussions and conclusions

CHAPTER 2

BACKGROUND

Medical imaging is the visualization process of the whole human body or some parts of it for clinical purposes and medical science. It provides a great knowlegde for scientist and physicians to reveal, diagnose or examine diseases and to study normal anatomy and physiology. Depending on patient population and a clinical problem, a variety of imaging modalities are available for use. Magnetic resonance imaging (MRI), computed tomography (CT), fluoroscopy, angiography, ultrasound, nuclear medicine and various imaging modalities can be given as some examples of medical imaging modalities [8].

Medical images designating the internal aspect of the body are supposed to be produced using nonivasive techniques since the physical penetration under the skin does not occur in the medical imaging modalities. In other words, we can not take a measurement at any internal point of the body in an invasive way. Nonetheless, by exposing the living tissue to some effects such as ultrasonic pressure waves and echos or X-ray radiation, we make observations outside of the body. Considering this sense, we can comprehend medical images as the outcome of mathematical inverse problems. Projection radiography is a good example to understand the medical images. X-rays, which are absorbed at different rates in different tissue types such as bone, muscle and fat, penetrate through the body, then X-ray radiation is observed on the other side. Since the same procedure is repeated multiple times with a different angle for each, one can reconstruct 2D images of the body slices using mathematical inverse problems.

In medical images, image intensities representing measurements is strictly related

to radiation absorption in X-ray imaging, RF signal amplitude in MRI, or acoustic pressure in ultrasound. Making a single measurement at each location of the part of the body brings about a scalar image. Making more than one measurement there (eg. dual-echo MRI) causes a multi channel image [9]. Images may be obtained in the continous domain or in discrete space. X-ray film and MRI can be given as examples of the former and the latter, respectively. In 2D discrete images, the location of each measurement is called a pixel and in 3D images, it is called a voxel.

A large number of images are acquired even after performing a single MR or CT scan. Moreover, the size of these images are growing day by day. Therefore, the use of computers have been required in order to facilitate the analysis of these images. Especially, computer algorithms to delineate anatomical structures and other regions of interest have been becoming increasingly significant in assisting and automating specific radiological tasks. Image enhancement, registration, segmentation, and visualization are the most common algorithms in biomedical-imaging applications [10]. These approaches are becoming more important in many medical applications, e.g., 1) diagnostics, 2) preoperative planning, 3) intraoperative navigation, 4) surgical robotics, 5) postoperative validation, 6) training, and 7) telesurgery [11].

Medical image segmentation, a widespread operation in the medical image applications, is the process of pixel or voxel classification in a medical image dataset. The outcome of the segmentation is the labeled image dataset indicating tissue type or anatomical structure of the pixel or voxel. The labels that result from this process have a wide variety of applications in medical research and visualization.

As far as the dimensionality is concerned, medical image segmentation algorithms operate in a 2D domain and a 3D domain. Some algorithms depending only on image intensities, such as thresholding, are independent of the image domain; however, certain algorithms, e.g., deformable models and region growing, incorporating spatial information may operate with a different way due to the dependency on the dimensionality of the image. In general, 2D methods are applied to 2D images, and 3D methods [12] are applied to 3D images. In some cases, on the other hand, 2D methods are applied sequentially to the slices of a 3D image [13]. This is prefered because of the fact that they are easy to implement, they have lower computational complexity,

and they need lower data storage capacity. Furthermore, certain structures are more easily defined along 2D slices.

Nowadays, tomographing imaging units, such as CT or MRI, are able to provide huge amounts of high resolution spatial data. Performing manual image segmentation on the data by experts takes so much time that the process in total would be inefficient and boring. In order to address this problem, automatic or semi-automatic segmentation algorithms, which extract the object of interest with none or little user interaction, are developed. Automatic methods can be applied succesfully within exactly defined bounds. However, semi-automatic approaches are more robust and preferable than automatic approaches since the formalization of the necessary priori knowledge about object of interest is challenging. On the other hand, finding efficient ways for information flow between user and computer is a essential challenge in semi-automatic approaches [1].

2.1 Applications of Medical Image Segmentation

Medical image segmentation have been applied to various medical image modalities. To illustrate, brain structures, such as the thalamus and the pallidum, can be segmented using MR images of the human brain or mass lesions can be detected in mammographic images. Applications of medical image segmentation can be classified in terms of the imaging modality in use and the object to be segmented. In the following paragraphs, some medical applications of the segmentation in the literature will be summarized by focusing on the applications related to this thesis.

The applications of medical image segmentation vary and are spread over many areas. We do not have any possibility to account for all of them in here due to fact that most of them are not in the scope of this dissertation. Segmentation of brain structures, segmentation in retinal imagery, and cardiac and lung segmentation are most common applications of medical imagery. In brain segmentation is commonly performed on MR images. Many of the studies [14–17] focus on the segmentation of brain structures. There are also some studies [18, 19] which emphasis on the detection of abnormal brain structures. In retinal segmentation, the applications emphasis on the

detection of retinal diseases [20] and crucial structures in the human retina [21, 22]. We account for cardiac and lung segmentation in detail with specific subsections. Other foremost applications of medical image segmentation are extraction of prostate [23], bone segmentation [24, 25], spinal MRI segmentation [26], segmentation of the structure with tubular surfaces [27–29], liver segmentation [30], segmentation of coronary vessels from digital subtracted angiograms [31], detection of variegated coloring in skin tumors [32, 33].

2.1.1 Cardiac Segmentation in CT and MR Images

Cardiac image segmentation is a growing and important research area of medical image analysis [34,35]. Segmentation and tracking of cardiac structures is used to assist physicians in various states of treatment of cardiovascular diseases. Because of the motion of the heart during a cardiac cycle and as a result of respiration, incorporating image data and temporal data becomes more important. Nowadays, 4D cardiac MRI and CT datasets are commonly used. For example, Lynch *et al.* [36] proposed a novel method for the segmentation of 4D (3D + time) cardiac MRI data using a priori knowledge about the temporal deformation of the myocardium. In addition, Dey *et al.* [12] developed a 3D semi-automated segmentation algorithm to segment the heart in the 4D-CT datasets acquired without contrast enhancement for use in estimating respiratory motion of the heart.

2.1.2 Lung Segmentation

In the literature, the segmentation studies related to human lung can be classified into two broad groups; human lung segmentation and pulmonary nodule segmentation. In the human lung segmentation, segmentation procedures aim to extract the whole lung or pulmonary structures no matter whether they suffer from some lung dieases or not. On the other hand, in the pulmonary nodule segmentation, segmentation procedures deal with only nodules in the lung. The identification of pulmonary nodules usually indicates the existence of lung cancer.

In computerized lung CT image analysis, automated lung segmentation is usually

realized a preprocessing step. Its robustness have influence on the performance of computerized noninvasive evaluation of numerous lung diseases, such as lung cancer, emphysema, pulmonary embolism, and interstitial lung disease (ILD), and other clinical practices, e.g., preoperative planning of lobectomies [37]. Thus, the development of an accurate automated lung segmentation scheme can be profoundly advantageous for clinical practices and quantitative lung image analysis. The strategies based on thresholding techniques [38] are relatively succesful to extract lung parenchyma from CT images because of obvious lower intensities of lung parenchyma in Hounsfield Unit (HU) than the surrounding structures. Using merely thresholding techniques brings about inaccurate segmentation results due to irregular boundaries. Sluimer *et al.* [39] presented a survey about computerized analysis of CT images of the lung. They summed up the problems causing inaccurate segmentation as high density pathologies such as juxtapleural nodules, posterior and anterior attachments between left and right lungs, false positive inclusion of other organs outside the lung (e.g., liver or spleen), and nonsmooth exclusion of the trachea and vessels in the hilar regions.

In order to make up for the errors from a single thresholding, there are a large number of approaches, such as [40–44], in the literature. Many of the studies were designed with consideration of specific applications, e.g., whole lung segmentation [37, 45], airway detection and analysis [46], lung nodule detection [47], lobe/fissure segmentation [48], and interstitial lung disease [49] and scleroderma disease [44] diagnosis. Masutani [50] proposed to address irregular segmentation boundary using radial basis functions (RBFs). Nonetheless, a set of points passing through the boundary of the anatomical structures should be manually specified by user.

The application of interest in this study is ultimately produce 3D geometric model of the torso; thus we need to segment all tissues with the torso. Infact, we focus on all tissues and organs in the upper torso and their 3D geometric model in this thesis.

2.2 Methods of Medical Image Segmentation

The medical image segmentation is considered a very difficult problem due to the imaging process itself and the complexity and variability of the anatomy that is rep-

resented in the images. In order to address this challenging problem, various segmentation methods have been developed in the literature. Most proposed methods are specific to a particular problem, e.g., anatomy specific assuming some *a priori* information that must either be built into the system or provided by a human operator. Therefore, the researchers dealing with this problem have no choice rather than to pay insufficient attention to most other problems.

In the literature, there are many papers reviewing various segmentation methods. In these papers, e.g., [9, 51], the methods of medical image segmentation were classified in many ways, according to different classification schemes. Nevertheless, we present here a broad classification of them into six groups; interactive thresholding [52, 53], region-based [16], boundary detection [54], clustering-based [55, 56], atlas-based [15, 57, 58] and hibrid [59, 60] methods. In thresholding, intensity values of all pixels in an image are compared with a threshold or the intensities that lie within a particular range. The algorithm decides whether the corresponding pixel belongs to foreground or background with respect to the comparison result. The key point to this algorithm is the selection of the threshold value. Boundary-based methods utilize such edge detection techniques as a gradient filter in order to to extract boundaries of different segments. On the other hand, this makes it sensitive to image noise. Region-based segmentation techniques, which are based on spatial-information, argue that pixels in the same segment have similar features. The general growing procedure begins with some initial points, called as seeds. Then, it compares every seed with its surrounding neighbors. Depending on a certain merging criterion is satisfied, the procedure decides whether the neighboring pixel is added to the same segment with the seed or not. Naturally, the selection of a merging criterion is superior to determine the success of the segmentation using this method. Moreover, most of the algorithms based on region splitting and merging are sensitive to the start position and the performance sequence of the initial seed points. Alternatively, Zheng *et al.* [61] proposed an unseeded region-growing algorithm for image segmentation. Clustering-based techniques are also widely used image segmentation. For example, k-means clustering is a classical and powerful clustering algorithm which is still applied to many applications [55, 56]. The advantages and disadvantages of all these segmentation techniques compel researcher to combine more of them into hybrid methods.

In the next subsections, methods of medical image segmentation are described in detail with six categories; thresholding, region-based methods, boundary detection methods, atlas-based methods, hybrid methods and other segmentation concepts.

2.2.1 Thresholding

Thresholding is a simple but effective technique to separate objects from the background in medical image segmentation [51]. In this technique, a range of intensity levels, whose boundaries are called *thresholds*, are used to create binary partition for each material class of a medical image. All pixels with intensities between these two thresholds are grouped together into one class and those with intensities staying out of the range are grouped together into another class. In other words, the output of the thresholding operation in the medical imagery is a binary image whose one state will indicate the foreground objects, i. e., some structures belonging to the body, while the complementary state will correspond to the background.

Various factors cause to complicate the thresholding operation. Nonstationary and correlated noise, ambient illumination, frequency of gray levels within the object and its background, inadequate contrast, and object size not commensurate with the scene can be given as some examples of the factors. In addition, the lack of objective measures to assess the performance of various thresholding algorithms, and the difficulty of extensive testing in a task-oriented environment, have been mentioned as other major handicaps [62].

Thresholding can be performed using single threshold or multiple thresholds. Using a single threshold, it is possible to generate a binary segmented image. In this case, the range is defined as the intensity levels between the single threshold and the minimum intensity level of an image, usually 0. If multiple thresholds are in use for the segmentation, this is known as multithresholding [63].

Any region in the histogram of an image is defined by two thresholds; a lower threshold and an upper threshold. Then, every pixel in the image is compared with this region. If the pixel lies into the range it will be tagged as foreground, and if it is out of the range as background (see Figure 2.1). In images with a very good contrast

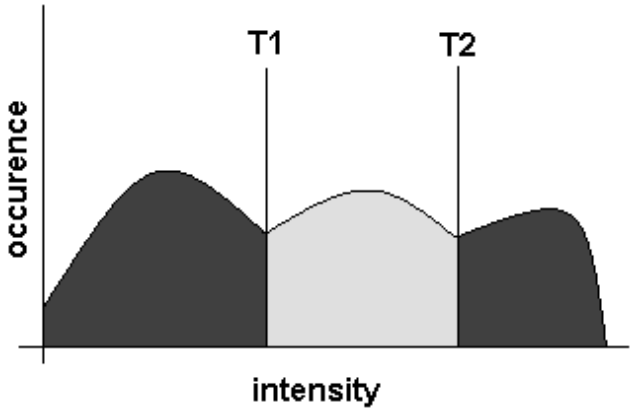


Figure 2.1: Histogram of an image with two thresholds T_1 and T_2 dividing the histogram into three seperate regions.

between regions, this technique is quite satisfactory to perform segmentation. The thresholds in use profoundly concern the quality of the segmentation results. In other words, a little change in the threshold values causes a different segmented region. This is the main drawback of this technique. In order to compensate for these effects, automatic, local and adaptive solutions can be applied [63]. Another drawback is that thresholding is very sensitive to noise and intensity inhomogeneities. For this reason, preprocessing on a raw image, such as filtering and image enhancement, is required to improve the results of the segmentation. However typically thresholding does not take into consideration the spatial characteristics of an image, some post-processing techniques brings to a succesful conclusion to cope with this.

In the literature, there is a remarkable number of surveys on thresholding. Sahoo *et. al.* [63] prepared a survey about nine thresholding algorithms and analized their performance in a comparative manner. Lee *et. al.* [64] carried out an analysis in order to compare five global thresholding methods with eachother and proposed useful criteria for assessment of thresholding performance. In addition, Glasbey [65] conducted an extensive statistical study to indicate the relationships and performance differences between 11 histogram-based algorithms. Sezgin and Sankur [62] described 40 thresholding algorithms with the idea underlying them, categorized them according to the information content used.

2.2.2 Region-Based Methods

Watershed transform and region growing are commonly known examples of region-based segmentation approaches. Watershed transform and region growing are shown as examples of region splitting and merging methods, respectively. In the following subsections, watershed transform and region growing methods are explained.

2.2.2.1 Watershed Transform

The watershed transform, originally proposed by Digabel and Lantuéjoul [66] and later improved by Beucher *et al.* [67], is the method of choice for image segmentation in the field of mathematical morphology. The intuitive description of this method comes from geography: a grey-level image may be considered as a landscape or topographic relief, where the grey level of pixel is interpreted as its altitude in the relief, and a drop of water falling a topographic relief flows along a path to finally reach to local minimum. At the end, the watersheds are represented as the lines seperating the drainage areas (also called catchment basins) over region [68]. Actually, instead of the original image, the watershed transform is applied to its gradient.

There are several disadvantages of watershed transform: (1) oversegmentation, (2) sensitivity to noise, (3) poor detection of significant boundaries with low contrast, (4) poor detection of thin structures.

Oversegmentation: The watershed transform produces plenty of small regions since instead of the original image, catchment basins are computed on its gradient. Number of resultant regions should be minimized to achieve the useful segmentation results. Using marker image, which causes to decrease in the number of minima in the image, is the solution to the minimization of region quantity [69, 70]. Moreover, The usage of a scale space approach to choose the regions of interest with such different filters as morphological operations [71] or nonlinear diffusions [72] is also conducted.

Sensitivity to noise: Local variations of the image causes the results of the transform to change dramatically. This effect deteriorates due to the use of the gradient

estimation amplifing the noise. In order to lower the problems stemmed from the local variations, anisotropic filters have been used [73].

Poor detection of significant boundaries with low contrast: If the region of interest has not high enough contrast boundaries, the watershed transform is not capable of the accurate segmentation. In addition, the detected contours does not always correspond to the contours of interest although the watershed transform intuitively extracts those with higher value between markers [59]. For the plain watershed transform, the boundary surface detection between white matter and gray matter is more difficult than such surfaces with higher contrast as the boundaries between gray matter and CSF or between CSF and bone, for example.

Poor detection of thin structures: The gradient estimation cause the image to smooth due to usual approach of storing gradient values only at the image pixel positions instead of storing them with sub-pixel accuracy when the watershed transform is performed on the gradient image. This brings about the difficulty in the detection of thin catchment basin areas, which is crucial for medical images segmentation.

A number of advantages of the watershed transform exits; it is fast, simple, and intuitive. The parallel implementation of this method is also possible. Moga and Gabbouj [74] reported that the process could be linearly speeded up for a number of processors up to 64, for instance. In addition, even if the contrast of the image is poor, its complete partitioning with separated regions is garanteed, so there is no need to join any kind of contour. In some studies [75, 76], the techniques to embed the watershed transform in a multiscale framework were proposed so that its advantages can be utilized to enhance the accuracy of their segmentation tools. Furthermore, some advantages of the watershed transform over active contours or surfaces were presented in [77]. In energy minimization methods, noise brings about erroneous results because of the fact that it can produce local minima. However, the watershed transform is not affected by lower-contrast edges since the watershed lines always correspond to the strongest edges between the markers. In addition to this, the watershed transform always extracts a contour between the markers even if there are only weak edges in that area. This contour will be positioned where the pixels have higher contrast.

2.2.2.2 Region Growing

Region growing is a bottom up technique which extracts an image region connected based on some similarity constraints. Region is iteratively grown from the seed subregion by adding in neighbouring subregions that are similar based on some predefined criteria, increasing the size of the region, where subregions usually correspond to pixels or voxels in the image or to catchment basins in watershed transform. The growing procedure is continued until all subregions belong to some region. The predefined criteria, actually similarity constraints, can be defined based on intensity information, edges in the image, and/or the output of any other segmentation algorithm [78]. In simplest region growing algoritm, a seed point that is manually selected by an operator is required and the extraction of all pixels connected to the initial seed based on any predefined criteria is realized. In order to determine the seed points, user interaction is required, which is its primary drawback. In other words, one must plant a seed for each region to be extracted. Split and merge is a region growing related algorithm which does not need a seed point [79]. Another drawback of the region growing is the sensitivity to noise. This causes the extracted regions to posses holes or become disconnected. On the contrary, partial-volume effects could lead to merger of separate regions. a homotopic region-growing algorithm [80] has been developed to address these problems, where the topology between an initial region and an outcome is planted. In addition, fuzzy analogies to region growing have been used [81]. In medical image processing, region growing is rarely used on its own. In general, it is used with other image processing operations. Especially, To delineate small and simple structures such as lesions and tumors are common medical applications of region growing [82, 83].

2.2.3 Boundary Dedection Methods

Boundary dedection methods have had a critical role in 2D and 3D medical image segmentation due to the fact that it has been commonly used for clinical purposes and medical science [84]. Over the last decades, the number of the applications of boundary detections have been growing owing to faster, more robust, and less artifacted scanning methods. Because of the complexity of medical structures, the

large variability in shapes, some kinds of noise, and limited body scanning methods, the boundary detection of anatomic structures in medical imagery is more difficult than in other imaging fields. Despite the before mentioned difficulties, there have been some studies obtaining faster and more accurate applications in 2D and 3D. In this section, we attempt to summarize the lastest techniques in 2D and 3D for boundary detections using deformable models, also known as active contour models.

In recent years, active contour models have been increasingly and widely used in image segmentation. These methods are basically described as follows. First of all, a close curve, also known as the evolving curve [4, 85, 86], is determined for the initialization in the image to be segmented. Then, driving the curve by a partial differential equation (PDE), it is evolved until the evolving curve converges. According to the representation of the evolving curve, active contour models can be classified as explicit [87, 88] and implicit [4, 89, 90] categories. Snake, a typical explicit active contour model, uses parametric equations to explicitly represent the evolving curve. Implicit active contour models, i.e., level set methods [86], replace the parametric curve with a signed distance function, i.e., the level set function. A level set function [86] is a real-valued function of multiple variables; when the function takes a constant value, e.g., zero, the obtained set is the zero level set which is used to represent the evolving curve. Thus, the evolving curve is implicitly represented by the zero set of the level set function. This representation results implicit active contour models that could handle the topological changing more conveniently than the explicit active contour models [90].

Based on the way defining boundaries, level set methods can further be categorized into either edge-based [89, 91] or region-based [4, 92] algorithms. Edge-based level set methods consider boundaries as a kind of discontinuation in gray values. To detect this discontinuation, these methods all define an edge indicator that is a positive and nondecreasing function. In theory, the evolving curve should stop where the edge indicator is equal to zero. However, in practice, due to the existence of noise, edge-based level set methods often cannot stop at a boundary location. Additionally, it is not easy for this kind of methods to detect weak and interior boundaries in images. To address these problems, Chan and Vese [4] proposed a region-based level set method called Chan-Vese (CV) method which incorporates the Mumford-Shah [93]

functional into a level set framework to give a piecewise constant representation to an image. The main idea of this method is to find the similarity in a given image, and it regards the subregion inside the evolving curve as a uniform object and the subregion outside as the background. The evolving curve is driven by an energy functional which is defined on the level set function and incorporates a "fitting" term. The "fitting" term defines the extent that the piecewise constant representation fits the initial image.

The CV method [4] utilizes a scalar, i.e., the gray value, to represent a pixel, but a scalar cannot comprehensively describe other information in an image, such as gradient and orientation information. Additionally, this method does not work well for images with noise, particularly the salt and pepper noise. Chan et al. [94] essentially extended the traditional CV method [4] from a scalar field to a vector field, but this method was mainly applied on the segmentation of multichannel images. That is to say, the representation of each pixel is still a scalar in each channel. Due to its vector form, this model ignores the spatial structure information among multichannel images. Wang and Vemuri [92] proposed a level set method, the Wang-Vemuri (WV) method, to segment a symmetrical tensor field, e.g., diffusion tensor magnetic resonance images (DTMRIs). This method can also be used to segment texture images by using a local structure tensor (LST) [95]. However, an LST just takes into account the horizontal and vertical direction information and ignores a basic important component, i.e., the gray value, which makes the application of this method limited. Additionally, this method is only applicable for the symmetry tensor field.

2.2.4 Atlas-Based Methods

Atlas-based segmentation methods have been effectively used for segmentation purposes in the various image modalities [96–98]. Compared with other segmentation methods, it has a significant advantage. That is to say, using a previously segmented image as a reference that shows how to make the segmentation of structures, it permits the use of a priori information on the simple shape and distribution of the segmented structures.

Segmentation procedure for new images can be realized in principle using only an

atlas image. On the other hand, recent studies, such as, [99–102] indicate that the use of multiple atlas images produces better segmentation results. Knowledge about a number of references could be merged to an average atlas [103] or to a so-called probabilistic atlas including probability values for each particular location [96, 100]. Thus, it is desired that the atlases consist of all the variability of a given population. Nevertheless, the multiple altases should be registered to the target image independently and the resultant segmentations combined in order to obtain maximum benefit of those [101].

An analogy can be made between the combination of segmentations derived independently from multiple atlas images and the combination of multiple independent classifiers in a generic classification problem [104]. In this analogy, each transformed atlas image can be regarded as a classifier, which assigns a label value to each voxel of the target image. The training process can be assimilated to the registration between the atlas image and the target image. It has been widely proven in the pattern recognition field that combining multiple classifiers can yield more robust and accurate results than using single classifiers [105], this fact being the main motivation for multi-atlas approaches.

The most widely used combination strategy in the literature is majority voting, also named majority rule, decision fusion or label voting. This approach weights each candidate segmentation equally and assigns to each voxel the label that most segmentations agree on [101, 102]. Another popular approach is called simultaneous truth and performance level estimation (STAPLE), which uses an expectation-maximization (EM) approach to reach the best possible final segmentation [106, 107]. STAPLE estimates the performance of each classifier iteratively and weights it accordingly. The two different methods presented in [107] are extensions of the one in [106] for images with multiple segmented structures. Shape-based averaging represents another way of combining segmentations [108] which is based on Euclidean distance maps computed for all structures in each candidate segmentation. The method was shown to keep structure regularity and contiguity better than majority voting. Another possibility is atlas selection: instead of combining segmentations, methods can be devised to select atlases a priori (before registration) or a posteriori (after registration) [101]. In [109], a number of atlases is selected for combination based on mutual informa-

tion [110]. In [111], atlas selection is done on a structure basis for the segmentation of brain magnetic resonance (MR) images. For each given structure, the atlas image with highest local mutual information is selected.

2.2.5 Hybrid Methods

In this subsection, we will look at the segmentation using something from more than one of the previous kinds of segmentation algorithms.

2.2.6 Other Segmentation Concepts

There are other segmentation methods, such as neural networks [112], Bayesian classifiers [20, 113], Hidden Markov Model [114], normalized cuts [115], LEGION [116], and fuzzy theory [117], which are commonly used in medical image analysis. In the following paragrahps, some of the aforementioned methods are described briefly since all these methods in this subsection are out of scope of this dissertation.

Neural Networks: In a neural network framework for image segmentation, neural network is usually applied with basic segmentation operations, such as thresholding, region growing, or etc, since neural networks requires some features describing the object to be segmented. Some examples of the features are area, circularity, inertial momentum, mean and standart deviation of radial length and intensity, entropy of intensity distribution, fractal index, eccentricity, anisotropy and etc. For example, Cascio *et al.* [112] proposed an algorithm depending on an edge-based threshold operator strategy for the mass segmentation. From the selected regions of interest, they extracted 12 features to discriminate between the two classes; pathological patients or healthy subjects.

Normalized Cuts: In the NCut, segmentation is considered as a graph-partitioning problem: "it maximizes both the total dissimilarity between the different groups and the total similarity within the groups [118]." The NCut segmentation technique has

a tendency to merge different features, e.g., windowed histograms, position, brightness. The combination of various features expand its application areas using different imaging modalities.

Fuzzy Connectedness: In images, regions belonging to object appear with a variety in intensities due to the actual object material property as well as due to such artifacts as noise, blurring, and variations in background orginated from the imaging modalities. Object regions can be recognized easily by observers as a gestalt despite this gradation of intensities. With a fuzzy topological consept called fuzzy connectedness, this scheme intends to mimic what the observer do. Fuzzy connectedness denotes how the image components be connected together spatially [119]. During recongnizing object in an image, the connectedness strength between every pair of the image components and all possible connection paths between the pair is considered. Modern workstations' computational speed give a chance to extract objects from an image using theoretical advances in fuzzy connectedness via dynamic programming in spite of high combinatorial complexity. Fuzzy connectedness framework are used in several medical applications in the areas of computer tomographic (CT), magnetic resonance (MR), angiography, mammography, and colonography as well as for detection of tumors and multiple sclerosis of the brain and upper airway disorders in children [117].

2.3 The Image Dataset Used In This Dissertation

In general, CT or MR Medical images have 12-16 bits resolution in gray scale. The dataset used in this study is obtained with CT medical imaging modality. The dataset comprise 61 CT images of upper torso in DICOM format, with features given Table 2.1.

In Figure 2.2, one of the images from the data set is depicted with mapping to 8-bits resolution. It is obvious that the contrast between organs or tissues (such as heart, fat and muscle tissue) to be segmented is lower with compared to between lungs and torso. Chest CT contains four significant anatomical tissues: bone, lung, heart and

Table 2.1: Features of the image data set

Photometric interpretation	Grayscale
Size	512x512
X Voxel Width	0.75mm
Y Voxel Width	0.75mm
Slice Thickness	5mm
Bits Allocated	12 Bits

soft tissue.

Figure 2.2: One of the images in the CT dataset with mapping to 8-bit resolution

2.4 Some Operations in Medical Image Segmentation

Anisotropic Filter: The way for effective suppression of imaging noise is anisotropic filtering, which preserves fine details such as region boundaries during this. In general, conventional linear filters, e.g., Gauss filters, makes sharp boundaries and small

structures of an image become blurry. In 1990, Perona [120] developed the Perona-Malik anisotropic diffusion method in order to remove the side effects of conventional linear filters. Perona also studied about the influence of the different parameters in filter functional on image boundaries. Multichannel anisotropic diffusion for smoothing in brain MR images are developed based on Perona's study by Gerig [121]. To effectively reduce noise in homogeneous regions and preserve object boundaries are its significant benefits. It also strengthens edges in the image. In fact, anisotropic filters is commonly used in the preprocessing steps of medical image segmentation [122].

We applied anisotropic diffusion filter proposed by Perona and Malik [120] with the parameters Table 2.2 before each segmentation frameworks.

Table 2.2: Parameters for the anisotropic diffusion filter

Diffusion Const. (K)	30
Lambda	0,25
Number of Iteration	50

Morphological Operations: Morphological operations involve filtering a labelmap such that the boundary of a labeled region either grows (dilation) or shrinks (erosion). Sequences of morphological operations can augment manual segmentation by filling in small holes or breaking connections between regions.

2.5 Evaluation of Segmentation Results

2.5.1 Manual Segmentation For Comparison Purpose

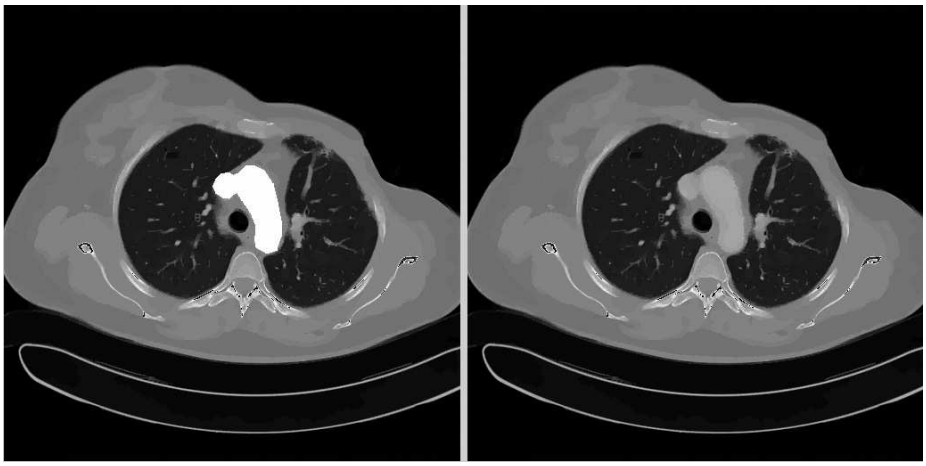
In order to evaluate quantitatively the performance of our proposed segmentation algorithms, the ground-truth reference images are required. One way to obtain the ground truth images is by manual segmentation performed by an expert. Then the performance can be measured by comparing the results of proposed segmentation algorithm with the manually segmented images. Although in this methodology, perfect ground truth images are not guaranteed due to the complete dependency on the

performance of the expert, the manual segmentation results are close to the ground truth that they can be used for validation of our results. Manual segmentation can be performed by delinating the boundaries or areas of the objects to be segmented with a completely manual manner or it can be realized by assisting to an expert with some computerized operations such as iso-intensity contours or watershed regions.

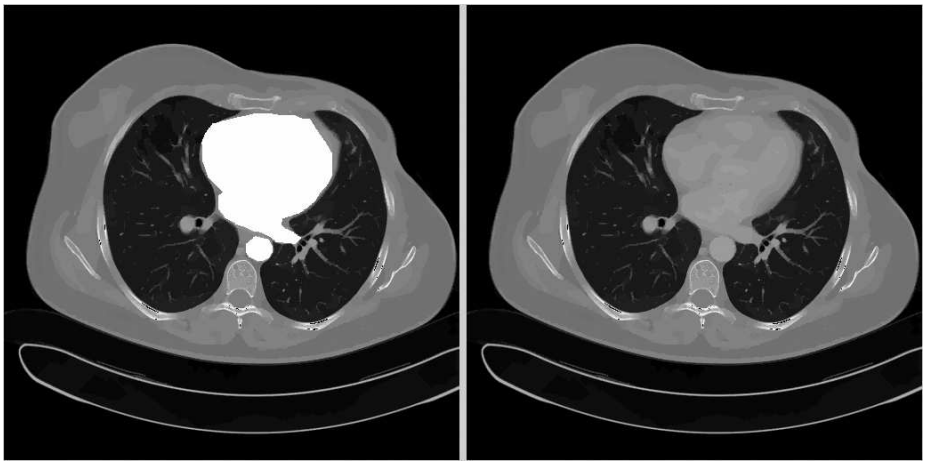
Completely Manual Segmentation: The completely manual segmentation is used to delineate the desired contours directly onto the raw image. In this kind of segmentation, the expert with knowledge of anatomy utilizes a mouse-based software to delineate the boundaries of the objects of interest on each slice in an image stack. The manual cadiac segmentation and bone segmentation for three different slices are depicted in Figure 2.3 and Figure 2.4, respectively. The manually segmented images and corresponding original images of lung for three different slices are shown together in Figure 2.5. In each image, the manually segmented objects are represented by the color white.

In real images, boundary of a segmented region is a smooth closed curve. The smooth closed curve can be represented by a sufficient number of straight lines to obtain piecewise approximation of the closed curve. In the completely manual segmentation, an expert selects a sufficient number of points to represent the closed curve. Here, the accurate selection and the sufficient number of these points are the critical issues. For this reason, unwillingly selecting faulty points or having insufficient numbers lead to the deformation of the real shape of the region to be segmented. To obtain fair representation of the boundaries in CT images, the number of pixels selected by an expert should be about 100. Thus, this method is very exhausting due to the fact that the expert spends too much time to perform completely manual segmentation. Because of the fatigue, this causes him or her to have a tendency to make errors. Therefore, we need to find an easier way for manual segmentation. In this study, we performed manual segmentation using watershed transform.

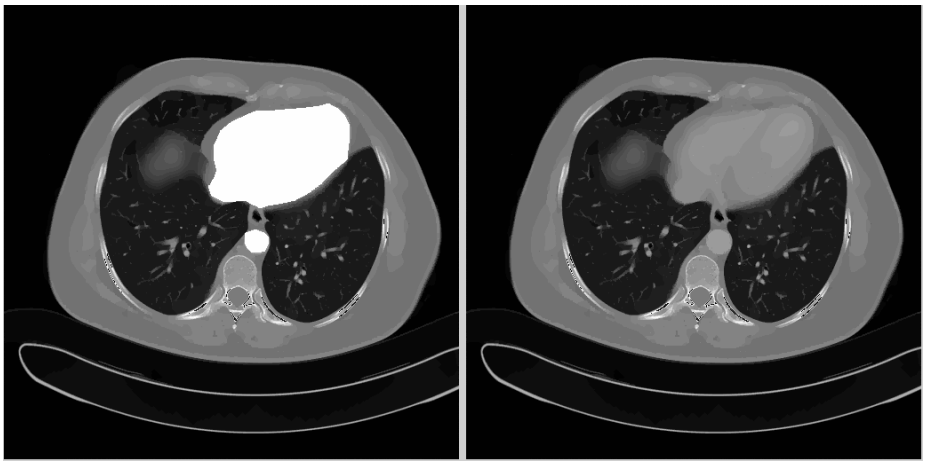
Manual Segmentation using Watershed Transform: In order to eliminate the drawbacks of the completely manual segmentation procedure, we developed a manual segmentation tool based on the watershed transform, which partitions the image



(a)

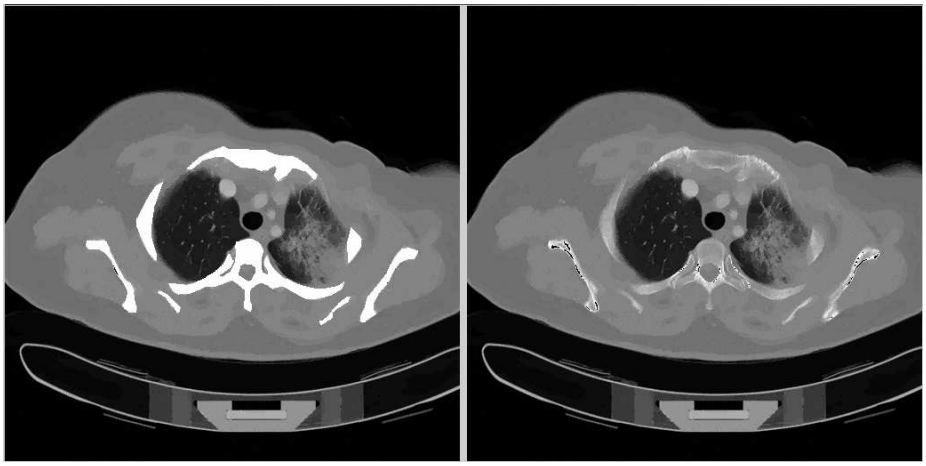


(b)

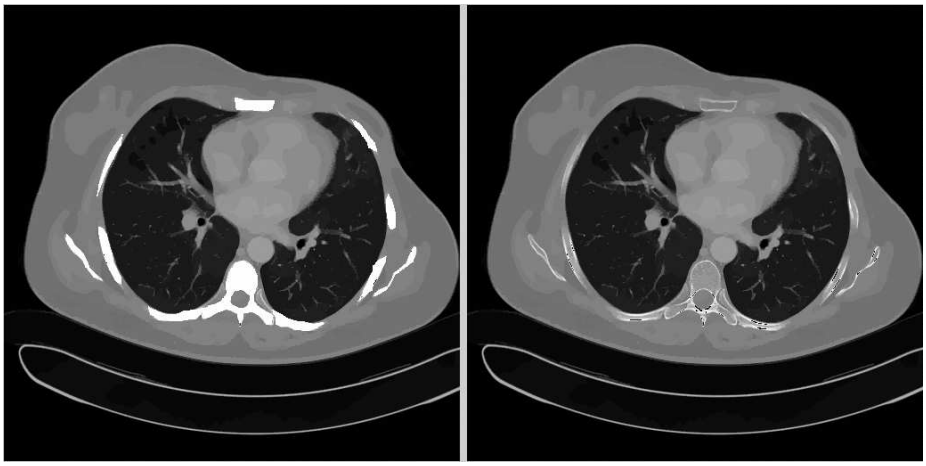


(c)

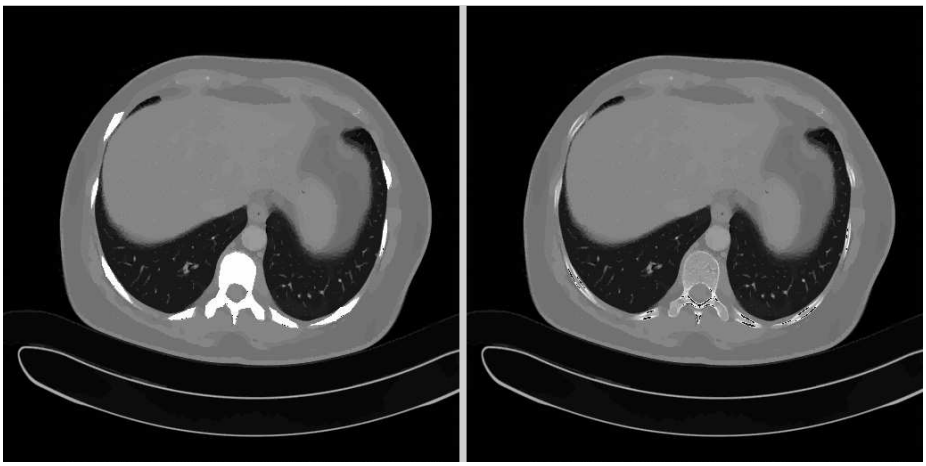
Figure 2.3: The original images (right) and the manual cardiac segmentation results (left) for (a) the slice 18, (b) the slice 30, and (c) the slice 38. The segmented objects are represented the color white.



(a)

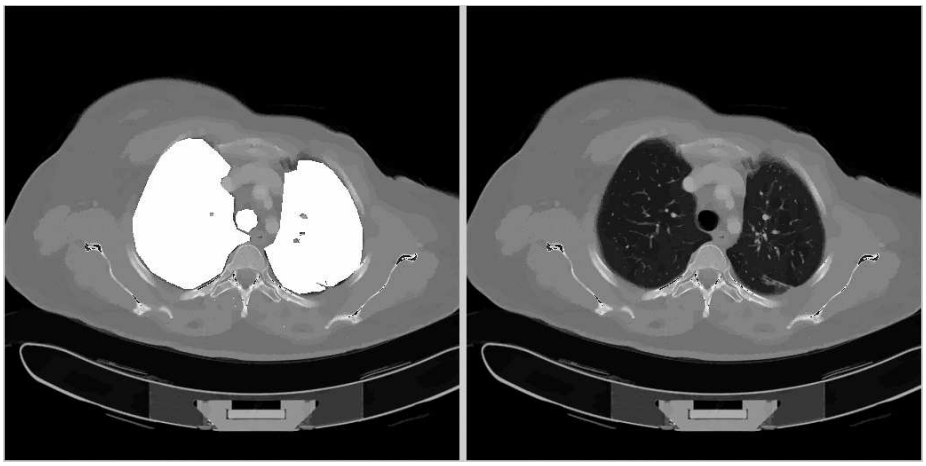


(b)

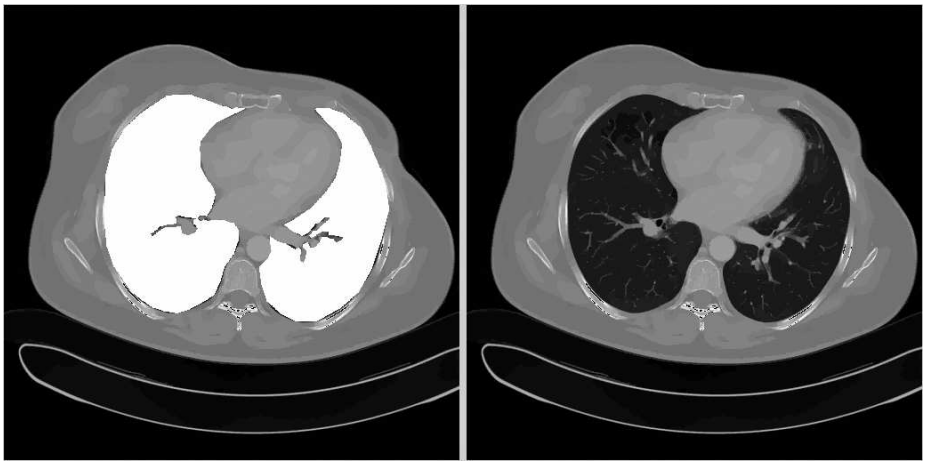


(c)

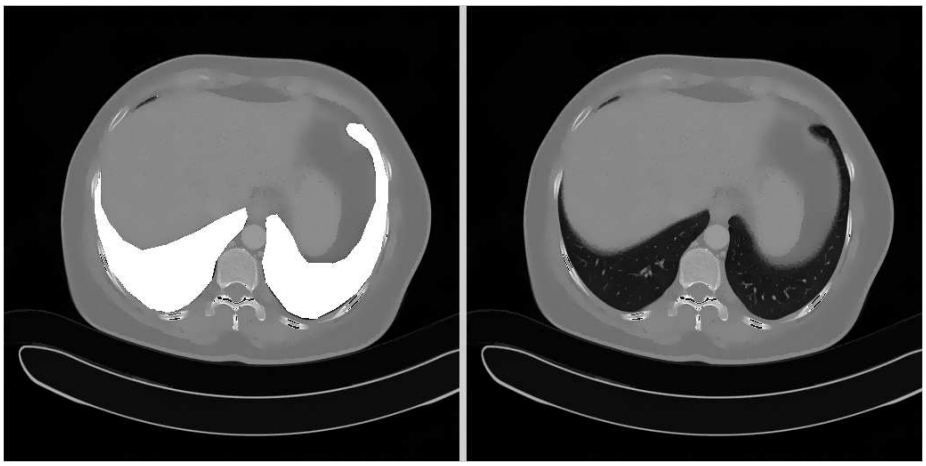
Figure 2.4: The original images (right) and the manual bone segmentation results (left) for (a) the slice 11, (b) the slice 29, and (c) the slice 44. The segmented objects are represented with the color white.



(a)



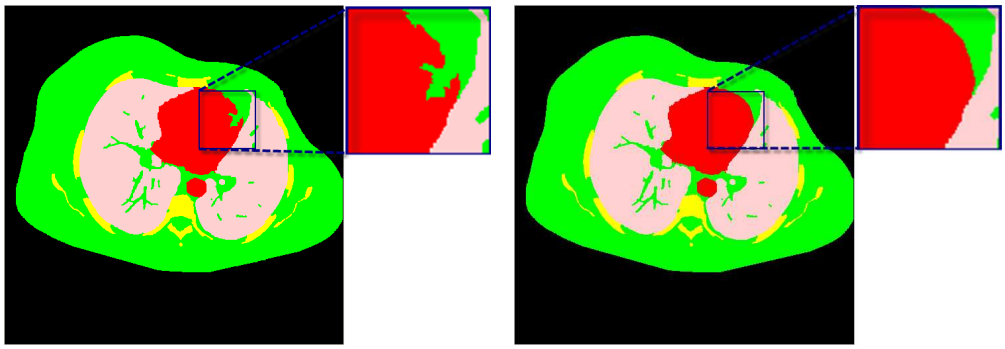
(b)



(c)

Figure 2.5: The original images (right) and the manual lung segmentation results (left) for (a) the slice 14, (b) the slice 31, and (c) the slice 45. The segmented objects are represented with the color white.

into disjoint homogeneous regions with respect to the pixel intensity value. This tool makes the manual segmentation operation significantly easier and more reliable. At the first step, the segmenter applies the watershed transform on the slice to be manually segmented. The watershed transform provides a few hundred homogeneous regions. Then, the expert can tag each region with the one of the predefined labels, namely, background, lungs, torso, bones or heart by clicking with a mouse. As an alternative, instead of manually labeling all regions one by one, the segmenter can change the label of any prelabelled region to a new one after loading the presegmented images. Thus, he/she can correct his/her faults, and leave the segmentation unfinished to continue sometime later. However, the expert sometimes want to tag the pixels belonging to the same region with different tags (see Figure 2.6(a)). For this reason, in order to make structure boundaries smooth, we give the expert a chance to define triangle regions so that he/she labels inside the triange with a predefined label via our segmenter tool (see Figure 2.6(b)).



(a) (b)

Figure 2.6: (a) The problem of manual segmentation only using watershed transform (b) The proposed solution for this problem.

During the manual segmentation operation, we worked with Dr. Mehmet Akif TEBER, who is a radiology specialist at Etlik Ihtisas Hospital. The manually segmented images for three different slices are depicted in Figure 2.7. Background, lungs, torso, bones, and heart are represented with such colors as black, blue, green, yellow, and red, respectively.

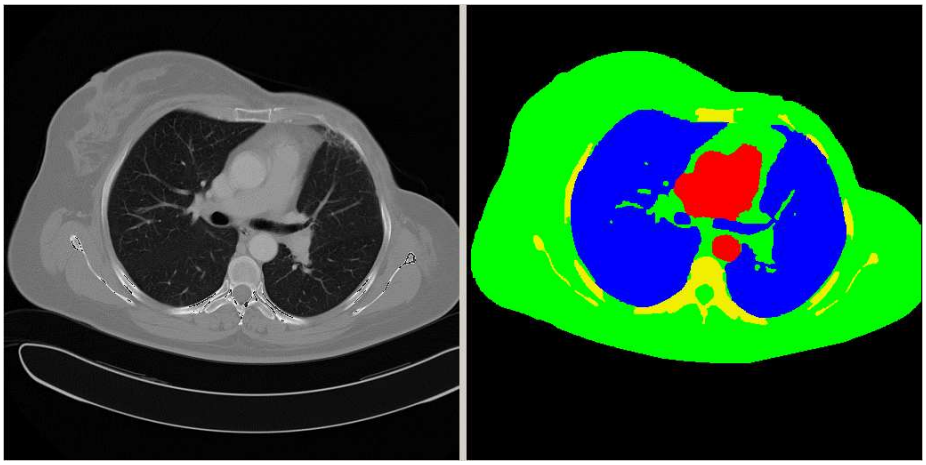
For the performance assesment of our segmentation frameworks, the manual segmentation was performed on every other slice in the CT image dataset by Dr. Mehmet

Akif Teber. That is to say, we made segmentation on 31 of 61 slices. Thus, we have 31 ground truth images. Every pixels in each image are classifed into five groups; background, lungs, torso, bones, and heart. For example, the manual segmentation result of the slice 31 is depicted in Figure 2.8.

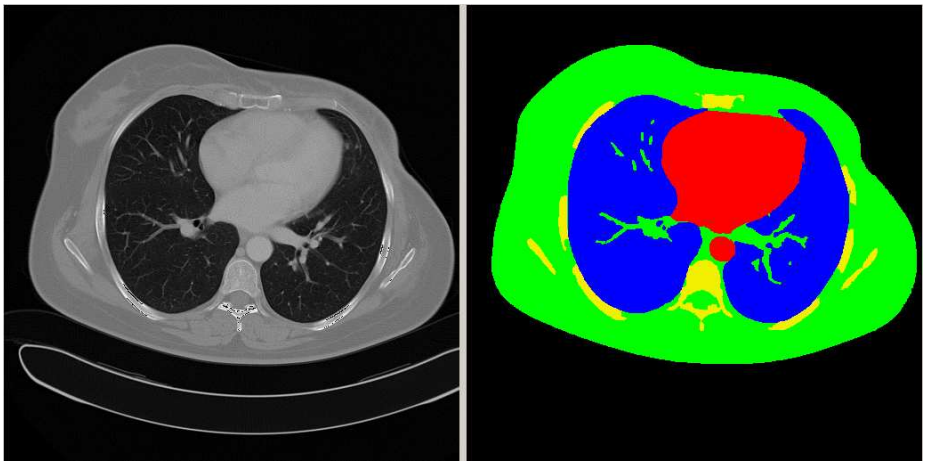
The left and right lungs including the trachea were segmented as lungs colored with blue. Since the upper part of the patient's right lung has been subject to an interstitial lung diease, it has higher intensities than the other parts. In manual lung segmentation, these pixels with higher intensities were also classified as lung. In the manual thoraxial segmentation, the thoraxial tissues excluding the lungs, the trachea, the heart, the beginning segments of the main vessels, and the bones were classified as torso and colored with green. The bones and the cartilages with higher densities were segmented as bones colored with yellow. In the manual cardiac segmentation, the beginning of the main vessels such as the superior vena cava, the inferior vena cava, the aorta, the pulmonary artery, and the pulmonary vein, were also segmented as a cardiac tissue. Therefore, the initial segments of these vessels were colored with red.

2.5.2 Quantitative Performance Measures

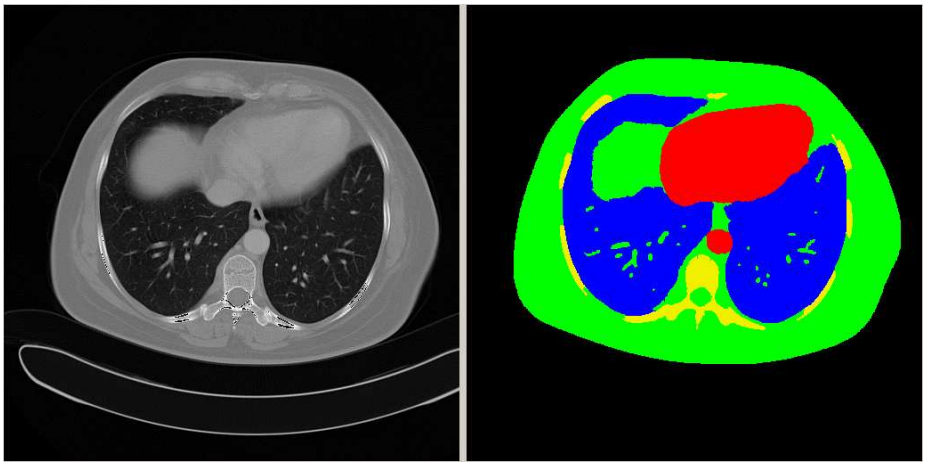
For quantitative performance measures, we use three metrics: Negative Rate Metric (NR), Overlap Index (OI) and Similarity Index (SI). All metrics are based on pixel-wise matches and mismatches between the ground truth image and the segmented image resulted from the proposed algorithm. For Negative Rate Metric, the lower the score the better the algorithm is at correctly segmenting foreground that matches the ground truth foreground segmentation. In the ideal case, NR Metric is equal to zero. NR Metric is the sum of two parts: a false negative score and a false positive score. A low false positive score means good delination of object region. A low false negative score means good delination of foreground internal to the object. As for Overlap and Similarity Index, the score close to 1 means that the algorithm provides good results. On the contrary, the score close to 0 means the segmentation is completely false. In the ideal case, both metrics are equal to one. (For details of the definition of these metrics, see Appendix A.)



(a)



(b)



(c)

Figure 2.7: The original images (right) and the manual segmentation results using the watershed transform (left) for (a) the slice 23, (b) the slice 31, and (c) the slice 39. Background, lungs, torso, bones, and heart are represented the color black, blue, green, yellow, and red, respectively.

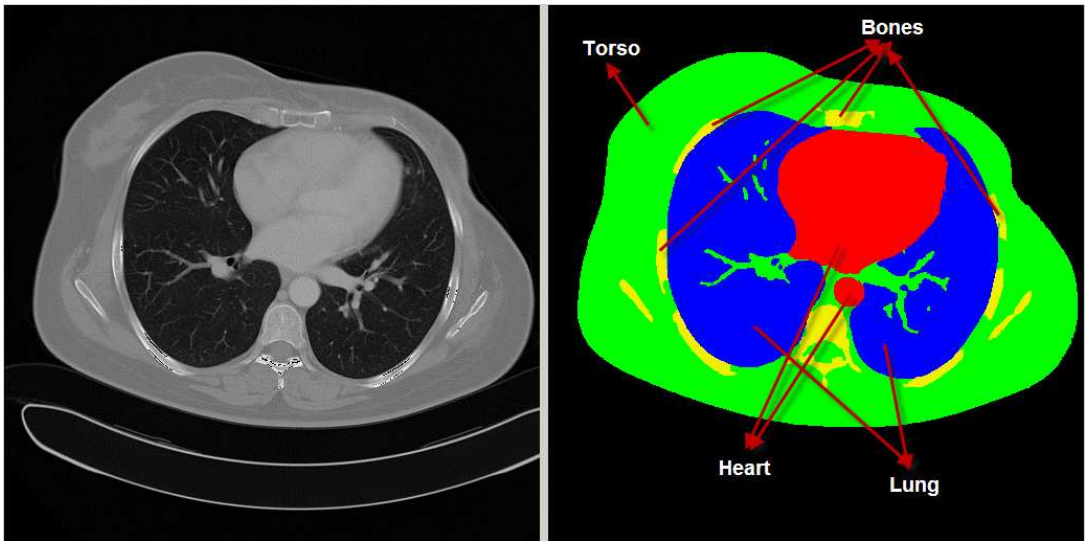


Figure 2.8: The original images (right) and the manual segmentation results (left) for the slice 31. Background, lungs, torso, bones, and heart are represented the color black, blue, green, yellow, and red, respectively.

CHAPTER 3

MEDICAL IMAGE SEGMENTATION USING THRESHOLDING

Brightness levels come from different surfaces change depending on their colors, their light absorption and reflection behavior. A threshold, that is brightness constant, can be determined to segment these surfaces to each other. Gray level thresholding is computationally cheap and fast and it is widely used in simple applications although it is oldest segmentation method.

In this section, first we begin to give mathematical information about OTSU's bi-level thresholding method, multi-level thresholding using extended version of OTSU's method and component labeling. Then, we will investigate the experiments on a slice with multi level thresholding. We will conclude the experiments of sequential segmentation on medical image slices.

3.1 Mathematical Theory

3.1.1 OTSU's Bi-Level Thresholding Method For Image Segmentation

This method has an aim to choose a threshold value that maximizes the between-class variances [2]. The maximization of between class variances also means to minimize within-class variance.

Let us assume that a gray level image has N pixels and the gray level value ranges from 0 to $L-1$. The number of pixels with gray level i ($0 \leq i \leq L-1$) is given with

f_i and the probability of having gray level i in image is given as:

$$P_i = \frac{f_i}{N}. \tag{3.1}$$

After bi-level thresholding, the pixels are divided into two classes. These classes are:

C_1: set of pixels with gray level value i, $0 \leq i \leq T$, where T is the calculated threshold value.
C_2: set of pixels with gray level value i, $T < i \leq L - 1$.

The corresponding probability distribution functions for classes C_1 and C_2 are

$$C_1 : \frac{P_0}{\omega_1}, ..., \frac{P_T}{\omega_1}, \tag{3.2a}$$
$$C_2 : \frac{P_{T+1}}{\omega_2}, ..., \frac{P_{L-1}}{\omega_2}. \tag{3.2b}$$

where ω_1 and ω_2 are zeroth-order cumulative moments of C_1 and C_2 respectively and given as

$$\omega_1 = \sum_{i=0}^{T} P_i = \frac{1}{N} \sum_{i=0}^{T} f_i, \tag{3.3a}$$
$$\omega_2 = \sum_{i=T+1}^{L-1} P_i = \frac{1}{N} \sum_{i=T+1}^{L-1} f_i. \tag{3.3b}$$

The mean values of classes C_1 and C_2 are given as,

$$\mu_1 = \sum_{i=0}^{T} \frac{iP_i}{\omega_1} = \frac{\sum_{i=0}^{T} i f_i}{\sum_{i=0}^{T} f_i}, \tag{3.4a}$$
$$\mu_2 = \sum_{i=T+1}^{L-1} \frac{iP_i}{\omega_2} = \frac{\sum_{i=T+1}^{L-1} i f_i}{\sum_{i=T+1}^{L-1} f_i}, \tag{3.4b}$$

and the mean value of the whole image intensity is given as,

$$\mu = \omega_1\mu_1 + \omega_2\mu_2 = \sum_{i=0}^{L-1} iP_i = \frac{1}{N}\sum_{i=0}^{L-1} i f_i, \tag{3.5}$$

the between-class variance of the thresholded image is defined as,

$$\sigma_B^2 = \omega_1 (\mu_1 - \mu)^2 + \omega_2 (\mu_2 - \mu)^2 , \tag{3.6a}$$

$$\sigma_B^2 = \omega_1\mu_1^2 - 2\omega_1\mu_1\mu + \omega_1\mu^2 + \omega_2\mu_2^2 - 2\omega_2\mu_2\mu + \omega_2\mu^2, \tag{3.6b}$$

$$\sigma_B^2 = \omega_1\mu_1^2 + \omega_2\mu_2^2 + \mu^2 \underbrace{(\omega_1 + \omega_2)}_{1} - 2\mu \underbrace{(\omega_1\mu_1 + \omega_2\mu_2)}_{\mu}, \tag{3.6c}$$

$$\sigma_B^2 = \omega_1\mu_1^2 + \omega_2\mu_2^2 - \mu^2. \tag{3.6d}$$

Otsu stated that the optimal threshold value T is chosen such that the between-class variance σ_B^2 is maximized. In other words, for every gray value, σ_B^2 is calculated, then T that maximizes the between-class variance is chosen.

$$\tilde{T} = argmax\left\{\sigma_B^2\right\}, \tag{3.7a}$$

$$\tilde{T} = argmax\left\{\omega_1\mu_1^2 + \omega_2\mu_2^2 - \mu^2\right\},$$

$$\tilde{T} = argmax\left\{\omega_1\mu_1^2 + \omega_2\mu_2^2\right\},$$

$$\tilde{T} = argmax\left\{\frac{1}{N}\left[\frac{\left(\sum_{i=0}^{T} i f_i\right)^2}{\sum_{i=0}^{T} f_i} + \frac{\left(\sum_{i=T+1}^{L-1} i f_i\right)^2}{\sum_{i=T+1}^{L-1} f_i}\right]\right\},$$

$$\tilde{T} = argmax\left\{\frac{\left(\sum_{i=0}^{T} i f_i\right)^2}{\sum_{i=0}^{T} f_i} + \frac{\left(\sum_{i=T+1}^{L-1} i f_i\right)^2}{\sum_{i=T+1}^{L-1} f_i}\right\},$$

where μ^2 and N are constant for each image. Thus, they can be ignored to determine the optimum threshold value T.

3.1.2 Iterative (Optimal) Threshold Selection For Bi-Level Thresholding Method

This threshold selection method works well even if the image histogram is not bi-modal. This method assumes that regions of two main grey levels are present in the

image. The algorithm (see Algorithm 1) is iterative, four to ten iterations usually being sufficient.

Algorithm 1 Iterative (optimal) threshold selection

1. Assuming no knowledge about the exact location of objects, consider as a first approximation that the four corners of the image contain background pixels only and the remainder contains object pixels.

2. At step t, compute μ_B^t and μ_O^t as the mean background and object grey level respectively, where segmentation into background and objects at step t is defined by the threshold value T^t determined in the previous step (equation 3.9);

$$\mu_B^t = \sum_{i \in C_B^t} \frac{iP_i}{\omega_B^t} = \frac{\sum_{i \in C_B^t} i f_i}{\sum_{i \in C_B^t} f_i}, \tag{3.8a}$$

$$\mu_O^t = \sum_{i \in C_O^t} \frac{iP_i}{\omega_O^t} = \frac{\sum_{i \in C_O^t} i f_i}{\sum_{i \in C_O^t} f_i}. \tag{3.8b}$$

3. Set

$$T^{t+1} = \frac{\mu_B^t + \mu_O^t}{2} \tag{3.9}$$

   T^{t+1} now provides an updated background / object distinction.

4. If $T^{t+1} = T^t$, halt; otherwise return to step 2.

The method performs well under a large variety of image contrast conditions. Its result is very close to the outcome of Otsu's bi-level thresholding method.

3.1.3 Multi-Level Thresholding Using Extended Version of OTSU's Method

In multi-level thresholding, bi-level thresholding formulation is extended by assuming that we have K classes instead of two, so we should have $K - 1$ threshold values $T_1, T_2, ..., T_{K-1}$. For example, the selection of two threshold for three-level thresholding is depicted in Figure 3.1. The pixels of the image are divided into K different classes:

C_1: set of pixels with gray level value i, $0 \leq i \leq T_1$, where T_1 is the calculated first

threshold value.

C_2: set of pixels with gray level value i, $T_1 < i \leq T_2$, where T_2 is the calculated second threshold value.

.

.

.

C_K: set of pixels with gray level value i, $T_{K-1} < i \leq L-1$, where T_{K-1} is the calculated last threshold value and L is the number of intensity level of image.

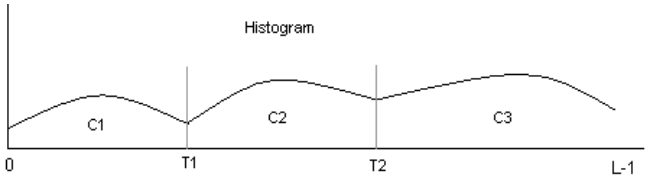


Figure 3.1: The selection of threshold values using Three-Level Extended Version of OTSU's Method

The corresponding probability distribution functions for classes are computed similarly. Then the zeroth order cumulative moments are:

$$\omega_k = \sum_{i \in C_i} P_i = \frac{1}{N} \sum_{i \in C_i} f_i \quad \text{for } k = 1, 2, ..., K \tag{3.10}$$

and the first order cumulative moments of classes are defined as:

$$\omega_k = \frac{1}{\omega_k} \sum_{i \in C_i} P_i = \frac{1}{N\omega_k} \sum_{i \in C_i} f_i \quad \text{for } k = 1, 2, ..., K \tag{3.11}$$

and the mean value of the whole image intensity is given as,

$$\mu = \sum_{k=1}^{K} \omega_k \mu_k. \tag{3.12}$$

Finally, we can define the between-class variance as:

$$\sigma_B^2 = \sum_{k=1}^{K} \omega_k (\mu_k - \mu)^2 = \left(\sum_{k=1}^{K} \omega_k \mu_k^2 \right) - \mu^2. \tag{3.13}$$

The optimum threshold values are chosen to maximize σ_B^2:

$$\left[\tilde{T}_1, \tilde{T}_2, ..., \tilde{T}_{K-1}\right] = argmax\left\{\sigma_B^2\right\} \tag{3.14a}$$

$$= argmax\left\{\left(\sum_{k=1}^{K} \omega_k \mu_k^2\right) - \mu^2\right\} \tag{3.14b}$$

$$= argmax\left\{\sum_{k=1}^{K} \omega_k \mu_k^2\right\} \tag{3.14c}$$

$$= argmax\left\{\frac{1}{N}\sum_{k=1}^{K}\left[\frac{\left(\sum_{i\in C_i} i f_i\right)^2}{\sum_{i\in C_i} f_i}\right]\right\} \tag{3.14d}$$

$$= argmax\left\{\sum_{k=1}^{K}\left[\frac{\left(\sum_{i\in C_i} i f_i\right)^2}{\sum_{i\in C_i} f_i}\right]\right\}, \tag{3.14e}$$

where μ^2 and N are constant for each image. Thus, they can be ignored to determine the optimum threshold values.

3.1.3.1 Experiments of Multi-Level Thresholding on A Single Slice

In our experiment, we use a medical image (see Figure 3.2) with 12 bits information in gray scale. It is obtained with CT medical imaging modality. This image is of upper torso in DICOM format. The gray level of original images is in the range of (0, 4096), and the histogram of the image is given in Figure 3.3.

First of all, we perform Otsu's bi-level thresholding method using Equation 3.7. As a result of this, we find a optimal threshold value as $T = 574$. In Figure 3.4, the white pixels are classifies as a *class* 1 and the remainders are as a *class* 0. Then, we perform iterative selection method. As we can see from Figure 3.5, the iterative selection method yields an identical result as bi-level thresholding method.

Figure 3.2: Original Image

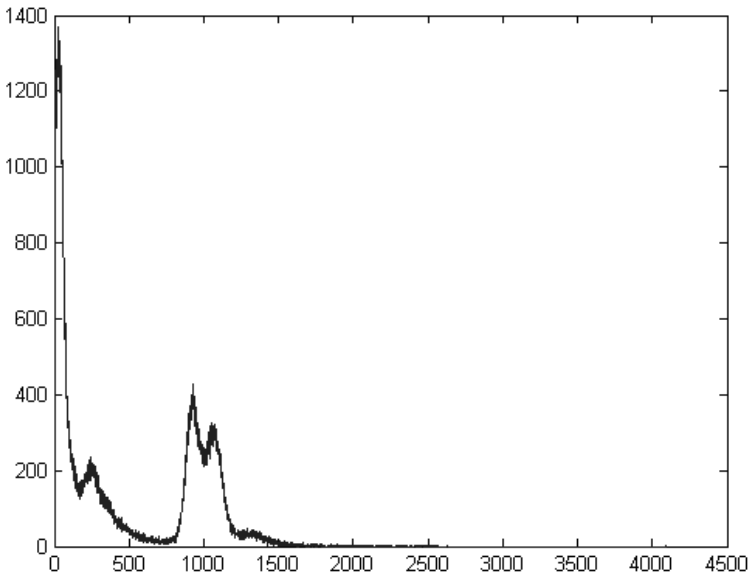


Figure 3.3: The histogram of the original image

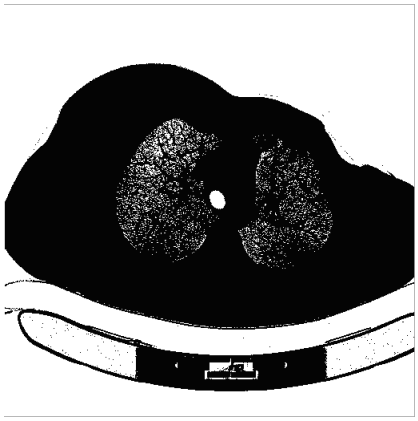
Figure 3.4: The outcome of the bi-level thresholding ($T = 574$). The pixel whose values are greater than the threshold are represented as white. The remains are black.

Figure 3.5: The result of the iterative selection method ($T = 574$), which is the same as that of the bi-level thresholding.

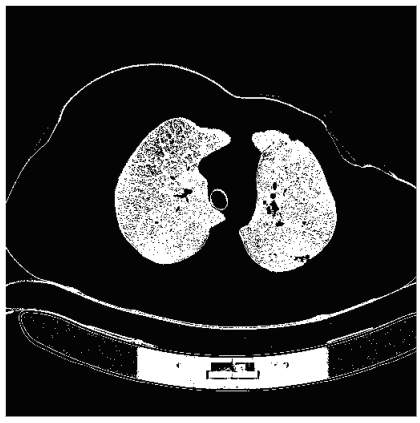
Figure 3.6: The outcome of the multi-level thresholding (T_1 = 196, T_2 = 682). The pixel whose values are greater than the threshold T_2 are represented as white. Other pixels whose values are between the thresholds are gray and the remains are black.

As far as multi-level thresholding is concerned, we make experiments for three and four level thresholding. In Figure 3.6, the result of three-level thresholding method is depicted. In Figure 3.7, the original image is classified into three classes. This classification is performed by the selection of the two threshold values. These classes are represented with the colors white, gray, and black. In the other figures, the image pixels are classified into two classes: the former is the set of pixels which belongs to the corresponding class representing by the color white and the latter is the set of pixels which does not belong to the class representing by the color black.

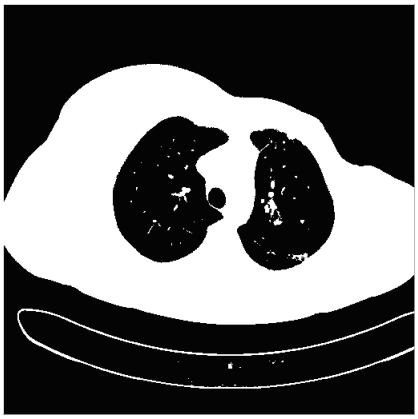
In Figure 3.8, the result of four-level thresholding method is depicted. In Figure 3.9, the original image is classified into four classes. This classification is performed by the selection of the three threshold values. These classes are represented with the colors white, two level of gray, and black. In the other figures, the image pixels are classified into two classes: the former is the set of pixels which belongs to the corresponding class representing by the color white and the latter is the set of pixels which does not belong to the class representing by the color black.



(a) The outcome of the multi-level thresholding (T_1 = 196, T_2 = 682). The pixel whose values are lower than the threshold T_1 T_1 are represented as white. The others are black. (*Pixelvalues* < 196)



(b) The outcome of the multi-level thresholding (T_1 = 196, T_2 = 682). The pixel whose values are between the threshold T_1 and T_2 represented as white. The others are black. (196 ≤ *pixelvalues* < 682)



(c) The outcome of the multi-level thresholding (T_1 = 196, T_2 = 682). The pixel whose values are greater than the threshold T_2 represented as white. The others are black. (682 ≤ *pixelvalues*)

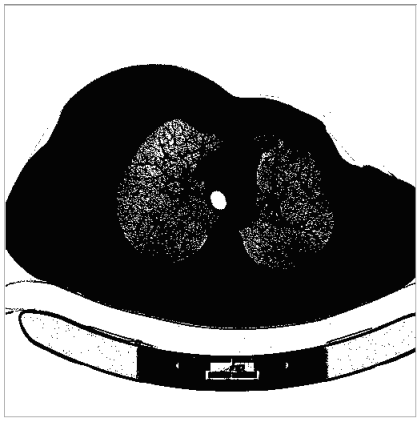
Figure 3.7: The outcome of the multi-level thresholding is shown as (a), (b) and (c).

Figure 3.8: The outcome of the multi-level thresholding ($T_1 = 196$, $T_2 = 682$, $T_3 = 1187$). The pixel whose values are greater than the threshold T3 are represented as white. Other pixels whose values are between the thresholds T_2 and T_3 are dark gray, others between the threshold T_1 and T_2 are light gray, and the remains are black.

Every possible combination of threshold values are analyzed to obtain the optimum threshold values. For this reason, the increase in the number of thresholds causes to take much more time to compute them. Since the bit depth in DICOM images is 12 Bit, and the gray level value range is between $(0, 4095)$ in our image, and therefore applying multilevel thresholding requires considerable amount of time. In the case of three threshold values, it takes about 5 minutes to obtain threshold values. However, for 4 threshold values it takes more than 4 hours for one image where 4 nested 'for-loops' are iterated from 1 to 4095 to obtain the maximum of the class variance values. Therefore, multi-level thresholding with more than 3 threshold values was not taken into account. In order to address the time problem, an iterative optimum threshold selection method can be applied. It is not logical to try all possible combinations so we can ignore some combinations using an iterative approaches for multi-level thresholding.

The resultant images may contain scattered and disconnected pixels because this thresholding technique does not take into account the spatial characteristics of the

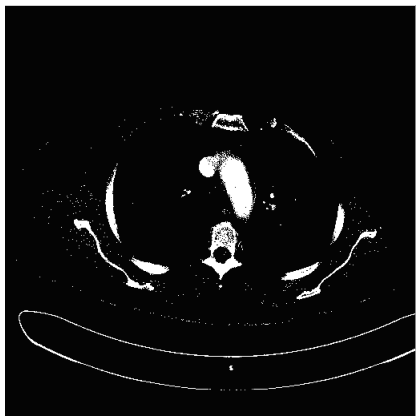
image, Furthermore, due to lack of spatial information, this method is very sensitive to noise and intensity inhomogenities.



(a) The representation of class 1 (pixel values < 194).

(b) The representation of class 2 (194 ≤ pixel values < 660).

(c) The representation of class 3. (660 ≤ pixel values < 1187).



(d) The representation of class 4. (1187 ≤ pixel values).

Figure 3.9: The outcome of the multi-level thresholding is shown as (a), (b), (c), and (d).

3.1.4 Component Labeling

A binary image is composed of pixels with two levels of intesity. Both levels represent whether the pixel belongs to object or background. Some of the pixels belonging to object are connected to each other. A set of connected pixels are called component. Each component on the image should be tagged with a unique label. Component Labeling algorithm searches all the connected pixels of the binary image and assigns a unique label to each component. In Figure 3.10, the binary image before and after labeling is shown. In many application, some characteristics of the components such as position, dimension, and orientation are calculated during the labeling process [6].

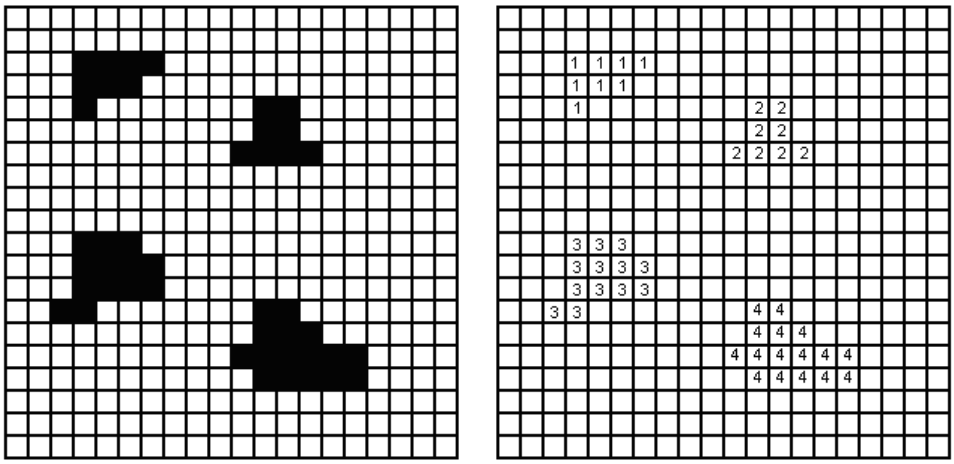


Figure 3.10: The binary image before and after labeling

Holding whole binary image in the memory requires more hardware specifications. In order to prevent this, sequential algorithm uses two rows of the image at a moment. For this reason, it can be performed even when the image are stored as a file and there is a space limitations in the memory; nevertheless, the process is completed after two passing over the image. In Algorithm 2 [6], the labeling process is clearly explained. In the algorithm, upper and left neighbours of the pixel of interest is examined and the pixel is supposed to be labeled as neighbours. If both neigbours have different labels, it is labeled with one of the labels and an equivalence table is prepared to maintain track of all labels that are equivalent. In the second pass, the equivalence table is used for the appointment a unique label to all pixels of a component.

Algorithm 2 Sequential Connected Components Algorithm using 4-connectivity

1. Scan the image from left to right and from top to bottom.
2. If the pixel belongs to the object, then examine the upper and left neighbours.
   - If one of them is labeled and the other is not, then tag with the same label.
   - If labels of both neighbours are same, then tag with the same label.
   - If labels of both neighbours are different, then tag with the upper's label and add labels to the equivalence table as equivalent labels. This means that both labels represent same component.
   - Otherwise assign a new label to this pixel and add this label to the equivalence table.
3. Perform again and again Step 2 until there are not any unlabeled pixel to consider.
4. Rescan the image to arrange the labels based on the equivalence table. Substitute the lowest label in equivalent set of each label for it.

Scanning an image from left to right and from top to bottom, the sequential algorithm deals with only two neighbours of a pixel: the one is above and the other is left. Note that these two pixels have already been inspected by the algorithm due to the direction of scanning. If none of these pixels belong to the object, then the pixel of interest is assigned a new label. If only one of these two pixels belongs to the object and has been tagged with label *L*, then the new pixel will be tagged with label *L*. If both pixels belong to the object and have been tagged with the same label *L*, then the new pixel will be tagged with label *L*. In the case where both neigbours have been tagged with different labels, then both labels have been used for the same component and they have to be merged, nonetheless. In this situation, the pixel is tagged with one of these two labels, usually the left neighbour's label that is the smallest one, and both labels are recorded as equivalent labels in the equivalence table. In Figures 3.11 and 3.12, the result of the first and second pass over an image are depicted, respectively.

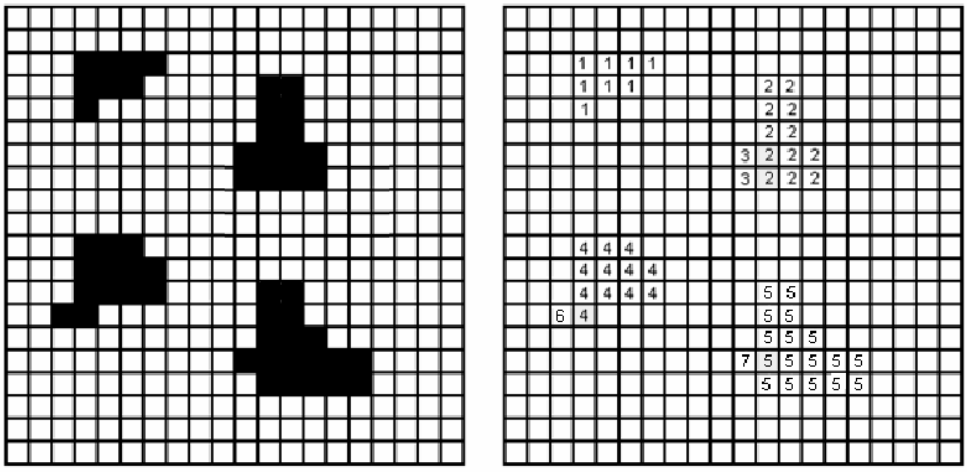


Figure 3.11: After first pass over an image

The equivalence table includes the information about all equivalent labels that are mapped to the same component. This information guarantees to assign unique label to each component. In the first pass, all those labels belonging to one component are proclaimed equivalent. In the second pass, one label from an equivalent set is used for

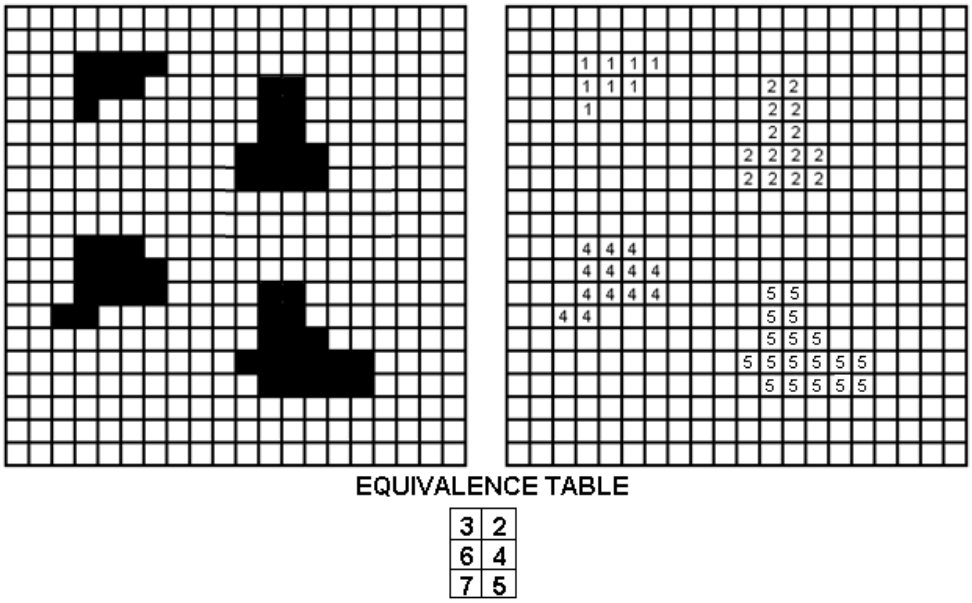


Figure 3.12: After second pass over an image

the appointment to all pixels of a component. The smallest label in the equivalent set is usually selected to represent a component. The assignment of a unique label to each component is realized in the second scan. After all process is over, the labels in use are renumbered so that gaps in the labels are eliminated. Some chracteristics such as first and second moments, area, and perimeter can be calculated for each component during these two scans.

3.2 A Framework For Sequential Segmentation on Medical Image Slices

In this section, we propose a semi-automatic sequential segmentation framework on CT Torso image set using the combination of multilevel thresholding and sequential labeling. In Figure 3.13, the proposed sequential segmentation framework is clearly explained. First of all, a user chooses any slice where the object of interest clearly appears from the slice set and then also selects any small window which encapsulates the object on the slice (see Figure 3.14). Excluding these user interferences, the whole procedure is performed automatically. After calculating a histogram of the window, the software determines the range of intesity which designate the object of

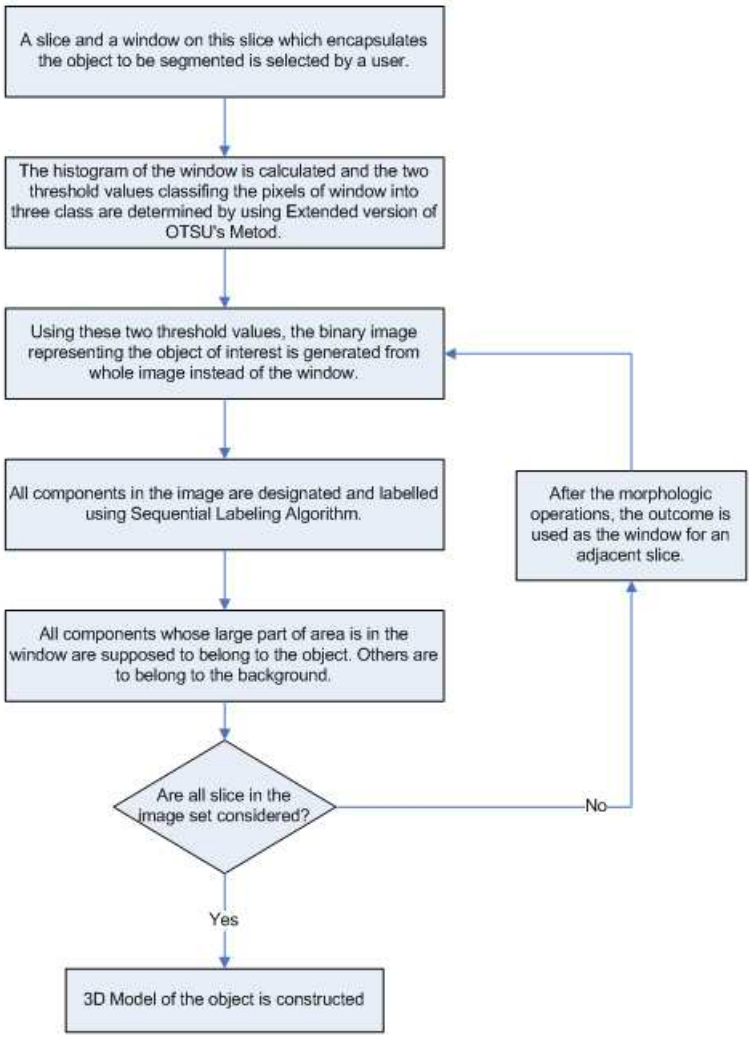


Figure 3.13: Flow Chart of The Proposed Framework

interest using Three-Level Extended Version of Otsu's Method (see Figure 3.15). For all slices in the image set, these calculated threshold values will be valid. In other words, it is not necessary to recalculate thresholds for each slice. If intensity value of a pixel stays in this range, then the pixel will belong to the object; otherwise, to the background. After considering all pixels of the image, the binary image representing object and background distinction is obtained (see Figure 3.16). Sequential Connected Components Algorithm is used for the labeling process of the binary image. Performing the algorithm, the components which have unique labels are achieved. The component whose large part of area correspond to the area of the window are perceived as belonging to the object. Therefore, other components are supposed to belong to the background (see Figure 3.17). So far, the segmentation procedure for a slice is completed. After the morphological operations, the outcome of the procedure for a slice is used as the search window for an adjacent slice since two succesive CT images resemble to each other. These procedure is performed for all slice in the medical image set so as to reconstruct the 3D model of the object (see Figure 3.18).

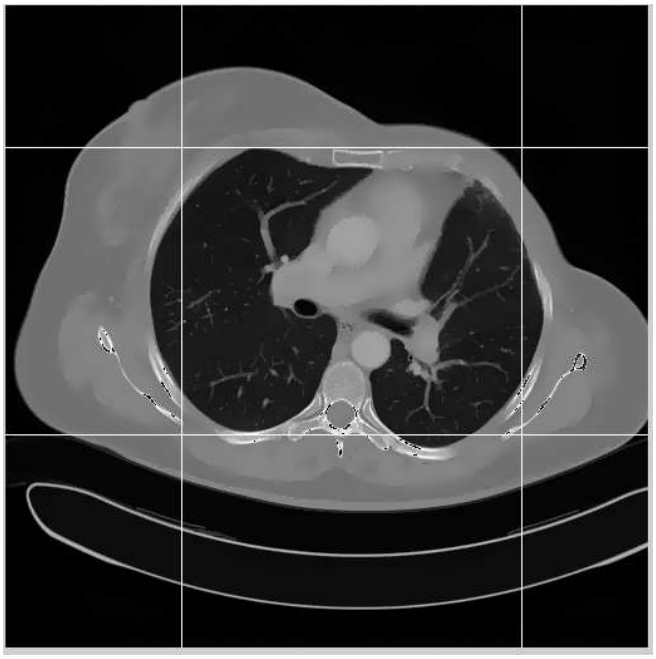


Figure 3.14: User Interference

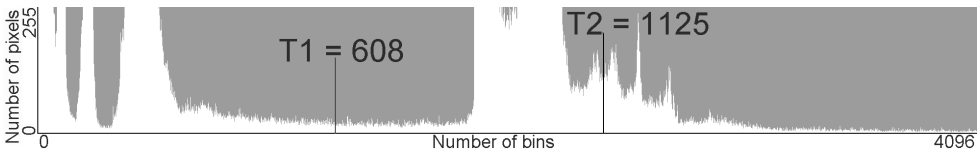


Figure 3.15: Determination of the range of intesity which designate the object of interest

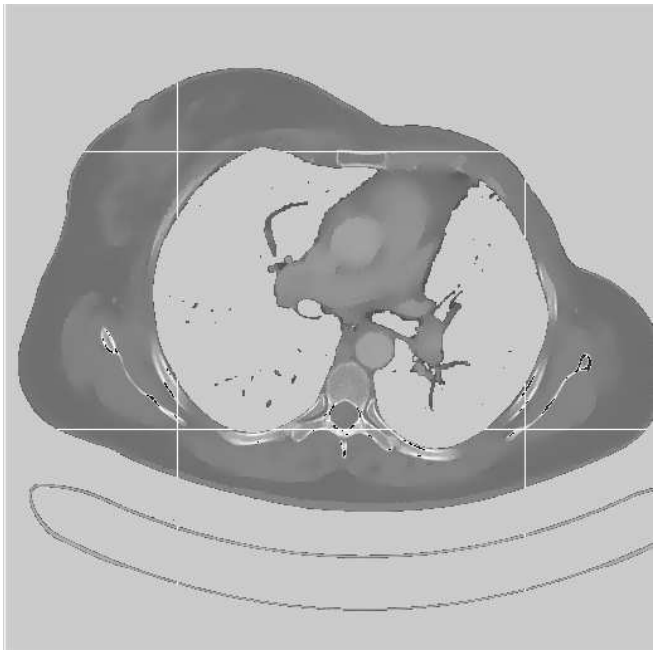


Figure 3.16: The binary image representing object and background distinction

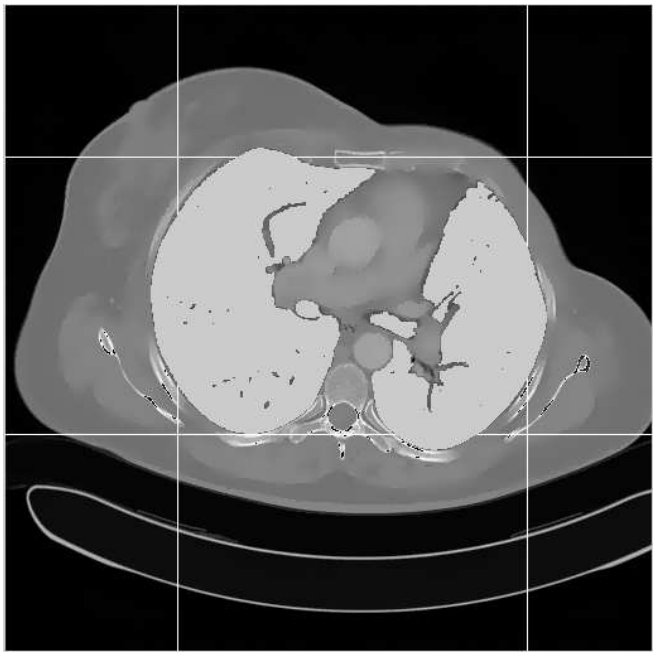


Figure 3.17: The outcome of the segmentation operation for a single slice

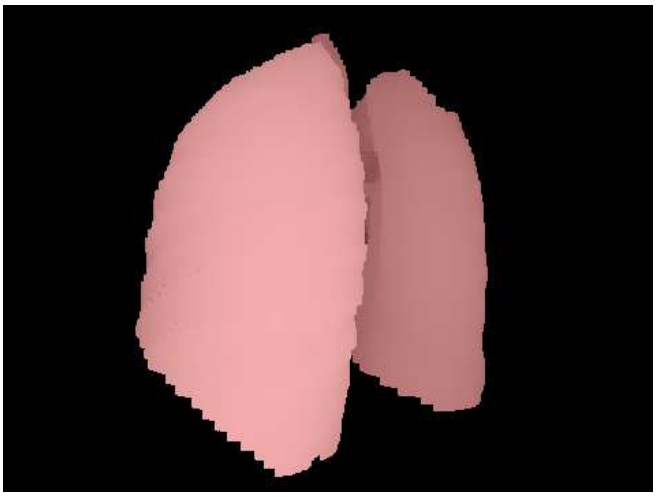
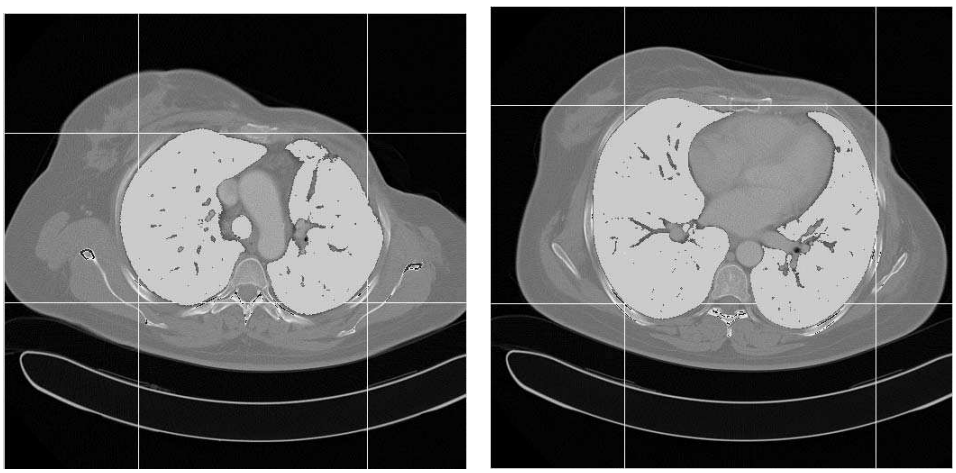


Figure 3.18: 3D representation of the object

3.3 Results

The proposed framework is performed on CT image set containing 61 images. In Figure 3.19, the segmented images of lungs for two different slices are given. In Figures 3.20 and 3.21, the segmented images of bones and heart for a slice are given, respectively.



(a) (b)

Figure 3.19: The segmentation of lungs for the slices (a) 18 and (b) 31

The average run-time durations are calculated based on implemented C++ source code run on an Intel(R) Core(TM)2Duo 3.00GHz CPU with 1.95GB RAM of PC and listed in Table 3.1. At the begining of the segmentation operation, the scheme calculates the threshold values; thus it requires additional time for an initial slice. For this reason, the segmentation for an initial slice spends much more time compared to the operation for another slice.

Table 3.1: The approximate durations for segmentation of lungs, heart and bones using thresholding

	Elapsed Time
Segmentation process for an initial slice	5 sec
Overall segmentation process	125 sec

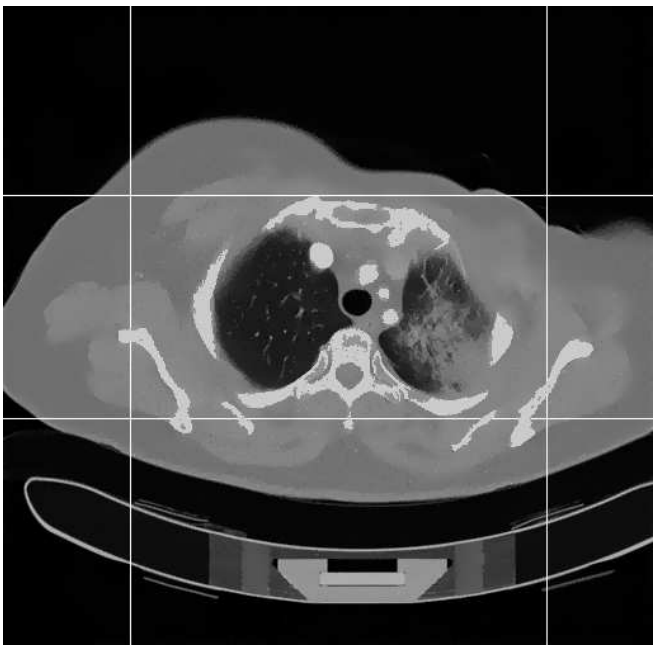


Figure 3.20: The segmentation of bones for the slice 11

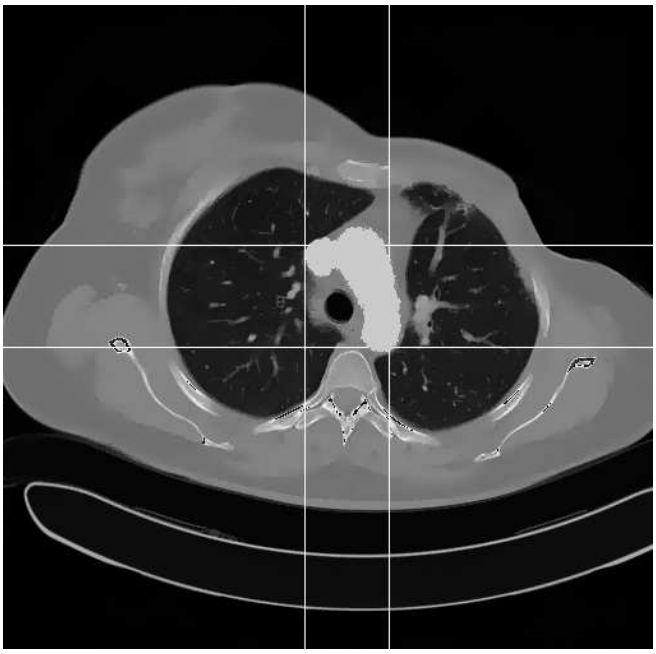


Figure 3.21: The cardiac segmentation for the slice 18

In Figures 3.22, 3.23, and 3.24, 3D representation of 2D segmentation of lungs, bones, and heart generated from all segmented images are depicted, respectively.

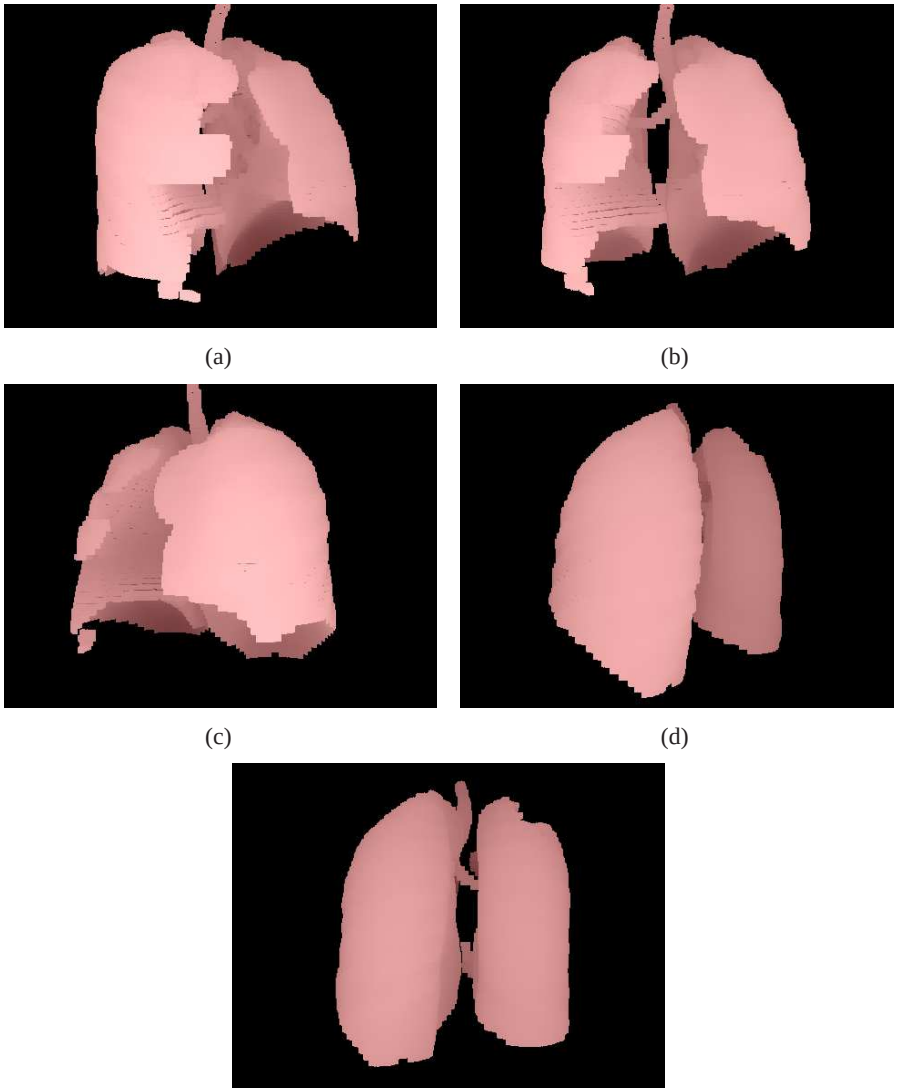
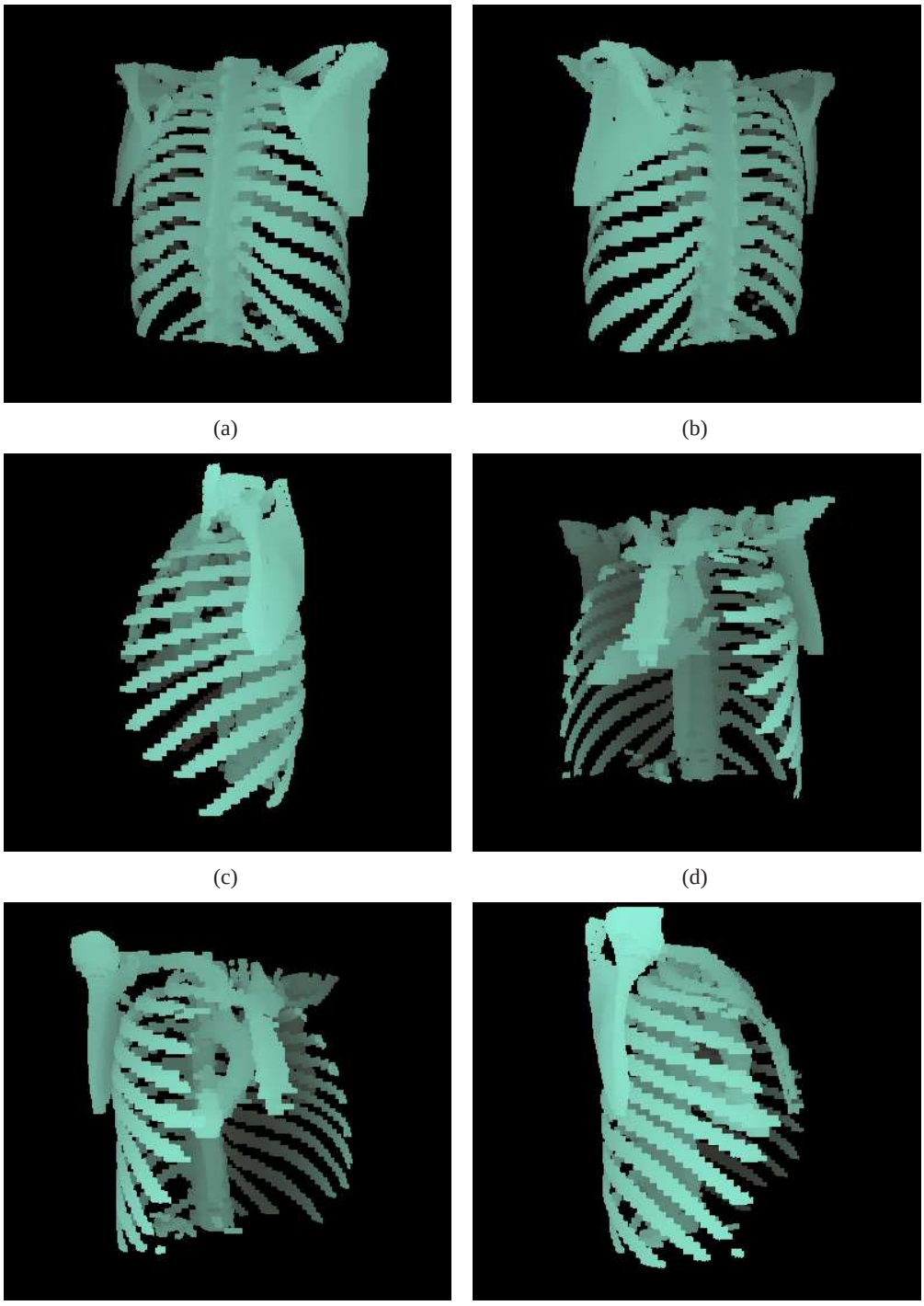


Figure 3.22: 3D representation of 2D segmentation of lungs from different views



(a) (b)

(c) (d)

(e) (f)

Figure 3.23: 3D representation of 2D segmentation of bones from different views

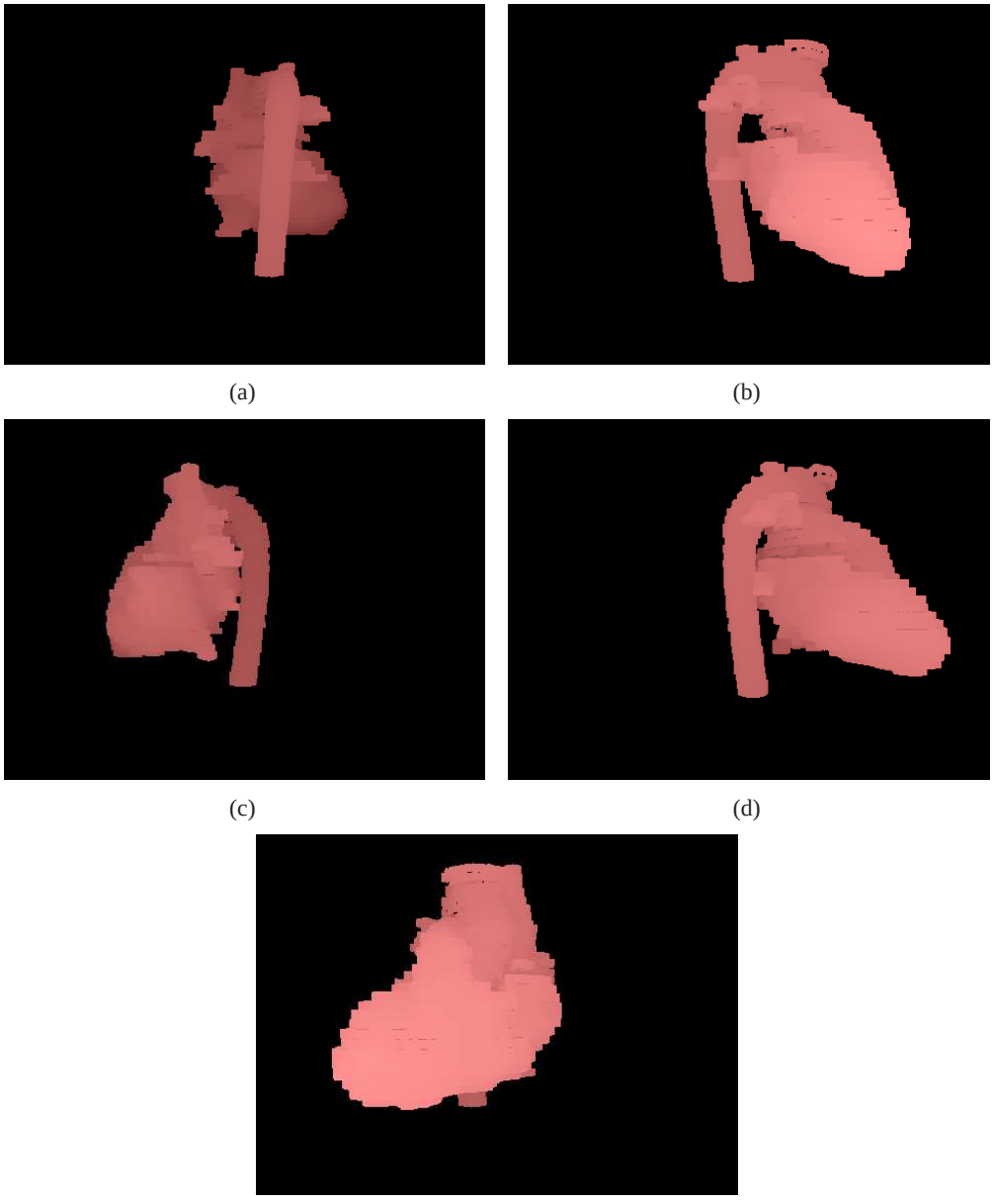


Figure 3.24: 3D representation of 2D segmentation of heart from different views

3.3.1 Quantitative Evaluation of Segmentation Results

As far as quantitative evaluations of the segmentation results are concerned, in this section, we compare the semi-automatically segmented images based on thresholding, with "Ground Truth" resulted from the manual segmentation based on watershed transform. Negative Rate Metric (NRM), Overlap Index (OI) and Similarity Index (SI) are used as explained in Chapter 2.5.2 and Appendix A.

In Figures 3.25, 3.26 and 3.27, negative rate metrics, missed, extra segmented, and overlapped rates and similarity indices are shown based on comparison between the automatically and manually segmented images for lungs.

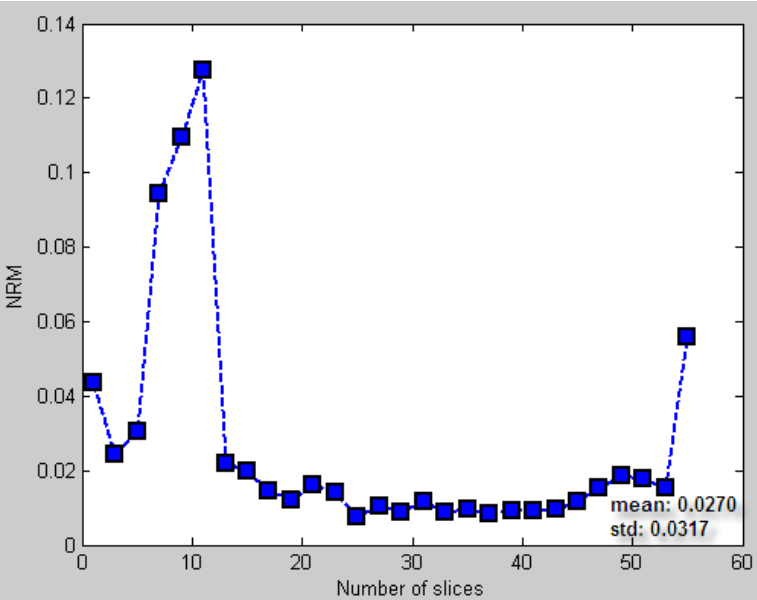


Figure 3.25: Negative Rate Metric results based on comparison between the automatically and manually segmented images for lungs.

As shown in Figures 3.25 and 3.27, all metrics have the same behaviour. The slices between 7 and 11 has the worst segmentation results in all lung segmentation results. Panel (a) of Figure 3.28 shows the image from slice 11 along with its segmentation results. Here, the colors red, green, and blue represent missed, extra segmented and overlapped regions, respectively. The numerical values of the evaluation metrics for these images are also given on the segmented images. In this slice, since the upper part of the patient's right lung has been subject to an interstitial lung disease, the

intesity values of the lung pixels are higher than their normal values. For this reason, they are segmented as torso using the proposed framework. In slice 55, shown in panel (b) of Figure 3.28, the relatively bad results are obtained for lung because of the high proportion of perimeter pixels to actual lung area. Namely, the increase in the proportion of perimeter to the lung area leads to the decrease in the segmentation quality since there is always faulty segmented pixels in the boundary regions of the structures. As for the remaining slices, we achieve good lung segmentation results; for example, in some slices such as slice 25, similarity index for lung reaches 0.98. Figure 3.29 shows lung segmentation results of two slices (25 and 37) with good values of evaluation metrics. As shown in the slice 37, the esophagus is segmented as lung due to its lower intensities, however.

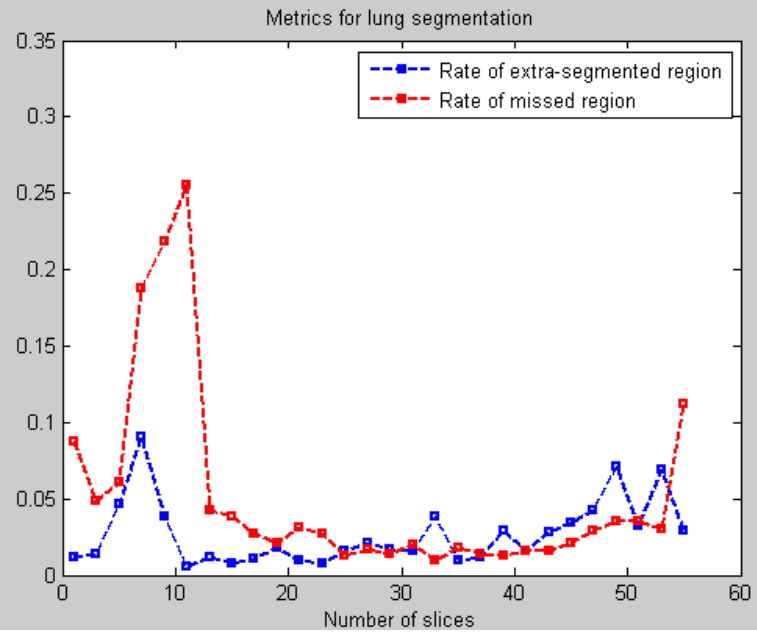


Figure 3.26: Missed and extra segmented rates based on comparison between the automatically and manually segmented images for lungs.

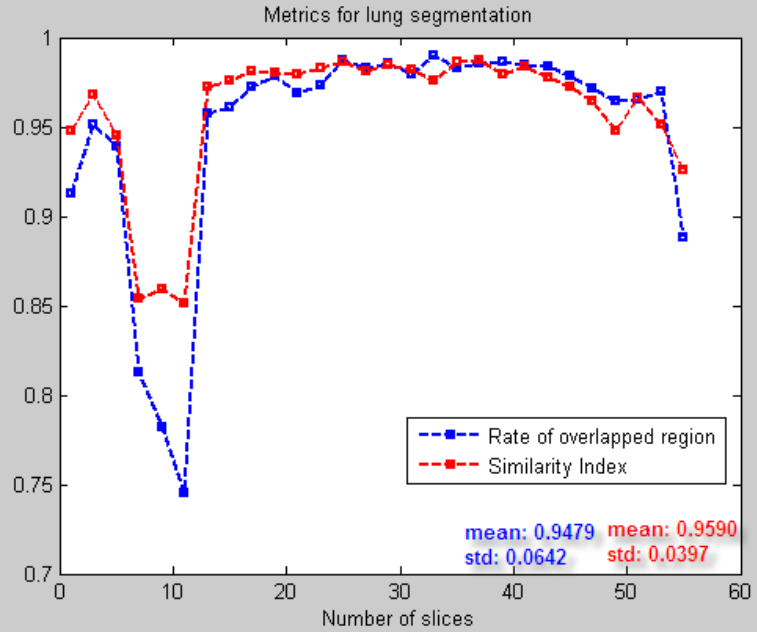
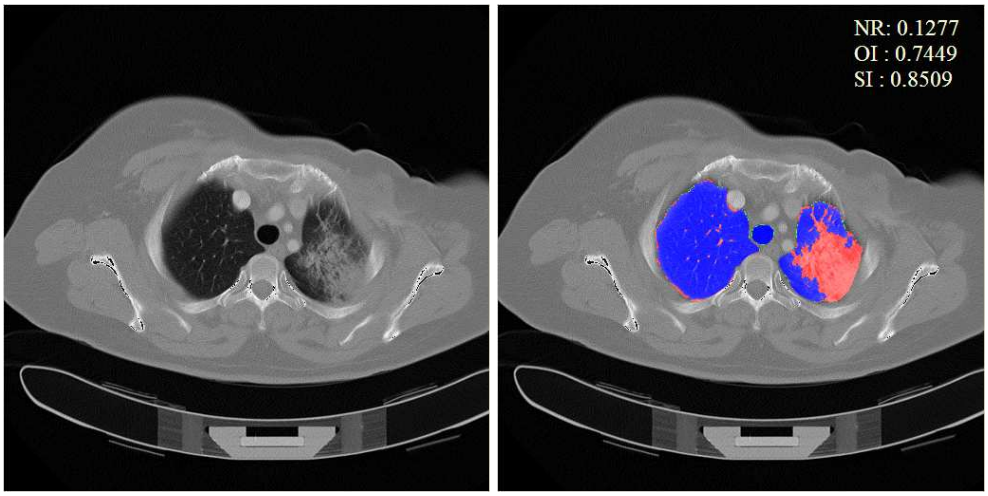
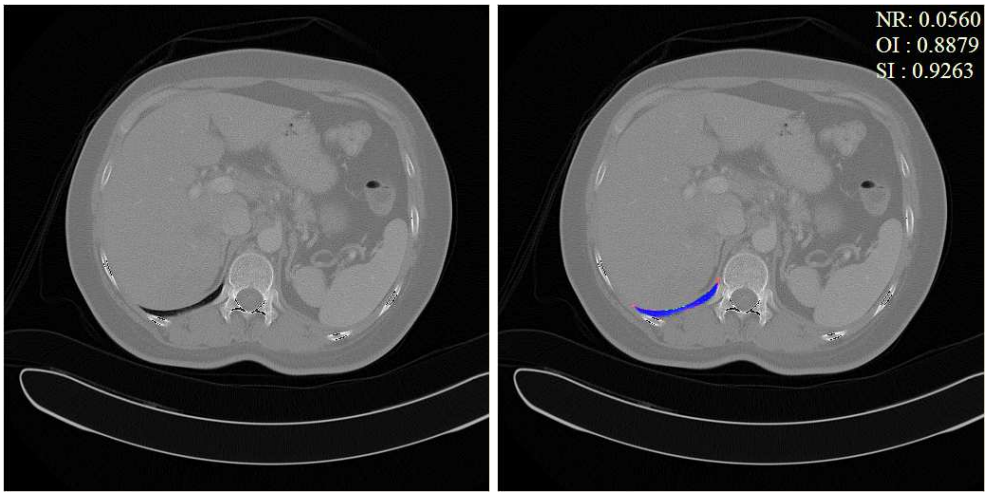


Figure 3.27: Overlapped rate and similarity index based on comparison between the automatically and manually segmented images for lungs.

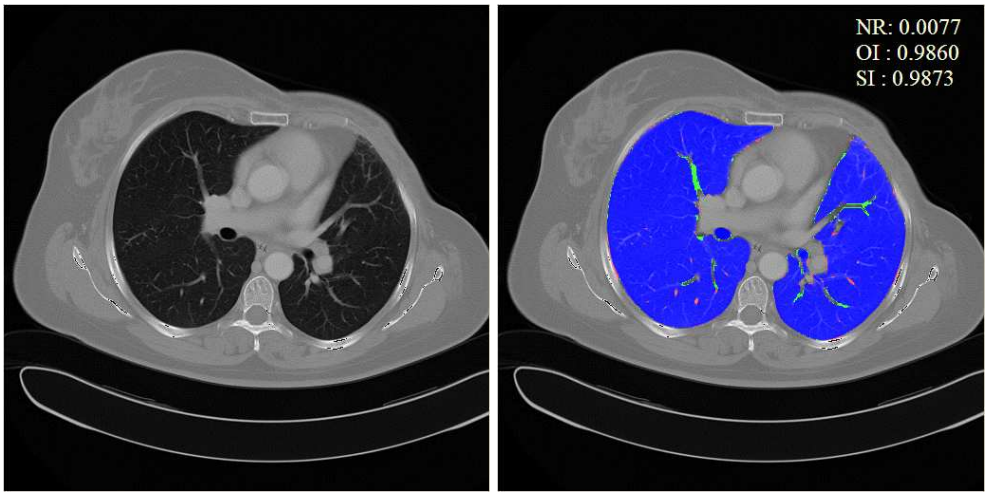


(a)

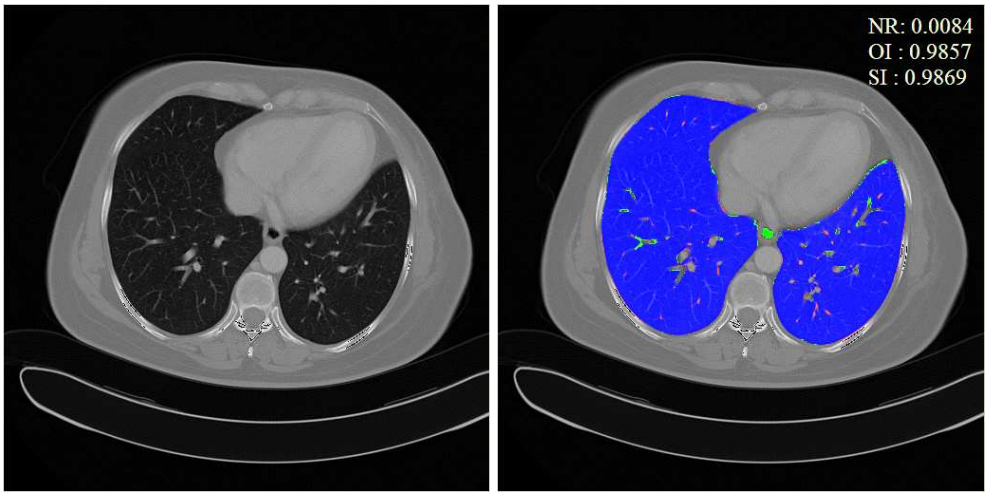


(b)

Figure 3.28: Two examples of bad lung segmentation results (right); the slices (a) 11 and (b) 55. The original images (left) are also shown.



(a)



(b)

Figure 3.29: Two examples of good lung segmentation results (right); the slices (a) 25 and (b) 37. The original images (left) are also shown.

Quantitative results for bone segmentation are given in Figures 3.30, 3.31 and 3.32. In these figures, negative rate metrics, missed, extra segmented, and overlapped rates and similarity indices are depicted, respectively, based on comparison between the automatically and manually segmented images. Two examples with poor evaluation metric values examples came from slices 01 and 57 (see Figure 3.33) and the ones with good results are from slices 23 and 45 (see Figure 3.34) according to both the similarity index and negative rate metric. In these figures also, the colors red, green, and blue represent missed, extra segmented and overlapped regions, respectively.

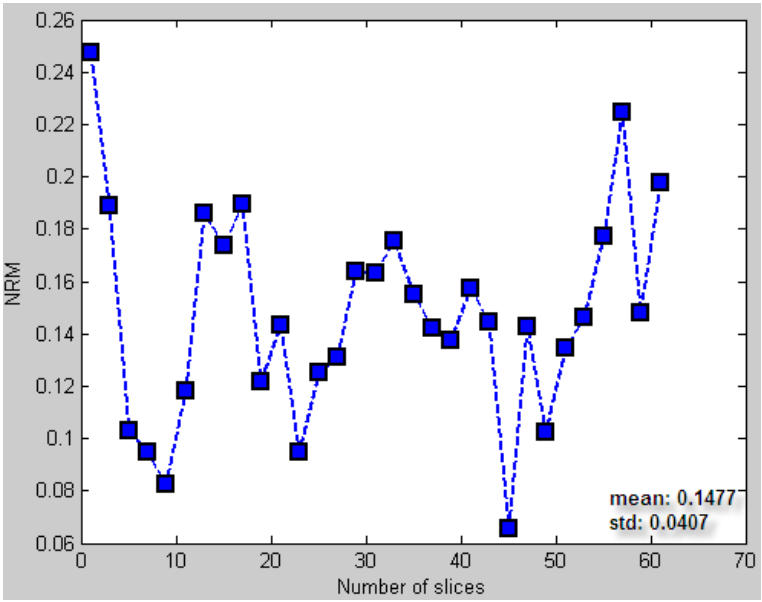


Figure 3.30: Negative Rate Metric results based on comparison between the automatically and manually segmented images for bones.

As shown in the Figures 3.33(a) and 3.33(b), the pixels corresponding to the spongy bone which is internal to the compact bone and the cartilage have lower intensities compared to the compact bone. Because of this, our algorithm misses the pixels belonging to the spongy bones and the cartilages. In addition, bones have large proportion of perimeter to area since they are scattered over the torso. For this reason, the the overall segmentation quality has a tendency to decrease (see the mean value of SI (Figure 3.32)). Despite these problems with the segmentation of bones, similarity index for bones rises over 0.90 in some slices, as seen in the Figure 3.34.

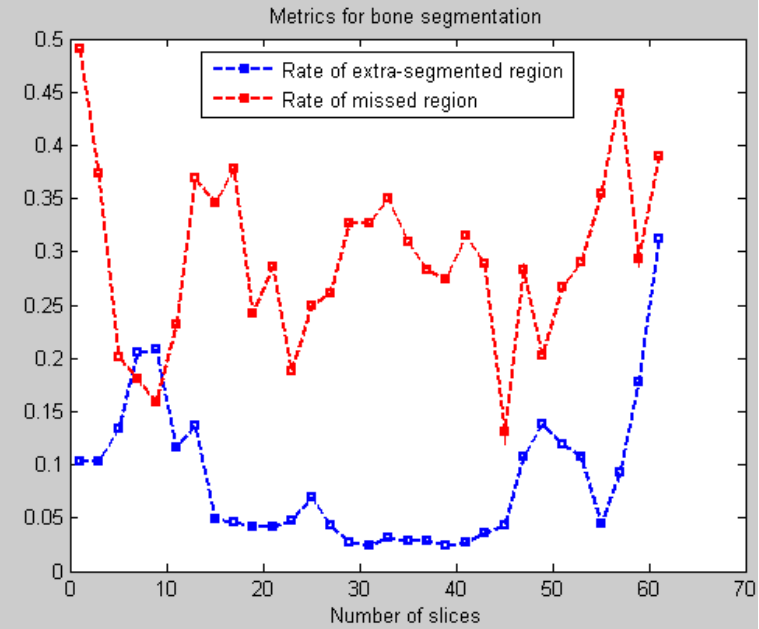


Figure 3.31: Missed and extra segmented rates based on comparison between the automatically and manually segmented images for bones.

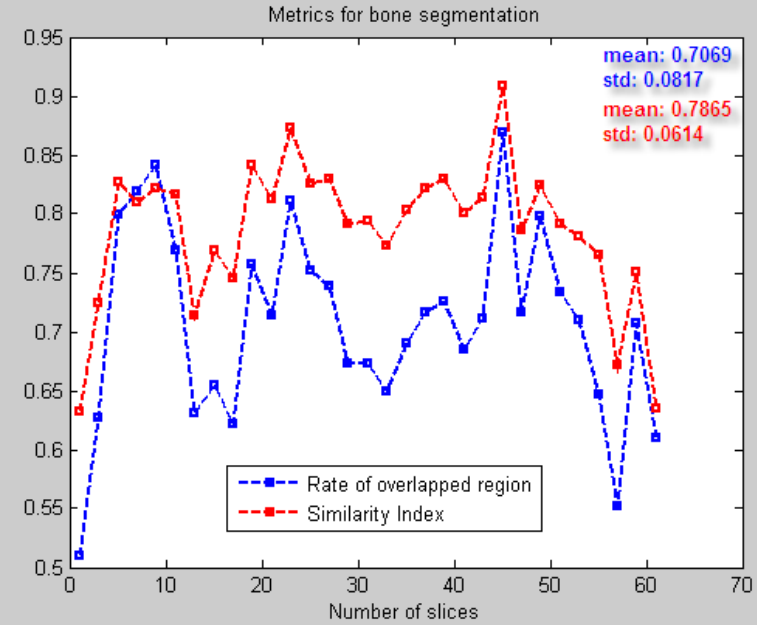
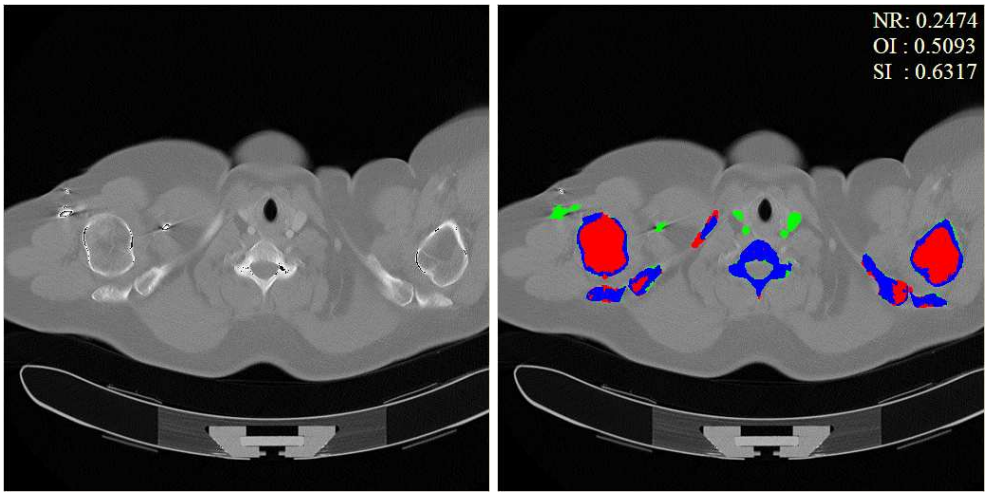
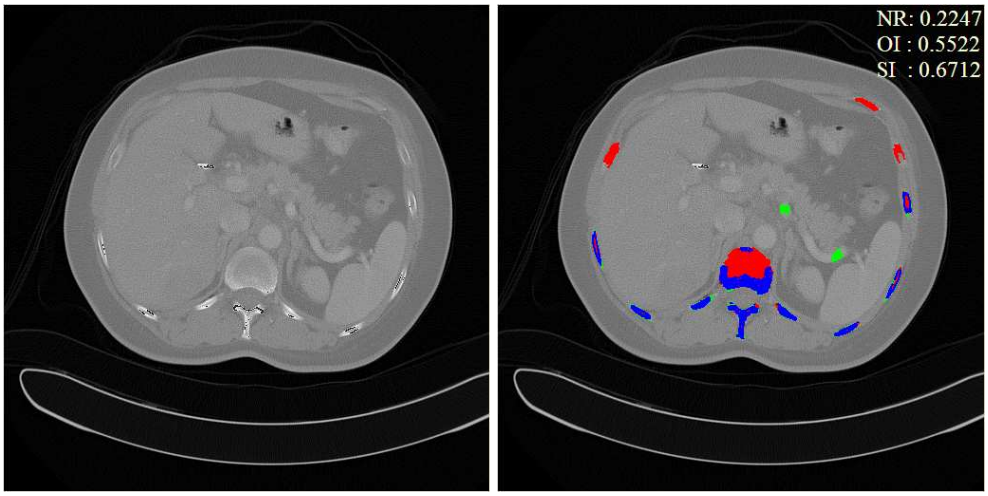


Figure 3.32: Overlapped rate and similarity index based on comparison between the automatically and manually segmented images for bones.

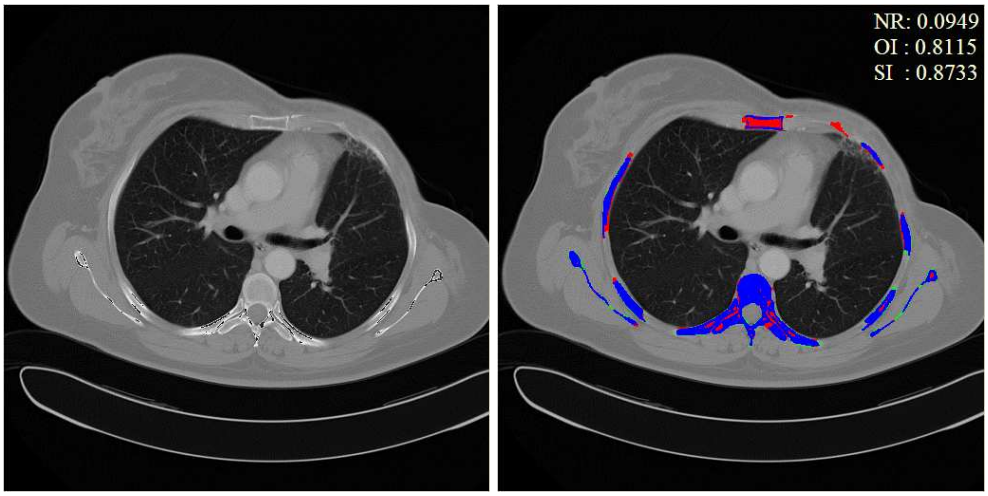


(a)

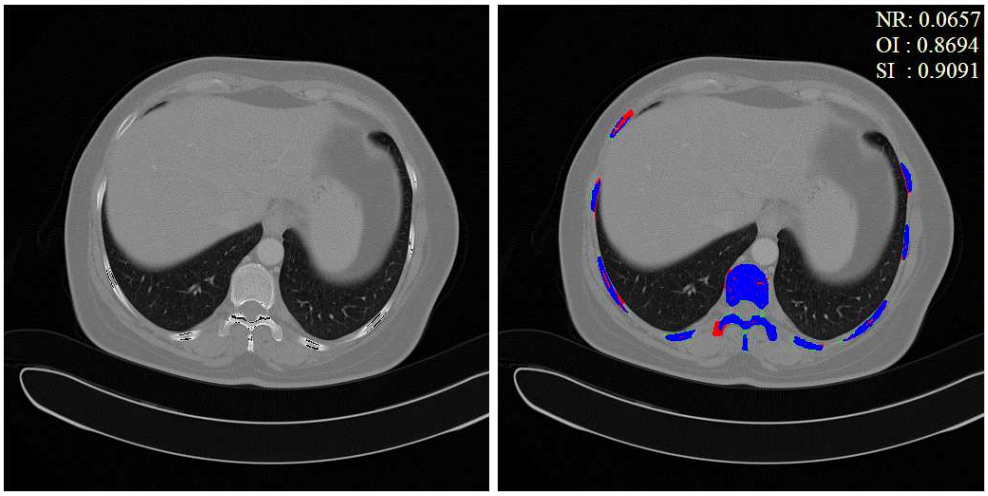


(b)

Figure 3.33: Two examples of bad bone segmentation results (right); the slices (a) 01 and (b) 57. The original images (left) are also shown.



(a)



(b)

Figure 3.34: Two examples of good bone segmentation results (right); the slices (a) 23 and (b) 45. The original images (left) are also shown.

In Figures 3.36 and 3.37, missed, extra segmented, and overlapped rates and similarity index are depicted based on comparison between the automatically and manually segmented images of the heart. With respect to both the similarity index and negative rate metric, bad results are observed in slices 25 and 41 (see 3.38) and the good ones are from the slices 17 and 45 (see 3.39). In these figures, the color coding is same as in lungs and bones.

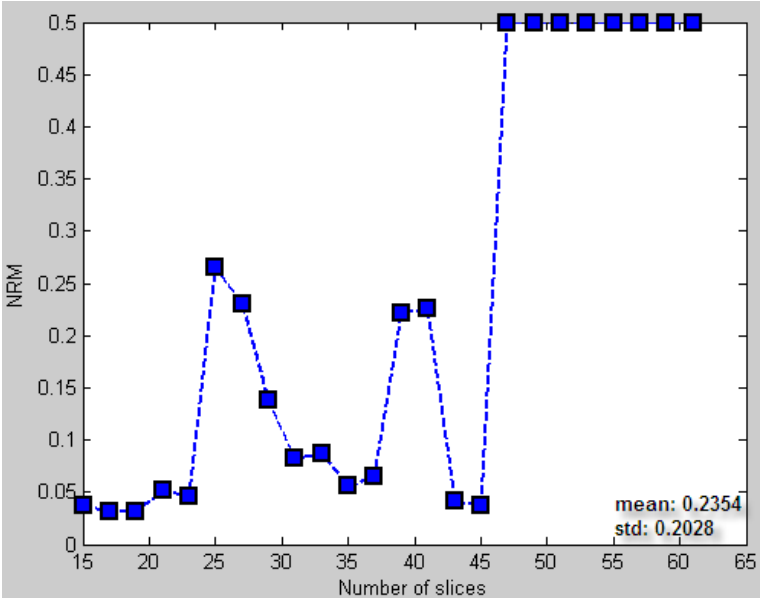


Figure 3.35: Negative Rate Metric results based on comparison between the automatically and manually segmented images for heart.

In this CT image dataset, intravenous contrast material was used to help highlight blood vessels. The cardiac muscles have lower intensities than the blood with contrast material. Therefore, our algorithm misses the pixels belonging to muscles (*i.e.*, the red regions) as seen in Figure 3.38(a). On the contrary, it catched the pulmonary vessels (*i.e.*, the green regions) due to their high intensities. In slice 41 shown in Figure 3.38(b), our algorithm did not produce good results since the heart does not have clear boundaries. The worst performances were obtained for slices from 47 to 61. For these slices, the algorithm was unable to segment the descending thoracic aorta, yielding high NRM values and low SI values. On the other hand, there are some slices such as 17 and 45, shown in Figure 3.39, for which the similarity index is larger than 0.95.

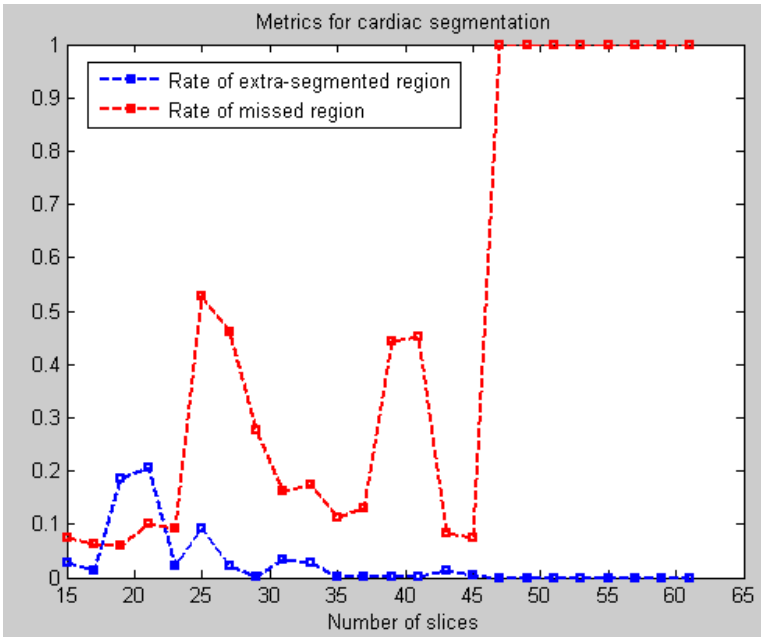


Figure 3.36: Missed and extra segmented rates based on comparison between the automatically and manually segmented images for heart.

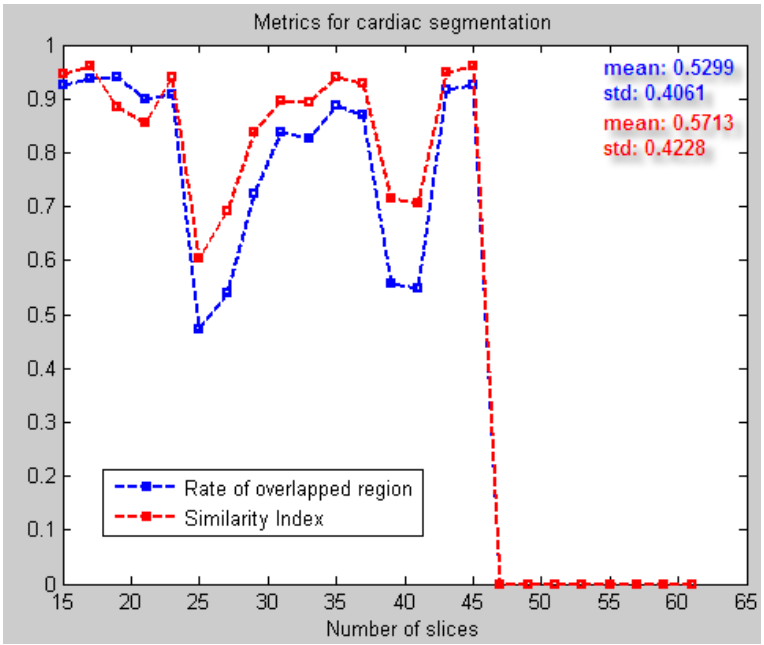
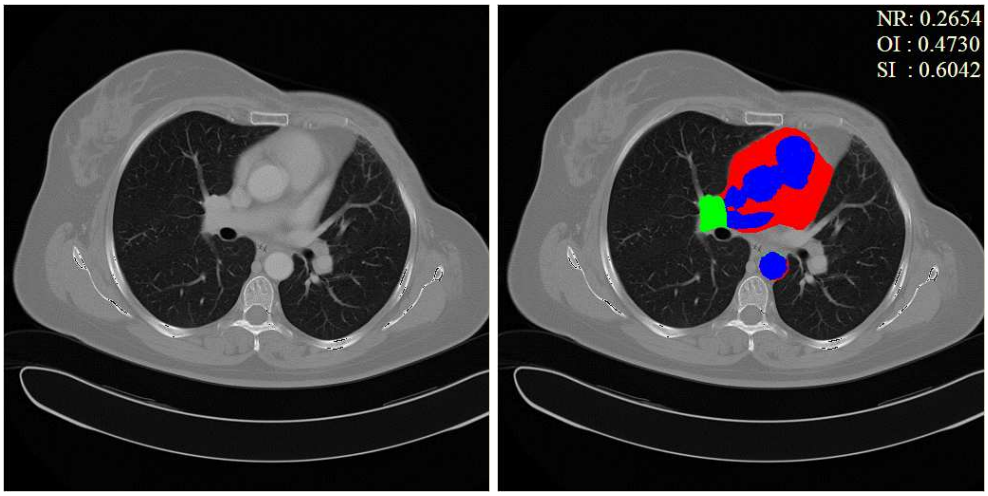
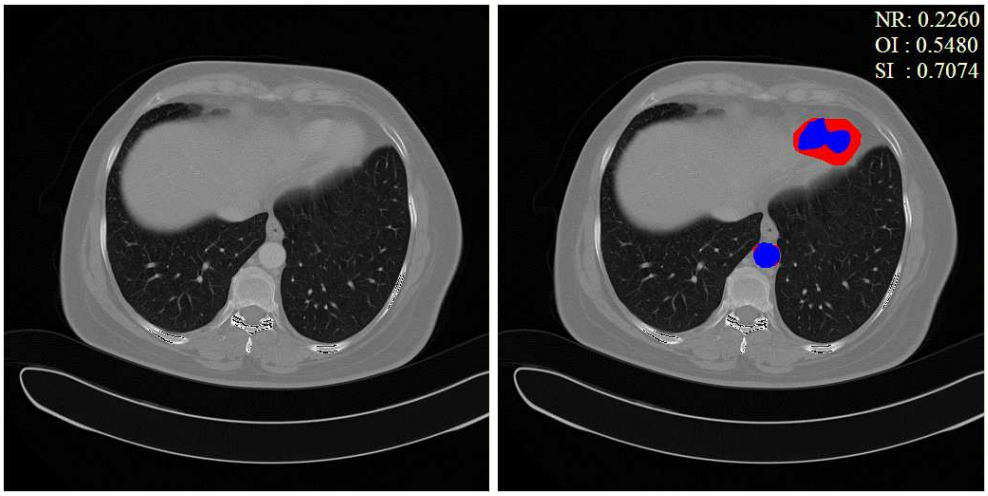


Figure 3.37: Overlapped rate and similarity index based on comparison between the automatically and manually segmented images for heart.

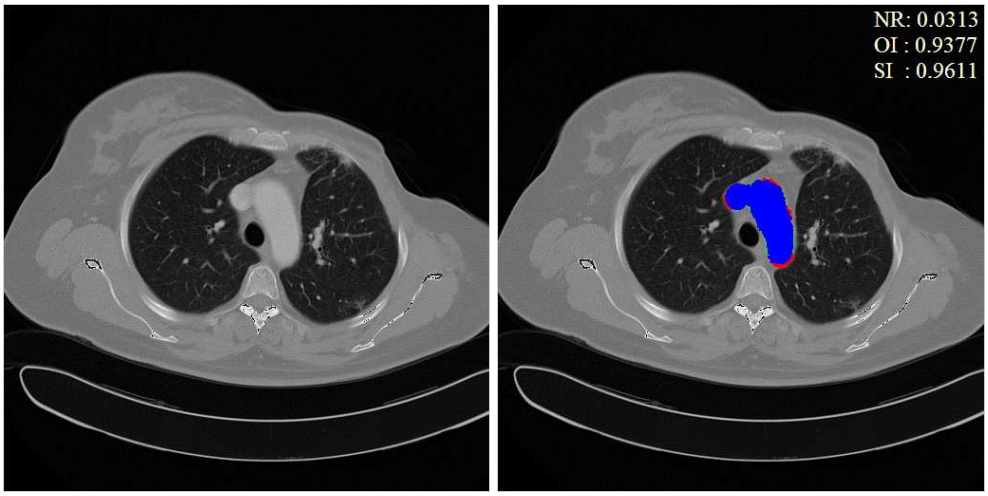


(a)

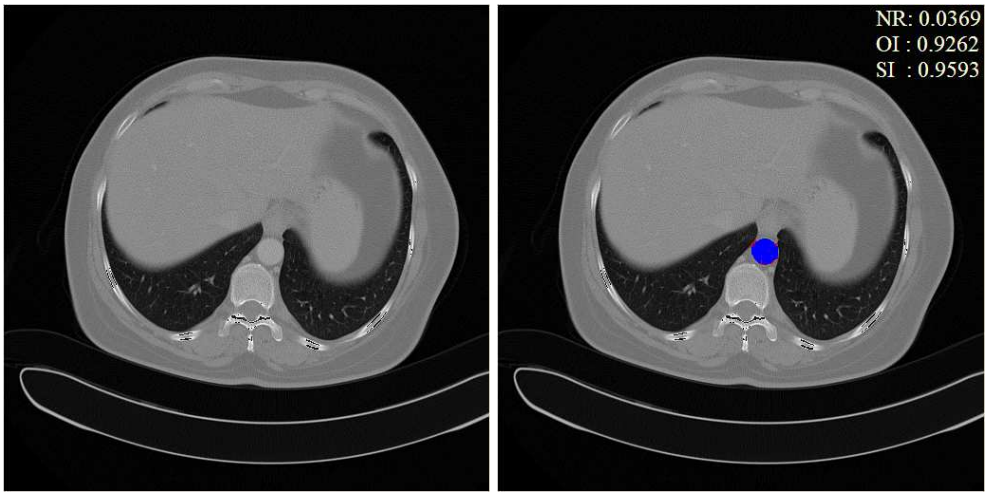


(b)

Figure 3.38: Two examples of bad cardiac segmentation results (right); the slices (a) 25 and (b) 41. The original images (left) are also shown.



(a)



(b)

Figure 3.39: Two examples of good cardiac segmentation results (right); the slices (a) 17 and (b) 45. The original images (left) are also shown.

For the segmentation of all tisues or organs in upper torso, the mean values and the standard deviations of NRM and SI are given in Table 3.2. These results indicate that the best result is obtained for lung segmentation because of the highest mean value and lowest standard deviation of SI. On the contrary, the worst results are obtained for cardiac segmentation due to the lowest mean value and the highest standard deviation of SI.

Table 3.2: The mean values and the standard deviations of NRM and SI for segmentation of lungs, heart and bones using thresholding

	NRM Mean/Std	**Overlapping Mean/Std**	**Similartity Index Mean/Std**
For lung	0.0270/0.0317	0.9479/0.0642	0.9590/0.0397
For bones	0.1477/0.0407	0.7069/0.0817	0.7865/0.0614
For heart	0.2350/0.2028	0.5299/0.4061	0.5713/0.4228

CHAPTER 4

MEDICAL IMAGE SEGMENTATION USING ACTIVE CONTOURS

Over the last decades, active contour models have been widely used in medical image segmentation. The basic idea in active contour models is evolving a close curve, also known as the evolving curve [4, 85, 86, 94], under the constraints from a given image until the curve converges to the boundary of the objects. One chooses an initial curve around the object to be segmented, then it converges to its closest boundary under the influence of external and internal forces. As far as the representation of the evolving curve is concerned, active contour models can be divided into explicit [87, 88] and implicit [4, 89, 90] categories. A typical example of explicit active contour models is snake [87, 88], where the explicit representation of the evolving curve is performed by parametric equations. Level set methods [86, 123, 124], which is an example of implicit active contour models, replace the parametric curve with a signed distance function, called as the level set function. A level set function [86], which is a real-valued function of multiple variables, can take a constant value, such as zero. In such a situation, the zero level set is obtained and used to represent the evolving curve. In other words, the zero set of the level set function implicitly represents the evolving curve. Implicit active contour models can cope with the topological deformations more conveniently than the explicit active contour models [90].

In this section, first we will introduce the Chan-Vese (CV) active contour model [4] as an example of geometric active contour models, and mathematical background about CV method and zeroth and first order level set methods, introduced by Osher and Sethian [123] are given in Appendix B.

4.1 Active Contour Model Without Edges

The active contour models depending on gradient of image is able to delineate only object boundaries marked with gradient of image. In the real world, the stopping function based on gradient never reaches zero on the edges since the discrete gradients are bounded. Thus, the curve cannot catch the boundaries. To address these handicaps, the active contour model without edges have been proposed by Chan and Vese [4], it is also called as Chan-Vese (CV) method.

Chan-Vese method is a powerful and flexible geometric active contour model that can detect objects whose boundaries are not necessarily marked with a gradient. Similar to other active contour models, it aims an energy function to be minimized. Unlike others, its stopping term has not a dependency on image gradient, however. For this reason, it can achieve the segmentation of tissue structures, some of which would be difficult to define object boundaries due to the complexity of the tissue structures.

Such minimal partition problem derived from active contour model can be formulated and solved using the level set method. Level set method can be used a single phase segmentation by using zero level set or can be extended to multiple phase segmentation by using multiple level set functions. With N level set functions one can represents up to 2N phases. For instant, it is possible to represent four distinct regions with two level set functions. The mathematical backgrounds for the active contours without edges and level set functions are clearly described in Appendix B.

4.2 A Framework For Segmentation on Consecutive Medical Image Slices

In this section, we propose a semi-automatic segmentation framework on a CT Torso image set using the combination of active contour model without edges and sequential labeling. In Figure 4.1, the proposed sequential segmentation framework is clearly explained. First of all, a user chooses any slice from the slice set where the objects to be segmented clearly appears, and then selects three pixels that generate two lines (see Appendix E). The part of the image under these lines are discarded since it is not related to thoraxial part of the image. The resultant image is depicted in Figure 4.2.

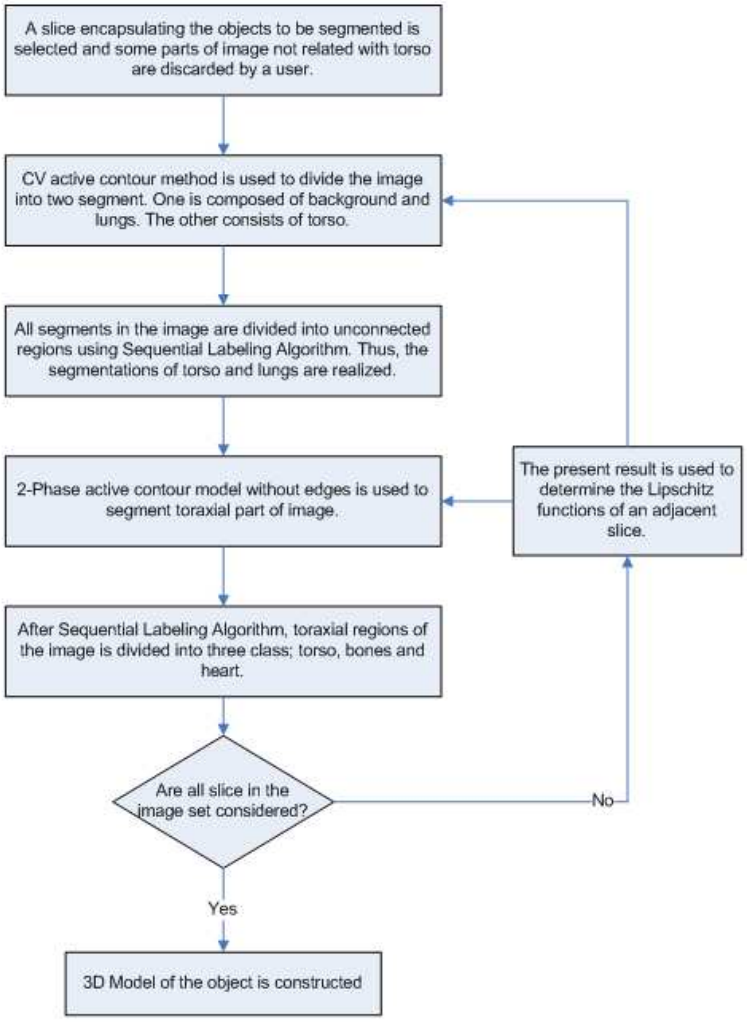


Figure 4.1: Flow Chart of The Proposed Framework

After applying CV active contour method on this resultant image in order to segment it into two groups of pixels, we realize the sequential labeling. As a result of this labeling, the segmentation of torso, lungs, and background is finalized for a slice. The remaining procedure is performed on the set of pixels belonging the torso. 2-Phase active contour model without edges is used in order to divide them into 4 classes. After applying sequential labeling algorithm to obtain the spatial information, we achieve the segmentation of bones and heart. With this last step, the segmentation procedure for a slice is completed (see Figure 4.3). Thus, we move on to the next slice. Because of the fact that two succesive CT images resemble each other, the results of the present slice is assigned as the initial values of the scalar Lipschitz functions for an adjacent slice. The same segmentation procedure is repeated for the current slice. Finally, all slices in the medical image set are segmented. From 2D segmentation of the individual slices, we can reconstruct the 3D model of the objects of interest (see Figure 4.4).

Figure 4.2: User Interference

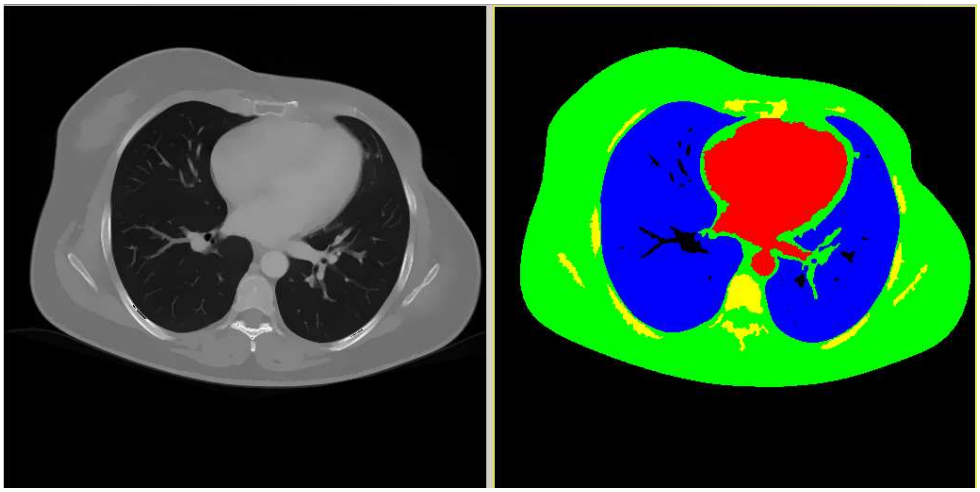
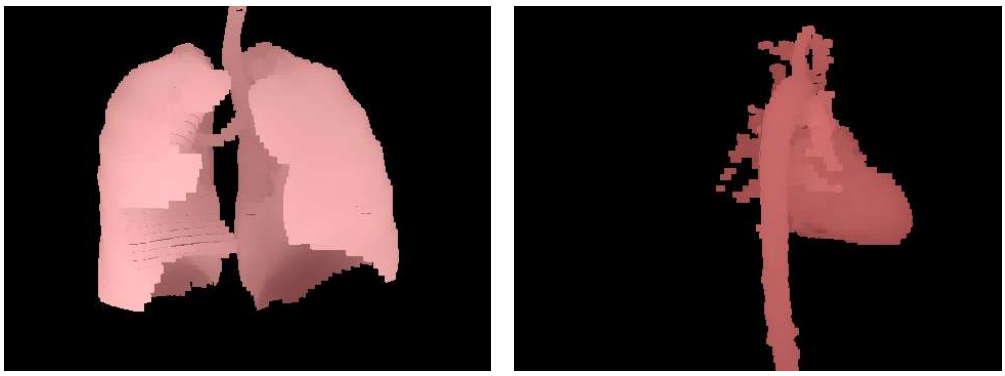


Figure 4.3: The outcome of the segmentation operation for a single slice



(a) (b)

Figure 4.4: 3D representations of the objects

4.3 Results

The proposed framework is performed on CT image set. In Figures 4.5, 4.6, 4.7, and 4.8, the segmentation results for slices 17, 35, 43, 54, respectively, are shown. In these figures, lungs are shown in blue, torso is shown in turquoise, bones are shown in yellow and heart is shown in red.

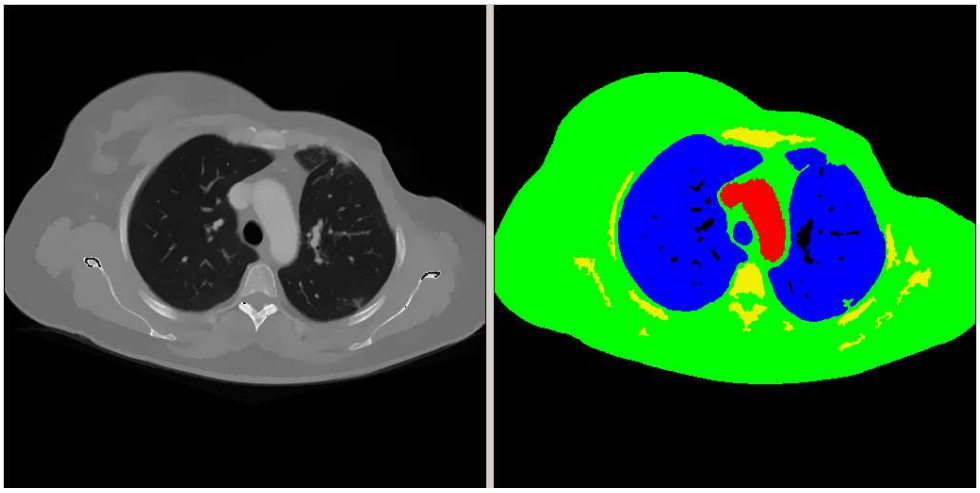


Figure 4.5: The segmentation results for slice 17

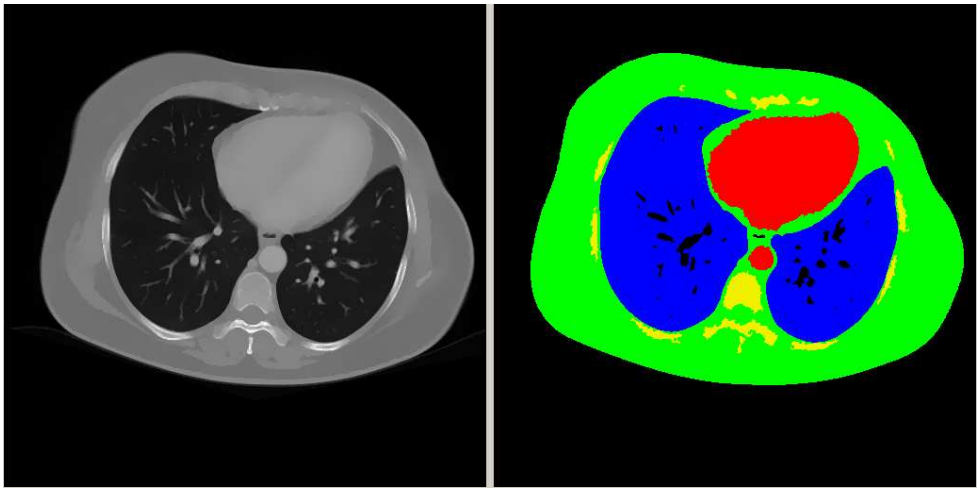


Figure 4.6: The segmentation results for slice 35

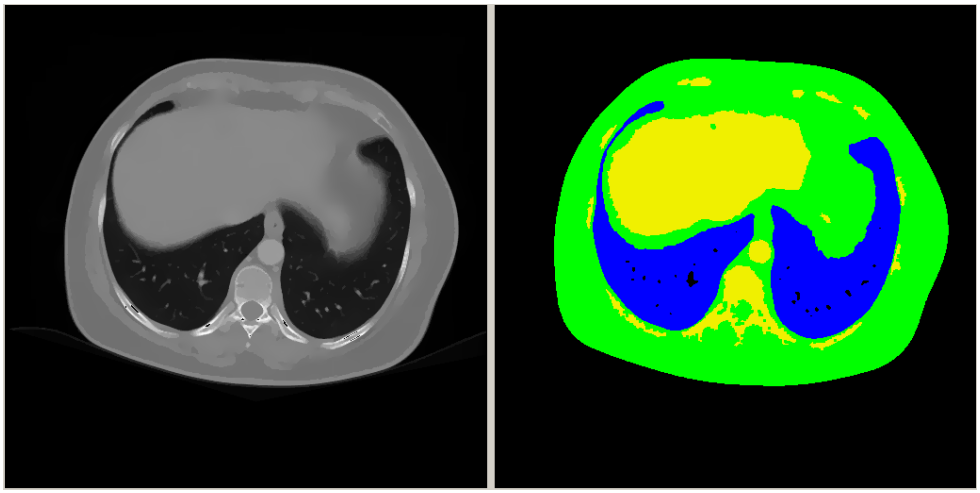


Figure 4.7: The segmentation results for slice 43

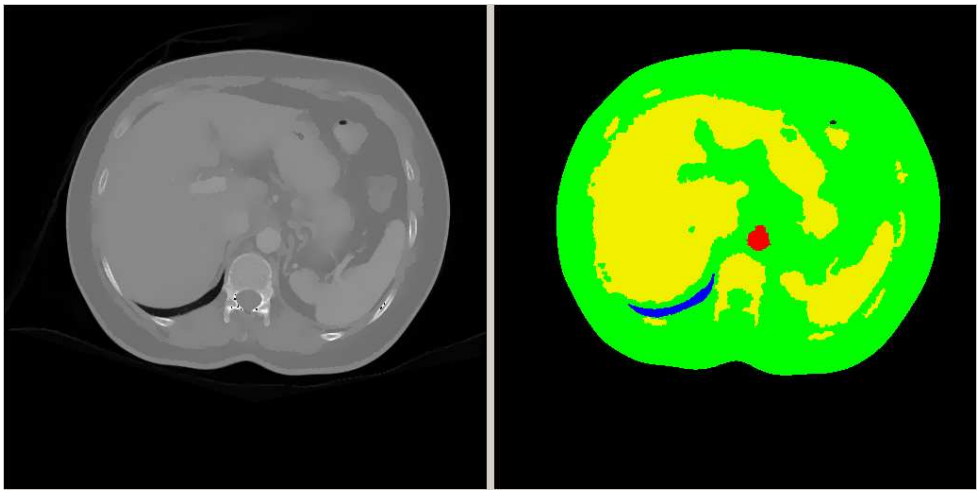


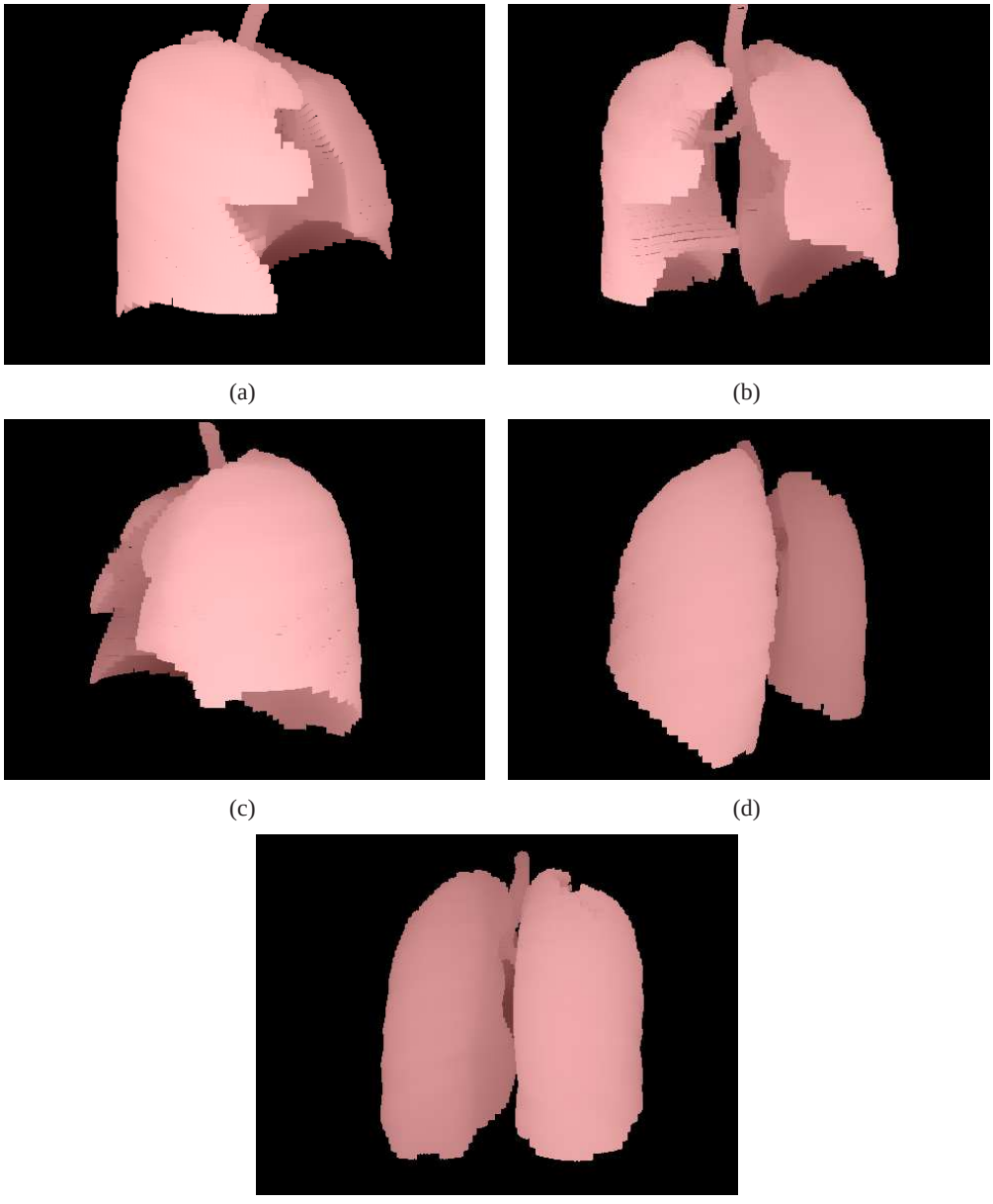
Figure 4.8: The segmentation results for slice 54

The average run-time durations are calculated based on implemented C++ source code run on an Intel(R) Core(TM)2Duo 3.00GHz CPU with 1.95GB RAM of PC and listed in Table 4.1.

Table 4.1: The approximated durations for the framework based on active contour model

	Elapsed Time
Segmentation process for an initial slice	39 sec
Overall segmentation process	1659 sec

In Figures 4.9, 4.10, 4.11, and 4.12, 3D representation of 2D segmentation of lungs, torso, bones, and heart generated from all segmentation images are depicted, respectively.

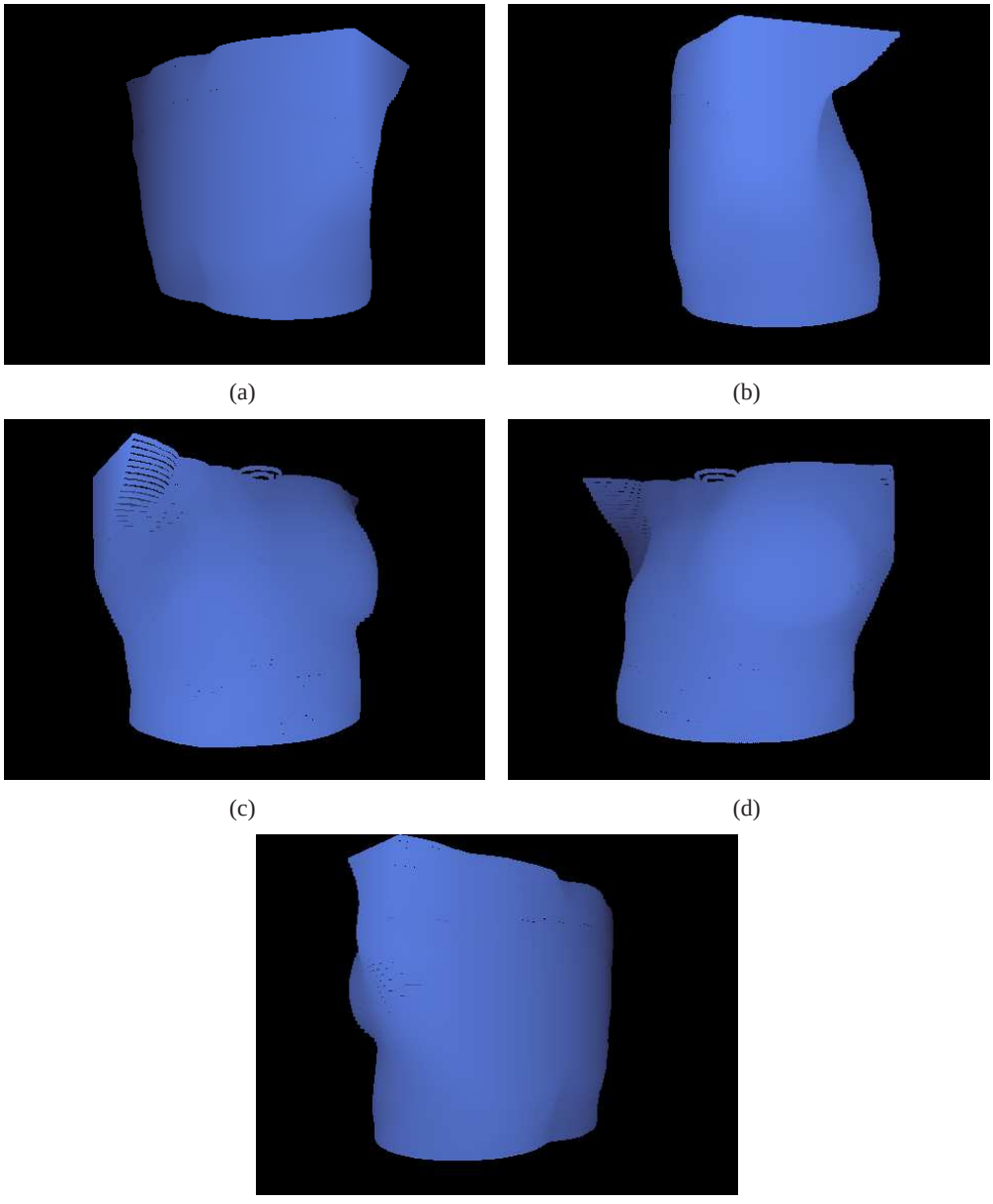


(a) (b)

(c) (d)

(e)

Figure 4.9: 3D representation of 2D segmentation of lungs from different views



(a) (b)

(c) (d)

(e)

Figure 4.10: 3D representation of 2D segmentation of torso from different views

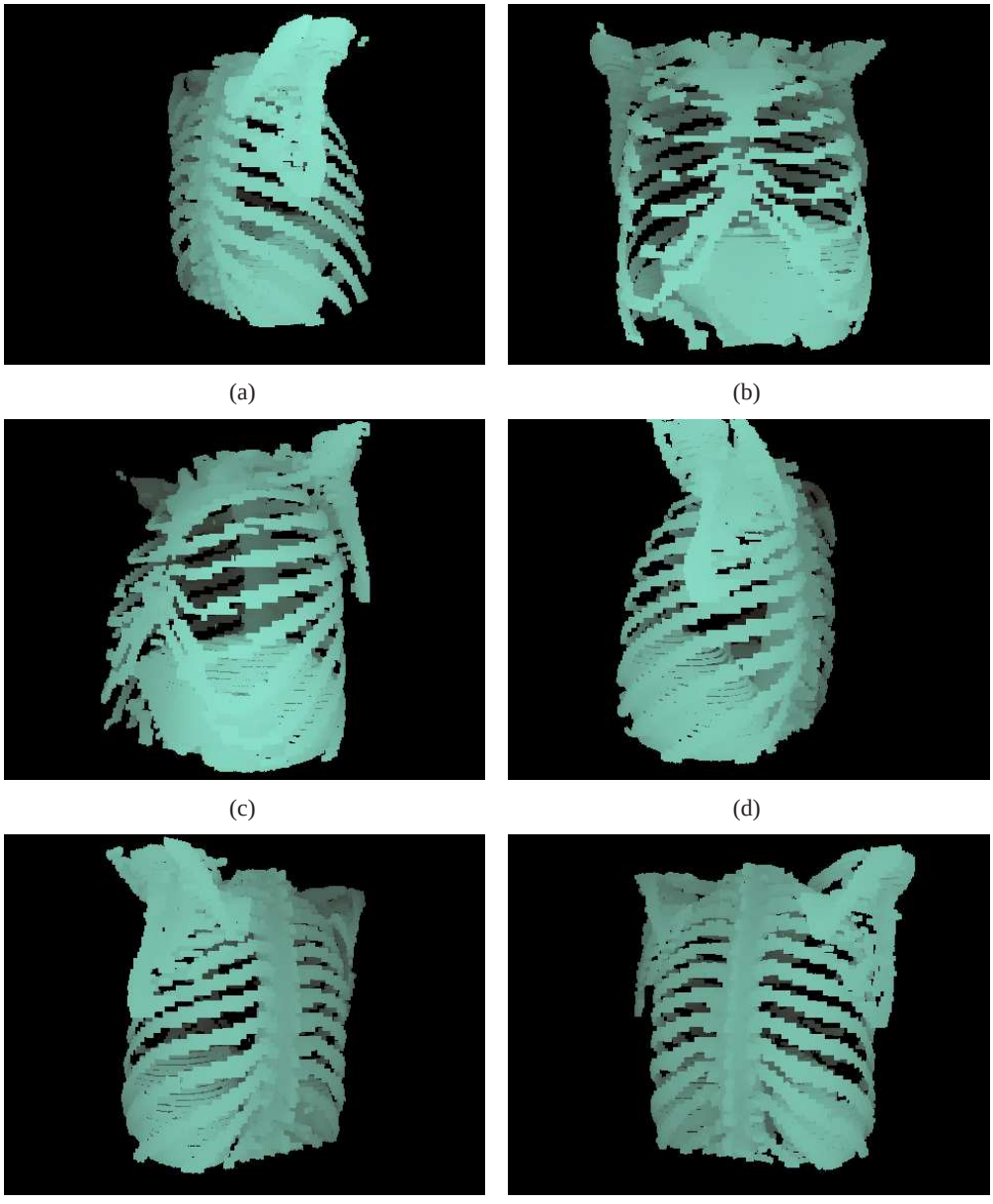


Figure 4.11: 3D representation of 2D segmentation of bones from different views

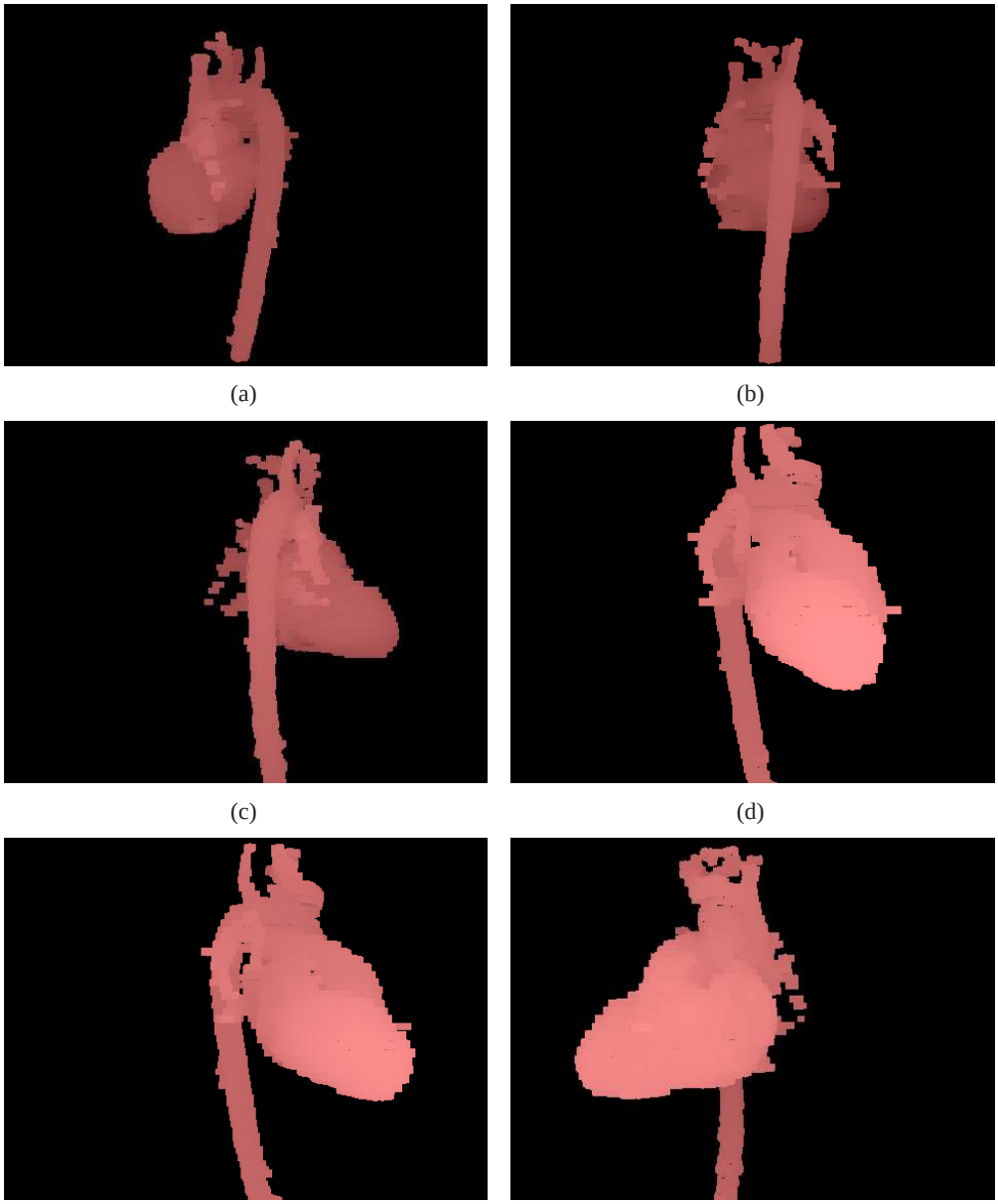


Figure 4.12: 3D representation of 2D segmentation of heart from different views

4.3.1 Quantitative Evaluation of Segmentation Results

As for quantitative evaluations of the segmentation results, in this section, we compare the semi-automatically segmented images based-on active contours with "Ground Truth" resulted from the manual segmentation based-on watershed transform. Negative Rate Metric (NRM), Overlap Index (OI) and Similarity Index (SI) are used as in Chapter 3.

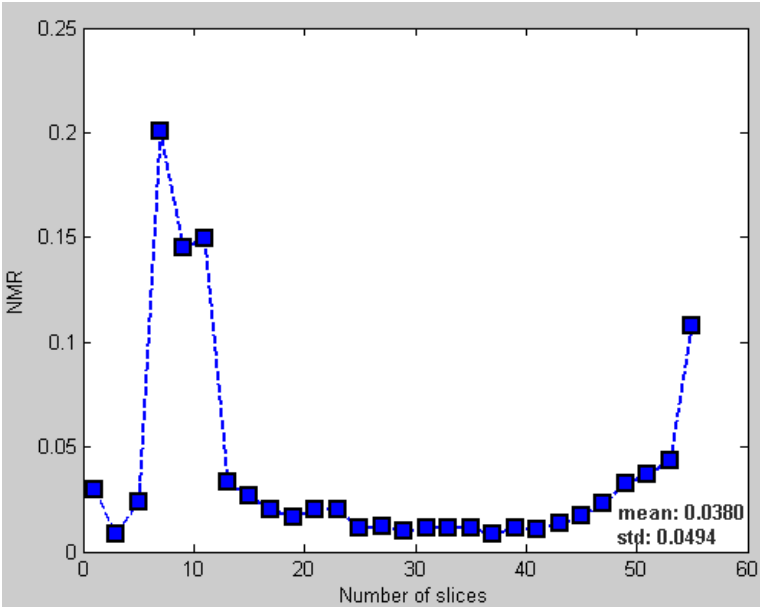


Figure 4.13: Negative Rate Metric results based on comparison between the automatically and manually segmented images for lungs.

In Figures 4.13, 4.14 and 4.15, negative rate metrics, missed, extra segmented, and overlapped rates and similarity indices are depicted based on comparison between the automatically and manually segmented images for lungs. These figures show that all metrics lead to similar conclusions; the slices between 7 and 11 have the worst segmentation results among all lung segmentation results. Panel (a) of Figure 4.16 shows the image from slice 7 along with its segmentation results. Here, the colors red, green, and blue represent missed, extra segmented and overlapped regions, respectively. The numerical values of the evaluation metrics for these images are also given on the segmented images. In this slice, since the upper part of the patient's right lung has been subject to an interstitial lung disease, those pixels are segmented

as torso using the proposed framework. In slice 55, shown in panel (b) of Figure 4.16, the relatively bad results are obtained for lung because of the high proportion of perimeter pixels to actual lung area. Namely, the increase in the proportion of perimeter to the lung area leads to the decrease in the segmentation quality since there is always faulty segmented pixels in the boundary regions of the structures. As for the remaining slices, we achieve good lung segmentation results; for example, in some slices such as slice 37, similarity index for lung reaches 0.99. Figure 4.17 shows lung segmentation results of two slices (29 and 37) with good values of evaluation metrics. As shown in the slice 37, contrary to the segmentation results using thresholding, the esophagus is not segmented as lung despite its lower intensities.

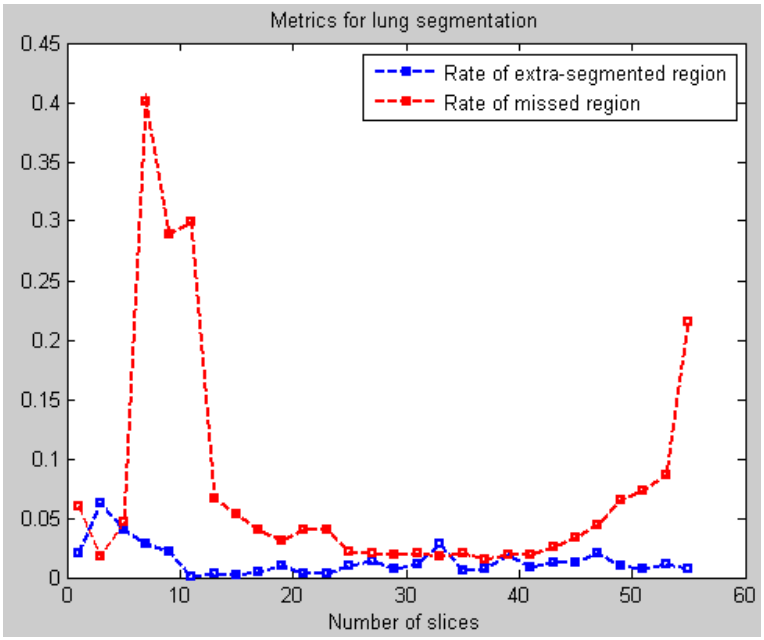


Figure 4.14: Missed and extra segmented rates based on comparison between the automatically and manually segmented images for lungs.

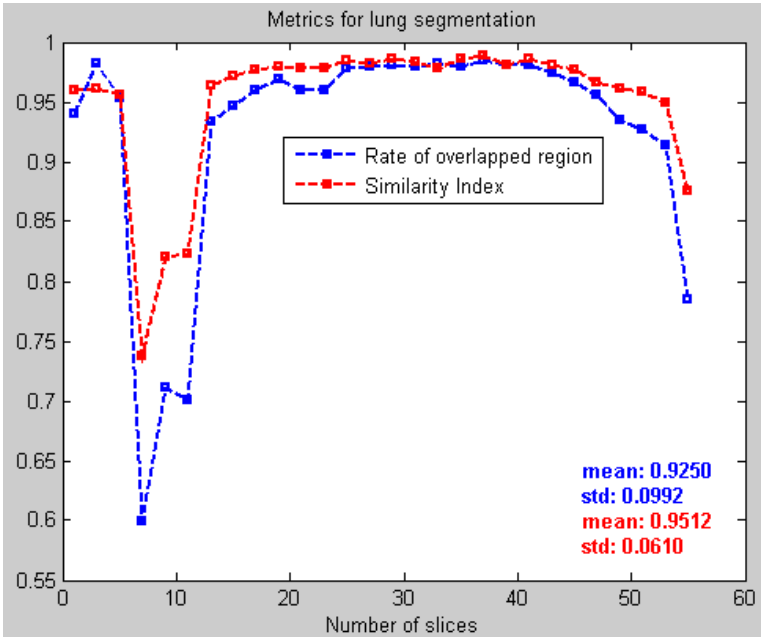
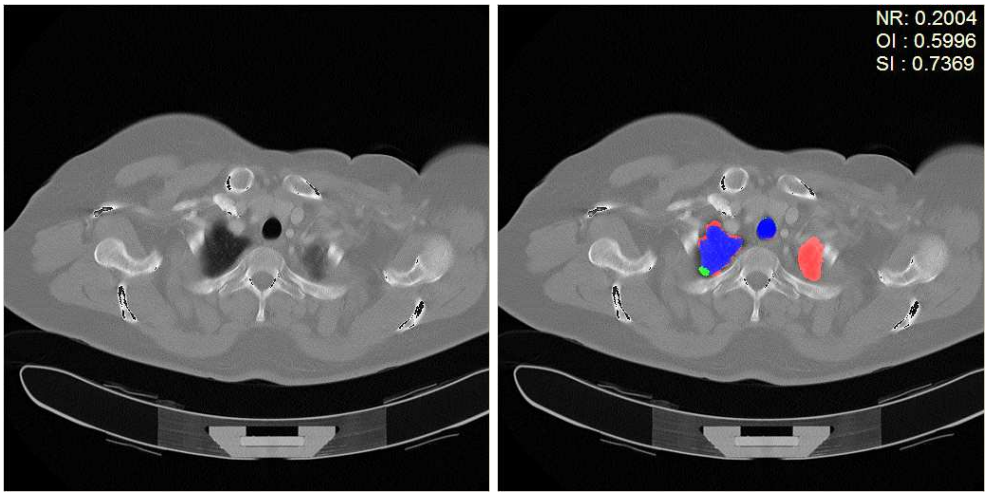
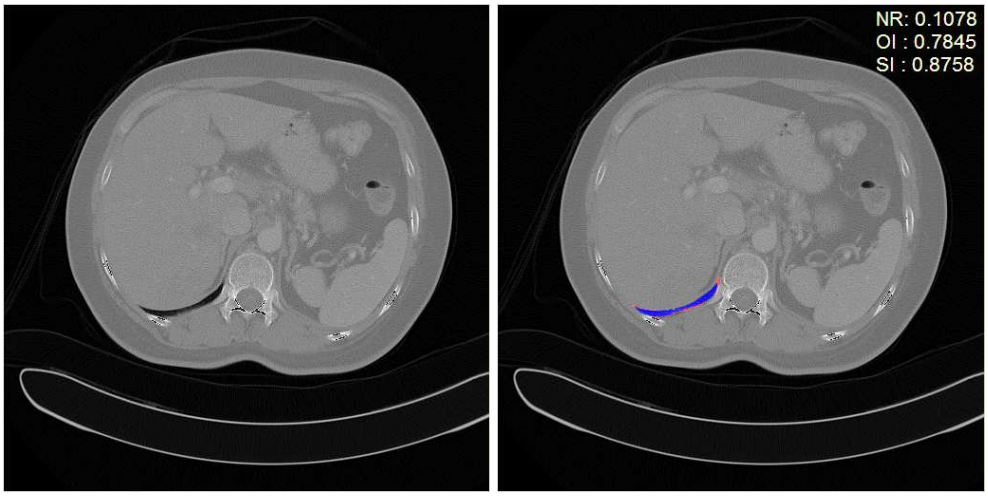


Figure 4.15: Overlapped rate and similarity index based on comparison between the automatically and manually segmented images for lungs.

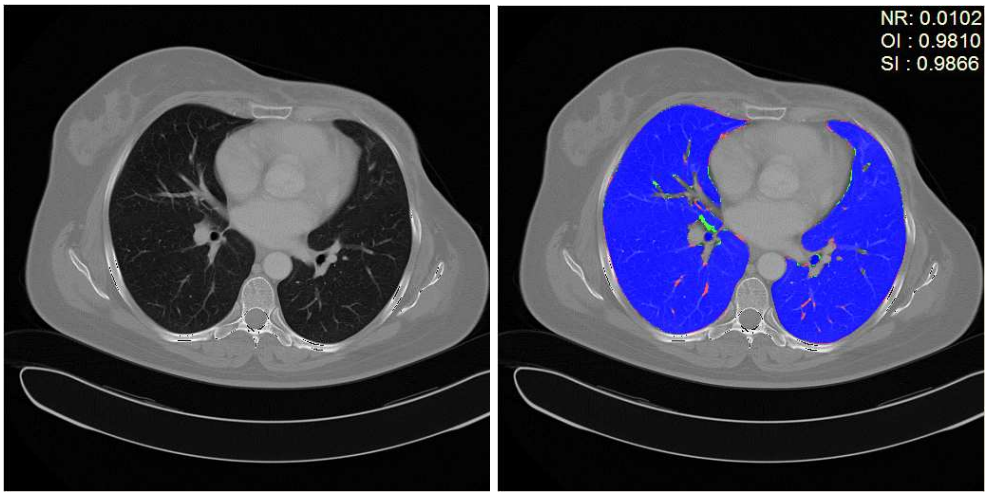


(a)

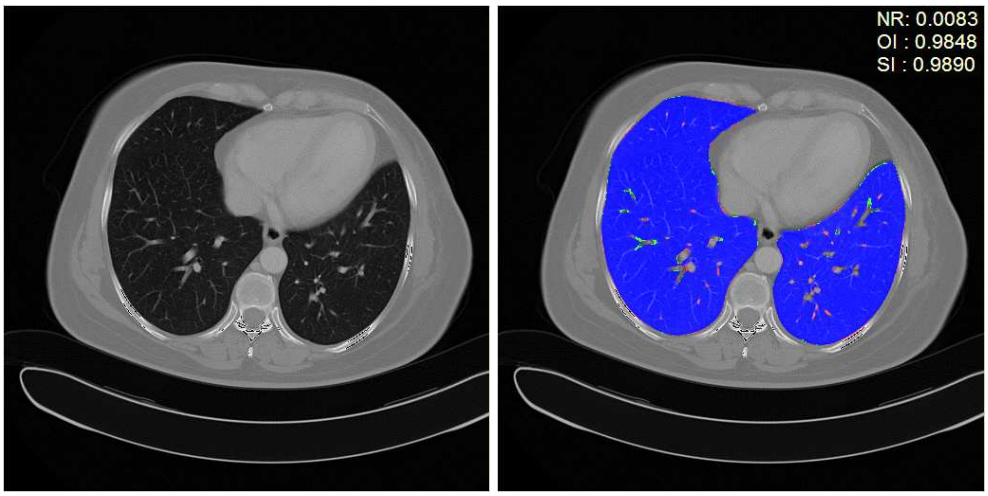


(b)

Figure 4.16: Two examples of bad lung segmentation results (right); the slices (a) 07 and (b) 55. The original images (left) are also shown.



(a)



(b)

Figure 4.17: Two examples of good lung segmentation results (right); the slices (a) 29 and (b) 37. The original images (left) are also shown.

Quantitative results for torso segmentation are given in Figures 4.18, 4.19 and 4.20. In these figures, negative rate metrics, missed, extra segmented, and overlapped rates and similarity indices are depicted, respectively, based on comparison between the automatically and manually segmented images.

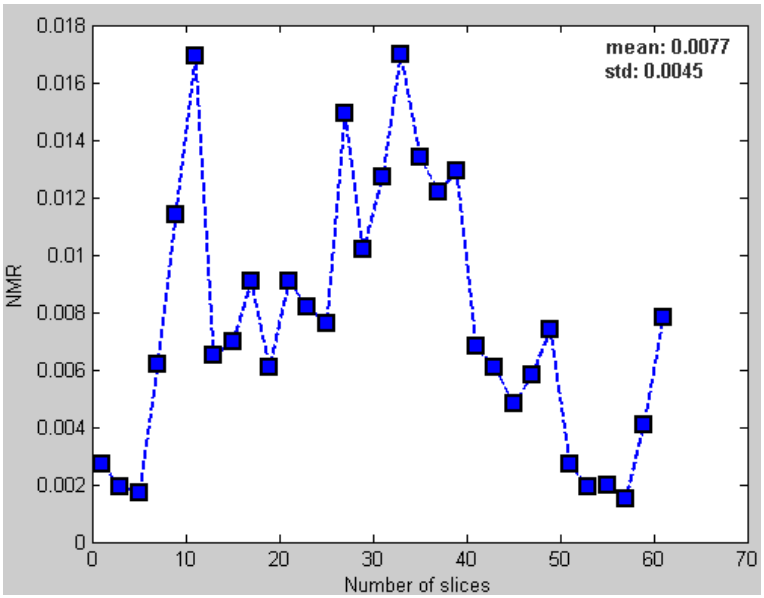


Figure 4.18: Negative Rate Metric results based on comparison between the automatically and manually segmented images for torso.

In the thoraxial segmentation, we define bones and heart as the thoraxial tissues. On the contrary, we exclude lungs from the thoraxial tissues. For this reason, the faulty segmented regions in lungs cause the segmentation performance for torso to decline. As shown in Figure 4.19, the rate of extra segmented regions peaks in the slices between 7 and 11 since the upper part of the patient's right lung has been subject to an interstitial lung disease. This also brings about the decrease in the similarity index.

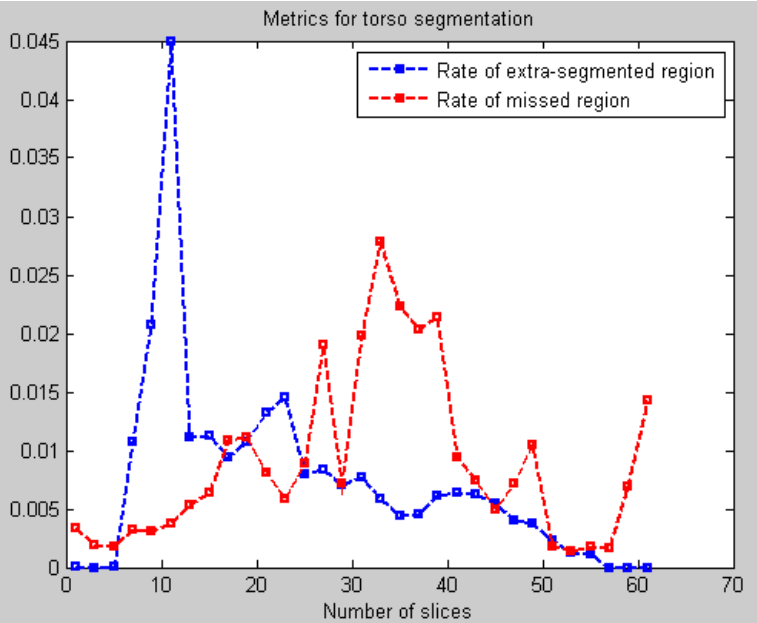


Figure 4.19: Missed and extra segmented rates based on comparison between the automatically and manually segmented images for torso.

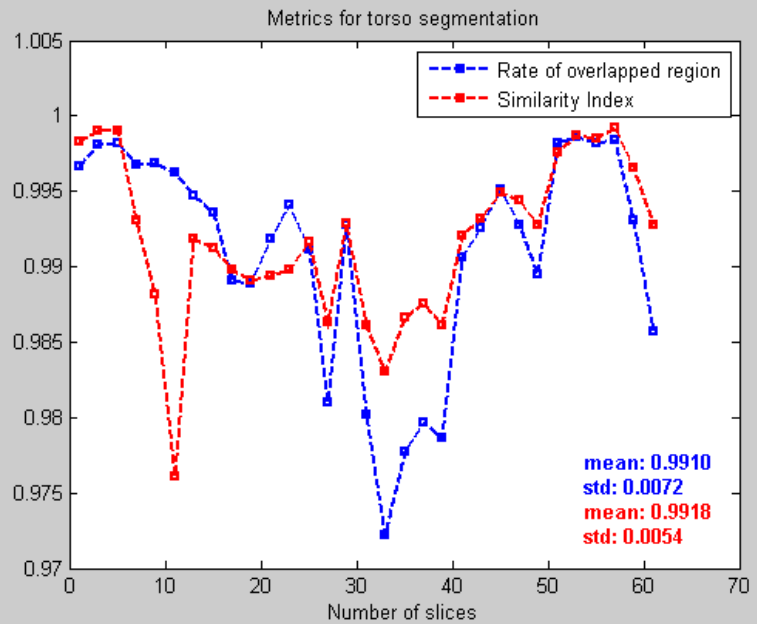


Figure 4.20: Overlapped rate and similarity index based on comparison between the automatically and manually segmented images for torso.

Quantitative results for bone segmentation are given in Figures 4.21, 4.22 and 4.23. In these figures, negative rate metrics, missed, extra segmented, and overlapped rates and similarity indices are depicted, respectively, based on comparison between the automatically and manually segmented images. Two examples with poor evaluation metric values came from slices 27 and 51 (see Figure 4.24) and the ones with good results are from slices 05 and 31 (see Figure 4.25) according to both the similarity index and negative rate metric. In these figures also, the colors red, green, and blue represent missed, extra segmented and overlapped regions, respectively.

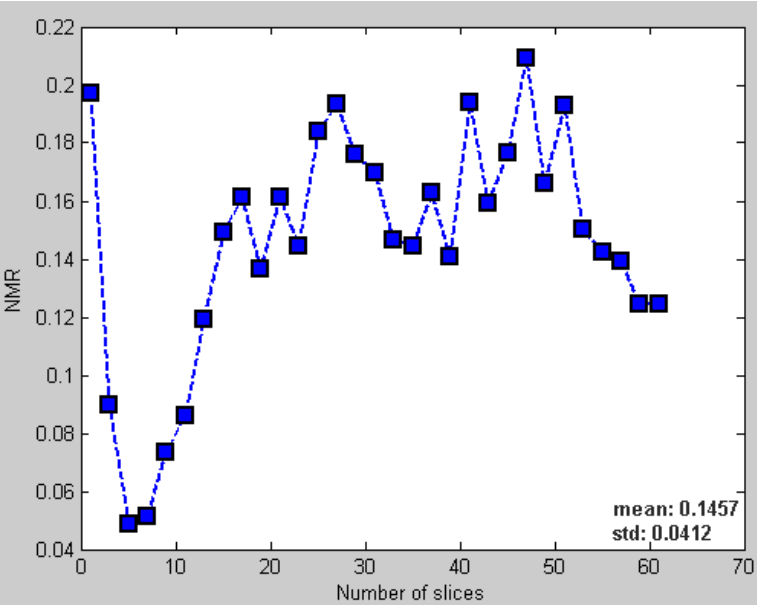


Figure 4.21: Negative Rate Metric results based on comparison between the automatically and manually segmented images for bones.

As shown in the Figures 4.24(a) and 4.24(b), some pixels have higher intensities belonging to liver, stomach, or other abdominal organs. Because of the similarities between the intensities of these tissues and the bones, our algorithm labels the pixels belonging to liver and stomach as bones. Moreover, it misses the pixels belonging to the spongy bones and the cartilages. Around these slices, the overall segmentation quality tends to decrease (see the mean value of SI (Figure 4.23)). Nevertheless, similarity index for bones rises over 0.80 in some slices, as seen in the Figure 4.25.

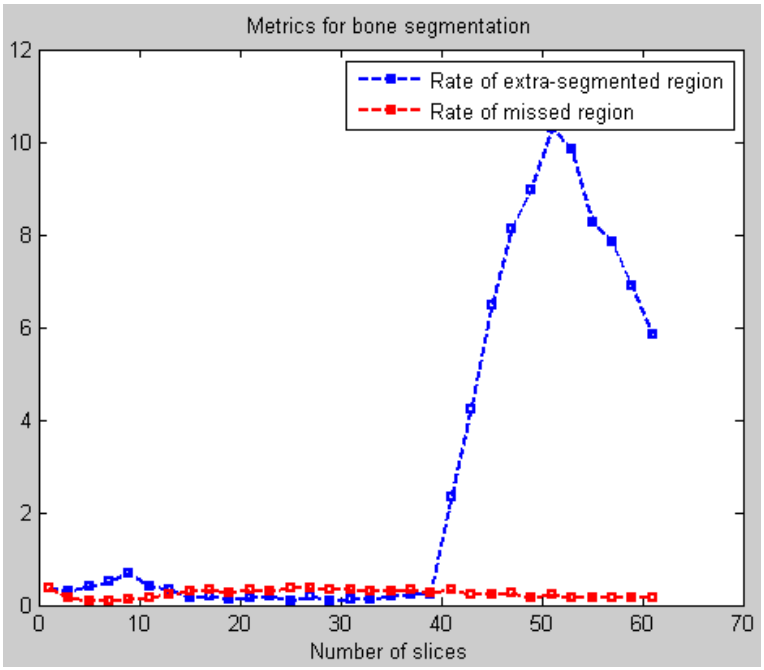


Figure 4.22: Missed and extra segmented rates based on comparison between the automatically and manually segmented images for bones.

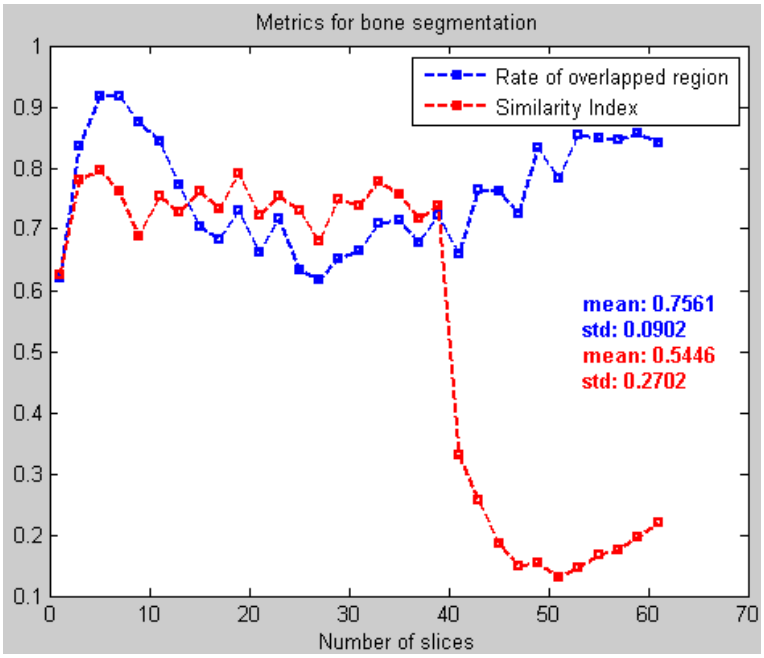
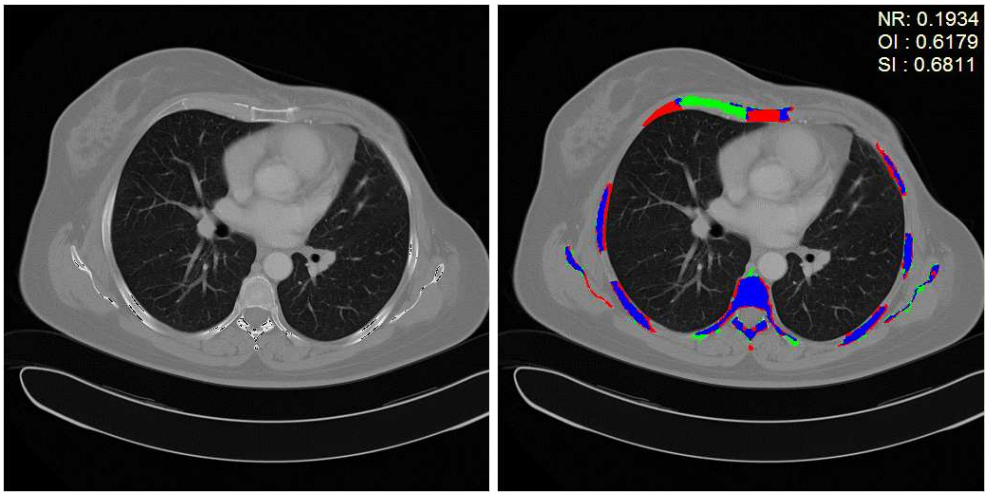
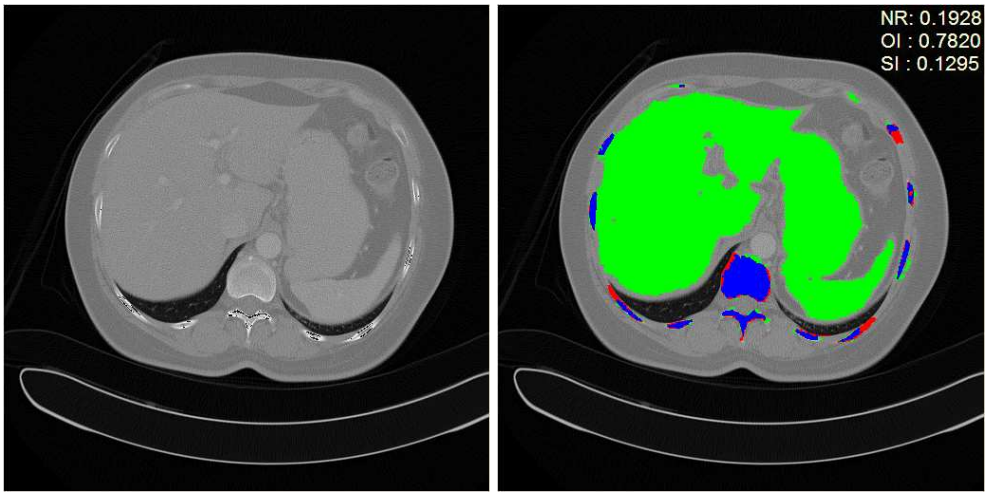


Figure 4.23: Overlapped rate and similarity index based on comparison between the automatically and manually segmented images for bones.

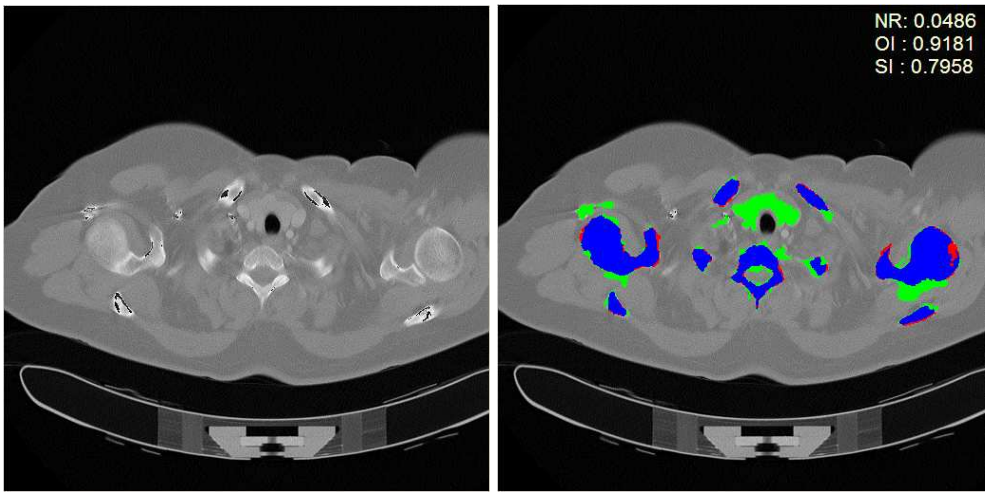


(a)

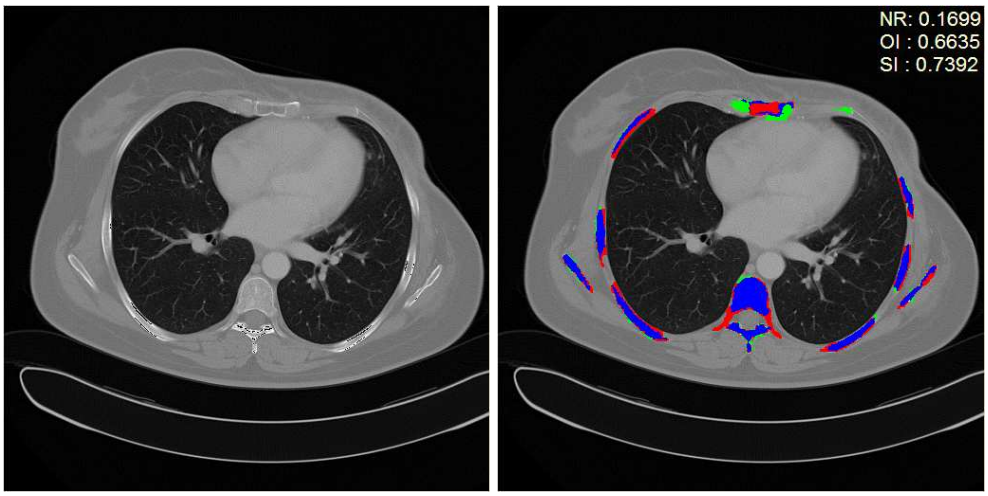


(b)

Figure 4.24: Two examples of bad bone segmentation results (right); the slices (a) 27 and (b) 51. The original images (left) are also shown.



(a)



(b)

Figure 4.25: Two examples of good bone segmentation results (right); the slices (a) 05 and (b) 31. The original images (left) are also shown.

In Figures 4.27 and 4.28, missed, extra segmented, and overlapped rates and similarity index are depicted based on comparison between the automatically and manually segmented images for the heart. According to both the similarity index and the negative rate metric, bad results are observed in slices 25 and 39 (see 4.29) and the good ones are from the slices 35 and 51 (see 4.30). In these figures, the color coding is same as in lungs and bones.

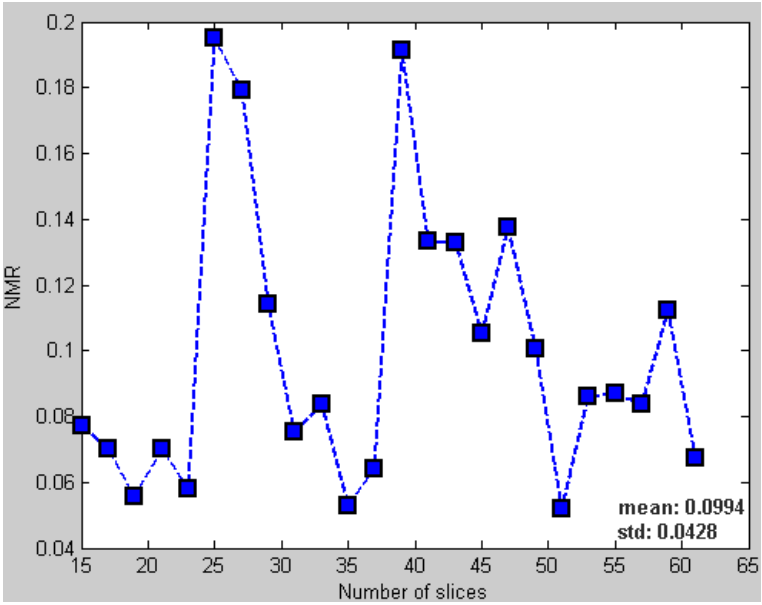


Figure 4.26: Negative Rate Metric results based on comparison between the automatically and manually segmented images for heart.

Because of the usage of intravenous contrast material, our algorithm misses the pixels belonging to muscle (*i.e.*, the red regions) as seen in Figure 4.29(a). On the contrary, it selected the pulmonary vessels (*i.e.*, the green regions) due to their high intensities. In slice 39 shown in Figure 4.29(b), our algorithm did not produce good results because of the uncertain boundaries of the heart. Nonetheless, there are some slices such as 35 and 51, where similarity index reaches to 0.95 as seen in Figure 4.30. Unlike the framework based on thresholding, the algorithm was able to segment the descending thoracic aorta.

For the segmentation of all tisues or organs in upper torso, the mean values and the standard deviations of NRM and SI are given in Table 4.2. These results indicate that the best results are obtained for lung segmentation because of the highest mean value

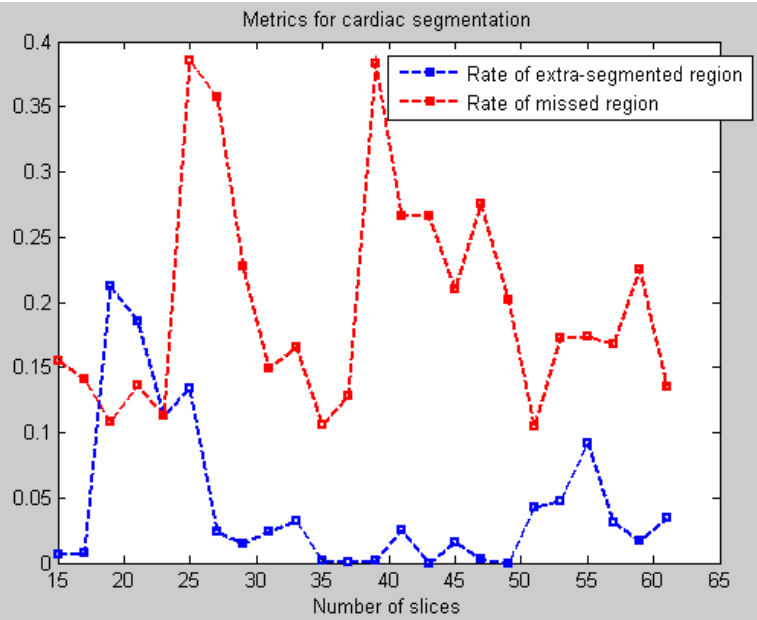


Figure 4.27: Missed and extra segmented rates based on comparison between the automatically and manually segmented images for heart.

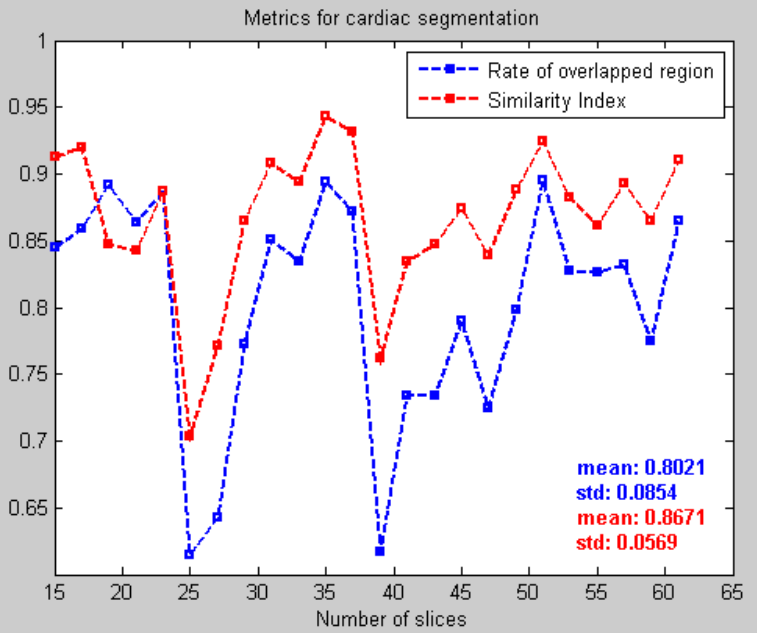
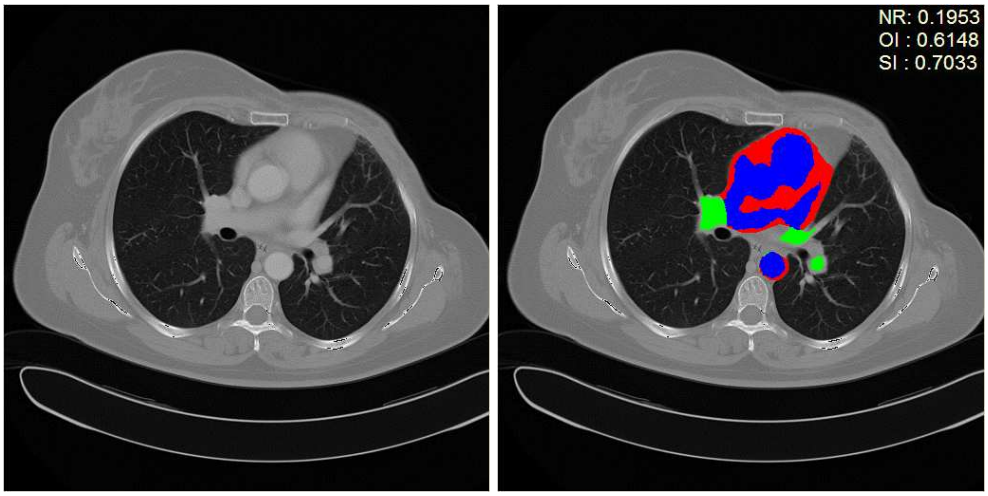


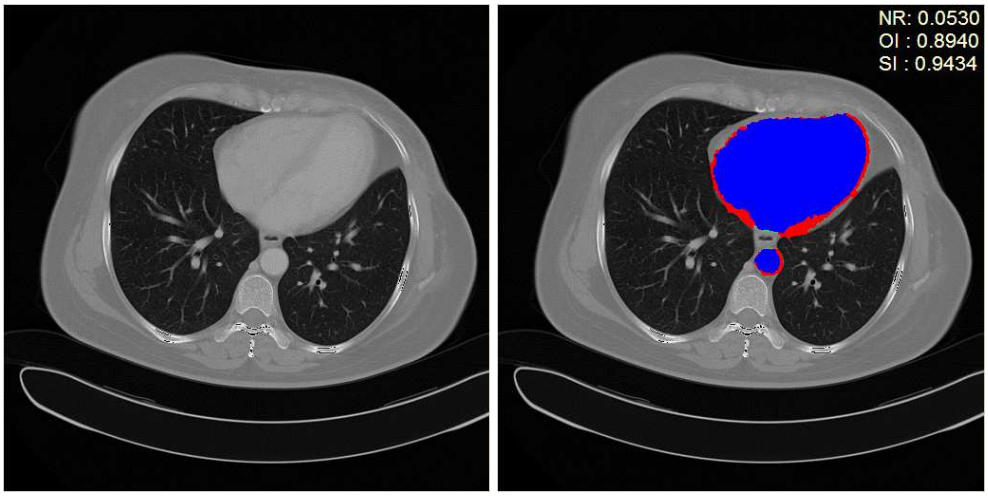
Figure 4.28: Overlapped rate and similarity index based on comparison between the automatically and manually segmented images for heart.



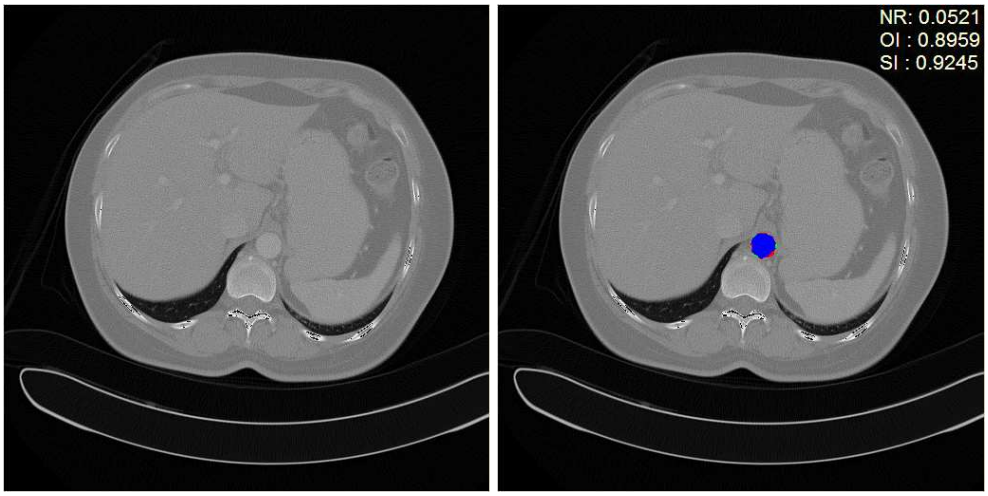
(a)

(b)

Figure 4.29: Two examples of bad cardiac segmentation results (right); the slices (a)25 and (b) 39. The original images (left) are also shown.



(a)



(b)

Figure 4.30: Two examples of good cardiac segmentation results (right); the slices (a)35 and (b) 51. The original images (left) are also shown.

and lowest standard deviation of SI. On the contrary, the worst results are obtained for cardiac segmentation due to the lowest mean value and the highest standard deviation of SI.

Table 4.2: The mean values and the standard deviations of NRM and SI for segmentation of lungs, heart and bones using thresholding

	NRM Mean/Std	**Overlapping Mean/Std**	**Similartity Index Mean/Std**
For lung	0.0380/0.0494	0.9250/0.0992	0.9512/0.0610
For bones	0.1457/0.0412	0.7561/0.0902	0.5446/0.2702
For heart	0.0994/0.0428	0.8021/0.0854	0.8671/0.0569

CHAPTER 5

MEDICAL IMAGE SEGMENTATION USING WATERSHED TRANSFORM

As in many fields of image processing, watershed transform has been widely used in medical image segmentation [125]. The intuitive description of this method coming from geography is that a grey-level image may be considered as topographic relief, where the intensity value of pixel is interpreted as its altitude in the relief. Let us consider that the rain is gradually falling on the terrain, then the watersheds are constructed in order that the catchment basins are totally separated. In general, this method is performed on the gradient of the image instead of itself.

The watershed transform, which is an intuitive method, has a number of advantages. In detail, it is simple to use, it is fast and can be parallelized. In addition to these, it produces plenty of seperated regions from the image even if the contrast is poor, so it does not require any kind of contour joining. On the other hand, it has some important disadvantages such as oversegmentation, sensitivity to noise, and poor detection of thin structures and significant areas with low contrast boundaries [59].

In this section, first we will introduce mathematical background about the watershed transform and k-means clustering. Then, we will show experimental result of these methods. Finally, we conclude the experiment of sequential segmentation on medical image slices.

5.1 Mathematical Theory

5.1.1 Watershed Definitions for Digital Images

Let us consider I is a 2D grey value image, the pixel value of the image $I(x,y)$ is between the range $(0,N)$, and $N_G(x,y)$ is the set of the neighbours of a pixel located at (x,y) with respect to the digital grid G.

There are two definitions of watershed for the digital images; the algorithmic definition by Vincent and Soille [126] and the definition by topographical distance by Meyer [127]. In this thesis, we focus on the algorithmic definition of the watershed transform by immersion proposed by Vincent and Soille [126].

Algorithmic definition by immersion: Let $I : D \rightarrow \mathtt{N}$ be a digital grey value image, which possesses the minimum value h_{min} and the maximum value h_{max}. When it undergoes a recursion with the grey level h increasing from h_{min} to h_{max}, the basins associated with the minima of I are successively extended. Let B_h denote the union of the set of basins computed at level h. A connected component of the threshold set T_{h+1} at level $h+1$ can be either a new minima, or an extension of a basin in B_h: in the latter case one computes the geodesic influence zone of B_h within T_{h+1}, resulting in an update $B_h + 1$. Let MIN_h denote the union of all regional minima at altitude h. Define the following recursion:

$$\left\{ \begin{array}{c} B_{h_{min}} = \{(x,y) \in D | I(x,y) = h_{min}\} = T_{h_{min}} \\ B_{h+1} = MIN_{h+1} \cup \text{Influence Zone}_{T_{h+1}}(B_h),\ h \in [h_{min}, h_{max}) \end{array} \right\} \tag{5.1}$$

In the Figure 5.1, an example of the watershed transform according to the recurrence is given, where A, B, and C are labels of basins and W is assigned to denote watershed pixels. Moreover, minima pixels in orijinal image are shown in bold. We note that 8-connectivity is used.

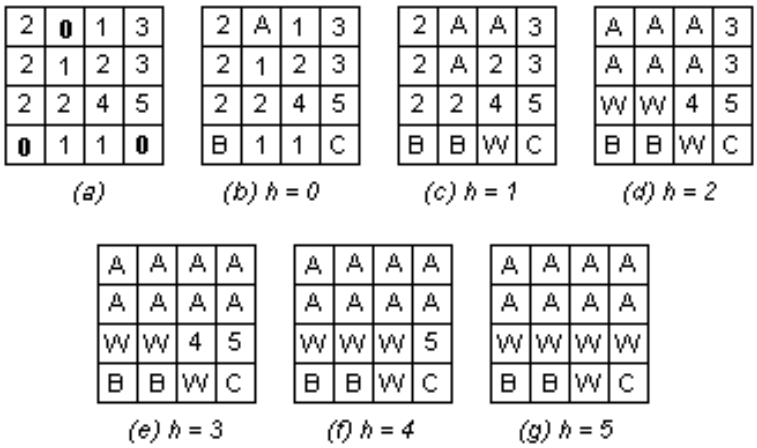


Figure 5.1: Watershed transform by immersion on the 8-connected grid. (a): Original image; (b-g): labelling steps based on 5.1.

5.1.2 K-Means Clustering

Roughly speaking, K-means clustering (also called c-means) is an unsupervise learning algorithm to classify samples based on their features into K number of classes, where K a is positive integer. The classification is performed by minimizing the sum of absolute distances between samples and the corresponding cluster centroid or mean. The final aim of K-means clustering is to classify the samples without supervising. An algorithm for K-means clustering is given in Algorithm 3.

Algorithm 3 An algorithm for K-means clustering

1. Set initial mean values for each of the classes.
2. Classify N samples by assigning the to “closest” mean.
3. Recompute the mean values as the average of samples in their classes.
4. Continue untill there is no change in the mean values.

5.2 A Framework For Sequential Segmentation on Medical Image Slices

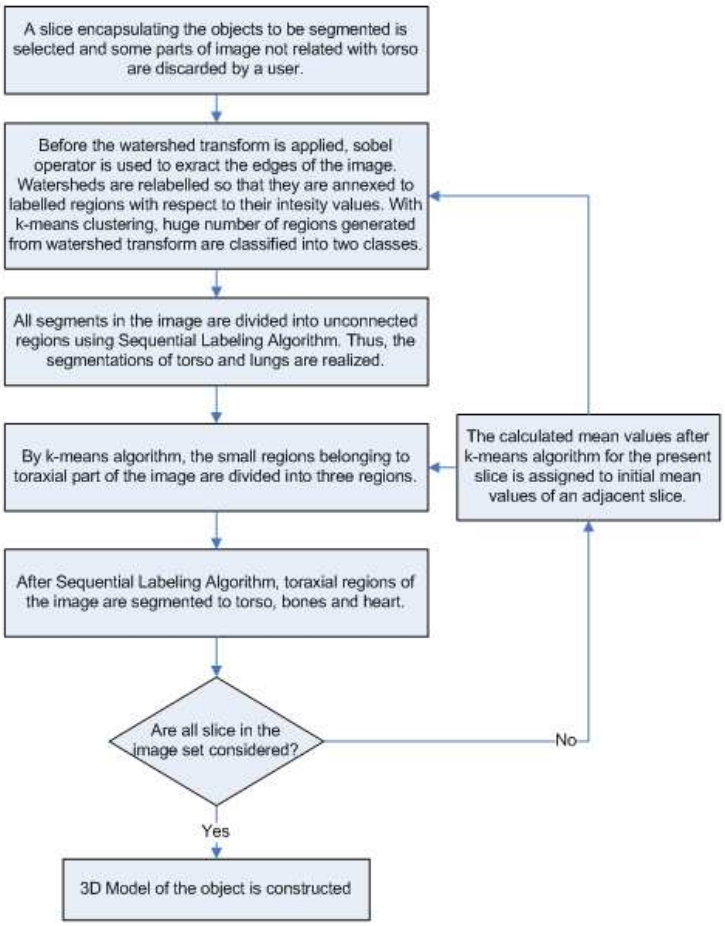


Figure 5.2: Flow Chart of The Proposed Framework

In this section, we propose a semi-automatic sequential segmentation framework on CT Torso image set using the combination of watershed transform, K-means clustering and sequential labeling. In Figure 5.2, the proposed sequential segmentation framework is clearly explained. First of all, a user chooses any slice from the slice set where the objects to be segmented clearly appears and then also selects three pixels that generate two lines (see Appendix E). The parts of the image under these lines are discarded since they are not related to thoraxial part of the image. The resultant image is depicted in Figure 5.3. Performing sobel operator on this resultant image in order to extract edges we realize the watershed transform to generate many small regions. This causes oversegmentation. In order to handle oversegmentation, we need

to merge these small regions. Using K-means clustering algorithm, we classify the regions into two groups. The outcome is used as an input for the sequential labeling. As a result of this, the segmentation of torso, lungs, and background is finalized for a slice. The remaining procedure is performed on the set of pixels belonging to the torso. A new K-means algorithm is performed on the thoraxial regions in order to divide them into 3 classes. After performing sequential labeling algorithm to obtain the spatial information, we achieve the segmentation of bones and heart. Thus, the segmentation procedure for a slice is completed (see Figure 5.4). Next, we need to pass the next slice. Because of the fact that two succesive CT images resemble each other, the mean value results of the present slice are assigned as the initial mean values for K-means clustering of an adjacent slice. With the same manner, all slices in the medical image set are segmented. Finally, we can reconstruct a 3D model of the objects of interest by combining the 2D segmentation results of all slices (see Figure 5.5).

Figure 5.3: User Interference

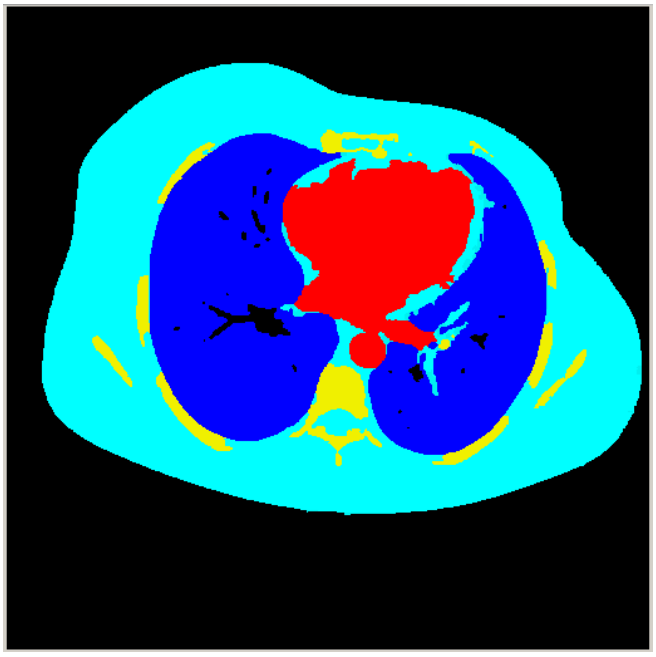
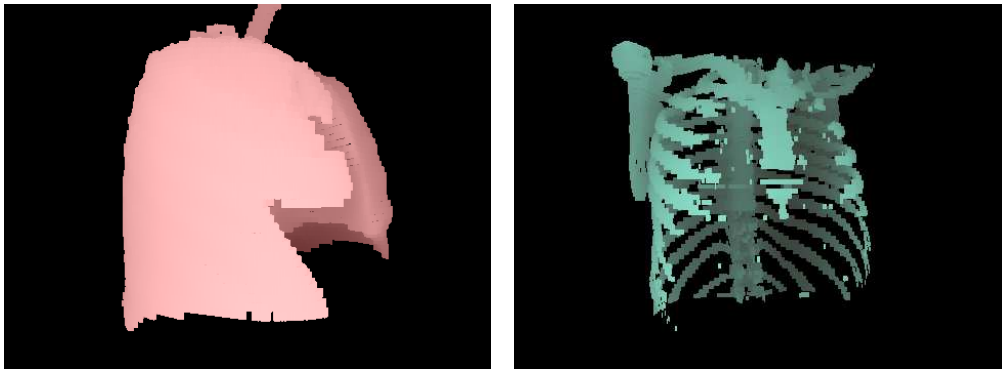


Figure 5.4: The outcome of the segmentation operation for a single slice



(a) (b)

Figure 5.5: 3D representations of the objects

5.3 Results

The proposed framework is performed on our CT image set. The segmentation results of lungs (blue), torso (turquoise), bones (yellow), and heart (red) are depicted in Figures 5.6, 5.7, 5.8, and 5.9, for the slices 17, 35, 43, and 54, respectively.

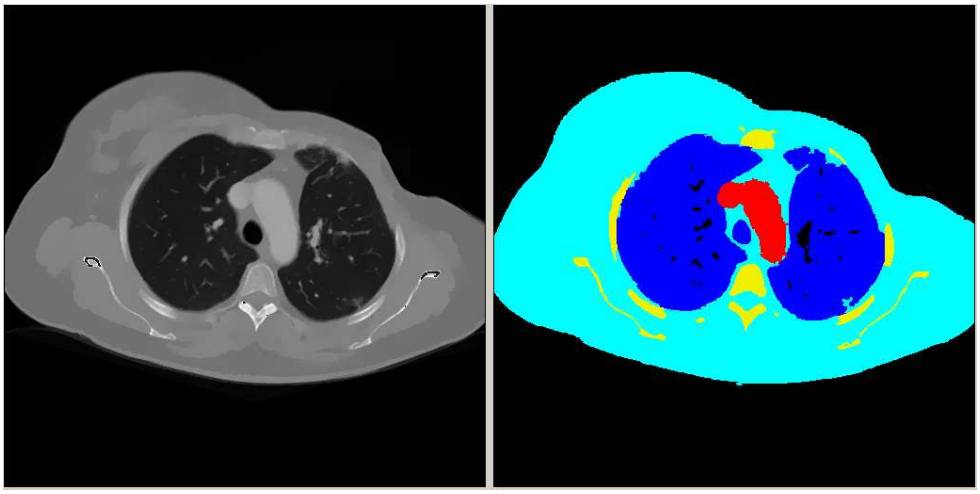


Figure 5.6: The segmentation results for slice 17

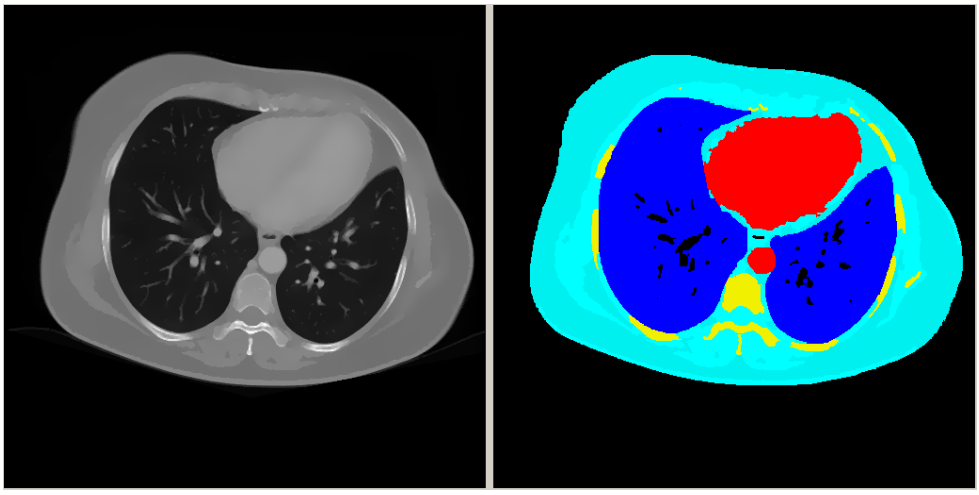


Figure 5.7: The segmentation results for slice 35

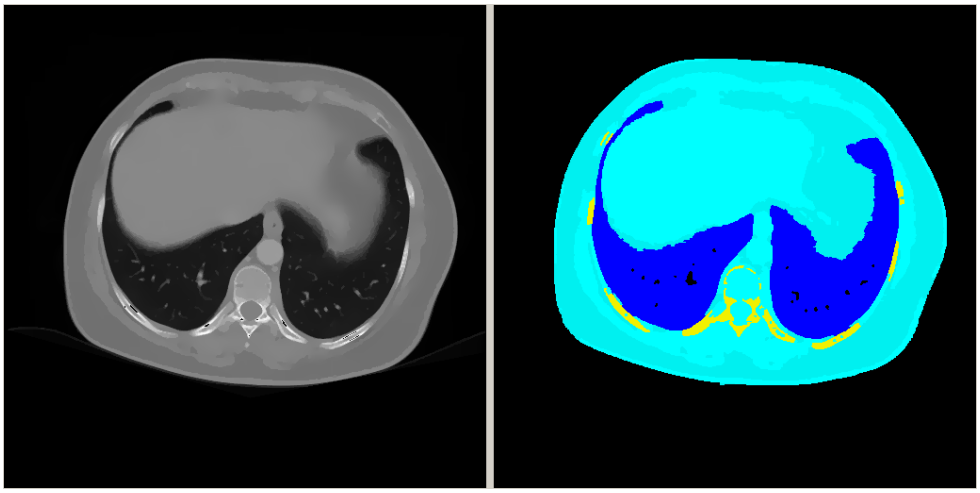


Figure 5.8: The segmentation results for slice 43

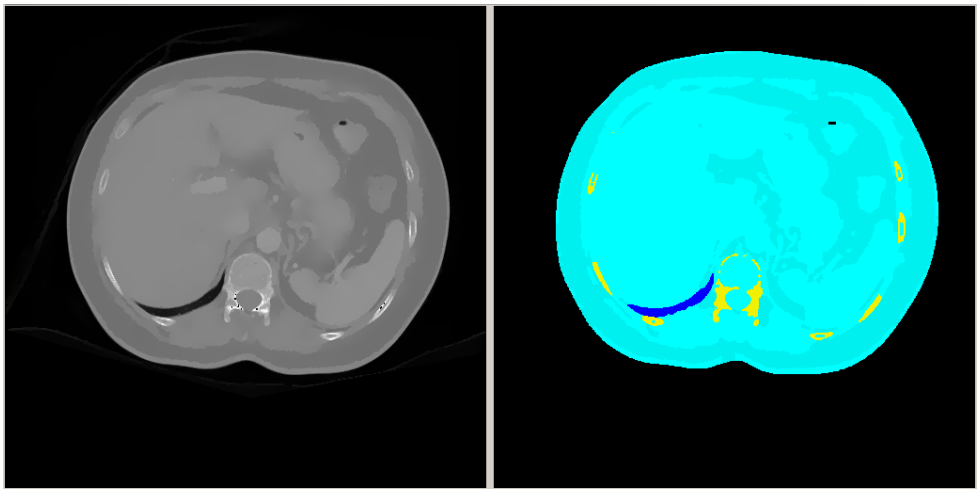


Figure 5.9: The segmentation results for slice 54

The average run-time durations are calculated based on implemented C++ source code run on an Intel(R) Core(TM)2Duo 3.00GHz CPU with 1.95GB RAM of PC and listed in Table 5.1.

Table 5.1: The approximated durations for the framework based on watershed transform

	Elapsed Time
Segmentation process for an initial slice	21 sec
Overall segmentation process	1281 sec

In Figures 5.10, 5.11, 5.12, and 5.13, 3D representation of 2D segmentation of lungs, torso, bones, and heart generated from all segmentation results of image set is depicted, respectively.

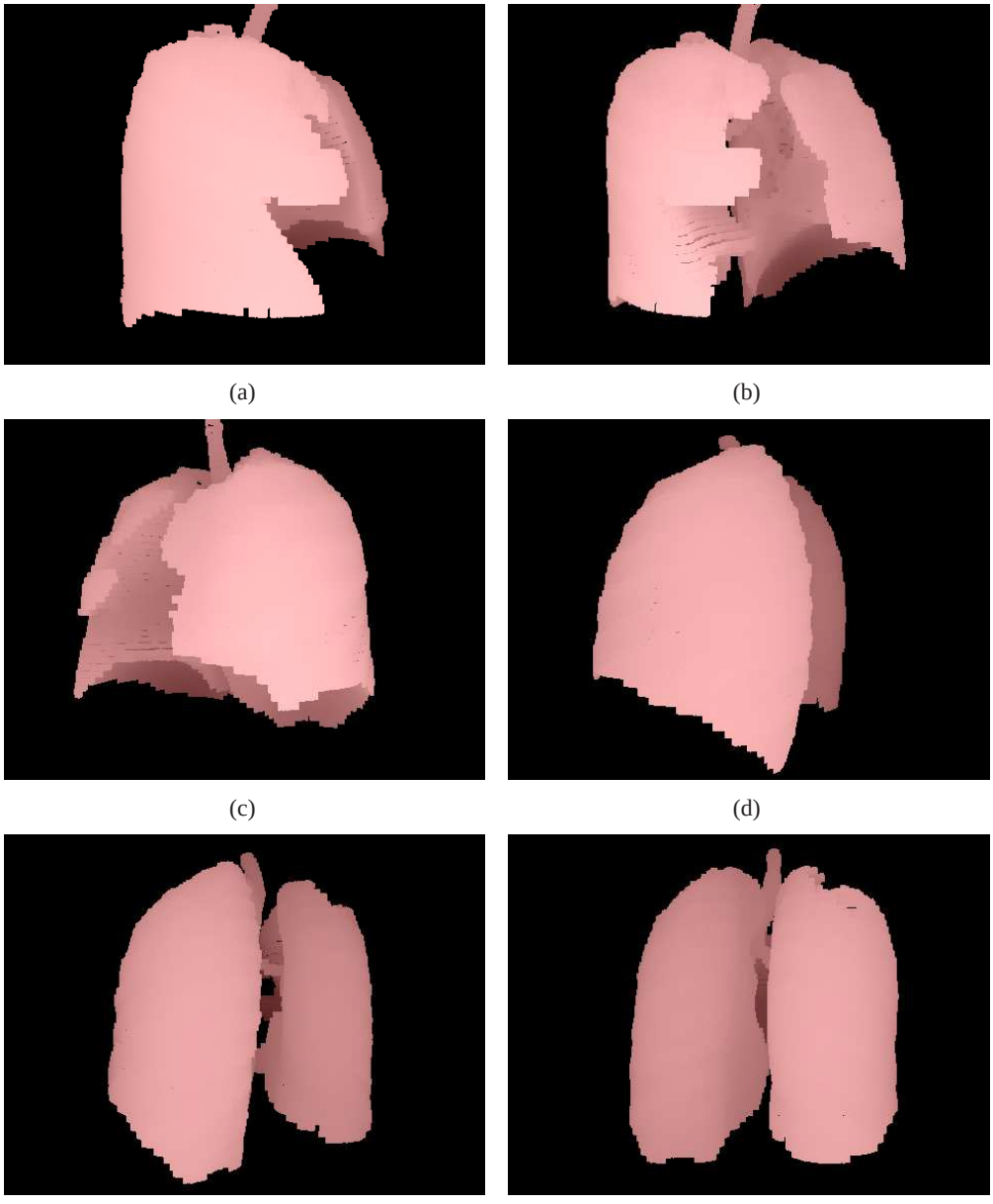
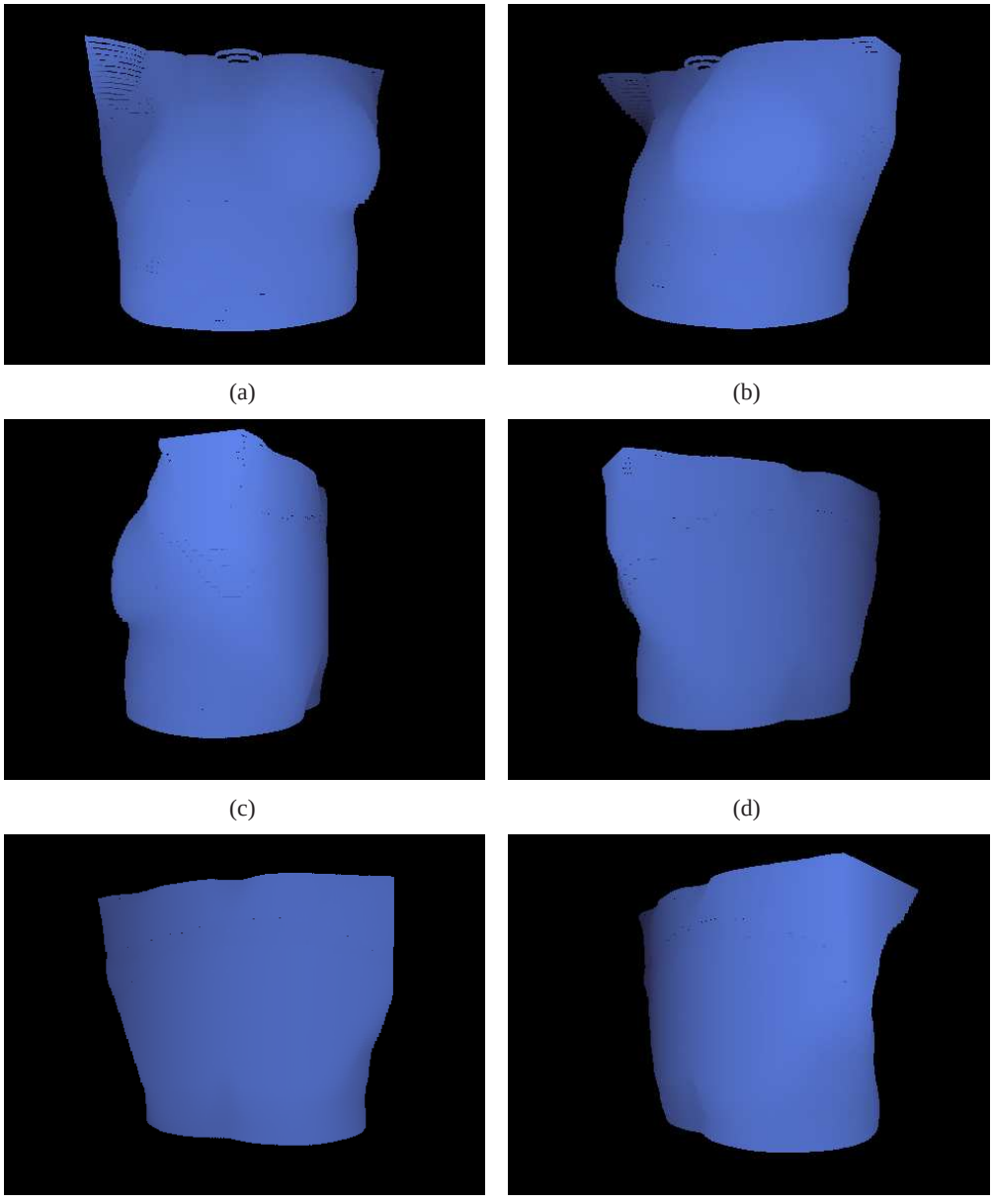


Figure 5.10: 3D representation of 2D segmentation of lungs from different views



(a) (b) (c) (d) (e) (f)

Figure 5.11: 3D representation of 2D segmentation of torso from different views

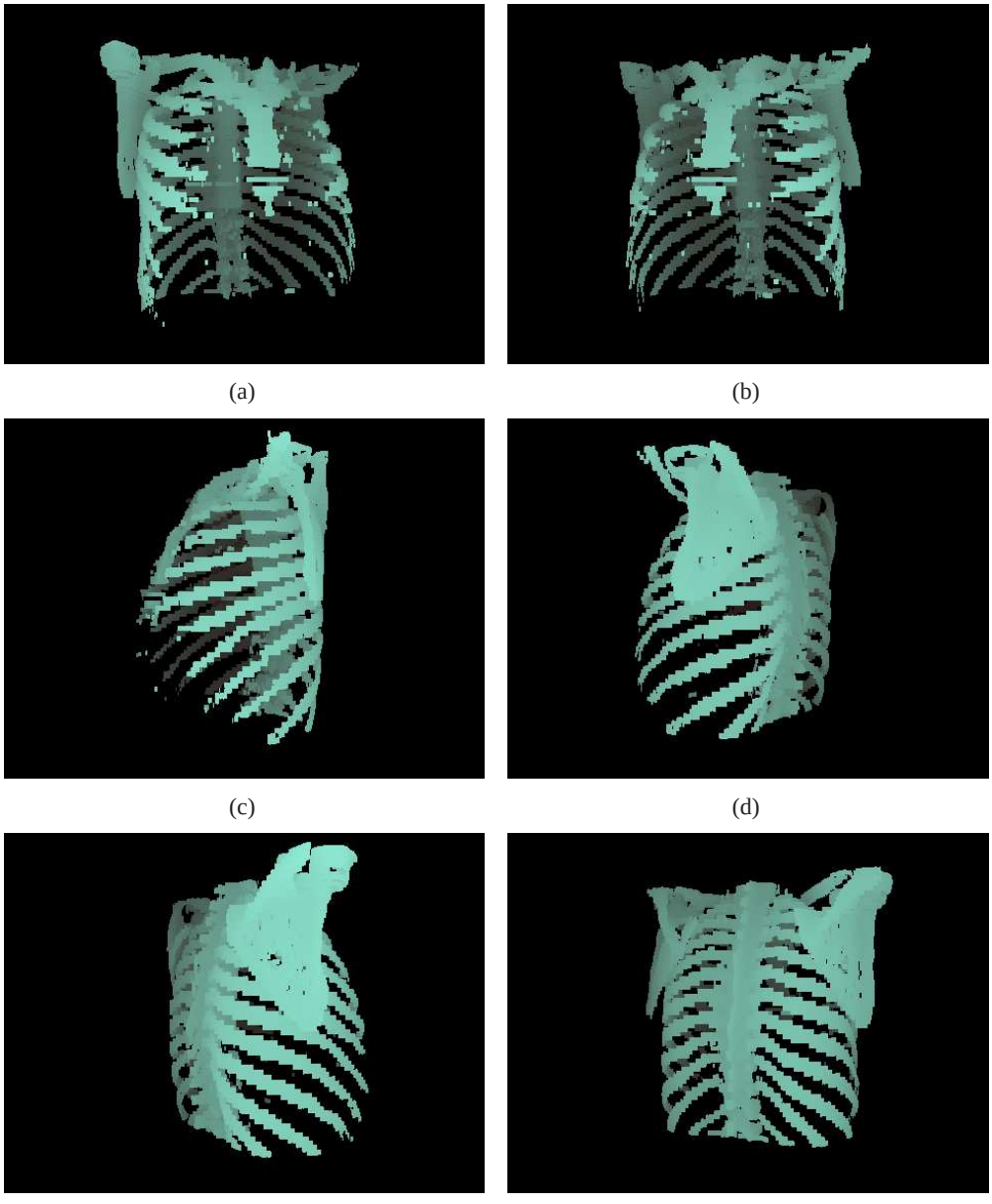


Figure 5.12: 3D representation of 2D segmentation of bones from different views

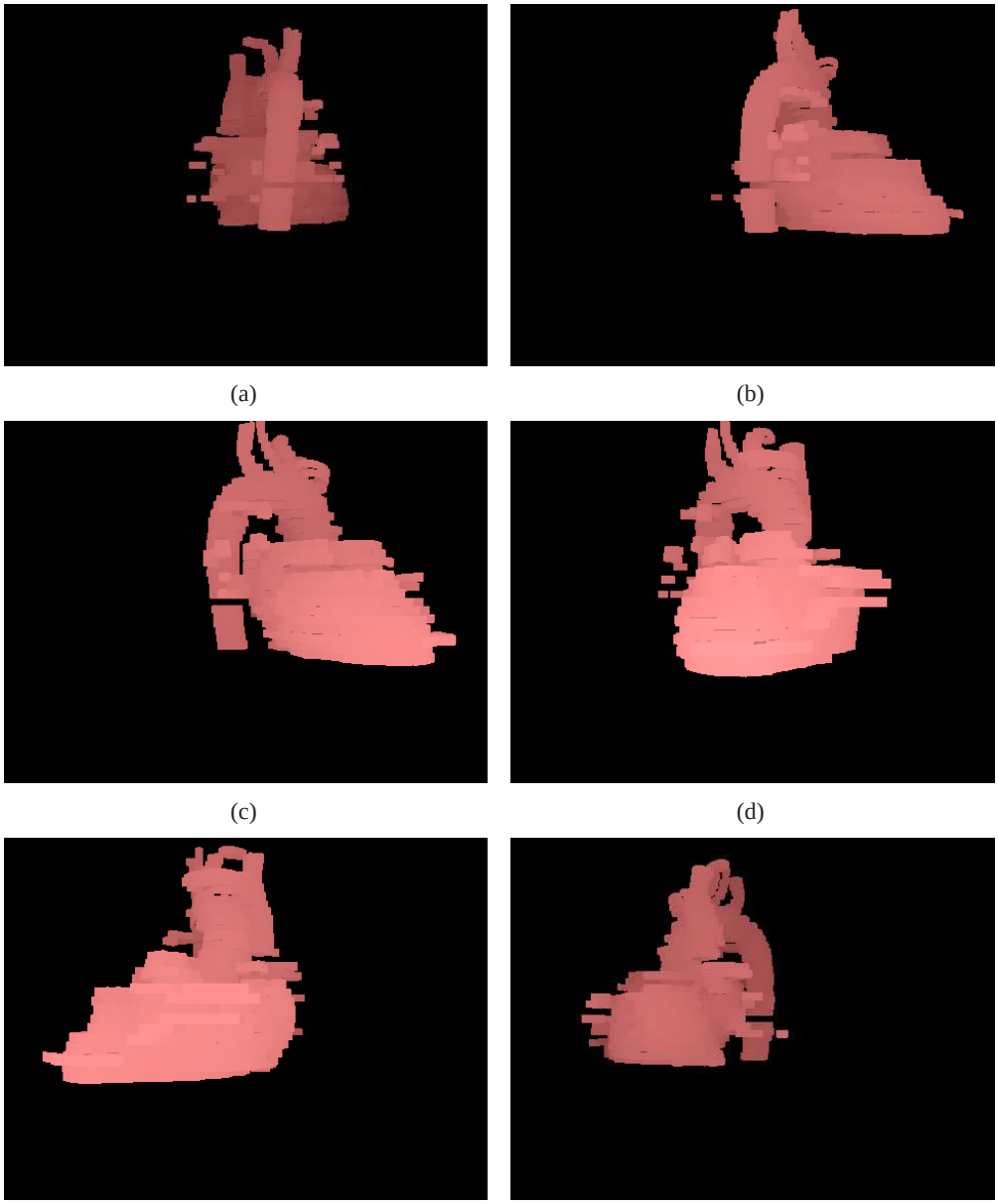


Figure 5.13: 3D representation of 2D segmentation of heart from different views

5.3.1 Quantitative Evaluation of Segmentation Results

In this section, we compare the semi-automatically segmented images based-on watershed transform with "Ground Truth" resulted from the manual segmentation. Negative Rate Metric (NRM), Overlap Index (OI) and Similarity Index (SI) are used like Chapters 3 and 4.

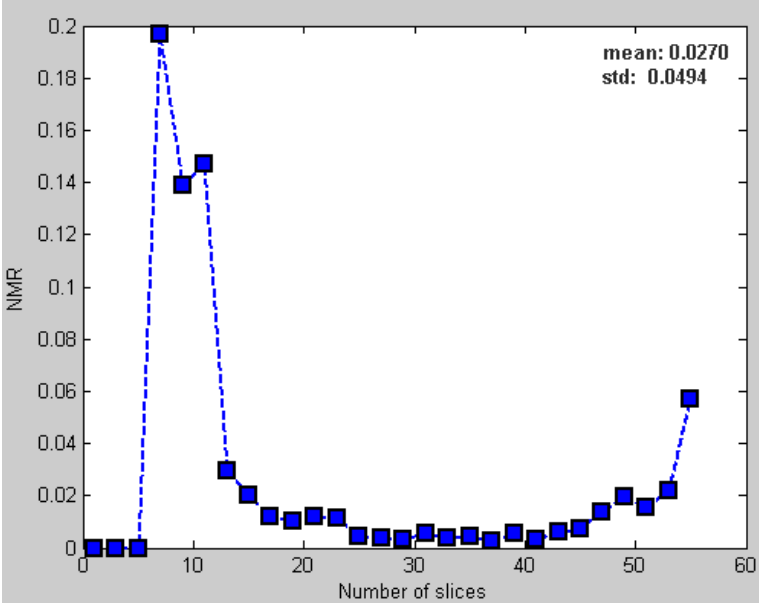


Figure 5.14: Negative Rate Metric results based on comparison between the automatically and manually segmented images for lungs.

In Figures 5.14, 5.15 and 5.16, negative rate metrics, missed, extra segmented, and overlapped rates and similarity indices are given based on comparison between the automatically and manually segmented images for lungs. These figures have consistent findings; all metrics imply similar conclusions on the performance of the watershed segmentation algorithm. The slices between 7 and 11 have the worst segmentation results among all lung segmentation results. Panel (a) of Figure 5.17 shows the image from slice 7 along with its segmentation results. Here, the colors red, green, and blue represent missed, extra segmented and overlapped regions, respectively. The numerical values of the evaluation metrics for these images are also given on the segmented images. In this slice, since the upper part of the patient's right lung has been subject to an interstitial lung disease. For this reason, they are segmented as torso using the

proposed framework. In slice 55, shown in panel (b) of Figure 5.17, the relatively bad results are obtained for lung because of the high proportion of perimeter pixels to actual lung area. Because of this, faulty segmented pixels are most likely to take place in the boundary regions of the structures. As for the remaining slices, we achieve good lung segmentation results; for example, in some slices such as slice 29, similarity index for lung reaches 0.99. Figure 5.18 shows lung segmentation results of two slices (29 and 41) with good values of evaluation metrics.

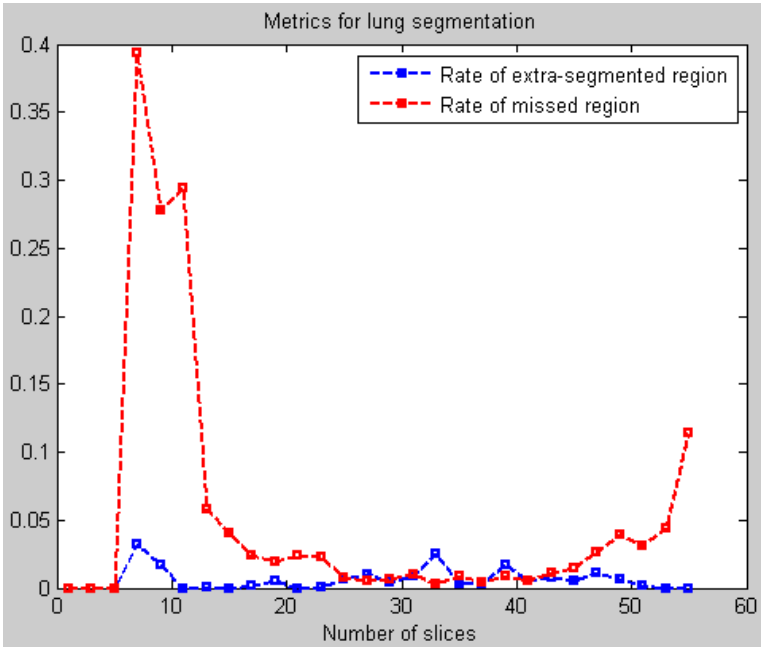


Figure 5.15: Missed and extra segmented rates based on comparison between the automatically and manually segmented images for lungs.

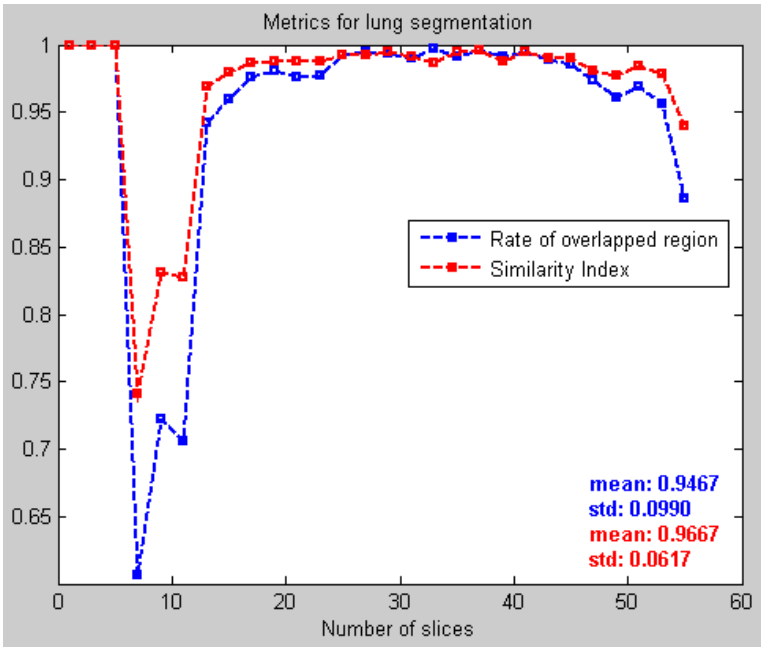
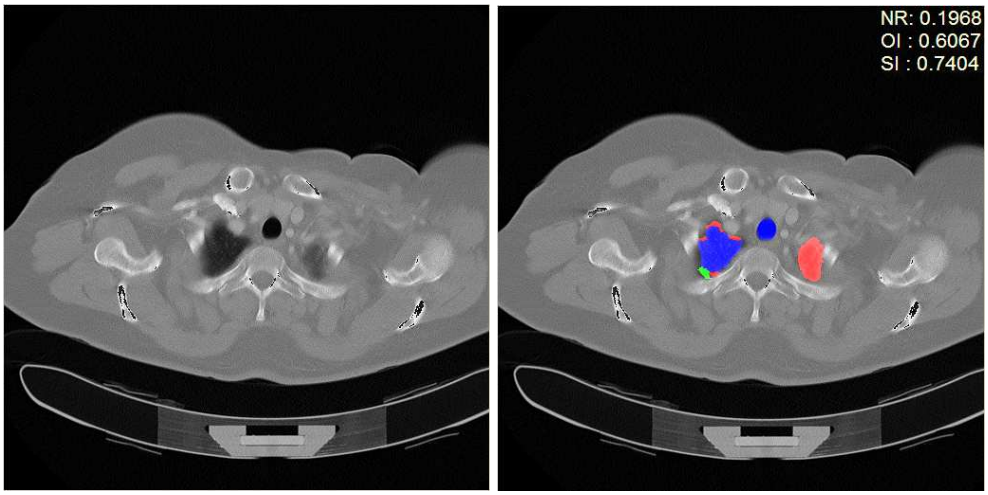
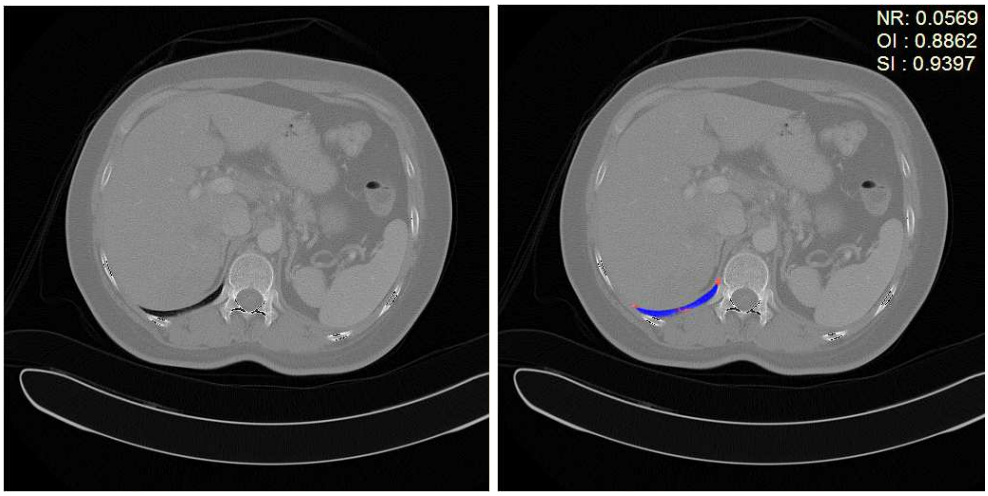


Figure 5.16: Overlapped rate and similarity index based on comparison between the automatically and manually segmented images for lungs.

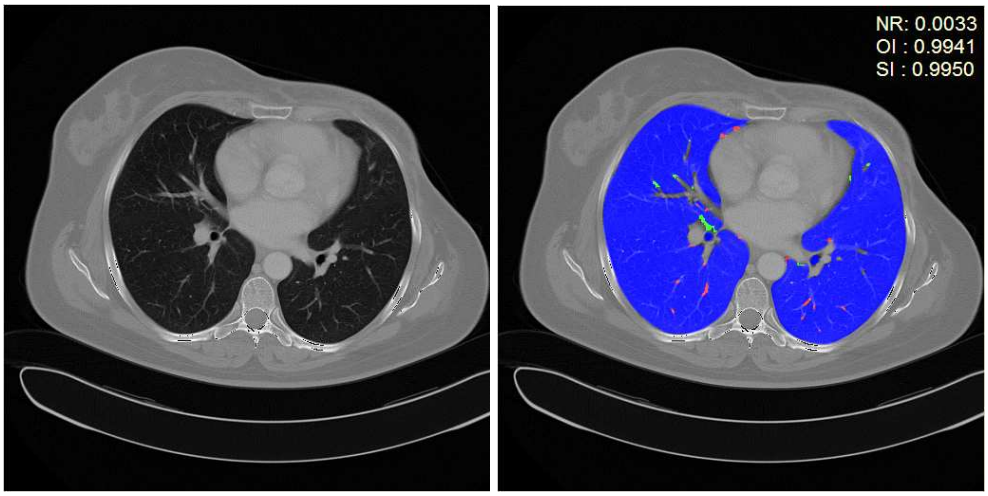


(a)

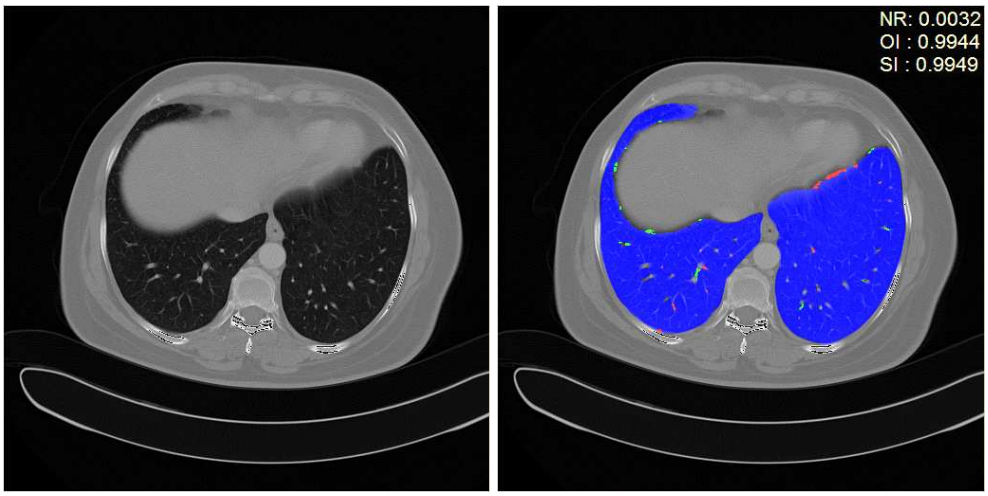


(b)

Figure 5.17: Two examples of bad lung segmentation results (right); the slices (a) 07 and (b) 55. The original images (left) are also shown.



(a)



(b)

Figure 5.18: Two examples of good lung segmentation results (right); the slices (a) 29 and (b) 41. The original images (left) are also shown.

Quantitative results for thoraxial segmentation are given in Figures 5.19, 5.20 and 5.21. In these figures, negative rate metrics, missed, extra segmented, and overlapped rates and similarity indices are depicted, respectively, based on comparison between the automatically and manually segmented images.

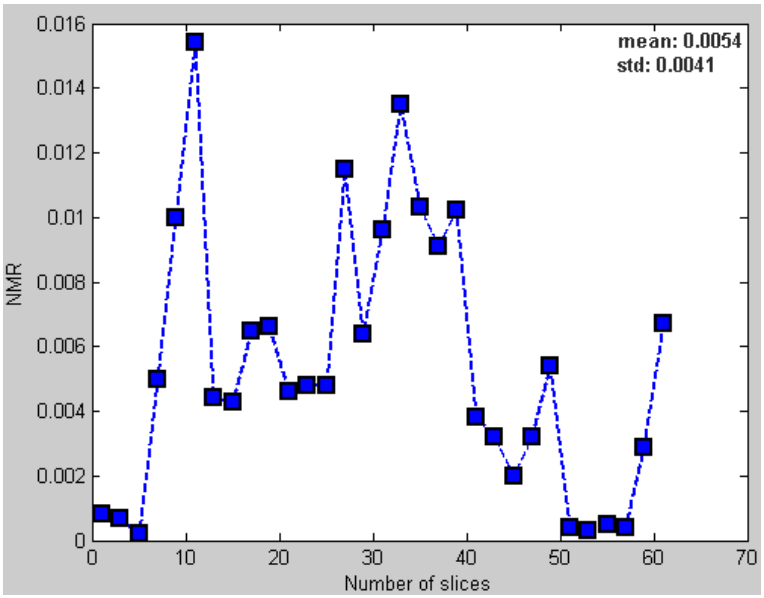


Figure 5.19: Negative Rate Metric results based on comparison between the automatically and manually segmented images for torso.

In the thoraxial segmentation, we define bones and heart as the thoraxial tissues. On the contrary, we exclude lungs from the thoraxial tissues. For this reason, the faulty segmented regions in lungs cause the segmentation performance for torso to decline. As shown in Figure 5.20, the rate of extra segmented regions peaks in the slices between 7 and 11 since the upper part of the patient's right lung has been subject to an interstitial lung disease. This also brings about the decrease in the similarity index.

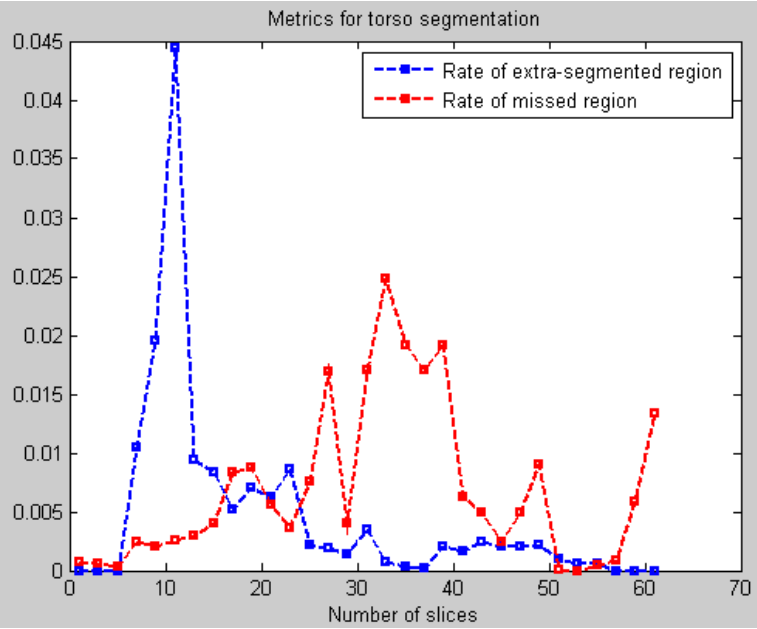


Figure 5.20: Missed and extra segmented rates based on comparison between the automatically and manually segmented images for torso.

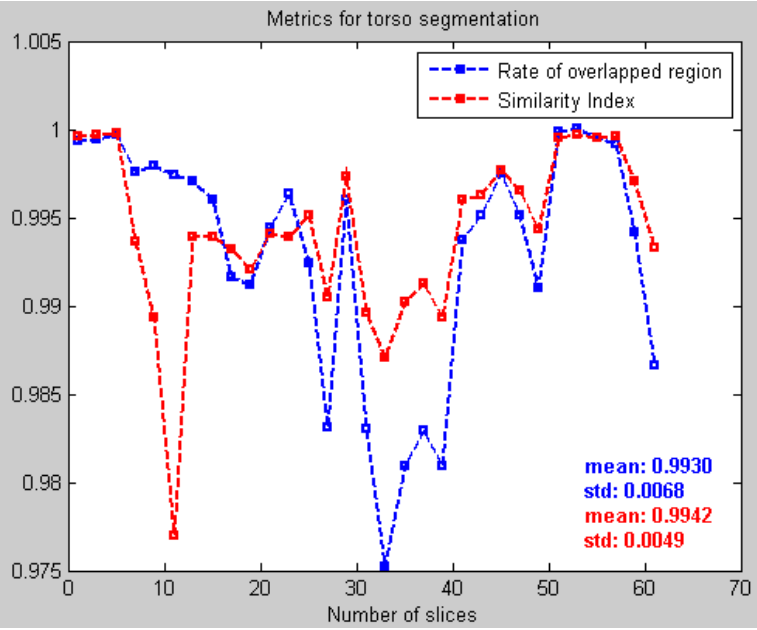


Figure 5.21: Overlapped rate and similarity index based on comparison between the automatically and manually segmented images for torso.

Quantitative results for bone segmentation are given in Figures 5.22, 5.23 and 5.24. In these figures, negative rate metrics, missed, extra segmented, and overlapped rates and similarity indices are depicted, respectively, based on comparison between the automatically and manually segmented images. Two examples with poor evaluation metric values examples came from slices 13 and 51 (see Figure 5.25) and the ones with good results are from slices 19 and 33 (see Figure 5.26) according to both the similarity index and negative rate metric. In these figures also, the colors red, green, and blue represent missed, extra segmented and overlapped regions, respectively.

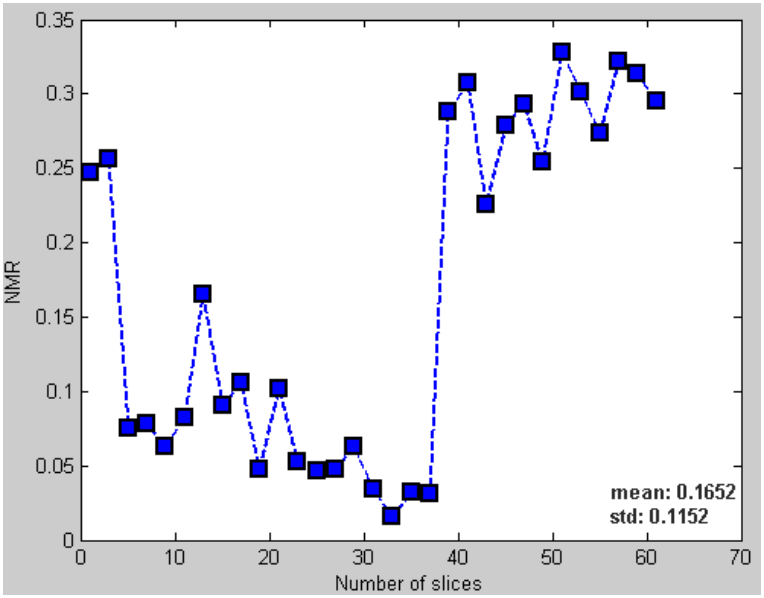


Figure 5.22: Negative Rate Metric results based on comparison between the automatically and manually segmented images for bones.

As shown in the Figures 5.25(a) and 5.25(b), this algorithm, unlike active contours, does not segment the pixels belonging to liver and stomach as bones. However, it misses the pixels belonging to the spongy bones and the cartilages. Around these slices, the overall segmentation quality tends to decrease (see the mean value of SI (Figure 5.24)). Nevertheless, similarity index for bones reaches to 0.80 in some slices, as seen in the Figure 5.26.

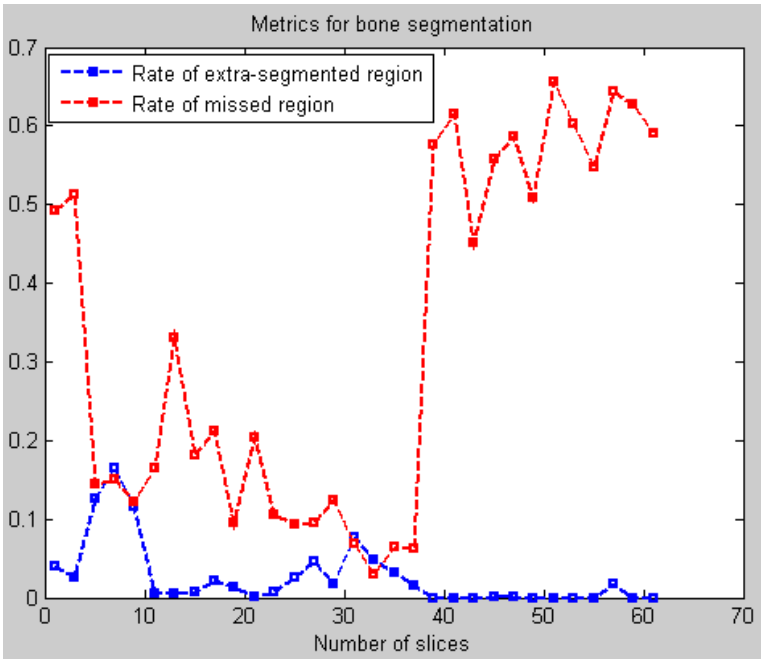


Figure 5.23: Missed and extra segmented rates based on comparison between the automatically and manually segmented images for bones.

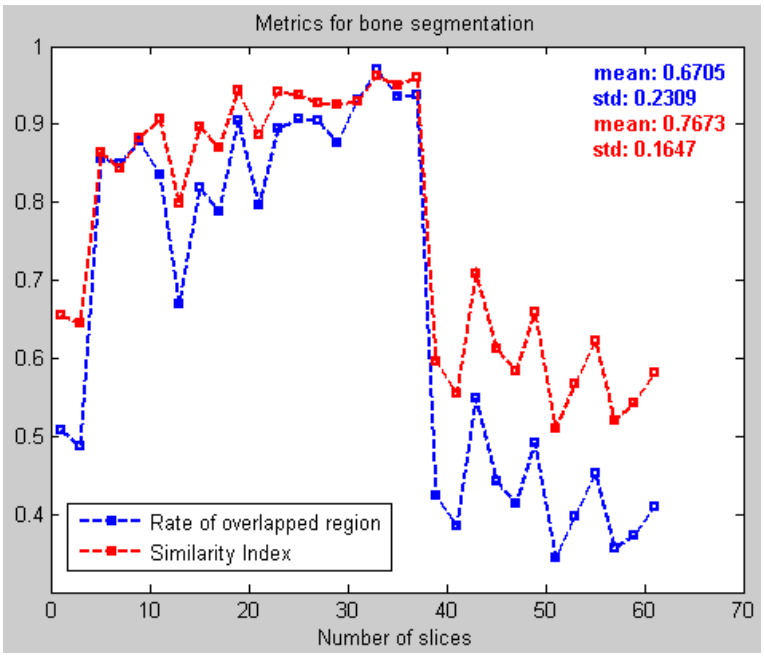
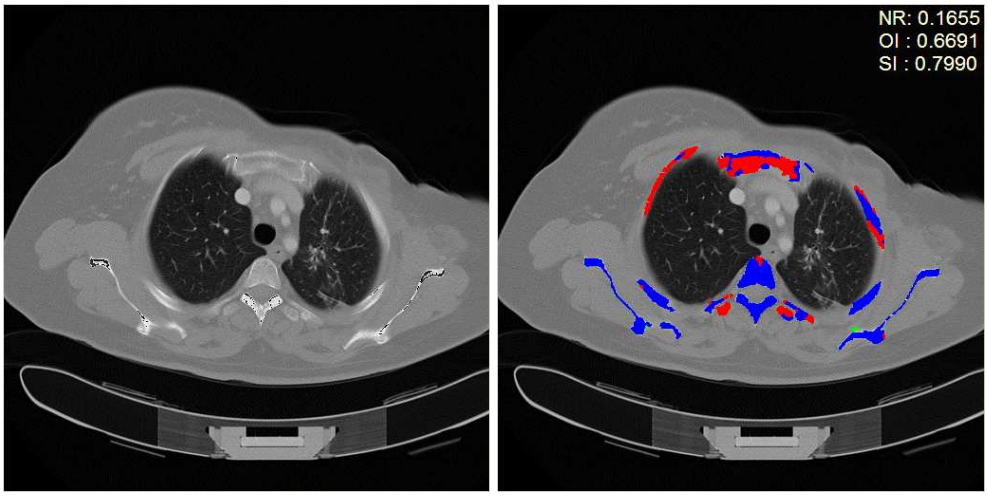
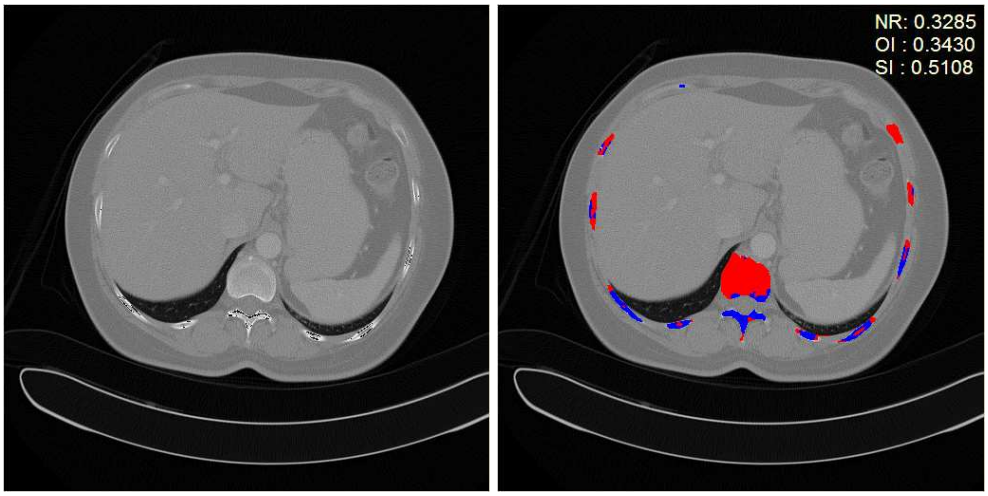


Figure 5.24: Overlapped rate and similarity index based on comparison between the automatically and manually segmented images for bones.

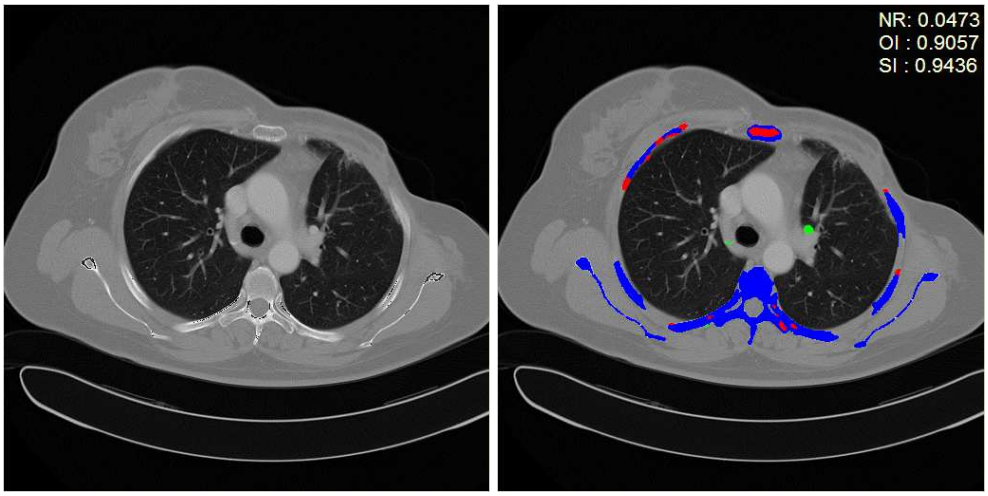


(a)

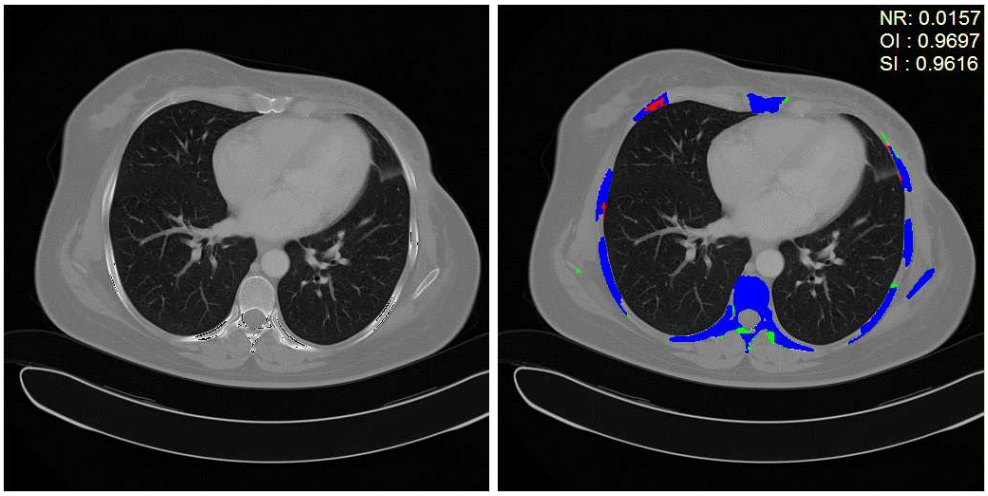


(b)

Figure 5.25: Two examples of bad bone segmentation results (right); the slices (a) 13 and (b) 51. The original images (left) are also shown.



(a)



(b)

Figure 5.26: Two examples of good bone segmentation results (right); the slices (a) 19 and (b) 33. The original images (left) are also shown.

In Figures 5.28 and 5.29, missed, extra segmented, and overlapped rates and similarity indices are depicted based on comparison between the automatically and manually segmented images for heart. According to both the similarity index and negative rate metric, bad results are observed in slices 21 and 39 (see 5.30) and the good ones are from the slices 19 and 37 (see 5.31). In these figures, the color coding is the same as in previous results.

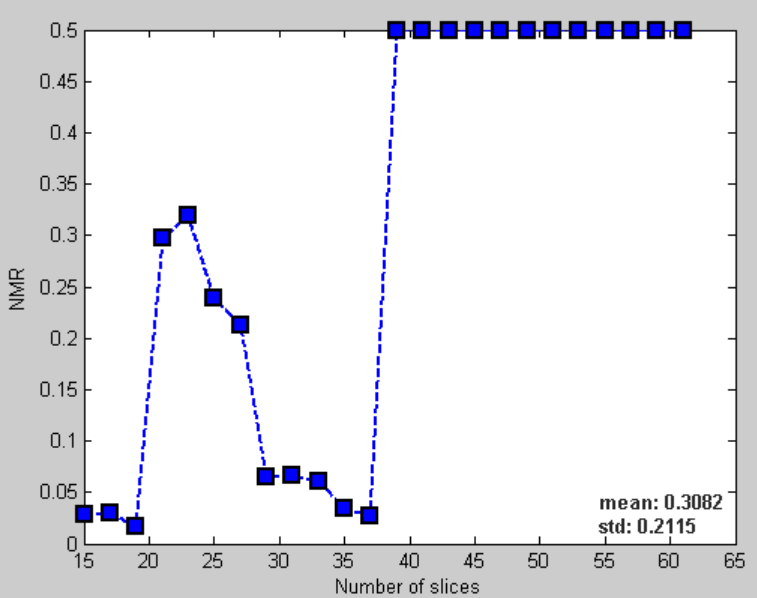


Figure 5.27: Negative Rate Metric results based on comparison between the automatically and manually segmented images for heart.

Because of the usage of intravenous contrast material, our algorithm misses the pixels belonging to muscle (*i.e.*, the red regions) as seen in Figure 5.30(a). On the contrary, it catches the pulmonary vessels (*i.e.*, the green regions) due to their high intensities. In slice 39 shown in Figure 5.30(b), our algorithm does not produce good results because of the uncertain boundaries of the heart. Nonetheless, there are some slices such as 19 and 37, where similarity index rises over 0.97 as seen in Figure 4.30. Like the framework based on thresholding, the algorithm was not able to segment the descending thoracic aorta.

For the segmentation of all tisues or organs in upper torso, the mean values and the standard deviations of NRM and SI are given in Table 5.2. These results indicate that the best results are obtained for lung segmentation because of the highest mean value

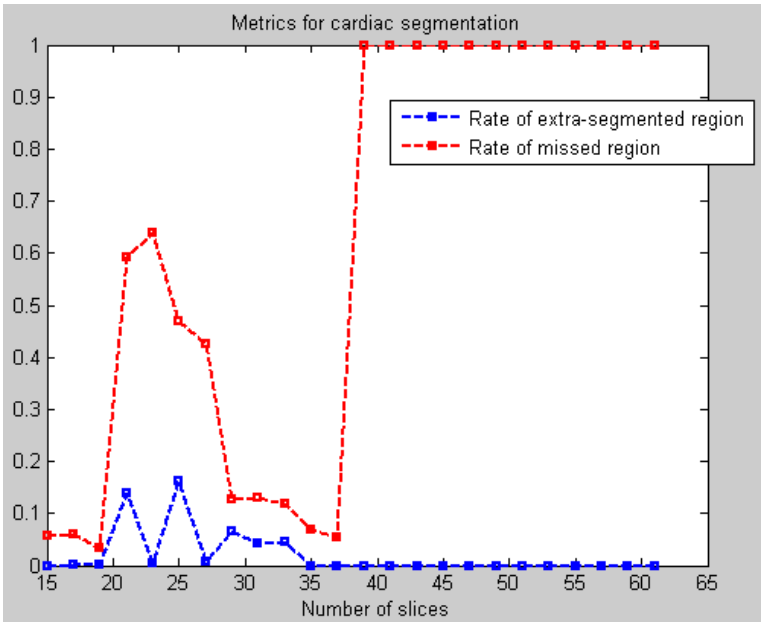


Figure 5.28: Missed and extra segmented rates based on comparison between the automatically and manually segmented images for heart.

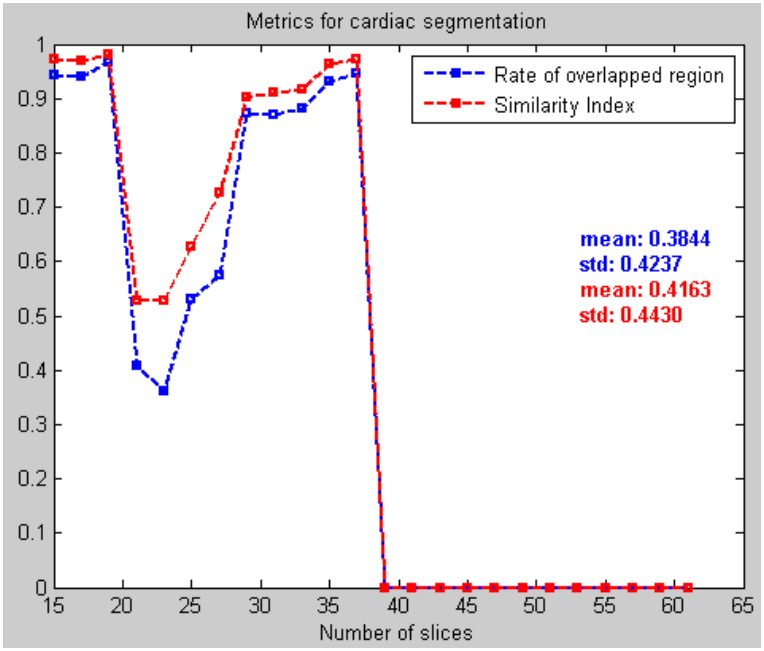
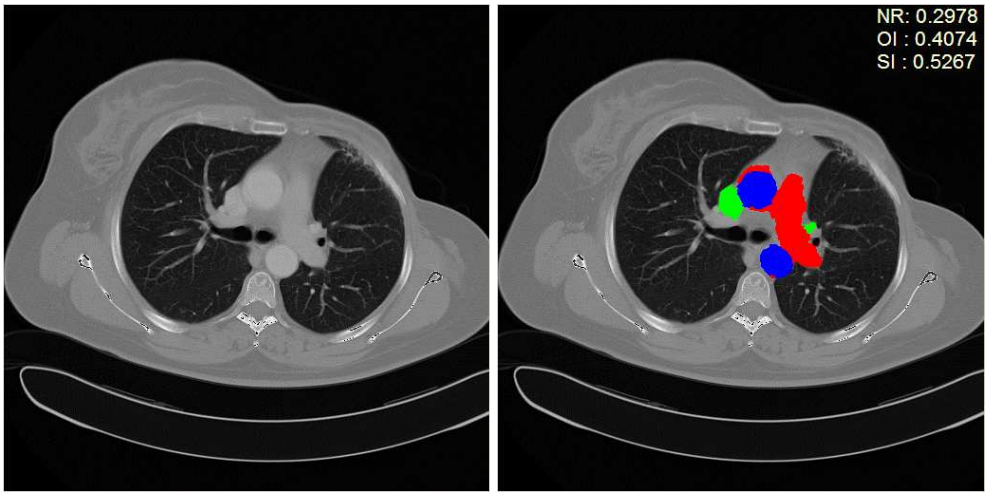
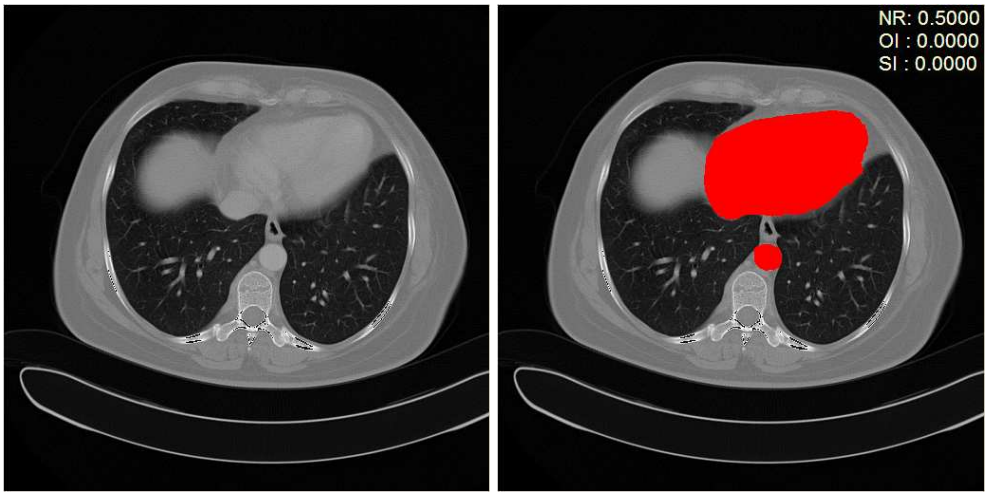


Figure 5.29: Overlapped rate and similarity index based on comparison between the automatically and manually segmented images for heart.

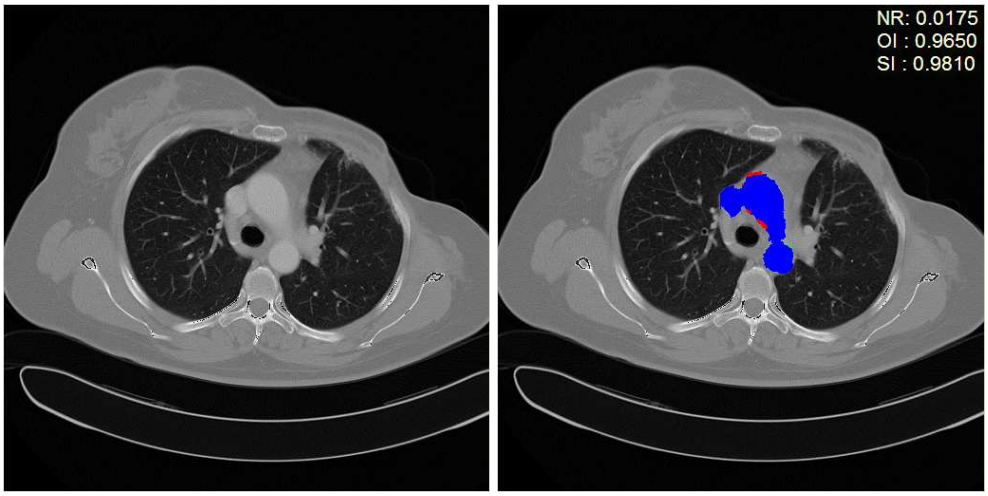


(a)

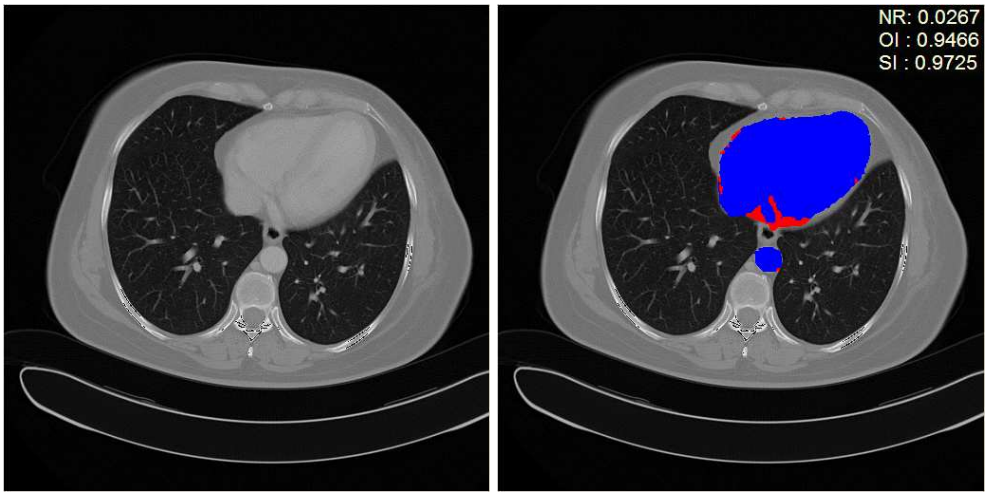


(b)

Figure 5.30: Two examples of bad cardiac segmentation results (right); the slices (a)21 and (b) 39. The original images (left) are also shown.



(a)



(b)

Figure 5.31: Two examples of good cardiac segmentation results (right); the slices (a)19 and (b) 37. The original images (left) are also shown.

and lowest standard deviation of SI. On the contrary, the worst results are obtained for cardiac segmentation due to the lowest mean value and the highest standard deviation of SI.

Table 5.2: The mean values and the standard deviations of NRM and SI for segmentation of lungs, heart and bones using thresholding

	NRM Mean/Std	**Overlapping Mean/Std**	**Similartity Index Mean/Std**
For lung	0.0270/0.0494	0.9467/0.0990	0.9667/0.0617
For bones	0.1652/0.1152	0.6705/0.2309	0.7673/0.1647
For heart	0.3082/0.2115	0.3844/0.4237	0.4163/0.4430

CHAPTER 6

DISCUSSIONS AND CONCLUSIONS

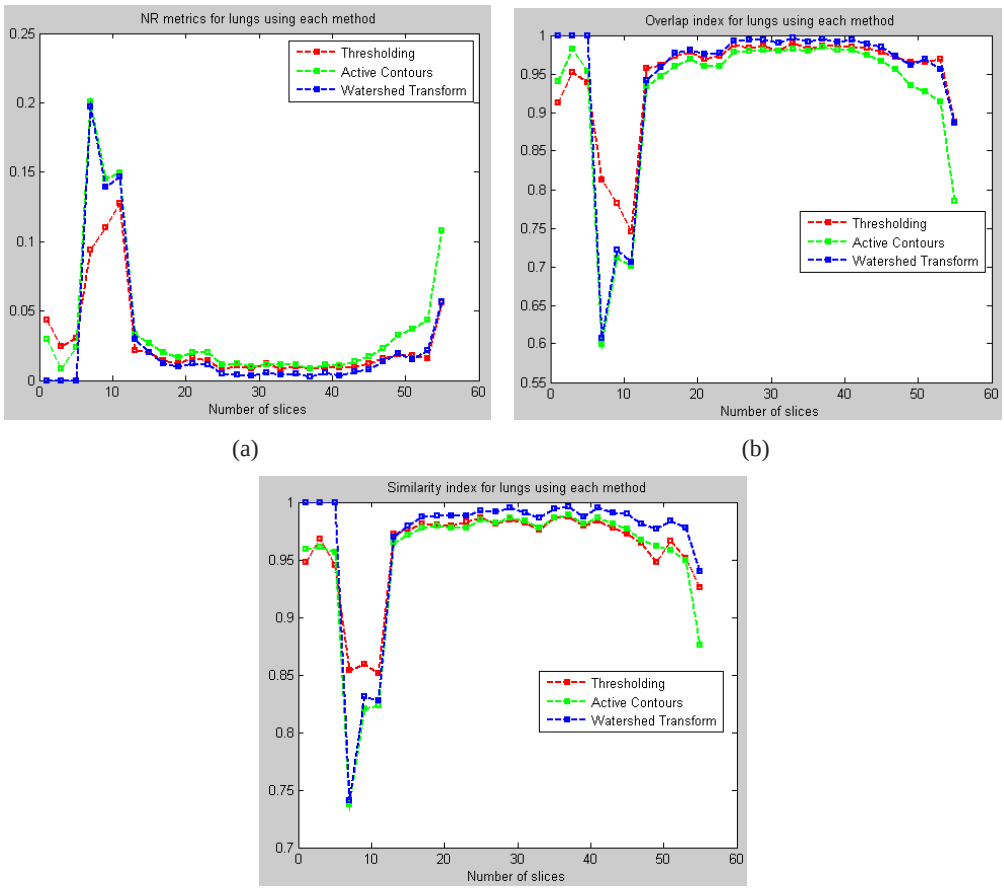
In this dissertation, three semi-automatic frameworks for the segmentation of main tissues and organs in the human upper torso were proposed and implemented on CT images. In the first framework, the main segmentation algorithm is thresholding; component labeling and morphological operations are used to prop up the theshold-ing framework. In the second framework, active contour model without edges (also known as Chan-Vese method) is applied with component labeling. In the third frame-work, watershed transform (as a segmentation algorithm) produces a large number of regions. This is an undesirable situation, known as oversegmentation. By using k-means clustering, these regions are merged with respect to their average values of pixel intensities. The performances of each of these three frameworks are appraised for each tissue in the upper torso by comparing the segmentation results quantitatively with manual segmentation results.

According to validation metrics for segmentation results produced by the three frame-works, none of the frameworks can give a complete and robust solution for segmen-tation of all the tissues or organs in upper torso. Each method has some advantages and disadvantages, and produces accurate results for segmentation of a specific organ or tissue.

Here, we make a comparison of the quantitative segmentation results for each tissue or organ in the upper torso with respect to methods used in this thesis.

Table 6.1: Numerical results for segmentation of lungs

	NRM Mean/Std	**Overlapping Mean/Std**	**Similartity Index Mean/Std**	**Elapsed Time**
Framework using thresholding	0.0270/0.0317	0.9479/0.0642	0.9590/0.0397	55 sec
Framework using active contours	0.0380/0.0494	0.9250/0.0992	0.9512/0.0610	187 sec
Framework using watershed transform	0.0270/0.0494	0.9467/0.0990	0.9667/0.0617	1280 sec



(a) (b)

(c)

Figure 6.1: The quantitative performance metrics for lungs using each method. (a) Negative rate, (b) Overlap Index, (c) Similarity Index

Lung Segmentation: According to the validation metrics and the elapsed time for lung segmentation, the framework based on watershed transform produces the best results but its elapsed time is too long. On the other hand, the framework based on thresholding produces the results very close to those of watershed transform and its elapsed time is about twenty times lower. Therefore, the most optimal solution to lung segmentation in the proposed methods is the framework based on thresholding.

As for the comparison between the results from this study and the previous work in the literature, the segmentation accuracy of each proposed scheme in this thesis is reasonable. Moreover, the time cost for thresholding is very low. As a previous work, Pu *et al.* state that the percentage of volume overlapping is 95.1 ± 2.0% when the differences in lung volume are considered for lung segmentation in the 43 CT examinations and the computational cost takes about 7 minutes for a CT examination using a desktop PC computer (AMD Athlon 64 × 2 Dual 2.11 GHz and 2 GB of RAM). Our proposed thresholding framework produce the segmentation results for lungs with the overlapping ratio approximately 95% in 0.5 min.

Table 6.2: Numerical results for segmentation of torso

	NRM Mean/Std	**Overlapping Mean/Std**	**Similartity Index Mean/Std**	**Elapsed Time**
Framework using active contours	0.0077/0.0045	0.9910/0.0072	0.9918/0.0054	187 sec
Framework using watershed transform	0.00054/0.0041	0.9930/0.0068	0.9942/0.0049	1260 sec

Thoraxial Segmentation: According to the performance metrics, like lung segmentation, the frameworks using active contours and watershed transform produce similar results for thoraxial segmentation. For this reason, the elapsed time plays a dominant role to choose the best method. According to Table 6.2, the framework using active contours is preferable due to its shorter duration.

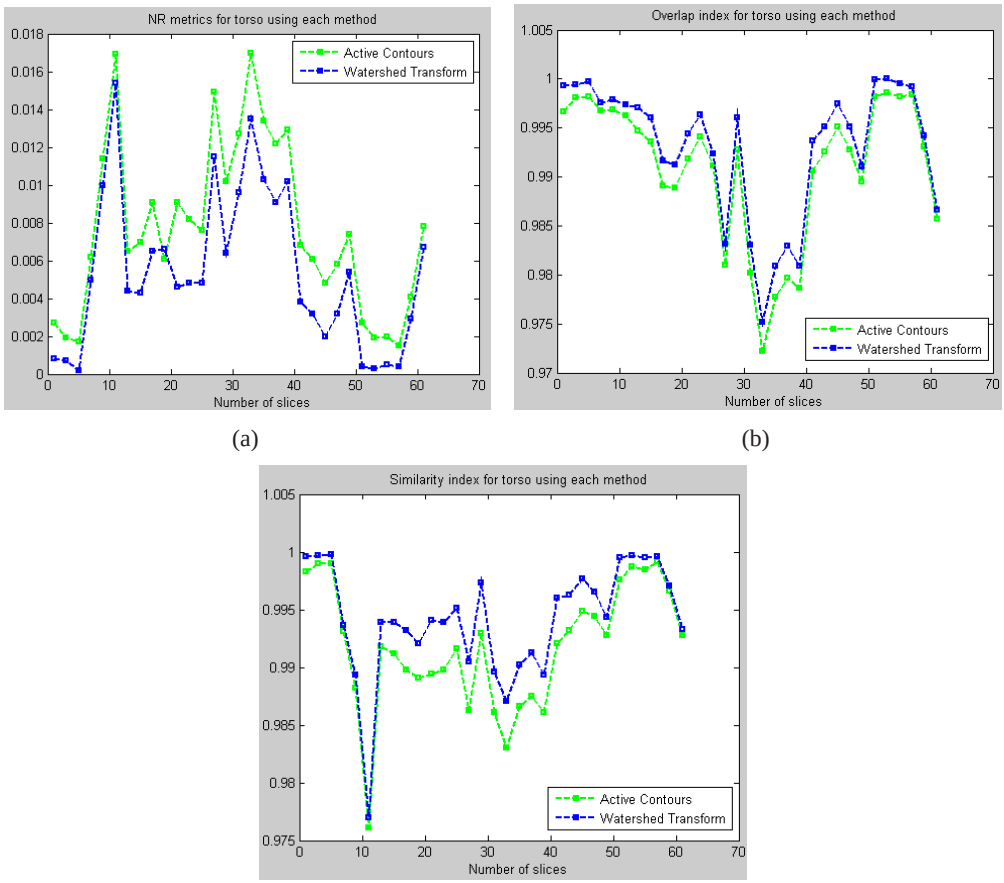
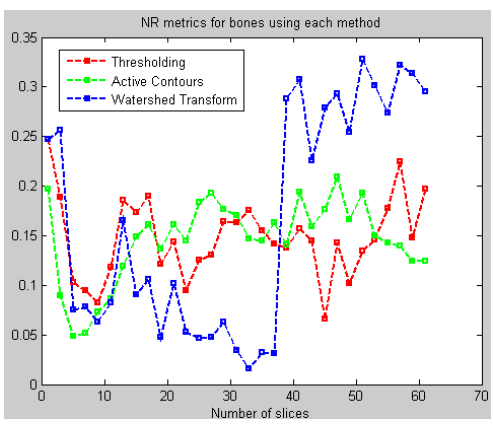


Figure 6.2: The quantitative performance metrics for torso using each method. (a) Negative rate, (b) Overlap Index, (c) Similarity Index

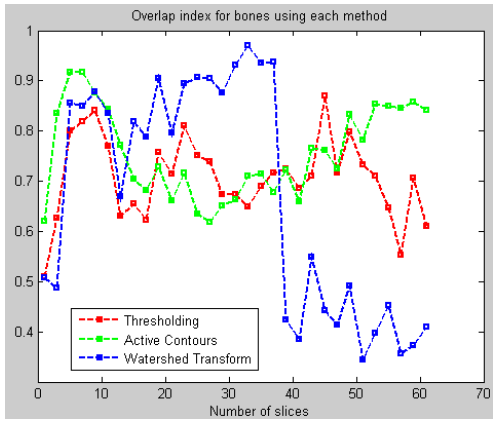
Bone Segmentation: According to the performance metrics shown in Figure 6.3, for the slices between 5 and 37, the framework using watershed transform provides good results but its performances for the number of slices larger than 37 gradually decline. For these slices, thresholdings gives the best results. As for overall performance, thresholding is the optimal method due to its duration and performance metrics given in Table 6.3.

Table 6.3: Numerical results for segmentation of bones

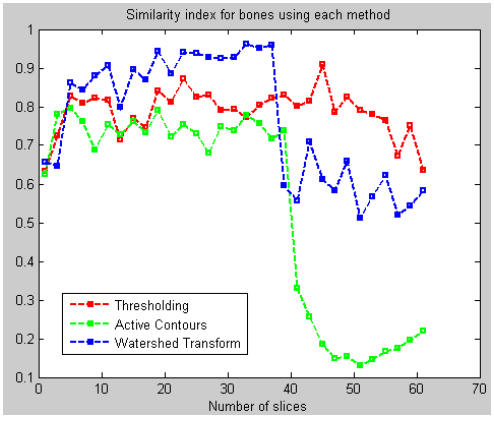
	NRM Mean/Std	**Overlapping Mean/Std**	**Similartity Index Mean/Std**	**Elapsed Time**
Framework using thresholding	0.1477/0.0407	0.7069/0.0817	0.7865/0.0614	40 sec
Framework using active contours	0.1457/0.0412	0.7561/0.0902	0.5446/0.2702	1659 sec
Framework using watershed transform	0.1652/0.1152	0.6705/0.2309	0.7673/0.1647	1281 sec



(a)



(b)

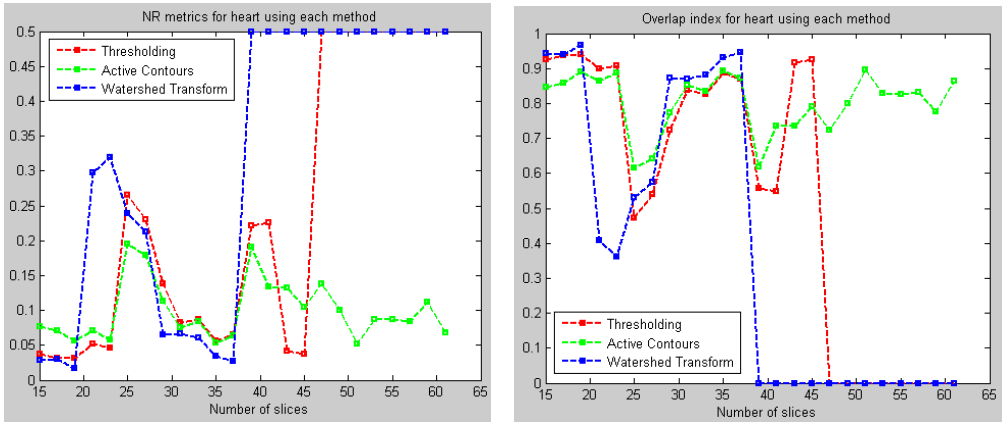


(c)

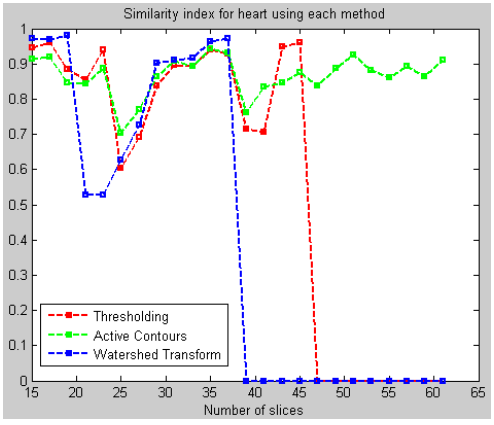
Figure 6.3: The quantitative performance metrics for bones using each method. (a) Negative rate, (b) Overlap Index, (c) Similarity Index

Table 6.4: Numerical results for segmentation of heart

	NRM Mean/Std	**Overlapping Mean/Std**	**Similartity Index Mean/Std**	**Elapsed Time**
Framework using thresholding	0.2354/0.2028	0.5299/0.4061	0.5713/0.4228	30 sec
Framework using active contours	0.0994/0.0428	0.8021/0.0854	0.8671/0.0569	1659 sec
Framework using watershed transform	0.3082/0.2115	0.3844/0.4237	0.4163/0.4430	1281 sec



(a) (b)



(c)

Figure 6.4: The quantitative performance metrics for heart using each method. (a) Negative rate, (b) Overlap Index, (c) Similarity Index

Cardiac Segmentation: According to the performance evaluation metrics shown in Figure 6.4, for the slices in the ranges 15-19 and 29-37, the framework using watershed transform provides good results but its overall performance is lower than the other methods. For the segmentation of the rest of the slices and for overall segmentation, active contour method produces relatively good results compared to the other methods as seen in Table 6.4.

As a conclusion, for each tissue or organ, we should use different methods. Moreover, the combination of these methods can be more efficient for even a single structure. For lung and torso segmentation, the thresholding provides sufficient segmentation results and its performance metrics are very good. On the other hand, for bone and cardiac segmentation, employing hybrid methods can be a solution to improve their segmentation performances. In addition, the local solutions instead of performing segmentation on the whole image, or employing full 3D segmentation techniques may lead to the enhancement of the segmentation results.

There are still unresolved problems and points to improve results in this thesis. In future work following studies can be performed for further improvements:

- The combination of different methods may be applied to obtain a complete effective and robust solution for cardiac and bone segmentation. Using the advantage of each method, we can improve the segmentation results.
- For the visualization of the 3D representation of the structures, surface meshing may be performed on the 2D segmentation results.
- 3D segmentation methods may improve the bone and cardiac segmentation results despite their high computational cost.

REFERENCES

[1] M. Harders and G. Szekely, "Enhancing human-computer interaction in medical segmentation," *Proceeding of IEEE*, vol. 91, no. 9, pp. 1430–1442, 2003.

[2] N. Otsu, "A threshold selection method from gray-level histogram," *IEEE Transactions on System Man Cybernetics*, vol. SMC-9, No:1, pp. 62–66, 1979.

[3] M. Cheriet, J. N. Said, and C. Y. Suen, "A recursive thresholding technique for image segmentation," *IEEE Transactions on Image Processing*, vol. 7, no. 6, pp. 918–921, 1998.

[4] T. F. Chan and L. A. Vese, "Active contours without edges," *IEEE Transactions on Image Processing*, vol. 10, no. 2, pp. 266–277, 2001.

[5] J. B. T. M. Roerdink and A. Meijster, "The watershed transform: Definitions, algorithms and parallelization strategies," in *Fundamenta Informaticae*, vol. 41, pp. 187–228, IOS Press, 2001.

[6] R. Jain, R. Kasturi, and B. G. Schunck, *Machine Vision*. McGraw-Hill Inc., 1995.

[7] T. K. Deyy, "Delaunay mesh generation of three dimensional domains." Research supported by NSF, USA (CCF-0430735 and CCF-0635008), 2008.

[8] J. D. Bronzino, *Biomedical Engineering Fundamentals (The Biomedical Engineering Handbook, Third Edition)*. CRC Press, 3 ed., April 2006.

[9] D. L. Pham, C. Xu, and J. L. Prince, "Current methods in medical image segmentation," *Annual Review of Biomedical Engineering*, vol. 2, pp. 315–338, 2000.

[10] A. F. Laine, "In the spotlight: Biomedical imaging," *IEEE Reviews in Biomedical Engineering*, vol. 1, pp. 4–7, 2008.

[11] R. Shahidi, R. Tombropoulos, and P. Grzeszczuk, "Clinical applications of three-dimensional rendering of medical data sets," *Proceeding of IEEE*, vol. 86, no. 3, pp. 555–568, 1998.

[12] J. Dey, T. Pan, D. J. Choi, D. Robotis, M. S. Smyczynski, P. H. Pretorius, and M. A. King, "Estimation of cardiac respiratory-motion by semi-automatic segmentation and registration of non-contrast-enhanced 4D-CT cardiac datasets," *IEEE Transactions on Nuclear Science*, vol. 56, no. 6, pp. 3662–3671, 2009.

[13] M. Ozkan, B. M. Dawant, and R. J. Maciunas, "Neural-network-based segmentation of multi-modal medical images: A comparative and prospective study," *IEEE Transactions on Medical Imaging*, vol. 12, no. 3, pp. 534–544, 1993.

[14] Z. Tu and X. Bai, "Auto-context and its application to high-level vision tasks and 3D brain image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. in press, pp. 1–14, 2010.

[15] X. Artaechevarria, A. Muñoz-Barrutia, and C. Ortiz-de Solórzano, "Combination strategies in multi-atlas image segmentation: Application to brain MR data," *IEEE Transactions on Medical Imaging*, vol. 28, no. 8, pp. 1266–1277, 2009.

[16] Z. Pan and J. Lu, "A bayes-based region-growing algorithm for medical image segmentation," *Computing in Science and Engineering, IEEE*, vol. 9, pp. 32–38, 2007.

[17] M. R. Sabuncu, B. T. T. Yeo, K. Van Leemput, B. Fischl, and P. Golland, "A generative model for image segmentation based on label fusion," *IEEE Transactions on Medical Imaging*, vol. in press, 2010.

[18] J. P. Thirion and G. Calmon, "Deformation analysis to detect and quantify active lesions in three-dimensional medical image sequences," *IEEE Transactions on Medical Imaging*, vol. 18, no. 5, pp. 429–441, 1999.

[19] A. Akselrod-Ballin, M. Galun, J. M. Gomori, M. Filippi, P. Valsasina, R. Basri, and A. Brandt, "Automatic segmentation and classification of multiple sclerosis in multichannel MRI," *IEEE Transactions on Biomedical Engineering*, vol. 56, no. 10, pp. 2461–2469, 2009.

[20] K. W. Tobin, E. Chaum, V. P. Govindasamy, and T. P. Karnowski, "Detection of anatomic structures in human retinal imagery," *IEEE Transactions on Medical Imaging*, vol. 26, no. 12, pp. 1729–1739, 2007.

[21] G. Ayala, T. León, and V. Zapater, "Different averages of a fuzzy setwith an application to vessel segmentation," *IEEE Transactions on Fuzzy Systems*, vol. 13, no. 3, pp. 384–393, 2005.

[22] M. Adel, A. Moussaoui, M. Rasigni, S. Bourennane, and L. Hamami, "Statistical-based tracking technique for linear structures detection: Application to vessel segmentation in medical images," *IEEE Signal Processing Letters*, vol. 17, no. 6, pp. 555–558, 2010.

[23] Y. Gao, R. Sandhu, G. Fichtinger, and A. Tannenbaum, "A coupled global registration and segmentation framework with application to magnetic resonance prostate imagery," *IEEE Transactions on Medical Imaging*, pp. 1–15, 2010.

[24] D. J. Michael and A. C. Nelson, "HANDX: a model-based system for automatic segmentation of bones from digital hand radiographs," *IEEE Transactions on Medical Imaging*, vol. 8, no. 1, pp. 64–69, 1989.

[25] R. Beichel, H. Bischof, F. Leberl, and M. Sonka, "Robust active appearance models and their application to medical image analysis," *IEEE Transactions on Medical Imaging*, vol. 24, no. 9, pp. 1151–1169, 2005.

[26] J. Carballido-Gamio, S. J. Belongie, and S. Majumdar, "Normalized cuts in 3-D for spinal MRI segmentation," *IEEE Transactions on Medical Imaging*, vol. 23, no. 1, pp. 36–44, 2004.

[27] T. Huysmans, J. Sijbers, and B. Verdonk, "Automatic construction of correspondences for tubular surfaces," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 4, pp. 636–651, 2010.

[28] T. Behrens, K. Rohr, and S. Stiehl, "Robust segmentation of tubular structures in 3-D medical images by parametric object detection and tracking," *IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics*, vol. 33, no. 4, pp. 554–561, 2003.

[29] K. Li, X. Wu, D. Z. Chen, and M. Sonka, "Optimal surface segmentation in volumetric images a graph theoretic approach," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 1, pp. 119–134, 2006.

[30] E. S. McCreedy, R. Cheng, P. F. Hemler, A. Viswanathan, B. J. Wood, and McAuliffe, "Radio frequency ablation registration, segmentation, and fusion tool," *IEEE Transactions on Information Technology in Biomedicine*, vol. 10, no. 3, pp. 490–496, 2006.

[31] S. A. Stansfield, "ANGY: a rule-based expert system for automatic segmentation of coronary vessels from digital subtracted angiograms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 2, pp. 188–199, 1986.

[32] S. E. Umbaugh, R. H. Moss, W. V. Stoecker, and G. A. Hance, "Automatic color segmentation algorithms with application to skin tumor feature identification," *IEEE Engineering in Medicine and Biology Magazine*, vol. 12, no. 3, pp. 75–82, 1993.

[33] G. A. Hance, S. E. Umbaugh, R. H. Moss, and W. V. Stoecker, "Unsupervised color image segmentation: With application to skin tumor borders," *IEEE Engineering in Medicine and Biology Magazine*, vol. 15, no. 1, pp. 104–111, 1996.

[34] A. Frangi, D. Rueckert, and J. Duncan, "New trends in three-dimensional cardiac image analysis," *IEEE Transactions on Medical Imaging*, vol. 21, no. 9, 2003.

[35] M. Sermesant, H. Delingette, and N. Ayache, "An electromechanical model of the heart for image analysis and simulation," *IEEE Transactions on Medical Imaging*, vol. 25, no. 5, pp. 612–625, 2006.

[36] M. Lynch, O. Ghita, and P. F. Whelan, "Segmentation of the left ventricle of the heart in 3-D+t mri data using an optimized nonrigid temporal model," *IEEE Transactions on Medical Imaging*, vol. 27, no. 2, pp. 195–203, 2008.

[37] J. Pu, D. S. Paik, X. Meng, J. E. Roos, and G. D. Rubin, "Shape break-and-repair strategy and its application to automated medical image segmentation," *IEEE Transactions on Visualization and Computer Graphics*, vol. 16, no. in press, pp. 1–10, 2010.

[38] S. G. Armato and W. F. Sensakovic, "Automated lung segmentation for thoracic CT," *Acad. Radiol.*, vol. 11, no. 9, pp. 1011–1021, 2004.

[39] I. Sluimer, A. Schilham, M. Prokop, and B. van Ginneken, "Computer analysis of computed tomography scans of the lung: A survey," *IEEE Transactions on Medical Imaging*, vol. 25, no. 4, pp. 385–405, 2006.

[40] D. Bartz, D. Mayer, J. Fischer, S. Ley, A. del Rio, S. Thust, C. Heussel, H. Kauczor, and W. Straßer, "Hybrid segmentation and exploration of the human lungs," *Proc. IEEE Visualization Conf.*, pp. 177–184, 2003.

[41] R. Garnavi, A. Baraani-Dastjerdi, H. A. Moghaddam, M. Giti, and A. A. Rad, "A new segmentation of lung HRCT images," *Proc. Digital Image Computing on Techniques and Applications*, pp. 52–59, 2005.

[42] H. Kim, H. Yoon, K. N. Trung, and G. S. Lee, "Automatic lung segmentation in CT images using anisotropic diffusion and morphology operation," *Proc. Seventh IEEE International Conference on Computer and Information Technology*, pp. 557–561, 2007.

[43] A. M. Ali and A. A. Farag, *Automatic Lung Segmentation of Volumetric Low-Dose CT Scans Using Graph Cuts*, vol. 5358, pp. 258–267. Springer, 2008.

[44] M. Prasad, M. Brown, F. Ahmad, S. andAbtin, J. Allen, I. da Costa, H. Kim, M. McNitt-Gray, J. Goldin, and S. Warfield, "Automatic segmentation of lung parenchyma in the presence of diseases based on curvature of ribs," *Proc. Seventh IEEE International Conference on Computer and Information Technology*, vol. 15, no. 9, pp. 1173–1780, 2008.

[45] A. A. Farag, A. S. El-Baz, and G. Gimelfarb, "Precise segmentation of multimodal images," *IEEE Transactions on Image Processing*, vol. 15, no. 4, pp. 952–968, 2006.

[46] W. Park, E. Hoffman, and M. Sonka, "Segmentation of intrathoracic airway trees: A fuzzy logic approach," *IEEE Transactions on Medical Imaging*, vol. 17, no. 4, pp. 489–497, 1998.

[47] G. D. Rubin, J. Lyo, D. S. Paik, A. Sherbondy, L. Chow, A. N. Leung, R. Mindelzun, S. E. Zinck, D. P. Naidich, and S. Napel, "Pulmonary nodules in MDCT scans: Impact of computer-aided detection," *Radiology*, vol. 235, pp. 274–283, 2005.

[48] J. Wang, M. Betke, and J. P. Ko, "Pulmonary fissure segmentation on CT," *Medical Image Analysis*, vol. 10, pp. 530–547, 2006.

[49] J. Wang, F. Li, and Q. Li, "Automated segmentation of lungs with severe interstitial lung disease in CT," *Medical Physics*, vol. 36, no. 10, pp. 4952–4599, 2009.

[50] Y. Masutani, "RBF-Based representation of volumetric data: Application in visualization and segmentation," *Proc. Medical Image Computing and Computer-Assisted Intervention (MICCAI 02)*, pp. 300–307, 2002.

[51] S. Lakare, "3D segmentation techniques for medical volumes." Research Proficiency Exam in Center for Visual Computing Department of Computer Science State University of New York at Stony Brook, 2000.

[52] P. K. Saha and J. K. Udupa, "Optimum image thresholding via class uncertainty and region homogeneity," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 7, pp. 689–706, 2001.

[53] S. Manay and A. Yezzi, "Anti-geometric diffusion for adaptive thresholding and fast segmentation," *IEEE Transactions on Image Processing*, vol. 12, no. 11, pp. 1310–1323, 2003.

[54] R. Goldenberg, R. Kimmel, E. Rivlin, and M. Rudzsky, "Cortex segmentation: A fast variational geometric approach," *IEEE Transactions on Medical Imaging*, vol. 21, no. 2, pp. 1544–1551, 2002.

[55] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu, "An efficient k-means clustering algorithm: Analysis and implementation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 881–892, 2002.

[56] L. Yingli, J. Tianzi, and Z. Yufeng, "A split-merge based region growing method for the analysis of fMRI data," *Human Brain Mapping*, vol. 22, no. 4, pp. 271–279, 2004.

[57] P. L. Bazin and D. L. Pham, "Homeomorphic brain image segmentation with topological and statistical atlases," *Medical Image Analysis*, vol. 12, pp. 616–625, 2008.

[58] Y. Zhou and J. Bai, "Multiple abdominal organ segmentation: An atlas-based fuzzy connectedness approach," *IEEE Transactions on Information Technology in Biomedicine*, vol. 11, no. 3, pp. 348–352, 2007.

[59] V. Grau, A. U. J. Mewes, M. Alcañiz, R. Kikinis, and S. K. Warfield, "Improved watershed transform for medical image segmentation using prior information," *IEEE Transactions on Medical Imaging*, vol. 23, no. 4, pp. 447–458, 2004.

[60] M. W. Hansen and W. E. Higgins, "Relaxation methods for supervised image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 9, pp. 949–962, 1997.

[61] L. Zheng, J. Jesse, and T. Hugues, "Unseeded region growing for 3D image segmentation," *Proc. ACM Pan-Sydney Workshop Visualization, ACM Press*, vol. 2, pp. 31–37, 2000.

[62] M. Sezgin and B. Sankur, "Survey over image thresholding techniques and quantitative performance evaluation," *Journal of Electronic Imaging*, vol. 13, no. 1, pp. 146–168, 2004.

[63] P. K. Sahoo, S. Soltani, A. K. C. Wong, and Y. C. Chen, "A survey of thresholding techniques," *Computer Vision, Graphics, and Image Processing*, vol. 41, no. 2, pp. 233–260, 1988.

[64] S. U. Le, S. Y. Chung, and R. H. Park, "A comparative performance study of several global thresholding techniques for segmentation," *Graph. Models Image Process.*, vol. 52, pp. 171–190, 1990.

[65] C. A. Glasbey, "An analysis of histogram-based thresholding algorithms," *Graph. Models Image Process.*, vol. 55, pp. 532–537, 1993.

[66] H. Digabel and C. Lantuéjoul, "Iterative algorithms," *In Actes du Second Symposium Européen d'Analyse Quantitative des Microstructures en Sciences des Matériaux, Biologie et Médecine, Caen*, pp. 85–99, 1978.

[67] S. Beucher and F. Meyer, "The morphological approach to segmentation: The watershed transform," *in Mathematical Morphology in Image Processing*, vol. 12, pp. 433–481, 1993.

[68] J. Serra, *Image Analysis and Mathematical Morphology*. New York: Academic Press, 1982.

[69] J. L. Vincent, "Morphological grayscale reconstruction in image analysis: Applications and efficient algorithms," *IEEE Trans. Image Processing*, vol. 2, pp. 176–201, 1993.

[70] S. Beucher, "Watershed, hierarchical segmentation and waterfall algorithm," in *in Mathematical Morphology and Its Applications to Image Processing*, (Dordrecht, The Netherlands), Kluwer, 1994.

[71] P. Jackway, "Gradient watersheds in morphological scale-space," *IEEE Trans. Image Processing*, vol. 5, pp. 913–921, 1996.

[72] J. Weickert, "Fast segmentation methods based on partial differential equations and the watershed transform," *in Proc. DAGM Symp.*, pp. 93–100, 1998.

[73] J. Sijbers, P. Scheunders, M. Verhoye, A. Van der Linden, D. Van Dyck, and E. Raman, "Watershed-based segmentation of 3d mr data for volume quantization," *Magn. Reson. Imag.*, vol. 15, pp. 679–688, 1997.

[74] A. N. Moga and M. Gabbouj, "Parallel image component labeling with watershed transformation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 19, pp. 441–450, 1997.

[75] J. M. Gauch, "Image segmentation and analysis via multiscale gradient watershed hierarchies," *IEEE Trans. Image Processing*, vol. 8, pp. 69–79, 1999.

[76] E. Dam and M. Nielsen, "Non-linear diffusion for interactive multiscale watershed segmentation," in *Lecture Notes in Computer Science*, vol. 1935, (Berlin, Germany), pp. 216–225, in MICCAI 2000: Fourth International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer-Verlag, 2000.

[77] H. T. Nguyen, M. Worring, and R. V. D. Boomgaard, "Watersnakes: Energy-driven watershed segmentation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 25, pp. 330–342, 2003.

[78] R. M. Haralick and L. G. Shapiro, "Image segmentation techniques," *Computer Vision, Graphics, and Image Processing*, vol. 29, no. 1, pp. 100–132, 1985.

[79] I. N. Manousakas, P. E. Undrill, G. G. Cameron, and T. W. Redpath, "Split-and-merge segmentation of magnetic resonance medical images: performance evaluation and extension to three dimensions," *Comput. Biomed. Res.*, vol. 31, pp. 393–412, 1998.

[80] J. Mangin, V. Frouin, I. Bloch, J. Regis, and J. Lopez-Krahe, "From 3D magnetic resonance images to structural representations of the cortex topography using topology preserving deformations," *J. Math. Imag. Vis.*, vol. 5, pp. 297–318, 1995.

[81] J. K. Udupa and S. Samarasekera, "Fuzzy connectedness and object definition: Theory, algorithms and applications in image segmentation," *Graph. Mod. Im. Proc.*, vol. 58, no. 3, pp. 246–261, 1996.

[82] P. Gibbs, D. L. Buckley, S. J. Blackband, and A. Horsman, "Tumour volume detection from MR images by morphological segmentation," *Phys. Med. Biol.*, vol. 41, pp. 2437–2446, 1996.

[83] S. Pohlman, K. A. Powell, N. A. Obuchowski, W. A. Chilcote, and S. G. Broniatowski, "Quantitative classification of breast tumors in digitized mammograms," *Med. Phys.*, vol. 23, pp. 1337–1345, 1996.

[84] J. S. Suri, S. Kamaledin Setarehdan, and S. E. Singh, *Advanced Algorithmic Approaches to Medical Image Segmentation*. State-of-the-Art Applications in Cardiology, Neurology, Mammography and Pathology Series, London, U.K.: Springer-Verlag, 1 ed., 2002.

[85] G. Sapiro, *Geometric Partial Differential Equations and Image Analysis*, ch. 2. Cambridge, U.K.: Cambridge Univ. Press, 2001.

[86] S. Osher and R. Fedkiw, *Level Set Methods and Dynamic Implicit Surfaces*, ch. 3–4. New York, U.S.A: Springer-Verlag, 2003.

[87] C. Xu and J. L. Prince, "Snake, shapes, and gradient vector flow," *IEEE Transactions on Image Processing*, vol. 7, no. 3, pp. 359–369, 1998.

[88] C. Chesnaud, P. Refregier, and V. Boulet, "Statistical region snake-based segmentation adapted to different noise models," *IEEE Transactions on Pattern Analysis Machine Intelligence*, vol. 21, no. 11, pp. 1145–1157, 1999.

[89] C. Li, C. Xu, C. Gui, and M. D. Fox, "Level set evolution without re-initialization: A new variational formulation," *in Proc. IEEE Conference Computer Vision Pattern Recognition*, vol. 1, pp. 430–436, 2005.

[90] J. S. Suri, K. Liu, S. Singh, S. N. Laxminarayan, X. Zeng, and L. Reden, "Shape recovery algorithms using level sets in 2D/3D medical imagery: A state-of-the-art review," *IEEE Transactions on Inf. Technol. Biomed.*, vol. 6, no. 1, pp. 8–28, 2002.

[91] X. Han, C. Xu, and J. L. Prince, "A topology preserving level set method for geometric deformable models," *IEEE Transactions on Pattern Analysis Machine Intelligence*, vol. 25, no. 6, pp. 755–768, 2003.

[92] Z. Wang and B. C. Vemuri, "DTI segmentation using an information theoretic tensor dissimilarity measure," *IEEE Transactions on Medical Imaging*, vol. 24, no. 10, pp. 1267–1277, 2005.

[93] D. Mumford and J. Shah, "Optimal approximations by piecewise smooth functions and associated variational problems," *Communications on Pure and Applied Mathematics*, vol. 42, no. 5, pp. 577–685, 1989.

[94] T. F. Chan, B. Y. Sandberg, and L. A. Vese, "Active contours without edges for vector-valued images," *Journal of Visual Communication and Image Represention*, vol. 11, no. 2, pp. 130–141, 2000.

[95] D. Tschumperle and R. Deriche, "Diffusion PDEs on vector-valued images," *IEEE Signal Processing Magazine*, vol. 19, no. 5, pp. 16–25, 2002.

[96] H. Park, P. Bland, and C. Meyer, "Construction of an abdominal probabilistic atlas and its application in segmentation," *IEEE Transactions on Medical Imaging*, vol. 22, no. 4, pp. 483–492, 2003.

[97] I. Sluimer, M. Prokop, and B. van Ginneken, "Toward automated segmentation of the pathological lung in CT," *IEEE Transactions on Medical Imaging*, vol. 24, no. 8, pp. 1025–1038, 2005.

[98] T. Rohlfing, R. Brandt, R. Menzel, and C. R. Maurer, "Evaluation of atlas selection strategies for atlas-based image segmentation with application to confocal microscopy images of bee brains," *NeuroImage*, vol. 21, no. 4, pp. 1428–1442, 2004.

[99] A. Klein, B. Mensh, S. Ghosh, J. Tourville, and J. Hirsch, "Mindboggle: Automated brain labeling with multiple atlases," *BMC Med. Imag.*, vol. 5, no. 1, 2005.

[100] C. Svarer, K. Madsen, S. Hasselbach, L. Pinborg, S. Haugbol, V. Frojaer, S. Holm, O. B. Paulson, and G. Knudsen, "MR-based automatic delineation of interest in human brain PET images using probability maps," *NeuroImage*, vol. 24, no. 4, pp. 969–979, 2005.

[101] T. Rohlfing, R. Brandt, R. Menzel, D. B. Russakoff, and C. R. Maurer, *Quo vadis, atlas-based segmentation?* The Handbook of Medical Image Analysis-Volume III: Registration Models, New York: Kluwer Academic/Plenum, 2005.

[102] R. A. Heckemann, J. V. Hajnal, P. Aljabar, D. Rueckert, and A. Hammers, "Automatic anatomical brain MRI segmentation combining label propagation and decision fusion," *NeuroImage*, vol. 33, no. 1, pp. 115–126, 2006.

[103] P. Kochunov, J. L. Lancaster, R. W. P. Thompson, J. Mazziotta, J. Hardies, and P. Fox, "Regional spatial normalization: Toward an optimal target," *J. Comput. Assist. Tomogr.*, vol. 25, no. 5, pp. 805–816, 2001.

[104] T. Rohlfing and C. R. Maurer, "Multi-classifier framework for atlasbased image segmentation," *Pattern Recognit. Lett.*, vol. 26, no. 13, pp. 2070–2079, 2005.

[105] L. I. Kuncheva, *Combining Pattern Classifiers: Methods and Algorithms*. New York, USA: Wiley, 1 ed., 2004.

[106] S. K. Warfield, K. H. Zou, and W. M. Wells, "Simultaneous truth and performance level estimation (staple): An algorithm for the validation of image segmentation," *IEEE Transactions on Medical Imaging*, vol. 23, no. 7, pp. 903–921, 2004.

[107] T. Rohlfing, D. Russakoff, and C. R. Maurer, "Performance-based classifier combination in atlas-based image segmentation using expectation-maximization parameter estimation," *IEEE Transactions on Medical Imaging*, vol. 23, no. 8, pp. 983–994, 2004.

[108] T. Rohlfing and C. R. Maurer, "Shape-based averaging," *IEEE Transactions on Image Processing*, vol. 16, no. 1, pp. 153–161, 2007.

[109] S. Klein, U. van der Heide, I. Lips, M. van Vulpen, M. Staring, and J. Pluim, "Automatic segmentation of the prostate in 3-D MR images by atlas matching using localised mutual information," *Med. Phys.*, vol. 35, no. 4, pp. 1407–1417, 2008.

[110] D. Mattes, D. R. Haynor, H. Vesselle, T. K. Lewellyn, and W. Eubank, "Non-rigid multimodality image registration," in *in Proc. SPIE Med. Imag. 2001: Image Process.*, vol. 4322, pp. 1609–1620, M. Sonka and K. M. Hanson, Eds., 2001.

[111] M. Wu, C. Rosano, P. Lopez-Garcia, C. S. Carter, and H. J. Aizenstein, "Optimum template selection for atlas-based segmentation," *NeuroImage*, vol. 34, no. 4, pp. 1612–1618, 2007.

[112] D. Cascio, F. Fauci, R. Magro, G. Raso, R. Bellotti, F. De Carlo, S. Tangaro, G. De Nunzio, M. Quarta, G. Forni, A. Lauria, M. E. Fantacci, A. Retico, G. L. Masala, P. Oliva, S. Bagnasco, S. C. Cheran, and E. Lopez Torres, "Mammogram segmentation by contour searching and mass lesions classification-with neural network," *IEEE Transactions on Nuclear Science*, vol. 53, no. 5, pp. 2827–2833, 2006.

[113] D. C. Stanford and A. E. Raftery, "Approximate bayes factors for image segmentation: The pseudolikelihood information criterion (PLIC)," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 11, pp. 1517–1520, 2002.

[114] A. Huang, R. Abugharbieh, and R. Tam, "A novel rotationally-invariant region-based hidden markov model for efficient 3D image segmentation (july 2008)," *IEEE Transactions on Image Processing*, vol. in press, 2010.

[115] J. Carballido-Gamio, S. J. Belongie, and S. Majumdar, "Normalized cuts in 3-D for spinal mri segmentation," *IEEE Transactions on Medical Imaging*, vol. 23, no. 1, pp. 36–44, 2004.

[116] N. Shareef, D. L. Wang, and R. Yagel, "Segmentation of medical images using legion," *IEEE Transactions on Medical Imaging*, vol. 18, no. 1, pp. 74–91, 1999.

[117] J. K. Udupa and P. K. Saha, "Fuzzy connectedness and image segmentation," *Proceedings of The IEEE*, vol. 91, no. 10, pp. 1649–1669, 2003.

[118] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 888–905, 2000.

[119] J. K. Udupa, P. K. Saha, and R. A. Lotufo, "Relative fuzzy connectedness and object definition: Theory, algorithms, and applications in image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 11, pp. 1485–1500, 2002.

[120] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Transactions on Pattern Analysis Machine Intelligence*, vol. 12, no. 7, pp. 629–639, 1990.

[121] G. Gerig, R. Kübler, and F. A. Jolesz, "Nonlinear anisotropic filtering of MRI data," *IEEE Transactions on Medical Imaging*, vol. 11, no. 2, pp. 221–232, 1992.

[122] J. Weickert, "Applications of nonlinear diffusion in image processing and computer vision," *Acta Mathematica Universitatis Comenianae*, vol. 70, no. 1, pp. 33–50, 2001.

[123] S. Osher and J. A. Sethian, "Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulation," *Journal of Computational Physics*, vol. 79, no. 1, pp. 12–49, 1988.

[124] R. Malladi, J. A. Sethian, and B. C. Vemuri, "Shape modeling with front propagation: A level set approach," *IEEE Transactions on Pattern Analysis Machine Intelligence*, vol. 17, no. 2, pp. 158–175, 1995.

[125] O. F. Olsen and M. Nielsen, "Multi-scale gradient magnitude watershed segmentation," in *Lecture Notes in Computer Science*, vol. 1310, (Berlin, Germany), pp. 6–13, in ICIAP'97-9th Int. Conf. on Image Analysis and Processing, Springer-Verlag, 1997.

[126] L. Vincent and P. Soille, "Watersheds in digital spaces: an efficient algorithm based on immersion simulations," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 13, no. 6, pp. 583–598, 1991.

[127] F. Meyer, "Topographic distance and watershed lines," *Signal Processing*, vol. 38, pp. 113–125, 1994.

[128] J. W. H. Meijs, O. W. Weier, M. J. Peters, and A. van Oosterom, "On the numerical accuracy of the boundary element method," *IEEE Transactions on Biomedical Engineering*, vol. 36, no. 10, pp. 1038–1049, 1989.

[129] G. Aubert and P. Kornprobst, *Mathematical Problems in Image Processing*. New York, USA: Springer-Verlag, 2002.

APPENDICES

Appendix A

SEGMENTATION VALIDATION METRIC

A.1 Negative Rate Metric

Negative rate metric evaluates a false negative rate $\left(NR_{fn}\right)$ and false positive rate $\left(NR_{fp}\right)$ as shown in Equation A.1. This metric is based on a pixel-wise mismatches between the base image and the segmented image.

$$NR = \frac{1}{2}\left(NR_{fn} + NR_{fp}\right) \tag{A.1}$$

where

$$NR_{fn} = \frac{N_{fn}}{N_{tp} + N_{fn}}$$

$$NR_{fp} = \frac{N_{fp}}{N_{fp} + N_{tn}}$$

N_{fp} (false positive) : Number of incorrectly detected pixels

N_{fn} (false negative) : Number of undetected pixels that belong to object of interest

N_{tp} (true positive) : Number of correctly detected pixels that belong to object of interest

N_{tn} (false negative) : Number of correctly rejected pixels

For the negative rate metric, the lower the score the better the algorithm is at correctly segmenting the object of interest that matches the ideal segmentation mask in the base

image. In the ideal case, NR is equal to zero.

A.2 Quantitative evaluation based on calculating missed, extra segmented and overlapped regions

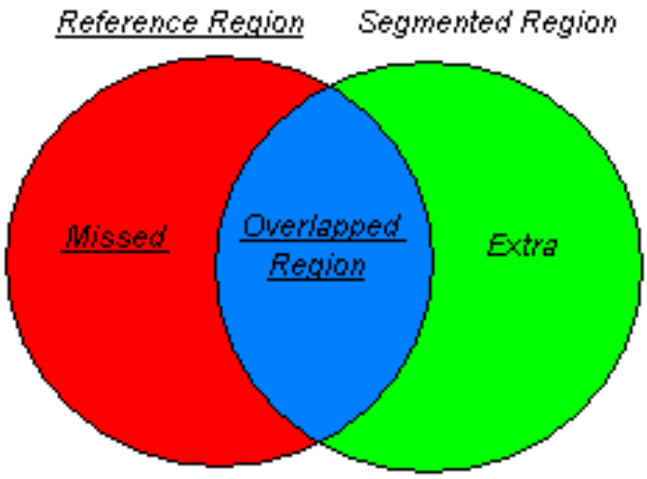


Figure A.1: Quantitative evaluation is performed by calculating missed, extra segmented and overlapped regions.

The measures used throughout the evaluation process are O (rate of overlapped region), E (rate of extra-segmented region), M (rate of missed region), Similarity Index (SI) [37] (see Figure A.1) and Relative Difference Measure [128]. Mathematical formulations of the measures are given below.

$$E \triangleq \frac{\text{Number of extra-segmented pixels}}{\text{Number of reference pixels}}$$

$$M \triangleq \frac{\text{Number of missed pixels}}{\text{Number of reference pixels}}$$

$$O \triangleq \frac{\text{Number of overlapped pixels}}{\text{Number of reference pixels}}$$

$$SI \triangleq \frac{\text{2 x Number of overlapped pixels}}{\text{Number of reference pixels + Number of segmented pixels}}$$

Appendix B

BACKGROUND: THE ACTIVE CONTOUR MODEL WITHOUT EDGES

B.1 The Chan-Vese Active Contour Model For Image Segmentation

Chan and Vese introduced a different approximation of the Mumford-Shah (MS) energy functional E_{MS} (equation B.1), which measures the degree of match between an image and its segmentation, so as to reveal boundaries of the object to be segmented without using the gradient of the image to stop the evolving curve on its boundaries and control points to interpolate the contour C [4]. Moreover, the selection of initial curve does not critical influence on the segmentation performance. This model is based on implicit surface evolution. Owing to this implicit representation, the topological changes such as splitting and merging can be detected automatically.

$$E_{MS}(u, C) = \alpha \int_{\Omega - C} |\nabla u|^2 \, \partial x \partial y + \beta \int_{\Omega} (u - I)^2 \, \partial x \partial y + \mu . \text{length}(C), \qquad \text{(B.1)}$$

where $\Omega \subset \Re^2$ is connected and bounded image domain. I is the orijinal image defined in the image domain Ω, $C \subset \Omega$ is the set of edge curves, u is a piecewise smooth approximation of the image I. In addition, α, β, and μ are the positive real constants which arrange the weights of the terms and settle the scale of the segmentation and smoothing.

The zero level set of this surface represents a moving curve and this curve captures boundaries or tracks interfaces in the image. The Chan-Vese method, which is a region-based segmentation method, is less sensitive to the initialization and more robust to noise than other active contour techniques such as snake. According to

this method, an image is separated into homogenous regions by using this level set function. These regions consist of the image parts with close mean values. An energy functional proposed by Chan and Vese [4] is defined as;

$$E_{CV}(C, c_1, c_0) = \mu.\text{length}(C) + \nu.\text{area}(\text{inside}(C)) + \lambda_1 \int_{inside(C)} |I - c_1|^2 \, \partial x \partial y + \lambda_2 \int_{outside(C)} |I - c_0|^2 \, \partial x \partial y, \tag{B.2}$$

where I is the given image, $C(s) : [0, 1] \rightarrow \Re^2$ is the curve, c_1 and c_0 are the real constants and they rely on C, μ and ν are fixed parameters and equal to or larger than zero, both λ_is are positive fixed parameters.

As for the relation between CV model and MS functional, the CV model is considered piecewise constant generalization of the minimal partion problem of the MS functional. It can be considered as the reduced form of the MS problem for $\nu = 0$, $\lambda_1 = \lambda_2 = \lambda$. In this particular situation, the boundary C is considered as a snake or an active contour and the approximation u of the image can be defined as

$$u = \left\{ \begin{array}{l} c_1, \ \text{inside}(C), \\ c_0, \ \text{outside}(C), \end{array} \right\}$$

where c_1 and c_0 are the average intensity values of the pixels inside C and outside C in the image, respectively

The CV model is based on the MS functional and both can be used for the image segmentation purpose. However, CV model and MS functional are slightly different. The main objective in the MS model is to generate an approximation of the initial image composed of homogenous regions. The Mumford-Shah model is appropriate to handle an image in which all boundaries are important and they are seperated by sharp edges. On the contrary, the aim in the CV model is to detect the boundaries of objects in an image. If one deals with an image in which specific objects are significant and we want to detect the boundaries of objects without sharp edges, one can use the Chan-Vese model [129].

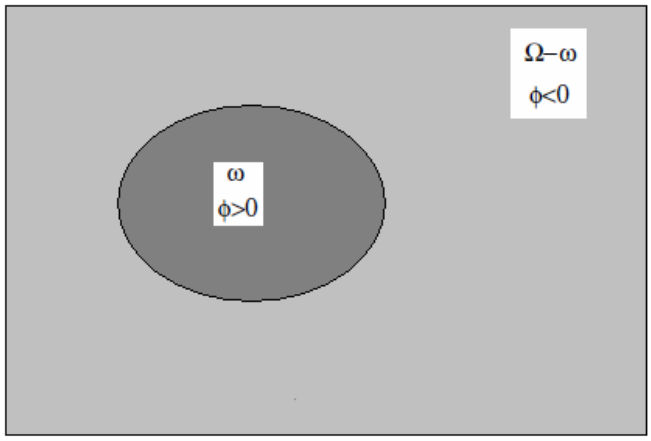


Figure B.1: A curve from the zero level set of the Lipschitz function ϕ is the boundary between the regions: $\{(x,y) : \phi(x,y) > 0\}$ and $\{(x,y) : \phi(x,y) < 0\}$.

The Level Set Formulation of the CV Model: The level set method is used to address the difficulty from the classic active contour models by minimizing the Chan-Vese energy functional. In the level set formulation, there is no explicit contour C and implicit representation is given by the zero level of a Lipschitz function ϕ instead of C. Therefore, the CV model is reformulated by substituting ϕ for C (equation B.3). This implicit curve physically overcomes the topological changes, such as splitting and merging. In addition, independent of the initial position, the curve automatically detects interior contours.

$$\begin{aligned} E_{CV}(C, c_1, c_0) &= \mu.\text{length}(\phi) + \nu.\text{area}(\text{inside}(\phi)) \\ &+ \lambda_1 \int_{\phi \geq 0} |I - c_1|^2 \, \partial x \partial y + \lambda_2 \int_{\phi < 0)} |I - c_0|^2 \, \partial x \partial y, \end{aligned} \tag{B.3}$$

The zero level set of scalar Lipschitz continuous functions is represented as $\phi(x,y) : \bar{\Omega} \rightarrow \Re$ where $\Omega \subset \Re^2$. The level set functions $\phi(x,y)$ satisfies the following conditions (see Figure B.1):

$$\left\{ \begin{array}{l} \text{inside}(C) = \omega = \{(x,y) \in \Omega : \phi(x,y) > 0\}, \\ \text{outside}(C) = \Omega / \bar{\omega} = \{(x,y) \in \Omega : \phi(x,y) < 0\}, \\ C = \partial\omega = \{(x,y) \in \Omega : \phi(x,y) = 0\}, \end{array} \right\}$$

Heaviside function H and the one dimensional Dirac measure δ are used to express the terms in the energy.

$$H(\phi) = \left\{ \begin{array}{l} 1 \text{ if } \phi \geq 0, \\ 0 \text{ if } \phi < 0, \end{array} \right\}$$

$$\delta(\phi) = \frac{\partial}{\partial \phi} H(\phi)$$

The terms related with length and area in the energy functional can be defined as:

$$\text{length}(\phi) = \int_{\Omega} \int |H(\phi)| \, \partial x \partial y = \int_{\Omega} \int \delta(\phi) |\nabla \phi| \, \partial x \partial y,$$

$$\text{area}(\phi) = \int_{\Omega} \int H(\phi) \, \partial x \partial y,$$

The remaining terms can be replaced as:

$$\int_{\phi \geq 0} \int |I - c_1|^2 \, \partial x \partial y = \int_{\Omega} \int |I - c_1|^2 H(\phi) \, \partial x \partial y,$$

$$\int_{\phi < 0} \int |I - c_0|^2 \, \partial x \partial y = \int_{\Omega} \int |I - c_0|^2 (1 - H(\phi)) \, \partial x \partial y,$$

Finally, the energy function can be written as:

$$\begin{aligned} E_{CV}(C, c_1, c_0) = \mu . \int_{\Omega} \int |H(\phi)| \, \partial x \partial y + \nu . \int_{\Omega} \int H(\phi) \, \partial x \partial y \\ + \lambda_1 \int_{\Omega} \int |I - c_1|^2 H(\phi) \, \partial x \partial y + \lambda_2 \int_{\Omega} \int |I - c_0|^2 (1 - H(\phi)) \, \partial x \partial y. \end{aligned}$$

The first term of the E_{CV} and also the solution of minimization problem cannot be computed because the Heaviside function H is not differentiable at 0. Therefore, one can use the regularized versions of the Heaviside function H and the dirac measure δ in order to obtain a global minimizer. There are two possible regualization of the Heaviside function H by $C^2\left(\overline{\Omega}\right)$ functions, as given in [4]. These are

$$H_{1,\epsilon}(\phi) = \left\{ \begin{array}{l} 1 \text{ if } \phi > \epsilon \\ 0 \text{ if } \phi < -\epsilon \\ \frac{1}{2}\left[1 + \frac{\phi}{\epsilon} + \frac{1}{\pi}\sin\left(\frac{\pi\phi}{\epsilon}\right)\right] \text{ if } |\phi| \leq \epsilon, \end{array} \right\}$$

and

$$H_{2,\epsilon}(\phi) = \frac{1}{2}\left(1 + \frac{2}{\pi}\arctan\left(\frac{\phi}{\epsilon}\right)\right).$$

In our experiments, we use the second regualization of the Heaviside function. As we know that $H'(\phi) = \delta$, we can compute the dirac measure δ as

$$\delta_\epsilon(\phi) = H'_\epsilon(\phi) = \frac{1}{\pi}\frac{\epsilon}{\epsilon^2 + \phi^2}.$$

When ϵ goes to zero, $H_\epsilon(\phi)$ and $\delta_\epsilon(\phi)$ converge to H and δ, respectively. By using the slightly regularized versions of the functions H and δ, the energy functional can be rewritten as

$$\begin{aligned} E_\epsilon(\phi, c_1, c_0) = \mu. \int_\Omega \int \delta_\epsilon(\phi) |\nabla\phi| \, \partial x \partial y + \nu. \int_\Omega \int H_\epsilon(\phi) \, \partial x \partial y \\ + \lambda_1 \int_\Omega \int |I - c_1|^2 H_\epsilon(\phi) \, \partial x \partial y + \lambda_2 \int_\Omega \int |I - c_0|^2 (1 - H_\epsilon(\phi)) \, \partial x \partial y. \end{aligned} \quad \text{(B.4)}$$

where H_ϵ and δ_ϵ are the regularized versions of H and δ and $|\nabla H_\epsilon(\phi)| = \delta_\epsilon(\phi)|\nabla\phi|$.

E_ϵ is minimized with respect to ϕ since c_0 and c_1 are fixed. Gateaux derivative of enegy functional in the direction ψ is used:

$$D_\phi E_\epsilon(\phi, c_1, c_0)[\psi] = lim_{t\to 0} \frac{E_\epsilon(\phi + t\psi, c_1, c_0) - E_\epsilon(\phi, c_1, c_0)}{t}. \quad \text{(B.5)}$$

Gateaux derivative of the term $\mu \int_\Omega \int \delta_\epsilon(\phi)|\nabla\phi| \, \partial x \partial y$ (see Appendix C) is

$$-\mu \int_\Omega \int \delta_\epsilon(\phi) \nabla \cdot \left(\frac{\nabla\phi}{|\nabla\phi|}\right) \psi \partial x \partial y \quad \text{(B.6)}$$

Gateaux derivative of the term $\nu \int_\Omega \int H_\epsilon(\phi)\, \partial x \partial y$ (see Appendix C) is

$$\nu \int_\Omega \int \delta_\epsilon(\phi)\, \psi \partial x \partial y \tag{B.7}$$

Gateaux derivative of the term $\lambda_1 \int_\Omega \int |I - c_1|^2 H(\phi)\, \partial x \partial y$ (see Appendix C) is

$$\lambda_1 \int_\Omega \int |I - c_1|^2 \delta_\epsilon(\phi)\, \psi \partial x \partial y \tag{B.8}$$

Gateaux derivative of the term $\lambda_2 \int_\Omega \int |I - c_0|^2 (1 - H(\phi))\, \partial x \partial y$ (see Appendix C) is

$$-\lambda_2 \int_\Omega \int |I - c_0|^2 \delta_\epsilon(\phi)\, \psi \partial x \partial y \tag{B.9}$$

Manipulating the derivatives mentioned by Equation B.6, B.7, B.8 and B.9, one can obtain the following

$$\begin{aligned} & D_\phi E_\epsilon(\phi, c_1, c_0)[\psi] \\ & \quad = lim_{t \to 0} \frac{E_\epsilon(\phi + t\psi, c_1, c_0) - E_\epsilon(\phi, c_1, c_0)}{t} \\ & \quad = -\mu \int_\Omega \int \delta_\epsilon(\phi) \nabla \cdot \left(\frac{\nabla \phi}{|\nabla \phi|} \right) \psi \partial x \partial y + \nu \int_\Omega \int \delta_\epsilon(\phi)\, \psi \partial x \partial y \\ & \quad + \int_\Omega \int \left(\lambda_1 |I - c_1|^2 - \lambda_2 |I - c_0|^2 \right) \delta_\epsilon(\phi)\, \psi \partial x \partial y \end{aligned} \tag{B.10}$$

Since $\frac{\partial \phi}{\partial n} = 0$,

$$\begin{aligned} & \int_\Omega \int \left(\mu \delta_\epsilon(\phi) \nabla \cdot \left(\frac{\nabla \phi}{|\nabla \phi|} \right) - \nu \delta_\epsilon(\phi) \right) \psi \partial x \partial y \\ & \quad + \int_\Omega \int \left(\lambda_1 |I - c_1|^2 - \lambda_2 |I - c_0|^2 \right) \delta_\epsilon(\phi)\, \psi \partial x \partial y = 0 \end{aligned} \tag{B.11}$$

By using the fundamental lemma of the calculus of variations, the following Euler Lagrange equation for ϕ is obtained.

$$\delta_\epsilon(\phi)\left(\mu\nabla\cdot\left(\frac{\nabla\phi}{|\nabla\phi|}\right)-\nu-\left(\lambda_1|I-c_1|^2-\lambda_2|I-c_0|^2\right)\right)=0 \tag{B.12}$$

$$\tfrac{\partial\phi}{\partial n}=0 \text{ on } \partial\Omega$$

Numerical Approximations: So as to find out the solution to the Euler-Lagrange equations, we apply the steepest descent algorithm in the distributed sense introducing an artificial time parameter t. The distributed iteration formula for the steepest descent algorithm is given as

$$\frac{\partial\phi}{\partial t}=\delta_\epsilon(\phi)\left(\mu\nabla\cdot\left(\frac{\nabla\phi}{|\nabla\phi|}\right)-\nu-\left(\lambda_1|I-c_1|^2-\lambda_2|I-c_0|^2\right)\right)=0$$

$$\phi(t,x,y)=\phi(0,x,y) \text{ in } \Omega$$

$$\tfrac{\delta_\epsilon(\phi)}{|\nabla\phi|}\tfrac{\partial\phi}{\partial n}=0 \text{ on } \partial\Omega$$

$$c_1=\frac{\int_\Omega\int IH_\epsilon(\phi)\,\partial x\partial y}{\int_\Omega\int H_\epsilon(\phi)\,\partial x\partial y}=average(I) \text{ in } \phi>0. \tag{B.13}$$

$$c_0=\frac{\int_\Omega\int I(1-H_\epsilon(\phi))\,dxdy}{\int_\Omega\int(1-H_\epsilon(\phi))\,\partial x\partial y}=average(I) \text{ in } \phi<0. \tag{B.14}$$

Method of finite differences is used to approximate the iteration formula. Let h be the space, time step Δt and $(x,y)=(ih,jh)$ be the grid points. Let $\phi^n=\phi(n\Delta t,ih,jh)$ be an approximation of $\phi(t,x,y)$ with $n\geq 0$. The finite differences are

$$\Delta_+^x\phi_{i,j}=\phi_{i+1,j}-\phi_{i,j}\ \Delta_-^x\phi_{i,j}=\phi_{i,j}-\phi_{i-1,j}$$

$$\Delta_+^y\phi_{i,j}=\phi_{i,j+1}-\phi_{i,j}\ \Delta_-^y\phi_{i,j}=\phi_{i,j}-\phi_{i,j-1}$$

Discretization of the Euler-Lagrange equation yields

$$\begin{aligned}\frac{\phi_{i,j}^{n+1}-\phi_{i,j}^{n}}{\Delta t} &= \delta_{\epsilon}\left(\phi_{i,j}^{n}\right)\left(\mu\chi\left(\phi_{i,j}^{n}\right)-\nu\right)\\ &\quad-\delta_{\epsilon}\left(\phi_{i,j}^{n}\right)\left(\lambda_1\left|I-c_1\right|^2-\lambda_2\left|I-c_0\right|^2\right)\end{aligned} \tag{B.15}$$

where

$$\begin{aligned}c_1 &= \frac{\sum IH_{\epsilon}\left(\phi_{i,j}^{n}\right)}{\sum H_{\epsilon}\left(\phi_{i,j}^{n}\right)}\\ c_0 &= \frac{\sum I\left(1-H_{\epsilon}\left(\phi_{i,j}^{n}\right)\right)}{\sum\left(1-H_{\epsilon}\left(\phi_{i,j}^{n}\right)\right)}\end{aligned}$$

$$\begin{aligned}H_{\epsilon}\left(\phi_{i,j}^{n}\right) &= \frac{1}{2}\left(1+\frac{2}{\pi}arctan\left(\frac{\phi_{i,j}^{n}}{\epsilon}\right)\right)\\ \delta_{\epsilon}\left(\phi_{i,j}^{n}\right) &= \frac{1}{\pi}\frac{\epsilon}{\epsilon^2+\left(\phi_{i,j}^{n}\right)^2}\\ \chi\left(\phi_{i,j}^{n}\right) &= \nabla\cdot\left(\frac{\nabla\phi}{\left|\nabla\phi\right|_{i,j}}\right)^{n}\\ \nabla\phi_{i,j}^{n} &= \begin{bmatrix}\frac{\phi^{n}_{i+\frac{1}{2},j}-\phi^{n}_{i-\frac{1}{2},j}}{h}\\ \frac{\phi^{n}_{i,j+\frac{1}{2}}-\phi^{n}_{i,j-\frac{1}{2}}}{h}\end{bmatrix}\\ \left|\nabla\phi_{i,j}^{n}\right| &= \sqrt{\left(\frac{\phi^{n}_{i+\frac{1}{2},j}-\phi^{n}_{i-\frac{1}{2},j}}{h}\right)^2+\left(\frac{\phi^{n}_{i,j+\frac{1}{2}}-\phi^{n}_{i,j-\frac{1}{2}}}{h}\right)^2}\\ K &= \left|\nabla\phi_{i,j}^{n}\right|\\ \left(\frac{\nabla\phi}{\left|\nabla\phi\right|}\right)_{i,j}^{n} &= \frac{1}{K}\begin{bmatrix}\frac{\phi^{n}_{i+\frac{1}{2},j}-\phi^{n}_{i-\frac{1}{2},j}}{h}\\ \frac{\phi^{n}_{i,j+\frac{1}{2}}-\phi^{n}_{i,j-\frac{1}{2}}}{h}\end{bmatrix}\\ \nabla\cdot\left(\frac{\nabla\phi}{\left|\nabla\phi\right|}\right)_{i,j}^{n} &= \frac{1}{K}\left(\frac{\phi^{n}_{i+1,j}-\phi^{n}_{i,j}}{h^2}-\frac{\phi^{n}_{i,j}-\phi^{n}_{i-1,j}}{h^2}+\frac{\phi^{n}_{i,j+1}-\phi^{n}_{i,j}}{h^2}-\frac{\phi^{n}_{i,j}-\phi^{n}_{i,j-1}}{h^2}\right)\\ \chi\left(\phi_{i+1,j}^{n}\right) &= \frac{1}{Kh^2}\left(\Delta_{+}^{x}\phi_{i,j}^{n}-\Delta_{-}^{x}\phi_{i,j}^{n}+\Delta_{+}^{y}\phi_{i,j}^{n}-\Delta_{-}^{y}\phi_{i,j}^{n}\right)\end{aligned}$$

Initial state of ϕ can be chosen an arbitrary curve around the objects of interest in the image.

B.2 2-Phase Piecewise Constant Active Contour Model Without Edges

In this section, the generalization of the 2-phase piecewise constant active contour model without edges [4, 94] and piecewise constant segmentation of images with more than two segments and junctions are using MS energy functional. As known that using only one level set function, one can represent only two phases or segments in the image. As a result of this, other geometrical features composed of more than two segments cannot be represented using only one level set function. With a multiphase level set model, one can represent more than two segments or phases, triple junctions and other complex topologies, in an efficient way. In order to represent n phases or segments with complex topologies, only $log_2 n$ level set functions are required. Since this partition is a disjoint decomposition and covering of the domain Ω by definition, it removes the problems of vacuum and overlap.

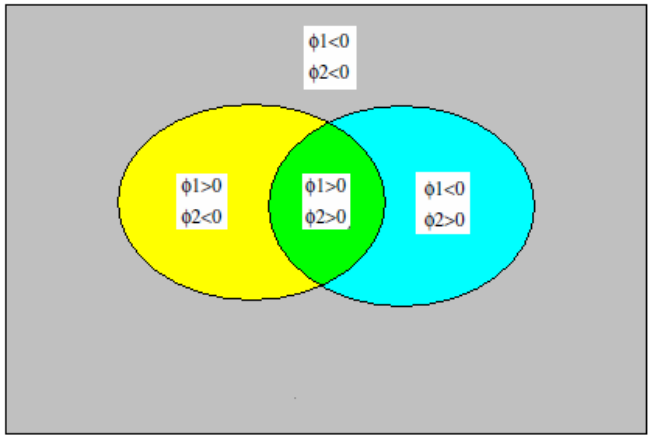


Figure B.2: An example of partition of the image in regions with boundaries represented via two level set of the Lipschitz functions $\{\phi_1 = 0\} \cup \{\phi_2 = 0\}$.

Let us consider $m = log_2 n$ level set functions $\phi_i : \Omega \rightarrow \Re$. The union of the zero level sets of ϕ_i represents the boundaries in the segmented image (see Figure B.2).

By using the regularized versions of the functions H and δ, the 2-phase energy func-

tional considered in CV active contour model without edges can written as

$$
\begin{aligned}
&E_{CV\epsilon}(\phi_1,\phi_2,c_{11},c_{10},c_{01},c_{00}) = \\
&\quad \mu_1 \int_\Omega \int \delta_\epsilon(\phi_1)\,|\nabla\phi_1|\,\partial x\partial y + \mu_2 \int_\Omega \int \delta_\epsilon(\phi_2)\,|\nabla\phi_2|\,\partial x\partial y \\
&\quad + \nu_1 \int_\Omega \int H_\epsilon(\phi_1)\,\partial x\partial y + \nu_2 \int_\Omega \int H_\epsilon(\phi_2)\,\partial x\partial y \\
&\quad + \lambda_{11} \int_\Omega \int |I-c_{11}|^2 H_\epsilon(\phi_2)\,H_\epsilon(\phi_1)\,\partial x\partial y \\
&\quad + \lambda_{10} \int_\Omega \int |I-c_{10}|^2 H_\epsilon(\phi_2)\,(1-H_\epsilon(\phi_1))\,\partial x\partial y \\
&\quad + \lambda_{01} \int_\Omega \int |I-c_{01}|^2 (1-H_\epsilon(\phi_2))\,H_\epsilon(\phi_1)\,\partial x\partial y \\
&\quad + \lambda_{00} \int_\Omega \int |I-c_{00}|^2 (1-H_\epsilon(\phi_2))\,(1-H_\epsilon(\phi_1))\,\partial x\partial y
\end{aligned}
\tag{B.16}
$$

where H_ϵ and δ_ϵ are the regularized versions of H and δ and $|\nabla H_\epsilon(\phi_i)| = \delta_\epsilon(\phi_i)\,|\nabla\phi_i|$. In addition, H_ϵ and δ_ϵ are

$$
H_\epsilon(\phi_i) = \frac{1}{2}\left(1+\frac{2}{\pi} arctan\left(\frac{\phi_i}{\epsilon}\right)\right) \tag{B.17}
$$

$$
\delta_\epsilon(\phi_i) = H'_\epsilon(\phi_i) = \frac{1}{\pi}\frac{\epsilon}{\epsilon^2+\phi_i^2} \tag{B.18}
$$

E_ϵ is minimized with respect to ϕ_i since c_{xx}s are fixed. Gateaux derivative of enegy functional in the direction ψ_i is used:

$$
D_{\phi_i}E_\epsilon(\phi_i,...)[\psi_i] = lim_{t\to 0}\frac{E_\epsilon(\phi_i+t\psi_i,...)-E_\epsilon(\phi_i,...)}{t}. \tag{B.19}
$$

Gateaux derivative of the term $\mu_i \int_\Omega \int \delta_\epsilon(\phi_i)\,|\nabla\phi_i|\,\partial x\partial y$ (see Appendix C) is

$$
-\mu_i \int_\Omega \int \delta_\epsilon(\phi_i)\,\nabla\cdot\left(\frac{\nabla\phi_i}{|\nabla\phi_i|}\right)\psi_i\partial x\partial y \tag{B.20}
$$

Gateaux derivative of the term $\nu_i \int_\Omega \int H_\epsilon(\phi_i)\,\partial x\partial y$ (see Appendix C) is

$$\nu_i \int_\Omega \int \delta_\epsilon (\phi_i) \psi_i \partial x \partial y \tag{B.21}$$

Gateaux derivative of the term $\lambda_{xx} \int_\Omega \int |I - c_{xx}|^2 H(\phi_i) ... \partial x \partial y$ in the direction ψ_i is

$$\begin{aligned}
&lim_{t \to 0} \lambda_{xx} \int_\Omega \int |I - c_{xx}|^2 \frac{H(\phi_i + t\psi_i) - H(\phi_i)}{t} ... \partial x \partial y \\
&= lim_{t \to 0} \lambda_{xx} \int_\Omega \int |I - c_{xx}|^2 \frac{H_\epsilon(\phi_i) + \frac{\partial H_\epsilon(\phi_i)}{\partial \phi_i} \cdot t\psi_i - H_\epsilon(\phi_i)}{t} ... \partial x \partial y \\
&= \lambda_{xx} \int_\Omega \int |I - c_{xx}|^2 H'_\epsilon(\phi_i) \psi_i ... \partial x \partial y \\
&= \lambda_{xx} \int_\Omega \int |I - c_{xx}|^2 \delta_\epsilon(\phi_i) \psi_i ... \partial x \partial y
\end{aligned} \tag{B.22}$$

Gateaux derivative of the term $\lambda_x \int_\Omega \int |I - c_{xx}|^2 (1 - H(\phi_i)) ... \partial x \partial y$ in the direction ψ_i is

$$\begin{aligned}
&lim_{t \to 0} \lambda_{xx} \int_\Omega \int |I - c_{xx}|^2 \frac{(1 - H(\phi_i + t\psi_i)) - (1 - H(\phi_i))}{t} ... \partial x \partial y \\
&= lim_{t \to 0} \lambda_{xx} \int_\Omega \int |I - c_{xx}|^2 \frac{-H_\epsilon(\phi_i) - \frac{\partial H_\epsilon(\phi_i)}{\partial \phi_i} \cdot t\psi_i + H_\epsilon(\phi_i)}{t} ... \partial x \partial y \\
&= -\lambda_{xx} \int_\Omega \int |I - c_{xx}|^2 H'_\epsilon(\phi_i) \psi_i ... \partial x \partial y \\
&= -\lambda_{xx} \int_\Omega \int |I - c_{xx}|^2 \delta_\epsilon(\phi_i) \psi_i ... \partial x \partial y
\end{aligned} \tag{B.23}$$

Manipulating the derivatives mentioned by Equation B.20, B.21, B.22 and B.23, one can obtain the following

$$
\begin{aligned}
&D_{\phi_i}E_\epsilon(\phi_1,\phi_2,c_{xx})[\psi_1] \\
&\quad = lim_{t\to 0}\frac{E_\epsilon(\phi_1+t\psi_1,\phi_2,c_{xx})-E_\epsilon(\phi_1,\phi_2,c_{xx})}{t} \\
&\quad = -\mu_1\int_\Omega\int\delta_\epsilon(\phi_1)\nabla\cdot\left(\frac{\nabla\phi_1}{|\nabla\phi_1|}\right)\psi_1\partial x\partial y+\nu_1\int_\Omega\int\delta_\epsilon(\phi_1)\psi_1\partial x\partial y \\
&\quad + \int_\Omega\int\left(\lambda_{11}|I-c_{11}|^2-\lambda_{10}|I-c_{10}|^2\right)\delta_\epsilon(\phi_1)\psi_1 H(\phi_2)\,\partial x\partial y \\
&\quad + \int_\Omega\int\left(\lambda_{01}|I-c_{01}|^2-\lambda_{00}|I-c_{00}|^2\right)\delta_\epsilon(\phi_1)\psi_1(1-H(\phi_2))\,\partial x\partial y
\end{aligned}
\tag{B.24}
$$

Since $\frac{\partial\phi_1}{\partial n}=0$,

$$
\begin{aligned}
&\int_\Omega\int\left(\mu_1\delta_\epsilon(\phi_1)\nabla\cdot\left(\frac{\nabla\phi_1}{|\nabla\phi_1|}\right)-\nu_1\delta_\epsilon(\phi_1)\right)\psi_1\partial x\partial y \\
&\quad + \int_\Omega\int\left(\lambda_{11}|I-c_{11}|^2-\lambda_{10}|I-c_{10}|^2\right)\delta_\epsilon(\phi_1)\psi_1 H(\phi_2)\,\partial x\partial y \\
&\quad + \int_\Omega\int\left(\lambda_{01}|I-c_{01}|^2-\lambda_{00}|I-c_{00}|^2\right)\delta_\epsilon(\phi_1)\psi_1(1-H(\phi_2))\,\partial x\partial y \\
&\quad = 0
\end{aligned}
\tag{B.25}
$$

By using the fundamental lemma of the calculus of variations, the following Euler Lagrange equation for ϕ_1 is obtained.

$$
\begin{aligned}
&\delta_\epsilon(\phi_1)\left(\mu_1\nabla\cdot\left(\frac{\nabla\phi_1}{|\nabla\phi_1|}\right)-\nu_1-\left(\lambda_{11}|I-c_{11}|^2-\lambda_{10}|I-c_{10}|^2\right)H(\phi_2)\right) \\
&\quad -\delta_\epsilon(\phi_1)\left(\left(\lambda_{01}|I-c_{01}|^2-\lambda_{00}|I-c_{00}|^2\right)(1-H(\phi_2))\right)=0
\end{aligned}
\tag{B.26}
$$

$$\frac{\partial\phi_1}{\partial n}=0 \text{ on } \partial\Omega_1$$

With the same manner, the following Euler Lagrange equation for ϕ_2 is obtained.

$$
\begin{aligned}
&\delta_\epsilon(\phi_2)\left(\mu_2\nabla\cdot\left(\frac{\nabla\phi_2}{|\nabla\phi_2|}\right)-\nu_2-\left(\lambda_{11}|I-c_{11}|^2-\lambda_{01}|I-c_{01}|^2\right)H(\phi_1)\right) \\
&\quad -\delta_\epsilon(\phi_2)\left(\left(\lambda_{10}|I-c_{10}|^2-\lambda_{00}|I-c_{00}|^2\right)(1-H(\phi_1))\right)=0
\end{aligned}
\tag{B.27}
$$

$$\frac{\partial\phi_2}{\partial n}=0 \text{ on } \partial\Omega_2$$

Numerical Approximations To find out the solution to the Euler-Lagrange equations, we apply the steepest descent algorithm in the distributed sense introducing an artificial time parameter t. The distributed iteration formula for the steepest descent algorithm is given as

$$\frac{\partial \phi_1}{\partial t} = \delta_\epsilon(\phi_1)\left(\mu_1 \nabla \cdot \left(\frac{\nabla \phi_1}{|\nabla \phi_1|}\right) - \nu_1 - \left(\lambda_{11}|I - c_{11}|^2 - \lambda_{10}|I - c_{10}|^2\right) H(\phi_2)\right) - \delta_\epsilon(\phi_1)\left(\left(\lambda_{01}|I - c_{01}|^2 - \lambda_{00}|I - c_{00}|^2\right)(1 - H(\phi_2))\right) = 0 \quad \text{(B.28)}$$

$$\frac{\partial \phi_2}{\partial t} = \delta_\epsilon(\phi_2)\left(\mu_2 \nabla \cdot \left(\frac{\nabla \phi_2}{|\nabla \phi_2|}\right) - \nu_2 - \left(\lambda_{11}|I - c_{11}|^2 - \lambda_{01}|I - c_{01}|^2\right) H(\phi_1)\right) - \delta_\epsilon(\phi_2)\left(\left(\lambda_{10}|I - c_{10}|^2 - \lambda_{00}|I - c_{00}|^2\right)(1 - H(\phi_1))\right) = 0 \quad \text{(B.29)}$$

For $k = 1$ and 2,

$$\phi_k(t, x, y) = \phi_k(0, x, y) \text{ in } \Omega_k$$

$$\frac{\delta_\epsilon(\phi_k)}{|\nabla \phi_k|}\frac{\partial \phi_k}{\partial n} = 0 \text{ on } \partial\Omega_k$$

$$c_{11} = \frac{\int_\Omega \int I H_\epsilon(\phi_2) H_\epsilon(\phi_1)\, \partial x \partial y}{\int_\Omega \int H_\epsilon(\phi_2) H_\epsilon(\phi_1)\, \partial x \partial y} = average(I) \text{ in } \phi_1 > 0 \text{ and } \phi_2 > 0 \quad \text{(B.30)}$$

$$c_{10} = \frac{\int_\Omega \int I H_\epsilon(\phi_2)(1 - H_\epsilon(\phi_1))\, \partial x \partial y}{\int_\Omega \int H_\epsilon(\phi_2)(1 - H_\epsilon(\phi_1))\, \partial x \partial y} = average(I) \text{ in } \phi_1 < 0 \text{ and } \phi_2 > 0 \quad \text{(B.31)}$$

$$c_{01} = \frac{\int_\Omega \int I(1 - H_\epsilon(\phi_2)) H_\epsilon(\phi_1)\, \partial x \partial y}{\int_\Omega \int (1 - H_\epsilon(\phi_2)) H_\epsilon(\phi_1)\, \partial x \partial y} = average(I) \text{ in } \phi_1 > 0 \text{ and } \phi_2 < 0 \quad \text{(B.32)}$$

$$c_{00} = \frac{\int_\Omega \int I(1 - H_\epsilon(\phi_2))(1 - H_\epsilon(\phi_1))\, \partial x \partial y}{\int_\Omega \int (1 - H_\epsilon(\phi_2))(1 - H_\epsilon(\phi_1))\, \partial x \partial y} = average(I) \text{ in } \phi_1 < 0 \text{ and } \phi_2 < 0 \quad \text{(B.33)}$$

Method of finite differences is used to approximate the iteration formula. Let h be the space, time step Δt and $(x.y) = (ih, jh)$ be the grid points. Let $\phi_k^n = \phi_k(n\Delta t, ih, jh)$ be an approximation of $\phi_k(t, x, y)$ with $n \geq 0$. The finite differences are

$$\Delta_+^x \phi_{i,j} = \phi_{i+1,j} - \phi_{i,j} \; \Delta_-^x \phi_{i,j} = \phi_{i,j} - \phi_{i-1,j}$$

$$\Delta_+^y \phi_{i,j} = \phi_{i,j+1} - \phi_{i,j} \; \Delta_-^y \phi_{i,j} = \phi_{i,j} - \phi_{i,j-1}$$

Discretization of the Euler-Lagrange equation yields

$$\begin{aligned}\frac{(\phi_1)_{i,j}^{n+1} - (\phi_1)_{i,j}^n}{\Delta t} &= \delta_\epsilon\left((\phi_1)_{i,j}^n\right)\left(\mu_1 \chi\left((\phi_1)_{i,j}^n\right) - \nu_1\right) \\ &- \delta_\epsilon\left((\phi_1)_{i,j}^n\right)\left(\lambda_{11}\left|I - c_{11}\right|^2 - \lambda_{10}\left|I - c_{10}\right|^2\right) H\left((\phi_2)_{i,j}^n\right) \\ &- \delta_\epsilon\left((\phi_1)_{i,j}^n\right)\left(\lambda_{01}\left|I - c_{01}\right|^2 - \lambda_{00}\left|I - c_{00}\right|^2\right)\left(1 - H\left((\phi_2)_{i,j}^n\right)\right) \end{aligned} \tag{B.34}$$

$$\begin{aligned}\frac{(\phi_2)_{i,j}^{n+1} - (\phi_2)_{i,j}^n}{\Delta t} &= \delta_\epsilon\left((\phi_2)_{i,j}^n\right)\left(\mu_2 \chi\left((\phi_2)_{i,j}^n\right) - \nu_2\right) \\ &- \delta_\epsilon\left((\phi_2)_{i,j}^n\right)\left(\lambda_{11}\left|I - c_{11}\right|^2 - \lambda_{01}\left|I - c_{01}\right|^2\right) H\left((\phi_1)_{i,j}^n\right) \\ &- \delta_\epsilon\left((\phi_2)_{i,j}^n\right)\left(\lambda_{10}\left|I - c_{10}\right|^2 - \lambda_{00}\left|I - c_{00}\right|^2\right)\left(1 - H\left((\phi_1)_{i,j}^n\right)\right) \end{aligned} \tag{B.35}$$

where

$$\begin{aligned} c_{11} &= \frac{\sum I H_\epsilon\left((\phi_2)_{i,j}^n\right) H_\epsilon\left((\phi_1)_{i,j}^n\right)}{\sum H_\epsilon\left((\phi_2)_{i,j}^n\right) H_\epsilon\left((\phi_1)_{i,j}^n\right)} \\ c_{10} &= \frac{\sum I H_\epsilon\left((\phi_2)_{i,j}^n\right)\left(1 - H_\epsilon\left((\phi_1)_{i,j}^n\right)\right)}{\sum H_\epsilon\left((\phi_2)_{i,j}^n\right)\left(1 - H_\epsilon\left((\phi_1)_{i,j}^n\right)\right)} \\ c_{01} &= \frac{\sum I\left(1 - H_\epsilon\left((\phi_2)_{i,j}^n\right)\right) H_\epsilon\left((\phi_1)_{i,j}^n\right)}{\sum\left(1 - H_\epsilon\left((\phi_2)_{i,j}^n\right)\right) H_\epsilon\left((\phi_1)_{i,j}^n\right)} \\ c_{00} &= \frac{\sum I\left(1 - H_\epsilon\left((\phi_2)_{i,j}^n\right)\right)\left(1 - H_\epsilon\left((\phi_1)_{i,j}^n\right)\right)}{\sum\left(1 - H_\epsilon\left((\phi_2)_{i,j}^n\right)\right)\left(1 - H_\epsilon\left((\phi_1)_{i,j}^n\right)\right)} \end{aligned}$$

$$
\begin{aligned}
H_\epsilon\left(\left(\phi_k\right)_{i,j}^n\right) &= \frac{1}{2}\left(1+\frac{2}{\pi}arctan\left(\frac{\left(\phi_k\right)_{i,j}^n}{\epsilon}\right)\right)\\
\delta_\epsilon\left(\left(\phi_k\right)_{i,j}^n\right) &= \frac{1}{\pi}\frac{\epsilon}{\epsilon^2+\left(\left(\phi_k\right)_{i,j}^n\right)^2}\\
\chi\left(\left(\phi_k\right)_{i,j}^n\right) &= \nabla\cdot\left(\frac{\nabla\phi_k}{\left|\nabla\phi_k\right|_{i,j}}\right)^n\\
\nabla\left(\phi_k\right)_{i,j}^n &= \begin{bmatrix} \frac{\left(\phi_k\right)_{i+\frac{1}{2},j}^n-\left(\phi_k\right)_{i-\frac{1}{2},j}^n}{h} \\ \\ \frac{\left(\phi_k\right)_{i,j+\frac{1}{2}}^n-\left(\phi_k\right)_{i,j-\frac{1}{2}}^n}{h} \end{bmatrix}\\
\left|\nabla\left(\phi_k\right)_{i,j}^n\right| &= \sqrt{\left(\frac{\left(\phi_k\right)_{i+\frac{1}{2},j}^n-\left(\phi_k\right)_{i-\frac{1}{2},j}^n}{h}\right)^2+\left(\frac{\left(\phi_k\right)_{i,j+\frac{1}{2}}^n-\left(\phi_k\right)_{i,j-\frac{1}{2}}^n}{h}\right)^2}\\
K_k &= \left|\nabla\left(\phi_k\right)_{i,j}^n\right|\\
\left(\frac{\nabla\phi_k}{\left|\nabla\phi_k\right|}\right)_{i,j}^n &= \frac{1}{K_k}\begin{bmatrix} \frac{\left(\phi_k\right)_{i+\frac{1}{2},j}^n-\left(\phi_k\right)_{i-\frac{1}{2},j}^n}{h} \\ \\ \frac{\left(\phi_k\right)_{i,j+\frac{1}{2}}^n-\left(\phi_k\right)_{i,j-\frac{1}{2}}^n}{h} \end{bmatrix}\\
\nabla\cdot\left(\frac{\nabla\phi_k}{\left|\nabla\phi_k\right|}\right)_{i,j}^n &= \frac{1}{K_k}\left(\frac{\left(\phi_k\right)_{i+1,j}^n-\left(\phi_k\right)_{i,j}^n}{h^2}-\frac{\left(\phi_k\right)_{i,j}^n-\left(\phi_k\right)_{i-1,j}^n}{h^2}+\frac{\left(\phi_k\right)_{i,j+1}^n-\left(\phi_k\right)_{i,j}^n}{h^2}-\frac{\left(\phi_k\right)_{i,j}^n-\left(\phi_k\right)_{i,j-1}^n}{h^2}\right)\\
\chi\left(\left(\phi_k\right)_{i+1,j}^n\right) &= \frac{1}{K_k h^2}\left(\Delta_+^x\left(\phi_k\right)_{i,j}^n-\Delta_-^x\left(\phi_k\right)_{i,j}^n+\Delta_+^y\left(\phi_k\right)_{i,j}^n-\Delta_-^y\left(\phi_k\right)_{i,j}^n\right)
\end{aligned}
$$

Initial state of each ϕ_i can be chosen an arbitrary curve around the objects of interest in the image.

Appendix C

GATEAUX DERIVATIVES OF THE CHAN-VESE ENERGY FUNCTIONAL TERMS

C.1 Gateaux derivative of the term $\mu \int_\Omega \int \delta_\epsilon(\phi) |\nabla\phi| \, \partial x \partial y$

Gateaux derivative of the term $\mu \int_\Omega \int \delta_\epsilon(\phi) |\nabla\phi| \, \partial x \partial y$ in the direction ψ is

$$lim_{t\to 0}\mu \int_\Omega \int \frac{\delta_\epsilon(\phi + t\psi) |\nabla(\phi + t\psi)| - \delta_\epsilon(\phi) |\nabla\phi|}{t} \partial x \partial y. \tag{C.1}$$

Subtracting and adding the term $\delta_\epsilon(\phi) |\nabla(\phi + t\psi)|$ to numerator, we obtain

$$\begin{aligned} lim_{t\to 0}\mu \int_\Omega \int & \frac{\delta_\epsilon(\phi + t\psi) |\nabla(\phi + t\psi)| - \delta_\epsilon(\phi) |\nabla(\phi + t\psi)|}{t} \\ & + \frac{\delta_\epsilon(\phi) |\nabla(\phi + t\psi)| - \delta_\epsilon(\phi) |\nabla\phi|}{t} \partial x \partial y \\ & = lim_{t\to 0}\mu \int_\Omega \int \frac{\delta_\epsilon(\phi + t\psi) - \delta_\epsilon(\phi)}{t} |\nabla(\phi + t\psi)| \, \partial x \partial y \\ & + lim_{t\to 0}\mu \int_\Omega \int \delta_\epsilon(\phi) \frac{|\nabla(\phi + t\psi)| - |\nabla\phi|}{t} \partial x \partial y \end{aligned} \tag{C.2}$$

$$\begin{aligned} \bullet lim_{t\to 0}\mu & \int_\Omega \int \frac{\delta_\epsilon(\phi + t\psi) - \delta_\epsilon(\phi)}{t} |\nabla(\phi + t\psi)| \, \partial x \partial y \\ & = lim_{t\to 0}\mu \int_\Omega \int \frac{\delta_\epsilon(\phi) + \frac{\partial \delta_\epsilon(\phi)}{\partial \phi} \cdot t\psi - \delta_\epsilon(\phi)}{t} |\nabla(\phi + t\psi)| \, \partial x \partial y \\ & = \mu \int_\Omega \int \frac{\partial \delta_\epsilon(\phi)}{\partial \phi} \psi |\nabla\phi| \, \partial x \partial y \end{aligned} \tag{C.3}$$

$$
\begin{aligned}
&\bullet lim_{t\to 0}\mu \int_\Omega \int \delta_\epsilon(\phi) \frac{|\nabla(\phi + t\psi)| - |\nabla\phi|}{t} \partial x \partial y \\
&= lim_{t\to 0}\mu \int_\Omega \int \delta_\epsilon(\phi) \frac{|\nabla\phi| + \frac{\partial|\nabla\phi|}{\partial\nabla\phi} \cdot t\nabla\psi - |\nabla\phi|}{t} \partial x \partial y \\
&= \mu \int_\Omega \int \delta_\epsilon(\phi) \frac{\nabla\phi}{|\nabla\phi|} \nabla\psi \partial x \partial y
\end{aligned} \tag{C.4}
$$

Substituting both results in Equation C.1, we may get

$$
\begin{aligned}
&lim_{t\to 0}\mu \int_\Omega \int \frac{\delta_\epsilon(\phi + t\psi)|\nabla(\phi + t\psi)| - \delta_\epsilon(\phi)|\nabla\phi|}{t} \partial x \partial y \\
&= \mu \int_\Omega \int \frac{\partial \delta_\epsilon(\phi)}{\partial\phi} \psi |\nabla\phi| + \delta_\epsilon(\phi) \frac{\nabla\phi \cdot \nabla\psi}{|\nabla\phi|} \partial x \partial y
\end{aligned} \tag{C.5}
$$

After Green's theorem is applied, we may obtain

$$
\begin{aligned}
&\bullet \mu \int_\Omega \int \delta_\epsilon(\phi) \frac{\nabla\phi \cdot \nabla\psi}{|\nabla\phi|} \partial x \partial y \\
&= \mu \oint_{\partial\Omega} \frac{\delta_\epsilon(\phi)}{|\nabla\phi|} \frac{\partial\phi}{\partial n} \psi \partial s - \mu \int_\Omega \int \psi \nabla \cdot \left(\delta_\epsilon(\phi) \frac{\nabla\phi}{|\nabla\phi|} \right) \partial x \partial y
\end{aligned} \tag{C.6}
$$

Substituting Equation C.6 in the right side of Equation C.5, we can get

$$
\mu \int_\Omega \int \frac{\partial \delta_\epsilon(\phi)}{\partial\phi} \psi |\nabla\phi| \, \partial x \partial y + \mu \oint_{\partial\Omega} \frac{\delta_\epsilon(\phi)}{|\nabla\phi|} \frac{\partial\phi}{\partial n} \psi \partial s - \mu \int_\Omega \int \nabla \cdot \left(\delta_\epsilon(\phi) \frac{\nabla\phi}{|\nabla\phi|} \right) \psi \partial x \partial y \tag{C.7}
$$

Developing the divergence operator

$$
\begin{aligned}
&\mu \int_\Omega \int \nabla \cdot \left(\delta_\epsilon(\phi) \frac{\nabla\phi}{|\nabla\phi|} \right) \psi \partial x \partial y \\
&= \mu \int_\Omega \int \frac{\partial \delta_\epsilon(\phi)}{\partial\phi} |\nabla\phi| \, \psi \partial x \partial y + \mu \int_\Omega \int \delta_\epsilon(\phi) \nabla \cdot \left(\frac{\nabla\phi}{|\nabla\phi|} \right) \psi \partial x \partial y
\end{aligned} \tag{C.8}
$$

We may write the right side of Equation C.5 as

$$\mu \int_\Omega \int \frac{\partial \delta_\epsilon(\phi)}{\partial \phi} |\nabla \phi| \psi \partial x \partial y + \mu \oint_{\partial \Omega} \frac{\delta_\epsilon(\phi)}{|\nabla \phi|} \frac{\partial \phi}{\partial n} \psi \partial s$$
$$- \mu \int_\Omega \int \frac{\partial \delta_\epsilon(\phi)}{\partial \phi} |\nabla \phi| \psi \partial x \partial y - \mu \int_\Omega \int \delta_\epsilon(\phi) \nabla \cdot \left(\frac{\nabla \phi}{|\nabla \phi|} \right) \psi \partial x \partial y \quad \text{(C.9)}$$

for all admissible ψ. Since $\frac{\partial \phi}{\partial n} = 0$, we eventually can get

$$- \mu \int_\Omega \int \delta_\epsilon(\phi) \nabla \cdot \left(\frac{\nabla \phi}{|\nabla \phi|} \right) \psi \partial x \partial y \quad \text{(C.10)}$$

C.2 Gateaux derivative of the term $\nu \int_\Omega \int H_\epsilon(\phi) \partial x \partial y$

Gateaux derivative of the term $\nu \int_\Omega \int H_\epsilon(\phi) \partial x \partial y$ in the direction ψ is

$$\begin{aligned} & lim_{t \to 0} \nu \int_\Omega \int \frac{H_\epsilon(\phi + t\psi) - H_\epsilon(\phi)}{t} \partial x \partial y \\ &= \nu \int_\Omega \int H'_\epsilon(\phi) \psi \partial x \partial y \\ &= \nu \int_\Omega \int \delta_\epsilon(\phi) \psi \partial x \partial y \end{aligned} \quad \text{(C.11)}$$

C.3 Gateaux derivative of the term $\lambda_1 \int_\Omega \int |I - c_1|^2 H(\phi) \partial x \partial y$

Gateaux derivative of the term $\lambda_1 \int_\Omega \int |I - c_1|^2 H(\phi) \partial x \partial y$ in the direction ψ is

$$\begin{aligned} & lim_{t \to 0} \lambda_1 \int_\Omega \int |I - c_1|^2 \frac{H(\phi + t\psi) - H(\phi)}{t} \partial x \partial y \\ &= lim_{t \to 0} \lambda_1 \int_\Omega \int |I - c_1|^2 \frac{H_\epsilon(\phi) + \frac{\partial H_\epsilon(\phi)}{\partial \phi} \cdot t\psi - H_\epsilon(\phi)}{t} \partial x \partial y \\ &= \lambda_1 \int_\Omega \int |I - c_1|^2 H'_\epsilon(\phi) \psi \partial x \partial y \\ &= \lambda_1 \int_\Omega \int |I - c_1|^2 \delta_\epsilon(\phi) \psi \partial x \partial y \end{aligned} \quad \text{(C.12)}$$

C.4 Gateaux derivative of the term $\lambda_2 \int_\Omega \int |I - c_0|^2 (1 - H(\phi)) \partial x \partial y$

Gateaux derivative of the term $\lambda_2 \int_\Omega \int |I - c_0|^2 (1 - H(\phi)) \partial x \partial y$ in the direction ψ is

$$
\begin{aligned}
& lim_{t \to 0} \lambda_2 \int_\Omega \int |I - c_0|^2 \frac{(1 - H(\phi + t\psi)) - (1 - H(\phi))}{t} \partial x \partial y \\
& = lim_{t \to 0} \lambda_2 \int_\Omega \int |I - c_0|^2 \frac{-H_\epsilon(\phi) - \frac{\partial H_\epsilon(\phi)}{\partial \phi} \cdot t\psi + H_\epsilon(\phi)}{t} \partial x \partial y \\
& = -\lambda_2 \int_\Omega \int |I - c_0|^2 H'_\epsilon(\phi) \psi \partial x \partial y \\
& = -\lambda_2 \int_\Omega \int |I - c_0|^2 \delta_\epsilon(\phi) \psi \partial x \partial y
\end{aligned} \tag{C.13}
$$

Appendix D

SOFTWARE USER GUIDES

D.1 Graphical user interface for filtering operation

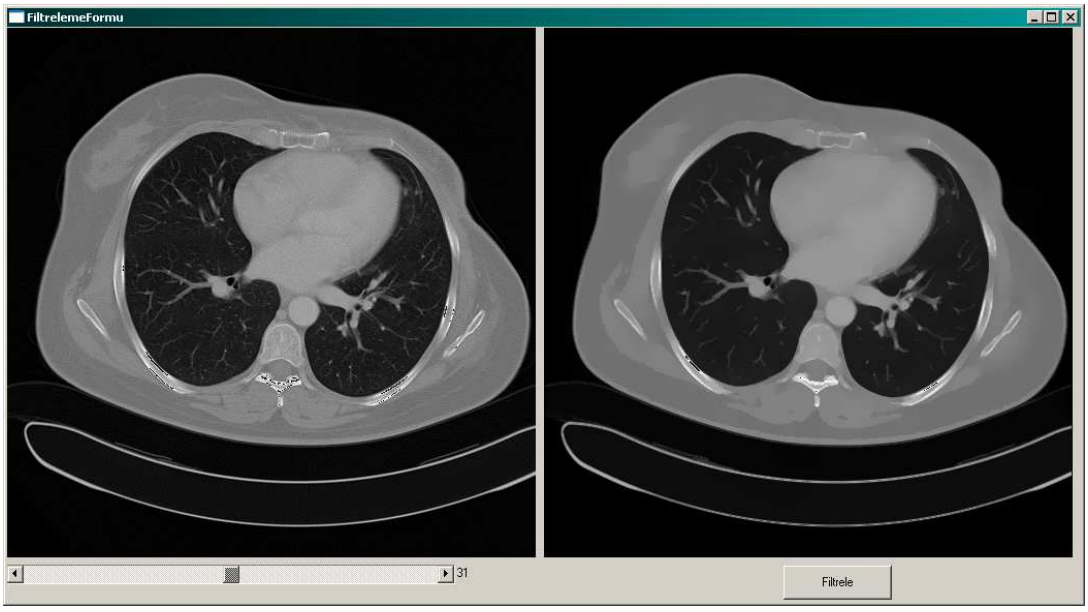
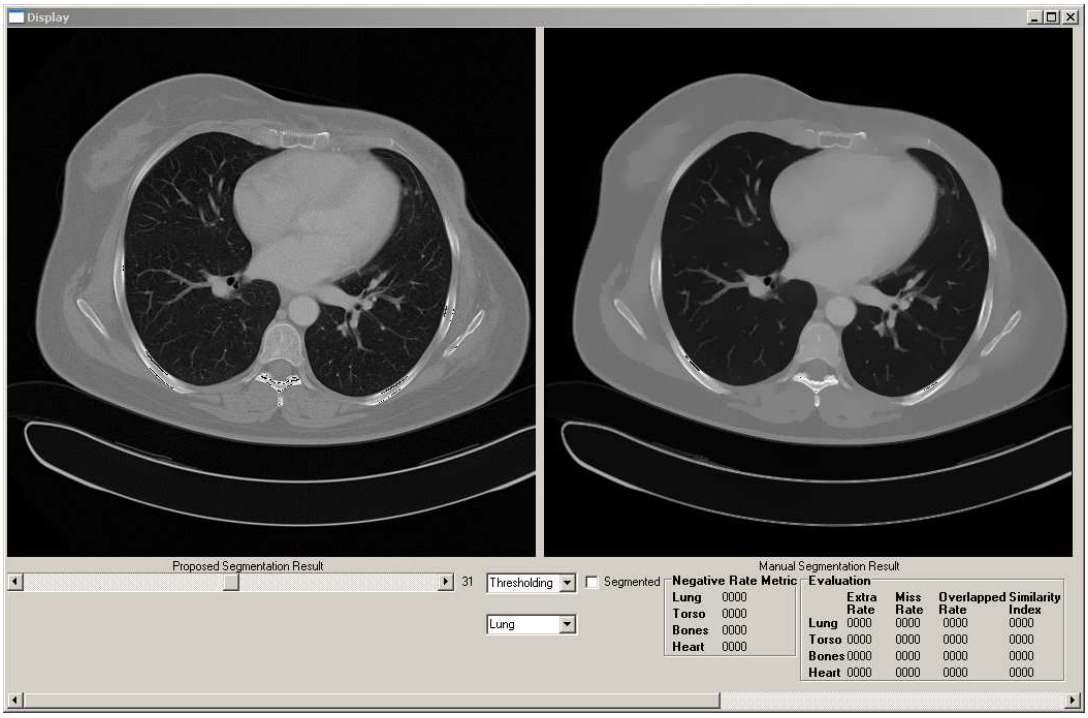
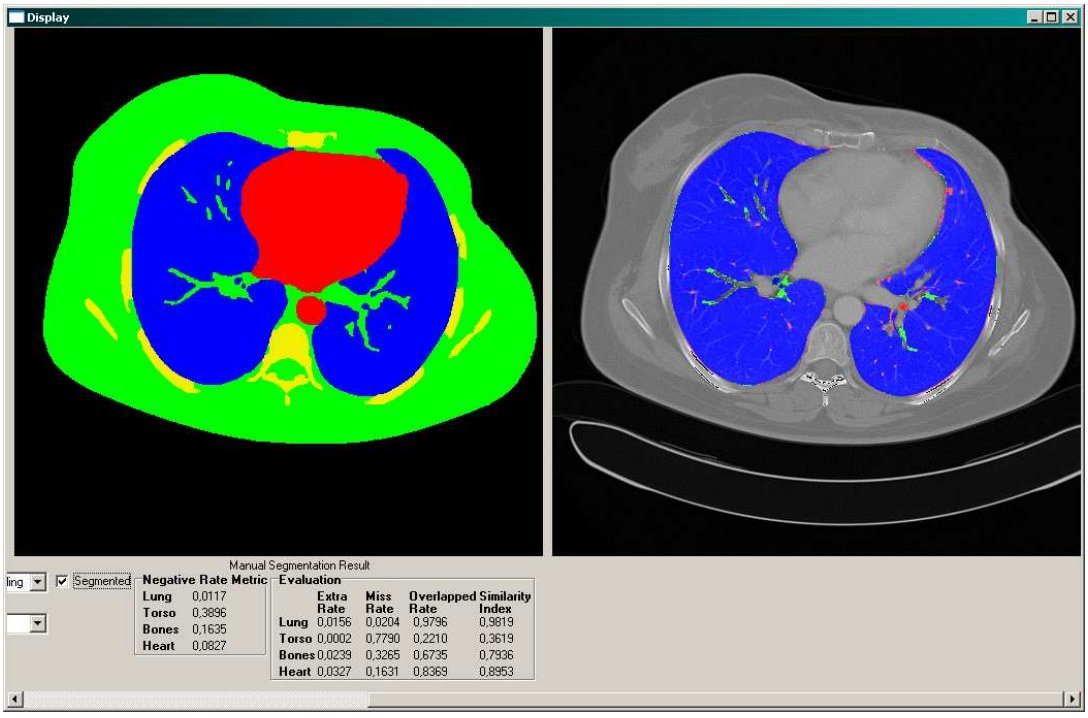


Figure D.1: GUI for filtering

The graphical user interface for filtering is shown in Figure D.1. After the user clicks the button called “Filtrele”, anisotropic filtering is applied on the medical image data set. The original medical images are stored in the folder named as “Orijinal”. The filtered images are stored with the file format “.bin” in the folder named as “Filtered”. In addition, the original and filtered images are depicted by using the scroll bar in the GUI.



(a)



(b)

Figure D.2: GUI to monitor the segmentation results

D.2 Graphical user interface to monitor the segmentation results

This interface shown in Figure D.2 is used to monitor the segmentation results and to calculate the metrics for the evaluation of segmentation. There are a scroll bar, two combo boxes, a check box, and three images. If the check box “segmented” is unchecked, the left two images show us the original and filtered version of the slice selected by using the scroll bar. If it is checked, the proposed and manual segmentation results are monitored in the first two images. In the right image, the missed, overlapped and extra segmented regions are shown for the structure determined by the user using the combo box. the proposed method is selected with the other combo box. Moreover, the evaluation metrics are calculated and motinored in the GUI.

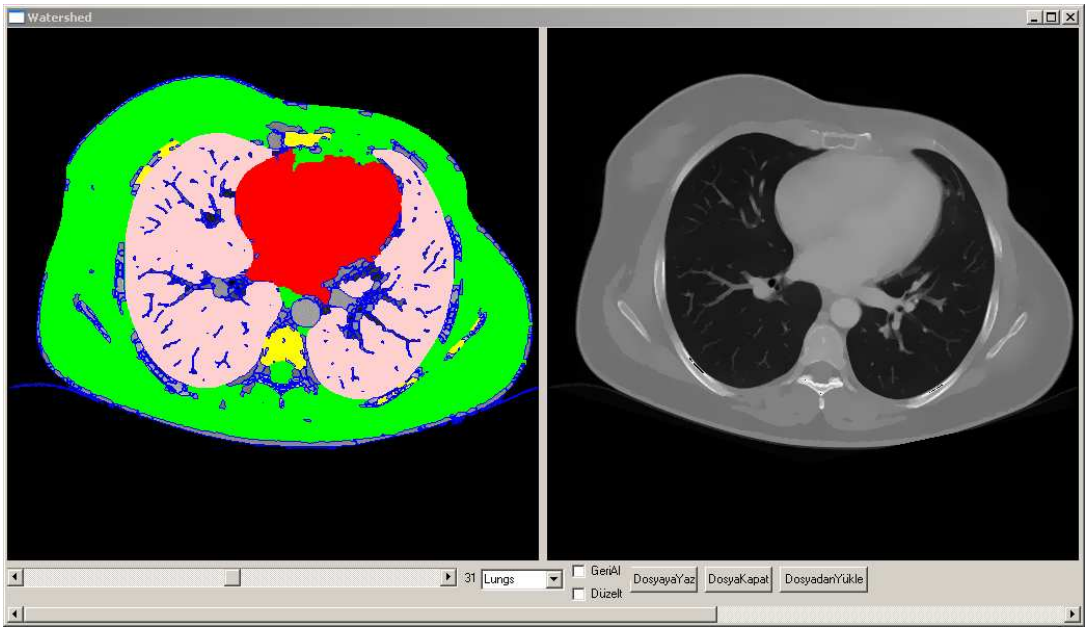


Figure D.3: GUI for manual segmentation using watershed transform

D.3 Graphical user interface for manual segmentation using watershed transform

This interface depicted in Figure D.2 is used to help an expert to make manual segmentation. There are a scroll bar, a combo box, two check boxes, three buttons, and

two images. By clicking three points with a mouse, an expert discards the table on which a patient lies from the image and the watershed transform is perform on the image. If the check boxes “Geri al” and “Duzelt” are unchecked, any region can be labeled by an expert with a mouse. If the check box “Geri al” is checked, an expert undoes labeling of the region. If the check box “Duzelt” is checked, the software gives a chance an expert to select any triangular region and to label the pixels inside the region with the structure determined by the user using the combo box.

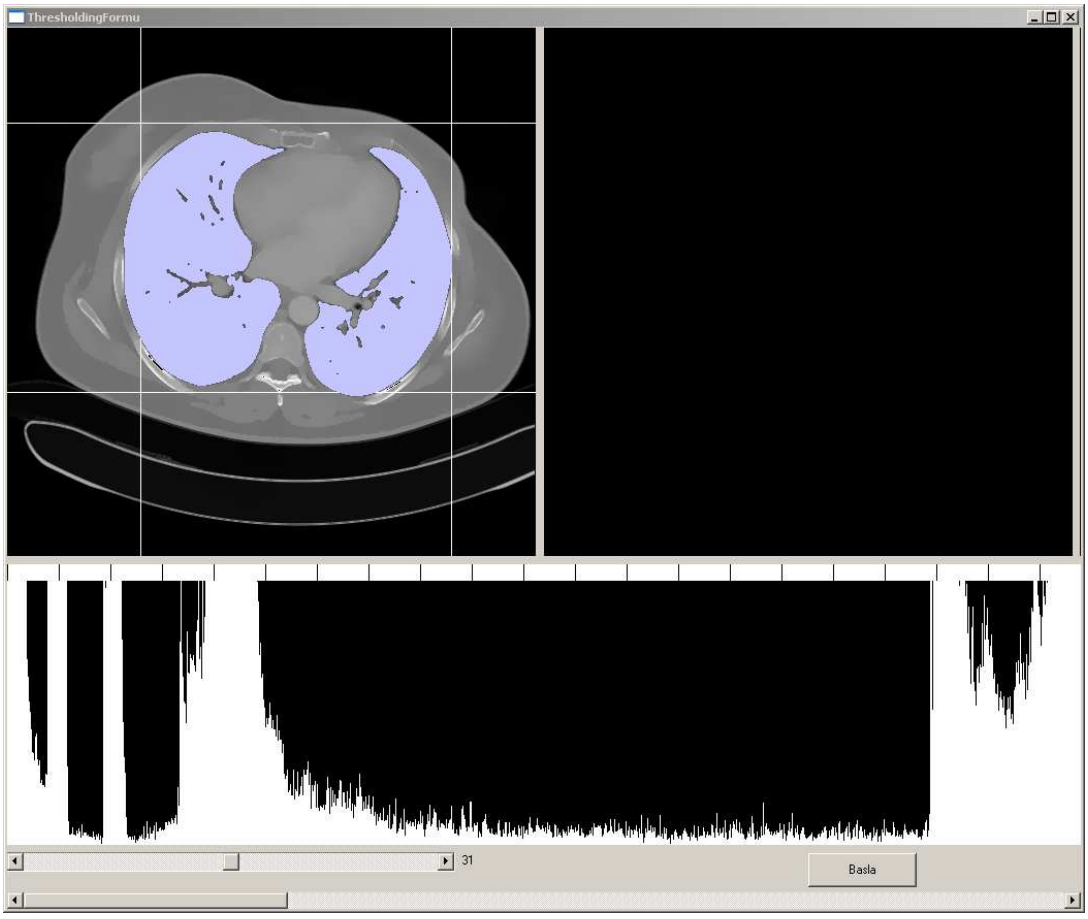


Figure D.4: GUI for the framework based on thresholding

D.4 Graphical user interface for the framework based on thresholding

This software depicted in Figure D.4 is used to make semi automatic segmentation based on thresholding. There are a scroll bar which is used for the selection of the

initial slice. By clicking two points with a mouse, the user describes the region of interest in the initial image. After defining the region of interest, the user chooses a pixel belongin to the structure to be segmented. Then the software performs segmentation of the structure selected by user for an initial slice. By clicking the button, the user continues the segmentation for the remaining slices.

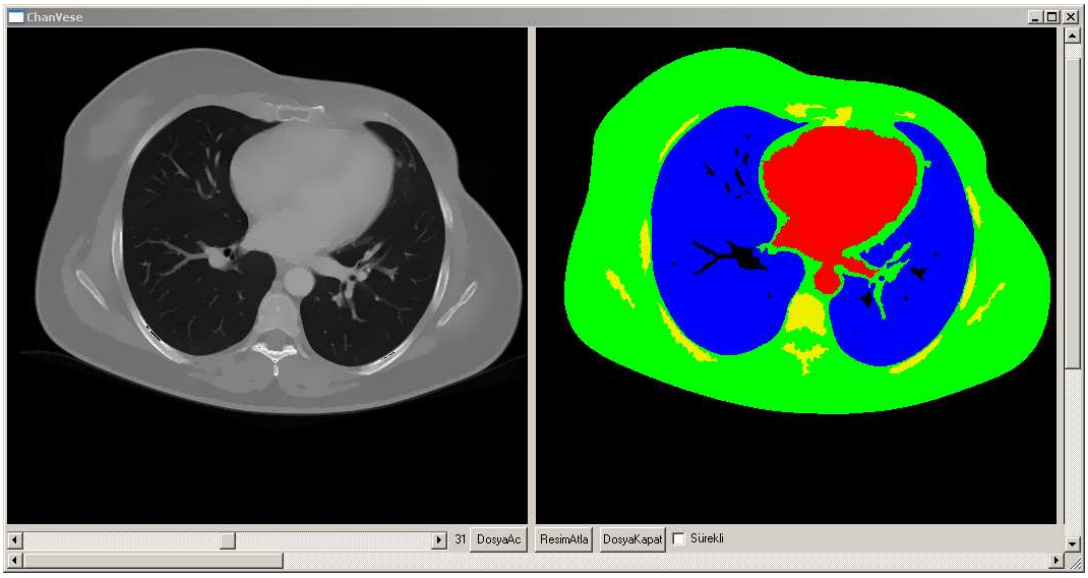


Figure D.5: GUI for the framework based on active contours

D.5 Graphical user interface for the frameworks based on active contours and watershed transform

The interfaces for the frameworks based on active contours and watershed transform depicted in Figures D.5 and D.6 are used to make semi-automatic segmentation based on active contours and watershed transform. A scroll bar is used for the selection of the initial slice. By clicking three points with a mouse, an expert discards the table on which a patient lies from the image and the watershed transform is perform on the image. Then the software performs segmentation of all of the structure for an initial slice. By clicking the button, the user continues the segmentation for the remaining slices.

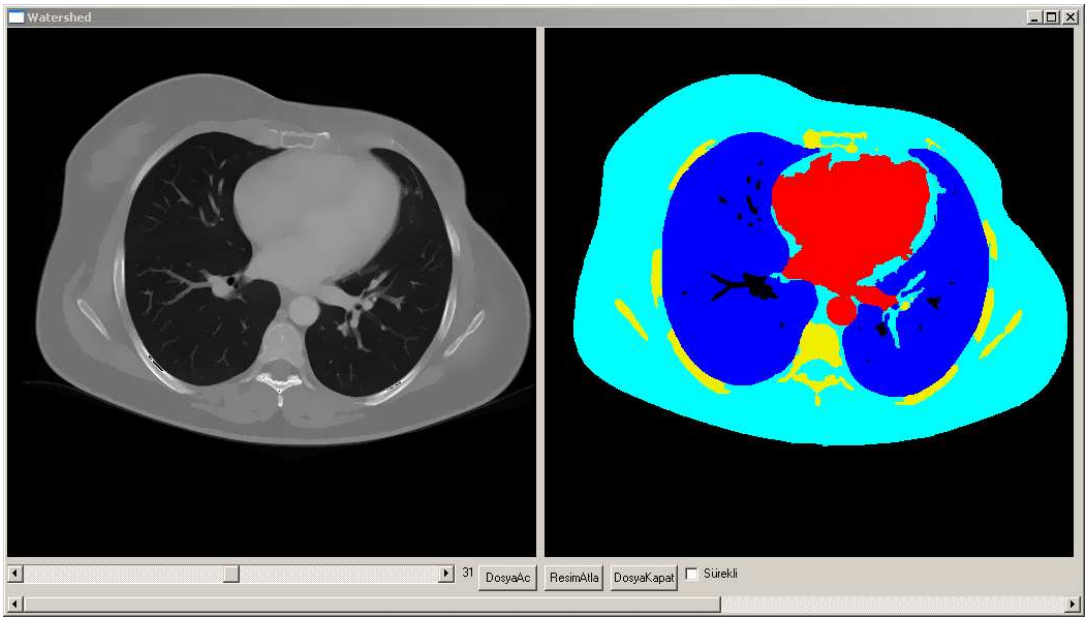


Figure D.6: GUI for the framework based on watershed transform

Appendix E

HOMOGENEOUS REPRESENTATION OF LINES AND POINTS

E.1 Representation of lines

A line in the plane is represented by an equation such as $ax + by + c = 0$. Different lines can be obtain by selecting different a, b and c. Therefore, a vector $(a, b, c)^T$ can represent a line. The equations $ax + by + c = 0$ and $kax + kby + kc = 0$ represent the same line, for non-zero constant k. For this reason, the vectors $(a, b, c)^T$ and $k(a, b, c)^T$ also represent the same line, for non-zero constant k. Indeed, two such vectors related by an overall scaling are considered as being equivalent.

E.2 Representation of points

A point $(x, y)^T$ lies on the line $(a, b, c)^T$ if and only if $ax + by + c = 0$. This equation may be written in terms of inner product of vectors; that is, $(x, y, 1)(a, b, c)^T$. Thus, the point $(x, y)^T$ in R^2 can be represented by a vector $(x, y, 1)$. Like line representation, two such vectors $(x, y, 1)$ and $(kx, ky, k1)$ represent the same point in R^2.

E.3 Line joining points

An expression for the line passing through two points $\vec{x}$ and $\vec{x'}$ could be driven by the definition of a line $\vec{l}$ by $\vec{l} = \vec{x} \times \vec{x'}$.

$$\vec{l} = \vec{x} \times \vec{x'} = \begin{vmatrix} i & j & k \\ x_i & x_j & x_k \\ x'_i & x'_j & x'_k \end{vmatrix}$$