



THE AUTOMATED BLOOD VESSEL SEGMENTATION FRAMEWORK WITH
HYBRID LEVEL SET METHOD

MR. TANANAN PATTANANGKUR

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR
THE DEGREE OF MASTER OF ENGINEERING (COMPUTER ENGINEERING)
FACULTY OF ENGINEERING
KING MONGKUT'S UNIVERSITY OF TECHNOLOGY THONBURI
2014

The Automated Blood Vessel Segmentation Framework with Hybrid Level Set Method

Mr. Tananan Pattanangkur B.Eng (Computer Engineering)

A Thesis Submitted in Partial Fulfillment of the Requirement for
the Degree of Master of (Computer Engineering)
Faculty of Engineering
King Mongkut's University of Technology Thonburi
2014

Thesis Committee

..... (Assoc. Prof. Suthep Madarasmi, Ph.D.)	Chairman of Thesis Committee
..... (Assoc. Prof. Tiranee Achalakul, Ph.D.)	Member and Thesis Advisor
..... (Santitham Prom-on, Ph.D.)	Member
..... (Sittapong Settapat, Ph.D.)	Member
..... (Chanikarn Wongviriyawong, Ph.D.)	Member

Thesis Title	The Automated Blood Vessel Segmentation Framework with Hybrid Level Set Method
Thesis Credits	12
Candidate	Mr. Tananan Pattanangkur
Thesis Advisor	Assoc. Prof. Dr. Tiranee Achalakul
Program	Master of Engineering
Field of Study	Computer Engineering
Department	Computer Engineering
Faculty	Engineering
Academic Year	2014

Abstract

Occlusive vascular disease is one of the most critical diseases nowadays. The important arteries and risks for this disease is a carotid artery. This artery is responsible for supplying the brain with oxygen blood. Medical visualization could assist doctors by visualizing the vessel structure of occlusive vascular disease patients in 3D model images. However, there are some obstacles for segmenting the vessel structure from medical images. As the range of intensity of blood vessels still overlaps with the range of intensity of bones, it is impossible to segment the blood vessel by using the conventional thresholding technique.

In this research, we intended to create novel algorithmic techniques that can perform the automatic segmentation of blood vessel in CTA images, without a user interaction or any initial points from the user. In addition, we also display our segmentation in high quality 3D model images. Moreover, we propose the modified version of Level Set Method which is called Hybrid LSM. This method is the combination between edge based and region based active contour model.

We performed the experiments by comparing Hybrid LSM with conventional method. Hybrid LSM can segment the boundary of the desired object, resist to the noisy image, work well despite the objects' unclear boundary and faster than the conventional methods. However, some errors can occur around the acute angle corners. In addition, our blood vessel segmentation framework was evaluated based on the evaluation criteria from Carotid Lumen Segmentation and Stenosis Grading Challenge (CLS). We performed the experiments by using fifteen CTA datasets. We obtained the average dice similarity at 87.09%, average sensitivity at 88.13% and average specification at 86.63% which shows that our method was efficient for vessel segmentation. However, some false positive occurred in the vessel reconstruction process

Keywords: Medical Images / Direct volume rendering / Ray Casting algorithm / Vessel segmentation / Vesselness filter / Level Set Method

หัวข้อวิทยานิพนธ์	การแยกชิ้นส่วนเส้นเลือด อย่างอัตโนมัติ ด้วยวิธีการ Hybrid Level Set Method
หน่วยกิต	12
ผู้เขียน	นายชนานันต์ พัฒนางกูร
อาจารย์ที่ปรึกษา	รศ.ดร.ธีรณี อจลากุล
หลักสูตร	วิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชา	วิศวกรรมศาสตร์
ภาควิชา	วิศวกรรมคอมพิวเตอร์
คณะ	วิศวกรรมศาสตร์
ปีการศึกษา	2557

บทคัดย่อ

ในปัจจุบัน โรคหลอดเลือดอุดตัน เป็นอีกโรคหนึ่งที่มีความร้ายแรงมาก ซึ่งเส้นเลือดที่มีความสำคัญ และเสี่ยงต่อการเกิดโรคหลอดเลือดอุดตันคือเส้นเลือด carotid artery โดยเส้นเลือดนี้จะทำหน้าที่ในการลำเลียงออกซิเจนไปสู่สมอง ในปัจจุบันเทคโนโลยีการสร้างภาพจำลองทางการแพทย์มีส่วนช่วยในการสร้างภาพจำลองโครงสร้างของเส้นเลือดที่มีการอุดตัน ในรูปแบบภาพสามมิติ แต่อย่างไรก็ตาม การแยกแยะเฉพาะส่วนที่เป็นโครงสร้างของเส้นเลือด ออกจากภาพถ่ายทางการแพทย์นั้น เป็นสิ่งที่ยังคงทำได้ยาก เนื่องจากช่วงค่าความเข้มสีของหลอดเลือด และกระดูก มักจะอยู่ในช่วงเดียวกัน ทำให้การแยกแยะโครงสร้างของเส้นเลือด ไม่สามารถทำได้ด้วยวิธีการ thresholding ทั่วไป

ในงานวิจัยนี้ คณะผู้วิจัยมีจุดประสงค์ที่จะพัฒนาวิธีการใหม่ ที่สามารถทำการแยกแยะโครงสร้างของเส้นเลือดจากภาพถ่ายทางการแพทย์ ประเภทภาพ CTA ได้โดยอัตโนมัติ โดยปราศจากการปรับแต่ง ค่า และ การกำหนดจุดเริ่มต้น หรือจุดสิ้นสุด จากผู้ใช้งาน และทำการแสดงผลลัพธ์ ภาพโครงสร้างของเส้นเลือดในรูปแบบภาพสามมิติ นอกจากนี้ คณะผู้วิจัยได้นำเสนอ วิธีการ Level Set Method ที่ได้รับการปรับปรุง โดยใช้ชื่อว่า Hybrid LSM ซึ่งเป็นการรวมคุณสมบัติเด่นของ วิธีการ edge based และ region based active contour model เข้าด้วยกัน

คณะผู้วิจัยได้ทำการทดลองเปรียบเทียบ การทำงานของ Hybrid LSM กับวิธีการ Level Set Method แบบเดิม ซึ่งพบว่า Hybrid LSM สามารถแยกแยะขอบเขตของวัตถุที่ต้องการ ได้ สามารถทำงานได้กับภาพที่มีสิ่งรบกวน และยังคงทำงานได้ดีแม้ว่าเส้นขอบของวัตถุที่ต้องการแยกแยะ จะไม่ชัดเจน นอกจากนี้ Hybrid LSM ใช้จำนวนรอบที่น้อยกว่าในการแยกแยะขอบเขตของวัตถุ แต่อย่างไรก็ตาม Hybrid LSM ยังคงไม่สามารถแยกแยะรายละเอียดส่วนที่เป็นมุมแหลมของวัตถุได้ถูกต้องนัก นอกจากนี้ คณะผู้วิจัยได้ทำการประเมินการทำงานของ กระบวนการแยกแยะโครงสร้างของเส้นเลือด โดยใช้วิธีการประเมินตามวิธีการของ

การแข่งขัน ในโครงการ Carotid Lumen Segmentation and Stenosis Grading Challenge (CLS) จากการทดลองกับชุดภาพถ่ายทางการแพทย์ประเภท CTA จำนวนสิบห้าชุด ผลลัพธ์ที่ได้มีค่าความแม่นยำโดยเฉลี่ย 87.09% ปริมาณโครงสร้างเส้นเลือดที่สามารถแยกแยะได้โดยเฉลี่ย 88.13% และ ปริมาณโครงสร้างเส้นเลือดที่สามารถแยกแยะได้ถูกต้องโดยเฉลี่ย 86.63% ซึ่งแสดงให้เห็นว่าวิธีการที่คณะผู้วิจัยได้พัฒนาขึ้นมีประสิทธิภาพเพียงพอสำหรับการแยกแยะโครงสร้างของเส้นเลือด แต่อย่างไรก็ตาม ค่าความผิดพลาดที่เกิดขึ้น มักจะเกิดขึ้นในระหว่างกระบวนการ ขึ้นรูปโครงสร้างของเส้นเลือด

คำสำคัญ : ภาพถ่ายทางการแพทย์ / Direct volume rendering / Ray Casting algorithm / การแยกชิ้นส่วนเส้นเลือด / Vesselness filter / Level Set Method

ACKNOWLEDGEMENTS

This project could not have been successful without the help of my advisor, Assoc. Prof. Dr. Tiranee Achalakul. She has always given me good advices, contributions and other helps during my study period. Her suggestion is my important motivation which drove me to work until I finished this project. I would also like to thank to all the committee members for their valuable suggestions and comments.

I would like to thank all my teammates, the MYRA team, for the best memories throughout the past two years during my master's degree study. We went through both happy and bad moments together. There were many obstacles along our paths but we fought against them with our teamwork. I owed my success to all of you. Those moments were the best part of my life which I will never forget. They always cheered me up when I was tired. Without you, the past two years could not have been so great.

Lastly, I would like to thank my family. They have always given me the love and morale when I feel dejected and hopeless. I could pass all of the difficulty because they are always beside me.

CONTENTS

	PAGE
ENGLISH ABSTRACT	i
THAI ABSTRACT	ii
ACKNOWLEDGEMENTS	iv
CONTENTS	v
LIST OF TABLES	vii
LIST OF FIGURES	viii
LIST OF EQUATIONS	x
LIST OF SYMBOLS	xii
LIST OF TECHNICAL VOCABULARY AND ABBREVIATIONS	xiii
CHAPTER 1 INTRODUCTION	1
CHAPTER 2 LITERATURE REVIEW	4
2.1 Related Works	4
2.2 Technical Issues	8
2.2.1 Level Set Method	8
2.2.1.1 A Geometric Active Contour Model Based on the Mean Curvature	9
2.2.1.2 Active Contour without Edge	12
2.2.2 Vesselness Filter	17
2.2.3 Center Line Extraction	19
2.2.4 Ray Casting Algorithm	21
CHAPTER 3 THE HYBRID LEVEL SET METHOD	27
3.1 The Ideas of the Hybrid LSM	27
3.2 Development of Hybrid LSM	29
3.2.1 Stopping Function	29
3.2.2 Initial Level Set Function	30
3.2.3 Dirac Delta Function	31
3.2.4 Reinitializing Function	31
3.3 Method Comparison and Evaluation	34
3.3.1 Experiment Methodology	34
3.3.2 Experimental Results and Discussion	38
CHAPTER 4 A FRAMEWORK FOR BLOOD VESSEL IMAGE SEGMENTATION	47
4.1 Input data	48
4.2 Bone Segmentation	51
4.2.1 Initiating the Level Set Function	51
4.2.2 Adjusting the Input Data	54
4.2.3 Applying the Hybrid LSM to Bone Segmentation	55
4.3 Histogram Analysis	58
4.4 Vessel Filtering	60
4.5 Vessel Reconstruction	61
4.6 Integration with Ray Casting Algorithm	63

CONTENTS (Cont.)

	PAGE
4.7 Evaluation Method	65
4.7.1 The Dice Similarity	65
4.7.2 Sensitivity	65
4.7.3 Specification	65
CHAPTER 5 RESULTS AND DISCUSSTION	66
5.1 Experimental Inputs	66
5.2 Visualization of Segmentation Results	66
5.3 Evaluation Results	69
5.4 Discussion	69
CHAPTER 6 CONCLUSION	71
REFERENCES	73
CURRICULUM VITAE	76

LIST OF TABLES

TABLE		PAGE
2.1	Possible patterns in 3D space	18
2.2	Patterns of surrounding neighbor	20
2.3	Hounsfield value for selected tissue types	22
3.1	Characteristic of Hybrid LSM, edge based and region based active contour model	28
3.2	Comparison of stopping function between Hybrid LSM, Edge Based and Region Based active contour model	30
3.3	Comparison of initial level set function between Hybrid LSM, Edge Based and Region Based active contour model	30
3.4	Comparison of Dirac delta function between Hybrid LSM, Edge Based and Region Based active contour model	31
3.5	Comparison of reinitializing function between Hybrid LSM, Edge Based and Region Based active contour model	33
3.6	The constant parameters	35
3.7	The input image, initial level set function and the referenced boundary of the desired object in each experiment	37
3.8	The experimental results	43
3.9	The segmentation results image in each experiment	45
4.1	The constant parameters for Hybrid LSM in bone segmentation process	55
4.2	The constant parameters for vesselness measurement model	61
4.3	The constant parameters for Hybrid LSM in vessel reconstruction process	62
5.1	Information about input datasets.	66
5.2	Visualization of segmentation results and references	67
5.3	Evaluation results	69
5.4	Average of evaluation results	69

LIST OF FIGURES

FIGURE	PAGE
1.1 Carotid arteries	1
2.1 An example of clear boundary input image	10
2.2 An initial surface of Level Set function (left), Contour at zero level of initial surface (right)	11
2.3 The contour at zero level after 500 iteration(a), 1,000 iteration(b), 2,000 iteration(c), and 4,000 iteration(d)	11
2.4 The final surface of Level Set function (left), The segmentation result (right)	11
2.5 Comparison between conventional and modified Dirac delta function	14
2.6 An example of unclear boundaries input image	16
2.7 An initial surface of Level Set function (left), Contour at zero level of initial surface (right)	16
2.8 The contour at zero level after 500 iteration(a), 1,000 iteration(b), 2,000 iteration(c), and 4,000 iteration(d)	16
2.9 The final surface of Level Set function (left), the contour at zero level (right)	16
2.10 The orientation of ideal tubular structure and direction of eigenvectors	20
2.11 The example of neighbor patterns	20
2.12 Diffuse reflection from different light directions	23
2.13 Normalized vector of surface, normalized vector of light, normalized vector in direction of the observer and normalized vector of reflected ray	24
2.14 Interpolation the sampled point.	25
2.15 Composition the sampling point along the ray	26
3.1 Distance Regularized function at steady state (left) and signed distance band (right)	32
3.2 Sequence of Level Set Method.	34
3.3 Iteration to convergence on experiment 1 and 2	44
3.4 Iteration to convergence on experiment 3 and 4	44
3.5 Iteration to convergence on experiment 5 and 6	44
3.6 Iteration to convergence on experiment 7 and 8	44
3.7 Iteration to convergence on experiment 9 and 10	45
4.1 The summarization of our methodology work flow	48
4.2 Schematic of the region of interest	49
4.3 Histogram of the partial volume of carotid lumen from the training datasets	49
4.4 The examples of slice of raw data	50
4.5 Histogram of bone (blue line) and carotid lumen (green line)	51
4.6 The example of slice of medical image	52
4.7 The example of slice after threshold by 1060 (a), and after fill the hole (b)	53
4.8 The example of slice after threshold by 1060 and 1545 (a), and after fill the hold (b)	53
4.9 The result of subtraction between Figure 4.5(b) and 4.6(b)	53
4.10 The example of slice of initial surface for level set function	53
4.11 Depiction of window level	54
4.12 Example of window level adjustment. Left image WC=1000 WW=500 Right image WC=1500 WW=2000	54
4.13 Example of slice of input data by setting WC=2000 WW=2000	55

LIST OF FIGURES (Cont.)

FIGURE	PAGE
4.14 The example of slice of contour at zero level after iteration 25 (a), iteration 50 (b) and iteration 100 (c)	56
4.15 The example of slice of bone segmentation result	56
4.16 The example of slice of original data after bones are eliminated	56
4.17 Summarization of bone segmentation process	57
4.18 Histogram of data after bones are eliminated	58
4.19 Histogram of data after filter by $1060 \leq \text{intensity value}$ and intensity value ≤ 1545	58
4.20 The example of slice of vessels mask	59
4.21 Example of slice of input data by setting WC=1200 WW=800	60
4.22 Scatter plot of data which are justified as vessels	61
4.23 Scatter plot of center line	62
4.24 Example of slice of segmentation output	63
4.25 Scatter plot of segmentation result	63
4.26 Graph of transfer function and opacity mapping (top) and graph of color transfer function (bottom)	64
4.27 Example of visualization result	64
5.1 An example of medical image slide around the bifurcation.	70
5.2 An example of cross section of the segmentation result and reference overlay on the medical image slide	70

LIST OF EQUATIONS

EQUATION	PAGE	
2.1	Edge detector function	9
2.2	Mean curvature	9
2.3	First derivative of ϕ with respect to t based on mean curvature	9
2.4	Initial level set function	10
2.5	Relationship between level set function and segmentation result.	10
2.6	Area of contour at zero level	12
2.7	Length of contour at zero level	12
2.8	Step function $H_0(\chi)$ and Dirac delta function $\delta_0(\chi)$	12
2.9	The energy function of Chan-Vee model	13
2.10	Definition of average intensity value inside the contour	13
2.11	Definition of average intensity value outside the contour	13
2.12	First derivative of ϕ with respect to t	13
2.13	Dirac Delta function	14
2.14	Chan-Vese's Dirac delta function	14
2.15	Chan-Vese's initial level set function	15
2.16	3-Dimension Gaussian function	17
2.17	Hessian Matrix	17
2.18	Element of Hessian Matrix	17
2.19	Order of eigenvalues	18
2.20	The pattern of eigenvalues for a voxel which belongs to the vessel	18
2.21	First measurement to evaluate blob-structure	18
2.22	Second measurement to distinguish between plate-like and line structure	18
2.23	Third measurement to distinguish the background and the foreground	19
2.24	Vesselness measurement model	19
2.25	Multi scale vesselness response	19
2.26	Brightness and contrast adjustment	21
2.27	Relation between density of element and intensity value in Hounsfield unit	22
2.28	Isovalue contour surfaces transfer function	22
2.29	The reflected intensity of ambient light	23
2.30	The reflected intensity of diffuse reflection	23
2.31	The reflected intensity of specular reflection	24
2.32	Phong Shading	24
2.33	Interpolation the sampled point	25
2.34	Composition	26
3.1	First derivative of ϕ with respect to t for Hybrid LSM	28
3.2	Modified stopping function	30
3.3	Conventional reinitializing function	31
3.4	Distance Regularized function	33
3.5	Regularization Term	33
3.6	First derivative of ϕ with respect to t for Distance Regularized Level Set Evolution model	35
3.7	First derivative of ϕ with respect to t for Active Contours without Edges model	35
3.8	The dice similarity	36
3.9	Sensitivity	36

LIST OF EQUATIONS (Cont.)

EQUATION		PAGE
3.10	Specification	36
4.1	Relationship between initial level set function and subtraction result.	52
4.2	Window level	54
4.3	Relationship between segmentation result and level set function.	55
4.4	Bone elimination	56
4.5	within-class variance	59
4.6	Between-class variance	59
4.7	Relationship between vesselness response, input data and vessel mask.	60
4.8	The dice similarity	65
4.9	Sensitivity	65
4.10	Specification	65

LIST OF SYMBOLS

ϕ_0	=	The initial level set function
ϕ	=	Level set function
Ω	=	Image domain
x	=	Specific location of data
$I(x)$	=	Intensity value of data at location x
λ_i	=	Eigen value
v_i	=	Eigen vector
WW	=	Window width
WC	=	Window center
δ_ε	=	Dirac delta function
G	=	Gaussian function
κ	=	Curvature
D_{si}	=	Dice similarity
TP	=	True Positive
FN	=	False Negative
FP	=	False Positive

LIST OF TECHNICAL VOCABULARY AND ABBREVIATIONS

CCA	=	Common carotid artery
ICA	=	Internal carotid artery
ECA	=	External carotid artery
PCI	=	Percutaneous Coronary Intervention
MRI	=	Magnetic Resonance Imaging
CT	=	Computed Tomography
CTA	=	Computerized Tomographic Angiography
DVR	=	Direct volume rendering
IVR	=	Indirect volume rendering
CLS	=	Carotid Lumen Segmentation and Stenosis Grading Challenge
LSM	=	Level Set Method
GPU	=	Graphics processing unit of a computer
HU	=	Hounsfield unit
3D	=	Three dimensions
2D	=	Two dimensions
SDB	=	Signed distance band

CHAPTER 1 INTRODUCTION

Occlusive vascular disease is one of the most critical diseases nowadays. More than ten million people around the world are killed by this every year [1][2]. And in the next five years, the number of victims will reach twenty million. More than fifty percent of the deceased lived in developing countries and southern Europe. One-third of the occlusive vascular disease patients displays no symptom until an acute heart failure. One of the important arteries and risks for this disease is a carotid artery. This artery is responsible for supplying the brain with oxygen blood. It starts from the heart passing through the neck and connecting to the brain. There are two common carotid arteries (CCA) in our body, left and right. Left CCA arises from the aortic arch; the right arises from the brachiocephalic artery [3]. Both of them bifurcate into an internal carotid artery (ICA) and external carotid artery (ECA), as illustrated in Figure 1.1. ICA delivers blood to the brain while ECA delivers blood to the other organs in the head. We can feel their pulsation by touching below the jaw bone [4]. There is only 1 percent of risk for people aged between 50 and 59 to have the occlusive vascular disease. However, the risk is 10 percent higher for the elderly aged between 80 and 89 [5].

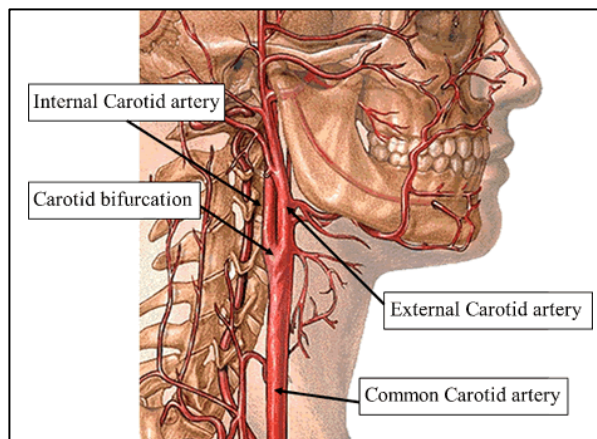


Figure 1.1 Carotid arteries

[Source: <http://thomasquinlan.com/2014/04/on-medical-issues/>]

At present, there are many modern techniques such as bypass surgery and Percutaneous Coronary Intervention (PCI) which are moderately successful. Medical image is one of the important tools for the analysis and diagnosis for occlusive vascular disease. It gives cross-section images of a human body by several imaging techniques. One of the most popular imaging techniques is Magnetic Resonance Imaging (MRI) and Computed Tomography (CT). Medical images are very useful in the medical field. Not only does it help doctors to diagnose and analyze the diseases accurately, but it is also used to plan treatment strategies. However, visualization of the blood vessel by taking medical images is more complicated than that of other organs. That is because blood vessels are transparent; they cannot be captured by using regular imaging techniques. The doctor has to inject the contrast media into the patient's blood vessels which is called angiography. Computerized tomographic angiography (CTA) is one of the medical imaging techniques which combines conventional CT scan and angiography. This technique is used to capture the detailed images of blood vessels.

Computer technology is a pre-requisite for the development of medical technologies. Image processing can be used to develop the visualization of medical image. The

medical visualization is the combination between medical image and image processing technique. It displays a part of human body in three-dimension (3D) model imaging which gives both doctors and patients an insight into medical data. In addition, medical visualization also presents details in aspects such as data exploration. There are two main techniques of medical visualization – direct volume rendering (DVR) and indirect volume rendering (IVR). DVR visualizes the 3D model images by using the specific transfer function to classify the medical image data, maps to opacities and shading colors, while IVR renders the polygonal model from the medical image data.

The employment of direct volume rendering for medical visualization could assist doctors by visualizing the vessel structure of occlusive vascular disease patients. Doctors can clearly address the stenosis points in the blood vessel structure, so they can diagnose and plan an operation with better accuracy. The system would reduce the radiologist's workload and increase the effectiveness of the treatment, because the doctor can find the anomaly area in the early stage. It is also helpful for surgeons who wish to avoid damaging the vascular structure or plan to reconstruct it. In addition, it also reduces the expenses of treatment since the cause of disease can be found at the early stage. Not only can the direct volume rendering visualize the blood vessel structure, it can also be adapted for organ segmentation which allows the doctor to visualize the anomaly in each organ easily.

Although the direct volume rendering is a useful technique for the medical field, there are some obstacles of implementation of DVR in clinical applications. The first one is the complexity of transfer function configuration. In organ segmentation, it is hard to configure the appropriate parameter for transfer function. The transfer function with a suitable parameter can generate the noiseless and low artifact three-dimension (3D) model image and it can clearly segment the organ. The transfer function should be adapted for each data set, which can be complicated for users. Moreover, blood vessels are a type of organ which cannot be segmented by using regular transfer function. Due to the complexity of blood vessel structure, which looks like a tree structure, it can be mistaken with other organs with similar intensity, the variety of vessel size and intensity, artifact and the contrast of vessel in medical image. Even though the CTA imaging technique permits us to identify blood vessels more clearly, the range of intensity of blood vessels still overlaps with the range of intensity of bones. It is impossible to segment the blood vessel by using the conventional thresholding technique. The second obstacle is the computation time of DVR. As the medical visualization is generated in high fidelity, the high resolution medical images are required. The high performance rendering algorithm should be performed with high computation time in the case of high quality image. Fortunately, this obstacle can be solved by using the parallelization computing which can improve the processing performance of Ray Casting algorithm, so the high quality image is generated in a few seconds.

Our research aims to automatically segment the blood vessels without any user interaction and any initial points from the user and visualize them in 3D model image. We acquire the medical image datasets from Carotid Lumen Segmentation and Stenosis Grading Challenge (CLS). They provide the datasets of the Carotid Bifurcation in a Computed Tomography Angiography (CTA). So, our purpose is to segment the carotid artery from the CTA images and evaluate our results by using the frameworks provided

by the organization. Moreover, we can compare our results with other authors and other methods.

In order to display the blood vessels structure in high quality 3D model images, this research proposes a framework to segment the location of blood vessels in the medical images by using the Level Set Method and vesselness function. The Level Set method is opted to distinguish between bones and vessels while the vesselness function is responsible for evaluating how likely the voxel in the datasets is to belong to the vessels structure. And for the visualization purpose, this research also proposes a framework to integrate the result from our solution with the Ray Casting algorithm.

Moreover, we propose the modified version of Level Set Method which is called Hybrid LSM. This method is the combination between edge based and region based active contour model. The Hybrid LSM not only preserves the advantages of both conventional methods but also improves the limitation and performance of LSM.

The rest of the paper is organized as follows. The next chapter reviews other published works about Ray Casting technique and vessel segmentation and describes the details of the proposed technique for this research, for example Level Set method, vesselness function, center line extraction and Ray Casting. In Chapter 3, we introduce our proposed method, Hybrid LSM, and describe the details of ideas and development. In addition, the benchmark results of our proposed method are also presented. Chapter 4 describes the details of our framework in a step-by-step manner and framework design. The results from our framework, evaluation method and evaluation results are shown in Chapter 5. Finally, in Chapter 6 the authors' conclusions are presented to sum up this research.

CHAPTER 2 LITERATURE REVIEW

In this section, a survey of related works is presented. We research the medical image visualization techniques, the useful techniques in order to visualize the medical image data in 3D model image, the solution for improvement of the computation speed and the quality of image. Besides, in order to develop the visualization of organ segment especially blood vessel structure, we research the vessel segmentation techniques. The details, advantages, and results of those researches are discussed in 2.1. In 2.2, we describe this noteworthy methodology to be adopted in our research.

2.1 Related Works

Nowadays, digital image data collection is useful for diagnosing diseases and planning treatment strategies. Even though the volume visualization can help clinicians in creating many utilities from the medical image data, there are many obstacles that obstruct the usability of the volume visualization in clinics. One of the obstacles is fidelity. Medical image requires high fidelity. It means that high performance rendering algorithm should be performed in the case of high quality image. Smelyanskiy et. al.[6] investigated the kind of volume rendering algorithm that is appropriate for modern parallel processing architectures. First, they compared three different methods of Direct Volume Rendering, which were Ray-Casting, Splatting, and Shear-Warp. In order to determine which method could generate a high fidelity medical image, they applied these three methods to render the high-resolution CT Angiography data. Each method should generate the image of the inferior epigastria artery (IEA). The image quality was determined by the visibility of the branches of the IEA and the sharpness of the vessel. They concluded that Ray-Casting method had the most acceptable quality for medical image processing. Furthermore, they evaluated the medical-quality from the result of Ray-Casting method on CPU (Intel Nehalem), GPU (Nvidia GeForce GTX280) and many-core architecture (Intel Larrabee). From the evaluation, the implementation of Ray-Casting on the quad-core Nehalem could improve the rendering speed up to 5.8x. The speed on Nvidia GeForce GTX280 can achieve 5x-8x. For the 16-core Larrabee, they could reach 10x speed-up over the single core Harpertown. For the 32- and 64-core Larrabee, the rendering speed was only 11x faster due to the overhead of data transfer. For the ideal speed-up, which, if the overhead was ignored, the 32- and 64-core Larrabee should achieve the 24x and 42x speed-up, respectively. In conclusion, we realize that Ray-Casting method can be the most appropriate medical image processing. Moreover, the parallel processing can help us improve the performance of volume visualization significantly.

Not only can the parallelization of Direct Volume Rendering method improve the rendering speed, it can also increase the image quality. In order to increase the sampling speed by reserving the image quality, Lee et. al. [7] proposed a technique that adopts the Volume Ray-Casting with Virtual Samplings. The traditional Volume Ray-Casting requires more sampling rate to increase the image quality and that also increases the computation cost. The Virtual Sampling technique is an approach to increase the sampling rate without increasing the computation cost by using a cubic spline curve interpolation. For the conventional ray-casting, a ray which is cast by image pixel transverses the volume data and samples the intensity. For the virtual sampling technique, instead of using the actual sampling points, it generates the virtual sampling points by using the cubic function to interpolate the two intensities. The virtual sampling is performed at one-fourth of the actual sampling interval, so it can increase

the sampling rate by four times. Although the cubic splines can be used for the virtual sampling point interpolation, it does not support the ‘local control’. Cubic splines use all of the input control points to formulate the spline curve. Therefore, the experimenters presented the Catmull-Rom splines which supported the local control properties. The Catmull-Rom splines can interpolate the curve by using piecewise cubic polynomials. The researchers compared the virtual sampling technique with the cubic sampling at a unit-voxel size interval (sampling rate 1x) and cubic sampling with a sampling rate 4x. The results showed that the cubic sampling with sampling rate 1x was the fastest rendering speed but it exhibited the highest amount of artifact. The virtual sampling technique had a faster rendering speed than the cubic sampling with a sampling rate 4x while the images qualities were equal. Thus, the virtual sampling technique can significantly improve the rendering speed. However, the disadvantage of the virtual sampling technique is the intensities of virtual sampling point can be out of range of the input volume data as the minimum and maximum block is required to control the virtual sampled intensities. However, the blocks can generate the loosely-bound minimum and maximum values which can introduce potential faults to the rendering performance.

Another technique that can improve the quality of image is a peak finding method proposed by Knoll et. al [8]. This method finds the peak within the ray and then creates the 2D look up table in order to replace the peak intensity to the integrated value. It is appropriate for a transfer function that has sharp features, such as the Dirac impulses. It can classify the noise data that the post-classification and the pre-integration cannot handle. Nevertheless, the peak finding method is more costly than the pre-integration method and it is also difficult to implement outside a Ray-Casting framework. However, the image quality of this technique will resemble the one from the conventional DVR if the transfer function is smooth.

In addition, Wang et. al. [9] proposed the hybrid strategy for volume rendering. This strategy uses the octree structure to organize the volumetric data and was implemented on the multiprocessor. The octree structure is used to index the data. Each parent node in the octree has eight children and contains the range of voxel value. Thus, the leave nodes of octree point to a voxel. In order to manage the data space, they implemented the Branch No-Need Octree (BONO), which can avoid allocating nodes of an empty subtree. The researchers were transversing the BONO in order to find the nodes that contained the range to cover the isovalue and pushed them into the list. Each list represented the ray which was cast by the image pixel. The experimenters suggested that the list of any pixel should contain fewer than 20 items. Then, they composted the voxels in each list similar to the conventional ray casting. For the implementation on the multithread processor, they divided the data into the equal regions; each processor was responsible for transversing the BONO and creating the list. Instead of separating the list for each image pixel, they created a list for four rays (2x2 pixel) since they found that the smaller number of lists could improve the performance of image processing. After the lists were prepared, they divided the screen into small images and applied the load balancing technique by mapping the image-tile with the equal number of processors and creating the sequence list which contains the pairs of image-tile and processor. Each processor would get the image-tiles index from the sequence list and preformed the ray casting by casting the ray through the list and composting the voxel values in the list. The process would be performed continuously until the sequence list would be empty. The advantage of this technique is that it can eliminate almost all

pixels that are nonintersecting the ray segment and create a shorter list for the remaining pixels.

The Ray-Casting method usually involves two issues. The first issue is that the opaque features might be alias with some other parts. The second issue is the under- and over-saturated result images which arose from the additive blending technique in compositing process. These two issues usually occur when an inappropriate transfer function was set. Marchesin et. al. [10] proposed the Per-Pixel technique. This technique is noteworthy because it does not use the transfer function and introduces the volume virtualization in a new aspect. The result of this method presented the visibility of the internal structure of the data set. They introduced the relevant function used in identifying the opacity of each voxel in data set instead of using the transfer function. The relevant function determines the opacity of each voxel by its gradient value. The high gradient area is more important than the regions with low variance (low gradient). The threshold of the gradient value is utilized as relevant function. The advantage of this technique is that it enables the data exploration concept that can represent the result in a different aspect. In addition, it is fully automatic choosing the appropriate relevant functions, which leaves user manipulation unnecessary. However, the result from this technique is a non-photorealistic that may not be useful in terms of clinical practice.

The organ segmentation is one of the useful features for visualization by using Direct Volume Rendering. However, it is hard to clearly segment the specific characteristic organs such as blood vessels by using the conventional transfer function, due to the complexity of the blood vessel structure that looks like the tree structure. They may be mistaken with other organs with similar intensity, vessel size and intensity, artifact and contrast of vessel in medical images. There are many advantages of blood vessel segmentation. The first one is the vessel structure visualization. Doctors can visualize the structure of blood vessel which allows them to address the stenosis points. Moreover, it can help with operation planning. They can avoid damaging the vessel structure or reconstructing them with this technique.

Marín et al. [11] reported that vessel segmentation method can be categorized into two main groups; rule-based method and supervised method. The rule-based method uses a mathematical model to search and match the characteristics of blood vessel while the supervised method is based on pixel or intensity classification. Marín et al.'s research presented a new supervised method by using gray-level and moment invariants based approach. Their research aimed to segment the blood vessel in retinal image of diabetic patient, in order to detect the diabetic retinopathy (DR) in early stages. Their method used the pixel classification technique and 7-D vector to extract vector from the medical image and send to a neural network. The neural network would determine which pixel was vessel. Finally, the pixel of output image was painted as a blood vessel. The knowledge about blood vessel location and characteristic is also required for the neural network training which helps to reduce the false positive in detection. For the method evaluation, their research used a dataset from the publicly available DRIVE and STARE database. The medical images from DRIVE database were used for training the neuron networks and the dataset from STARE to test the accuracy of the method. The experimenter found that the accuracy was improved from 0.9452 for DRIVE database to 0.9526 for STARE database. In addition, this method provided high accuracy results when compared with the previously-published vessel segmentation method. They found that the performance of the previous method dropped when tested with an unseen

database while the proposed method still provided a good performance. It proves that the proposed method is robust for any unseen dataset. The total computing time for one medical image is one minute and thirty seconds approximately.

On the other hand, many researchers have developed the rule-based model for vessel structure extraction. Shang et al. [12] proposed a new technique for the vessel segmentation by adopting the rule-based method. Their research reported that active contour models were the most successful image segmentation technique. It could be categorized into three types; edge based, region based and higher level knowledge based. Their research introduced a region competition-based active contour model which is used to segment thick vessels. For thin and weak vessels, they defined the vascular vector field. Finally, their method also used curvature strategy to smooth the surface of the vessel. This method was developed to extract the vessel tree in the lung, liver and coronary artery from high solution medical images such as MRI and CT scan. For the method evaluation, the experimenters compared their method with the classic active contour models. There are GAC, CURVES and C-V. The test dataset was the simulated cylindrical body connected with five sizes of tubes. GAC and CURVES could extract only three out of five tubes while the C-V could not segment the thin tube and the proposed model could segment all five tubes. The experimenter also tested the model with real dataset which was the vessel tree of liver images. The proposed model and C-V could extract the small artery in high details while GAC and CURVES model performed poorly. The experimenter discussed that because GAC and CURVES are edge-base models, their performance dropped for weak edges of thin and weak vessels. C-V and the proposed model are region-based active contour models so, they used the intensity of the vessel to detect the boundary which were robust for small vessels. Moreover, the proposed model had higher accuracy than the C-V model because it adopted the vascular vector field to detect the thin vessel. However, the computation speed of proposed model is slower when compared with the other model. It uses 8 minutes for 120 slice volume.

Zou et al. [13] is the one who researched about the vessel segmentation by using the rule-based method. They proposed a consecutive scanline tracking method which was fitted by using the vessel profiles such as vessel central point, radius, edge, location and direction. An adaptive tracking strategy was adapted with this model to determine the appropriate stopping criteria. Besides, the models also use look-ahead detection scheme to detect the continuous point of the same vessel. The proposed algorithm aimed to extract the vessel network automatically, without any human interaction and initial starting point. And it generated the patient's vessel model by using the geometric information. The consecutive scanline is a significant method for the tracking process. Their research used five scanlines to detect central points of the blood vessel. Those central points would be fitted by a quadratic function which could calculate the estimated vessel direction. The tracking process might stop when it was disrupted by noise and poor image quality so, it could not do the tracking until the end of the vessel. This research developed termination criteria from the knowledge about blood vessel characteristics in order to help the tracking process determine the appropriate stopping point. Although the consecutive scanline process was stopped by termination criteria, it does not mean that it reached the end of the vessel. This research solved this problem by adopting the look-ahead detection scheme (LAD). It was used to search the possible continuation points in the same vessel segment. The researcher evaluated their method by testing it with the synthetic data and comparing it to the previous method. They

reported that their method could provide higher accuracy than the other algorithm and work well with 2-D subtracted X-ray angiograms (DSAs) of low image quality.

In conclusion, as we mentioned earlier, the volume visualization for the medical image processing is a useful application for clinicians. The medical image requires the fidelity which depends on the quality of image and volume rendering algorithm. The Ray-Casting method is one of the direct volume rendering (DVR) methods that can provide a high quality image. However, the quality of image is usually compromised with the performance. Thus, we would implement the Ray-Casting method on the GPU-based CUDA architecture, which is a parallel computing platform created by NVIDIA. It allows us to program the multiprocessing units on the graphic card which can improve the processing performance of Ray-Casting. Another interesting technique is virtual sampling. This technique generates the virtual sampling points from the actual sampling points by using the Cubic splines interpolation. It enables us to increase the sampling rate without increasing the computation cost. Moreover, the rule-based method for vessel segmentation is also interesting. This technique generates the model fitted by using a vessel profile. It enables us to extract the blood vessel structure from the medical image dataset. Besides, it can detect even the weak vessel in the image and handle the noisy and poor quality image.

Therefore, we propose to develop a framework to segment the location of blood vessels in the medical images by using the Level Set Method and vesselness function and apply the Ray Casting algorithm on the segmentation results for the visualization purpose. Another intention is that we would develop the modified version of Level Set Method in order to improve the performance and limitation of the conventional LSM. In addition, we propose to evaluate our framework and modified version of LSM based on the evaluation criteria from Carotid Lumen Segmentation and Stenosis Grading Challenge (CLS).

2.2 Technical Issues

In this section, we will discuss the most important methodology for this research. In order to segment the blood vessels in medical images, we apt for the level set method (LSM) to distinguish the bones and the vessels in CTA images. We describe the detail of this technique in Section 2.2.1. After the bones are eliminated, we are to discriminate the blood vessels from other tissues. The vesselness filter is employed to evaluate how likely it is for a voxel to belong to the vessel structure. We discuss about this technique in Section 2.2.2. Next, in Section 2.2.3, we discuss about center line extraction used to generate the initialization of vessels. Moreover, in order to implement the direct volume rendering technique, we are interested in Ray Casting technique. We describe the detail of this algorithm in Section 2.2.4.

2.2.1 Level Set Method

Level Set Method (LSM) was introduced in 1998 by Osher and Sethian [14]. This technique is used to identify the boundary of the objects in the image and distinguish between foreground and background. At present, many research works are applying this technique to the medical images to segment the organs boundary [15, 16, 17].

The concept of this method is representing the boundary of the objects by using the contour of the surface at zero level. This surface is called level set function. This method runs iteratively, in order to evolve the surface until the contour at zero level

reaches the specific boundary. The evolution of surface depends on the forces which drive the surface to expand and shrink. The development of surface directly affects the contour shape to split and merge.

In this section, we discuss about two types of Level Set Method. There are edge based and region based active contour models. In order to describe the equations and explain how the model works, we exemplify the geometric active contour model based on the mean curvature motion to represent the edge based active contour model and active contour without edge to represent the region based active contour model. Moreover, we show the examples of results and discuss advantages and disadvantages of both techniques.

2.2.1.1 A Geometric Active Contour Model Based on the Mean Curvature

In the classical active contour model, the evolution of model is based on the edge of the object. The edge of the object is used to stop the evolution of level set function when the contour at zero level reaches the boundary of the desired object. The edge detector function depends on the gradient of the image. Equation 2.1 shows the example of edge detector function, where $I(x)$ represents the intensity value of input image at location x and $G(x, \sigma)$ represents the Gaussian function. The gradient of the image $|\nabla G(x, \sigma) * I(x)|$ is high at the boundary of the object where the intensity value of the object and intensity value of background are different. So, $g(x)$ becomes closer to zero at the boundary of the object and closer to one inside the object.

$$g(x) = \frac{1}{1 + |\nabla G(x, \sigma) * I(x)|}$$

where G is the Gaussian function.

$I(x)$ is the intensity value of an input image at location x .

Equation 2.1 Edge detector function

$$\kappa(x) = \text{div} \left(\frac{\nabla \phi(x)}{|\nabla \phi(x)|} \right)$$

where ϕ is the level set function.

Equation 2.2 Mean curvature

$$\frac{\partial \phi}{\partial t} = g |\nabla \phi| \left(\text{div} \left(\frac{\nabla \phi}{|\nabla \phi|} \right) + v \right)$$

where g is the edge detector function.

v is the constant parameter.

Equation 2.3 First derivative of ϕ with respect to t based on mean curvature

The mean curvature κ is defined by Equation 2.2. $\kappa(x)$ represents the curvature of the level set function ϕ which passes through the location x .

The evolution of level set function which is based on the mean curvature motion is defined by Equation 2.3 where the mean curvature κ is the main force which drives the evolution of the level set function while g is the stopping function used to terminate the evolution when the contour at zero level reaches the boundary of the desired object.

v is the parameter which controls the direction and speed of evaluation. The sign of this parameter affect the evolution of the level set function. If v is negative, the surface tends to shrink. On the other hand, if v is positive, the surface tends to expand. In addition the magnitude of v affects the propagation speed.

The gradient of level set function $|\nabla\phi|$ is used to control the evolution response around the border of contour at zero level. Only the border of the contour at zero level evolves in each iteration while the surface inside and outside the contour does not take the evolution effect. This term enables the model to control the growth of the contour and support the stopping function to prevent the contour to leak out the desired boundary by terminating the evolution process when the contour reaches the desired boundary.

Setting the initial level set function, ϕ_0 appropriately is an important step for edge based active contour model. Since the stopping function prevents the contour at zero level from leaking out of the desired boundary, the initial contour of level set function should be inside or surround the boundary of the desired object. Equation 2.4 shows the definition of initial level set function, where c_0 is the constant value.

$$\phi_0(x) = \begin{cases} c_0 & \text{if } x \text{ is the location of initial point} \\ -c_0 & \text{otherwise} \end{cases}$$

where c_0 is the constant parameter.

Equation 2.4 Initial level set function

An example of input image is shown in Figure 2.1. It is a white spiral image with a gray background. The initial level set function and initial contour at zero level are shown in Figure 2.2. Figure 2.3 shows the results of the geometric active contour model based on the mean curvature after 500, 1,000, 2,000 and 4,000 iterations, respectively. Figure 2.4 shows the segmentation result and the level set function after the level set method finishes. For this example, we set the constant parameters $v=0.5$.

$$I_{output}(x) = \begin{cases} 1 & \phi(x) \geq 0 \\ 0 & \phi(x) < 0 \end{cases}$$

where ϕ is the level set function.

Equation 2.5 Relationship between level set function and segmentation result.

Equation 2.5 defines the relationship between level set function and segmentation result. Where $I_{output}(x)$ represents the intensity value of segmentation result at position x .



Figure 2.1 An example of clear boundary input image

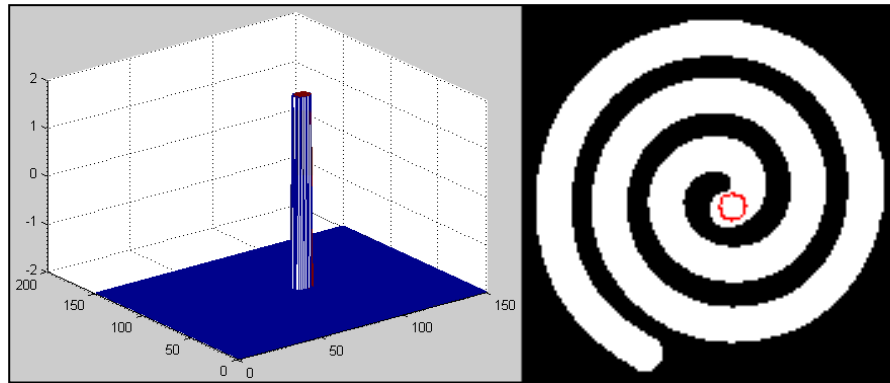


Figure 2.2 An initial surface of Level Set function (left),
Contour at zero level of initial surface (right)

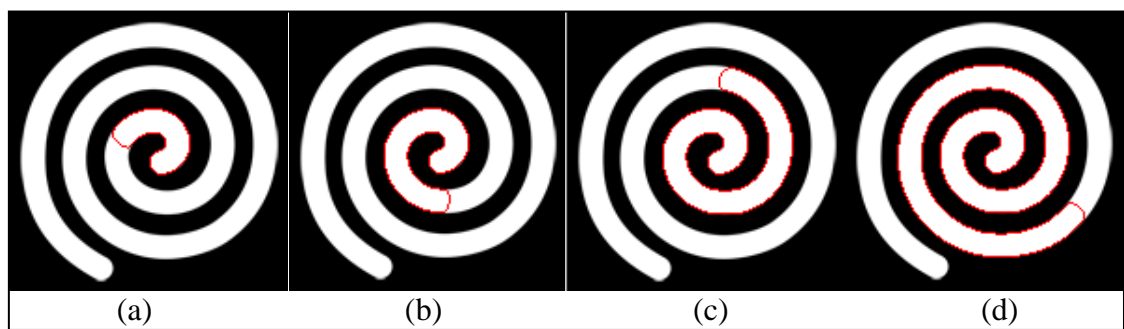


Figure 2.3 The contour at zero level after 500 iteration(a), 1,000 iteration(b),
2,000 iteration(c), and 4,000 iteration(d)

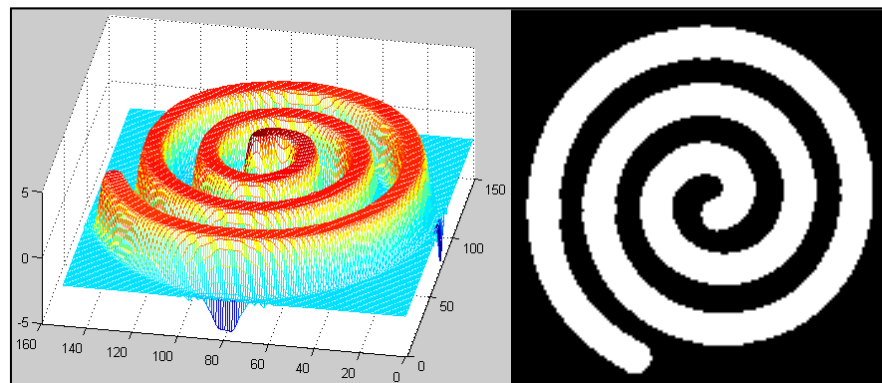


Figure 2.4 The final surface of Level Set function (left),
The segmentation result (right)

The advantage of this technique is that it can be used to segment only the boundary of the desired object by setting the appropriate initial level set function. As this model applies the edge detector as a stopping function and the gradient of level set function to control the evolution at the border of contour at zero level, the evolution process can be terminated when the contour at zero level reaches the desired boundary. So, we can use this technique to segment only the desired object, even though there are many objects on the foreground image.

However, this technique is not appropriate for the segmentation of objects which have unclear boundary. As the stopping function g depends on the gradient of the image, the unclear boundary causing the stopping function is not closer to zero on the edges of the object and it cannot be used to terminate the evolution of level set function. In addition,

this technique is not robust to the noise. If the input image is very noisy, the Gaussian filter has to be strong to remove the noise; in so doing, the object edges are also removed.

2.2.1.2 Active Contour without Edge

The active contour without edge technique is one of the most widely used level set methods. This technique still works well even though the edge of the segmented object is unclear.

Before covering the active contour without edge technique, it is appropriate to introduce the notations that are used throughout this section

ϕ represents the level set function

Ω represents the image domain. In this term, the image domain is 2-Dimension for one image and 3-Dimension for the whole dataset.

x represents the specific location: (x,y) for 2D and (x,y,z) for 3D

The area of the contour at zero level, $\phi \geq 0$, is defined by Equation 2.6 and length of edge around the contour, $\phi = 0$, is defined by Equation 2.7.

$$Area\{\phi \geq 0\} = \int_{\Omega} |\nabla H_0(\phi(x))| dx$$

Equation 2.6 Area of contour at zero level

$$\begin{aligned} Length\{\phi = 0\} &= \int_{\Omega} |\nabla H_0(\phi(x))| dx \\ &= \int_{\Omega} \delta_0(\phi(x)) \nabla \phi(x, y) dx \end{aligned}$$

Equation 2.7 Length of contour at zero level

Where $H_0(\chi)$ is step function which is defined by Equation 2.8. And Dirac delta function $\delta_0(\chi)$ is first derivative of $H_0(\chi)$ with respect to χ

$$H_0(\chi) = \begin{cases} 1 & , \chi \geq 0 \\ 0 & , \chi < 0 \end{cases} \quad \delta_0(\chi) = \frac{d}{d\chi} H_0(\chi)$$

Equation 2.8 Step function $H_0(\chi)$ and Dirac delta function $\delta_0(\chi)$

Active contour without edge is proposed by Chan-Vese [18]. This technique focuses on minimizing an energy function by using the region properties. The energy function is shown in Equation 2.9.

$$\begin{aligned}
E_\varepsilon(c_1, c_2, \phi) &= \mu \cdot E(\phi) + v \cdot A(\phi) + \lambda \cdot P(c_1, c_2, \phi) \\
E_\varepsilon(c_1, c_2, \phi) &= \mu \int_{\Omega} \delta_\varepsilon(\phi(x)) \nabla \phi(x) dx + v \int_{\Omega} H_\varepsilon(\phi(x)) dx \\
&\quad + \lambda_1 \int_{\Omega} (I(x) - c_1)^2 H_\varepsilon(\phi(x)) dx \\
&\quad + \lambda_2 \int_{\Omega} (I(x) - c_2)^2 (1 - H_\varepsilon(\phi(x))) dx
\end{aligned}$$

Equation 2.9 The energy function of Chan-Vee model

$$c_1(\phi) = \frac{\int_{\Omega} I(x) H_\varepsilon(\phi(x)) dx}{\int_{\Omega} H_\varepsilon(\phi(x)) dx}$$

Equation 2.10 Definition of average intensity value inside the contour

$$c_2(\phi) = \frac{\int_{\Omega} I(x) (1 - H_\varepsilon(\phi(x))) dx}{\int_{\Omega} (1 - H_\varepsilon(\phi(x))) dx}$$

Equation 2.11 Definition of average intensity value outside the contour

Where $I(x)$ is the intensity value of input image at position x , c_1 and c_2 are the average intensity value inside and outside the contour respectively. The definition of c_1 and c_2 are defined by Equation 2.10 and 2.11. μ , v , λ_1 and λ_2 are the constant parameters which control the sensitivity of each term.

The energy function consists of three terms, Edge term $E(\phi)$, Area term $A(\phi)$ and Propagation term $P(c_1, c_2, \phi)$. Minimization of this energy function means that we would like to find the area where the deviation of intensity value inside and outside the contour is minimized while the area and the length of edge of this area are also minimized.

By solving the Euler-Lagrange equation, we obtain the equation which explains the evolution of level set function over the time as defined in Equation 2.12.

$$\frac{\partial \phi}{\partial t} = \delta_\varepsilon(\phi) \left[u \operatorname{div} \left(\frac{\nabla \phi}{|\nabla \phi|} \right) - v - \lambda_1 (I - c_1)^2 + \lambda_2 (I - c_2)^2 \right]$$

where δ_ε is the Dirac delta function.

ϕ is the level set function.

I is the intensity value of input image.

u, v, λ_1 and λ_2 are the constant parameters.

c_1 is the average intensity value inside the contour.

c_2 is the average intensity value outside the contour.

Equation 2.12 First derivative of ϕ with respect to t

$$\delta_\varepsilon(\chi) = \begin{cases} \frac{1}{2\varepsilon} [1 + \cos\left(\frac{\pi\chi}{\varepsilon}\right)] & |\chi| \leq \varepsilon \\ 0 & |\chi| > \varepsilon \end{cases}$$

where ε is the constant parameter.
 χ is the input value.

Equation 2.13 Dirac Delta function

Since v is the parameter which controls the sensitivity of Area term, the sign of this parameter affects the evolution of the surface. If v is negative, the surface tends to expand. On the other hand, if v is positive, the surface tends to shrink.

Dirac delta function $\delta_\varepsilon(\phi)$ is used to stimulate the evolution response around the border of contour at zero level. That means the area around the border of contour takes the evolution effect more than other parts of the surface. In the conventional Dirac delta function, only the border of the contour at zero level evolves in each iteration while the surface inside and outside the contour does not take the evolution effect, as defined in Equation 2.13. Chan-Vese [18] introduce the Dirac delta function which allows the surface inside and outside the contour to take the evolution effect. He suggests that the modified Dirac delta function allow the model to obtain the global minimizer. The modified Dirac delta function is defined by Equation 2.14. Figure 2.5 depicts the comparison between conventional Dirac delta function and modified Dirac delta function where d1 (solid line) represents the conventional Dirac delta function which $\varepsilon=1.5$ and d2 (dash line) represents Chan-Vese's Dirac delta function which $\varepsilon=1.5/\pi$. Notice that the value of conventional Dirac delta where $|\chi| > \varepsilon$ are equal to zero, while the value of modify Dirac delta where $|\chi| > \varepsilon$ are more than zero.

$$\delta_\varepsilon(\chi) = \frac{1}{\pi\varepsilon} \left(\frac{1}{1 + \left(\frac{\chi}{\varepsilon}\right)^2} \right)$$

where ε is the constant parameter.
 χ is the input value.

Equation 2.14 Chan-Vese's Dirac delta function [18]

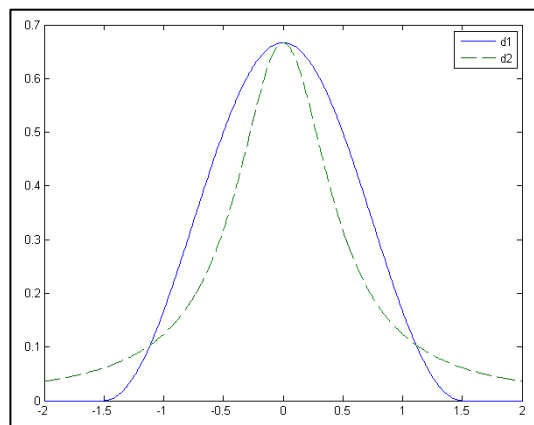


Figure 2.5 Comparison between conventional and modified Dirac delta function [19]

As u is the parameter which controls the weight of Edge term, the value of u affects the characteristic of the model. If u is large, the model will try to strictly minimize the length of boundary of the contour.

The Propagation term is the major force which drives the evolution of surface. This term controls the characteristic of the model which attempts to minimize the deviation of intensity value inside and outside the contour at zero level.

Setting Initial level set function is not strict for the region based active contour model. As the model does not base on the boundary of the object, we can set the initial contour of level set function everywhere on the image. At the end of the process, the model can specify the boundary of all objects in the foreground image. Chan-Vese introduces the initial level set function which does not need the reinitializing process while we perform the level set method. The conventional level set function uses the reinitializing process to prevent the level set function from becoming too flat. Equation 2.15 shows the definition of Chan-Vese's initial level set function, where x_0 and y_0 are the location of the center point and r_0 is the radius of the initial contour of level set function. Notice that there is no flat surface over the initial level set function.

$$\phi_0(x, y) = -\sqrt{(x - x_0)^2 + (y - y_0)^2} + r_0$$

where ϕ_0 is the initial level set function.

x_0 and y_0 are the location of the center point.

r_0 is the radius.

x and y are the specific location.

Equation 2.15 Chan-Vese's initial level set function

An example of input image is shown in Figure 2.6. It is an image of two balls of which the boundary is unclear. The initial level set function and initial contour at zero level are shown in Figure 2.7. Figure 2.8 shows the results of active contour without edge after 500, 1,000, 2,000 and 4,000 iterations, respectively. Figure 2.9 shows the segmentation result and the level set function after the level set method finishes. For this example, we set the constant parameters $\nu = -0.5$, $\mu = 1$, $\lambda_1 = \lambda_2 = 1$ and $\varepsilon = 1.5/\pi$. The relationship between level set function and segmentation result is defined in Equation 2.5.

The advantage of this technique is that it can define the object boundary even though the boundary is unclear or there is no edge around the object, as shown in Figure 2.6. This is because this technique uses the information based on the region properties instead of the stopping function based on the gradient of image. In addition, this technique is also robust to noisy images. Moreover, the initial level set function can be set everywhere on the image. Since this model allows the contour at zero level to grow over the image domain, the model can find the suitable boundary of the object of which the deviation of intensity value inside and outside the contour is minimized

However, this technique is not appropriate to segment the desired object in the image which has many objects in foreground. Due to the fact that the active contour without edge focuses on minimizing the energy function by using the region properties, if we perform this method until the end or meet the convergence point, this method will segment all objects in the foreground, rather than segment only the desired object. There is no appropriate parameter used to terminate the evolution of level set function when the contour at zero level reaches the desired boundary.

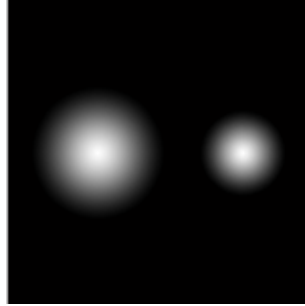


Figure 2.6 An example of unclear boundaries input image

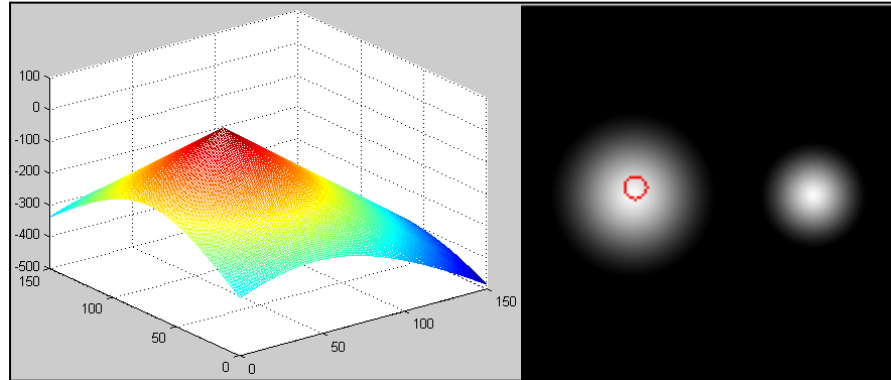


Figure 2.7 An initial surface of Level Set function (left),
Contour at zero level of initial surface (right)

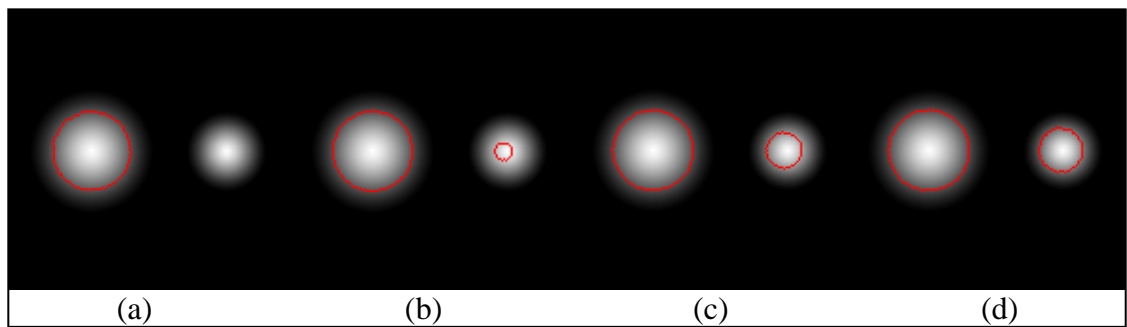


Figure 2.8 The contour at zero level after 500 iteration(a), 1,000 iteration(b),
2,000 iteration(c), and 4,000 iteration(d)

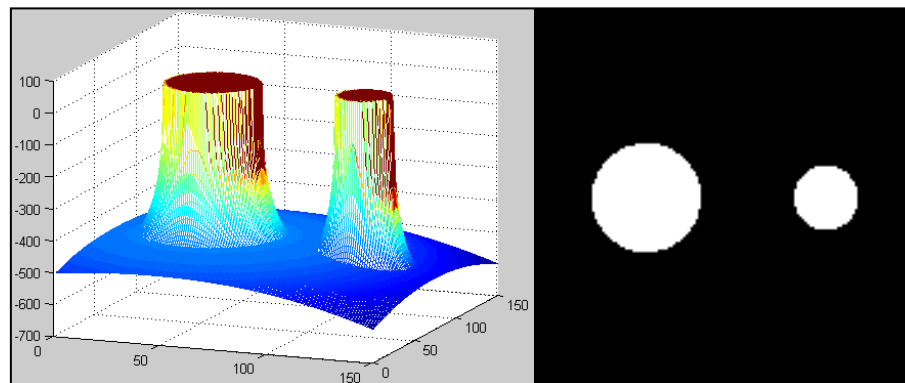


Figure 2.9 The final surface of Level Set function (left),
the contour at zero level (right)

2.2.2 Vesselness Filter

The vesselness filter was proposed by Frangi et al. [20] in 1998. This filter applies the Hessian matrix to extract features from image data. Not only can it measure how likely it is for a point to belong to vessels, but it can also detect the orientation of the vessels by using the eigenvalues and eigenvectors from the Hessian matrix.

The vesselness filter uses two properties of the vessel to evaluate the likelihood of point that belongs to vessels. The first property is the intensity value of that point and its neighbors are stable through the vessel direction. The second property is the intensity value of that point and its neighbors are changeable in the normal direction of the vessel. In addition, the vesselness filter also adopts the Gaussian filter which enables it to collect the intensity value information from the neighboring points and makes it resist to noises.

The vesselness filter is a voxel-based filter; every voxel in the dataset is calculated to evaluate the vesselness value. So, the computation time depends on the number of voxel in the dataset.

$$G(x, \sigma) = \frac{1}{\sqrt{(2\pi\sigma^2)}^3} e^{-\frac{(x^2+y^2+z^2)}{2\sigma^2}}$$

where σ is the parameter for Gaussian function.
 x is the specific location in a 3D space.

Equation 2.16 3-Dimension Gaussian function

$$H(x) = \begin{bmatrix} I_{xx}(x) & I_{xy}(x) & I_{xz}(x) \\ I_{yx}(x) & I_{yy}(x) & I_{yz}(x) \\ I_{zx}(x) & I_{zy}(x) & I_{zz}(x) \end{bmatrix}$$

Equation 2.17 Hessian Matrix

$$I_{uv}(x, \sigma) = \left(\frac{\partial^2 G(x, \sigma)}{\partial u \partial v} \right) * \sigma^\gamma I(x)$$

where G is the Gaussian function.

γ is the normalized parameter.

$I(x)$ is the intensity value of the image data at location x .

Equation 2.18 Element of Hessian Matrix

The Hessian matrix for each voxel is shown in Equation 2.17 and 2.18. x represents a voxel location in 3D space, $I(x)$ is the intensity value of the image data at location x , $G(x, \sigma)$ is the 3-Dimension Gaussian function which is defined as Equation 2.16 and γ is the normalized parameter [20] which is important in the case that we want to compare the response of different operators at multiple scales.

The vessel filtering process starts by calculating the Hessian matrix of each voxel. Each element of Hessian matrix is the second derivative of Gaussian function convolution with the intensity value of image data. Then, we can calculate the eigenvalues and the eigenvectors from the Hessian matrix. For the Hessian matrix in 3-Dimension, there are three eigenvalues λ_1, λ_2 and λ_3 and three eigenvectors v_1, v_2 and v_3 . The eigenvalues are sorted as show in Equation 2.19.

$|\lambda_1| \leq |\lambda_2| \leq |\lambda_3|$
 where λ_1, λ_2 and λ_3 are the eigenvalues.

Equation 2.19 Order of eigenvalues

Normally, if we measure the vesselness of bright vessel on a dark background, a voxel which belongs to the vessel has λ_1 that is very small and nearly zero while λ_2 and λ_3 are negative but their absolute values are large and both of them are nearly equal, as shown in Equation 2.20. The possible patterns in 3D space for eigenvalues [20] are summarized in Table 2.1.

$$\begin{aligned} |\lambda_1| &\approx 0 \\ |\lambda_1| &\ll |\lambda_2| \\ \lambda_2 &\approx \lambda_3 \end{aligned}$$

where λ_1, λ_2 and λ_3 are the eigenvalues.

Equation 2.20 The pattern of eigenvalues for a voxel which belongs to the vessel

Table 2.1 Possible patterns in 3D space [20] (H=high, L=low, N=noisy, usually small, +/- indicate the sign of eigenvalues)

λ_1	λ_2	λ_3	Orientation Pattern
N	N	N	Noisy, no preferred direction
L	L	H-	Plate-like structure (bright)
L	L	H+	Plate-like structure (dark)
L	H-	H-	Tubular structure (bright)
L	H+	H+	Tubular structure (dark)
H-	H-	H-	Blob-like structure (bright)
H+	H+	H+	Blob-like structure (dark)

In order to identify the tubular structure, there are three measures which make use of the eigenvalues.

The first measure R_B is used to identify the pixel which belongs to the blob-like structure but it cannot distinguish between a line and plate-like structure. Response from this measure will be large when the pixel belongs to blob-structure. The equation is shown in Equation 2.21.

$$R_B = \frac{\text{Volume}/(\frac{4\pi}{3})}{(\text{Largest Cross Section}/\pi)^{3/2}} = \frac{|\lambda_1|}{\sqrt{|\lambda_2\lambda_3|}}$$

where λ_1, λ_2 and λ_3 are the eigenvalues.

Equation 2.21 First measurement to evaluate blob-structure

The second measure R_A is used to solve the drawback of the first measure. It can distinguish between plate-like and line structure. Response from this measure will be zero if the pixel belongs to line structure, as shown in Equation 2.22.

$$R_A = \frac{(\text{Largest Cross Section Area})/\pi}{(\text{Largest Axis Semi-length})^2} = \frac{|\lambda_2|}{|\lambda_3|}$$

where λ_1, λ_2 and λ_3 are the eigenvalues.

Equation 2.22 Second measurement to distinguish between plate-like and line structure

The final measure S makes use of the prior knowledge in which the vessel structure in the medical image is usually brighter than the background. So, this measure is used to distinguish the background from the foreground. Response of the measure will be low in the background meaning that there is no structure present. The equation is shown in Equation 2.23 where D is the dimension of the image.

$$S = \sqrt{\sum_{j \leq D} \lambda_j^2}$$

where λ_j is the eigenvalue.

Equation 2.23 Third measurement to distinguish the background and the foreground

In order to combine the advantage of those measures, Frangi et al. introduce the vesselness function [20] as shown in Equation 2.24.

$$v(x, \sigma) = \begin{cases} 0 & \text{if } \lambda_2 > 0 \text{ or } \lambda_3 > 0 \\ (1 - \exp(-\frac{R_A^2}{2\alpha^2})) \exp(-\frac{R_B^2}{2\beta^2}) (1 - \exp(-\frac{S^2}{2c^2})) & \end{cases}$$

where λ_2 and λ_3 are the eigenvalues.

α, β and c are the constant parameters.

Equation 2.24 Vesselness measurement model

According to the equation, α, β and c are constants which are used to control the sensitivity of those measures. The response of vesselness function will be maximized when the property of the pixel follows all three criteria.

- A voxel should not be likely to belong to a blob-like structure. So, response of the first measure R_B is low.
- A voxel should belong to a plate-like structure. On the other hand, it should not belong to a line structure. So, response of the second measure R_A is high.
- A voxel should not belong to the background and has intensity value higher than the background. So, response of the third measure S is high.

$$v_{multi-scale}(x) = \max_{\sigma} v(x, \sigma)$$

where v is the vesselness response.

σ is the scale factor.

x is the specific location.

Equation 2.25 Multi scale vesselness response

The response of vesselness function also depends on the value of scale factor, σ . It will be maximized when the scale factor matches the size of the detected vessel. So, in order to filter the multiscale vessels, we compare the vesselness response from every scale value as shown in Equation 2.25

2.2.3 Center Line Extraction

The center line extraction or vessel thinning is the process that we use to specify the voxel which lies at the center of the vessels. In this process, we apply the Non-maximum Suppression technique. According to the vessel filter process, the vesselness response of the voxels at the center of the vessels is usually higher than the vesselness

response of the voxels at the border. So, the voxel will be suppressed if its vesselness response is smaller than the vesselness response of any neighbor around it in the vessel orientation.

In a 3-Dimension space, there are 13 major patterns of surrounding neighbor which are summarized in Table 2.2. The first column is the orientation of the vessels which can be identified by using the first eigenvector of Hessian matrix which is usually perpendicular to the vessels orientation, as shown in Figure 2.10. The second column is the list of the neighbors surrounding in corresponding orientation. The example of neighbor patterns is shown in Figure 2.11 where the arrow represents the direction of vessels orientation and the boxes represent the neighbors.

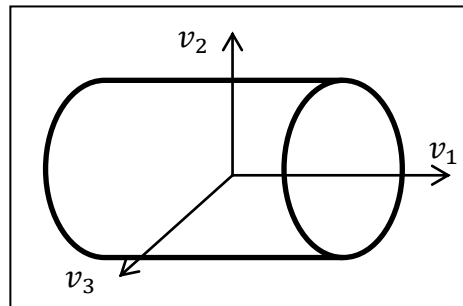


Figure 2.10 The orientation of ideal tubular structure and direction of eigenvectors

Table 2.2 Patterns of surrounding neighbor [21]

Vessels Orientation	Surrounding Neighbors
(0, 0, 1)	(0, 1, 0), (1,-1, 0), (1, 0, 0), (1, 1, 0), (-1,-1, 0), (-1, 0, 0),(-1, 1, 0), (0,-1, 0)
(0, 1,-1)	(0, 1, 1), (1,-1,-1), (1, 0, 0), (1, 1, 1), (-1,-1,-1), (-1, 0, 0),(-1, 1, 1), (0,-1,-1)
(0, 1, 0)	(0, 0, 1), (1, 0,-1), (1, 0, 0), (1, 0, 1), (-1, 0,-1), (-1, 0, 0),(-1, 0, 1), (0, 0,-1)
(0, 1, 1)	(0, 1,-1), (1,-1, 1), (1, 0, 0), (1, 1,-1), (-1,-1, 1), (-1, 0, 0),(-1, 1,-1), (0,-1, 1)
(1,-1,-1)	(0, 1,-1), (1, 0, 1), (1, 1, 0), (-1,-1, 0), (-1, 0,-1), (0,-1, 1)
(1,-1, 0)	(0, 0, 1), (1, 1,-1), (1, 1, 0), (1, 1, 1), (-1,-1,-1), (-1,-1, 0),(-1,-1, 1), (0, 0,-1)
(1,-1, 1)	(0, 1, 1), (1, 0,-1), (1, 1, 0), (-1,-1, 0), (-1, 0, 1), (0,-1,-1)
(1, 0,-1)	(0, 1, 0), (1,-1, 1), (1, 0, 1), (1, 1, 1), (-1,-1,-1), (-1, 0,-1),(-1, 1,-1), (0,-1, 0)
(1, 0, 0)	(0, 0, 1), (0, 1,-1), (0, 1, 0), (0, 1, 1), (0,-1,-1), (0,-1, 0), (0,-1, 1), (0, 0,-1)
(1, 0, 1)	(0, 1, 0), (1,-1,-1), (1, 0,-1), (1, 1,-1), (-1,-1, 1), (-1, 0, 1),(-1, 1, 1), (0,-1, 0)
(1, 1,-1)	(0, 1, 1), (1,-1, 0), (1, 0, 1), (-1, 0,-1), (-1, 1, 0), (0,-1,-1)
(1, 1, 0)	(0, 0, 1), (1,-1,-1), (1,-1, 0), (1,-1, 1), (-1, 1,-1), (-1, 1, 0),(-1, 1, 1), (0, 0,-1)
(1, 1, 1)	(1, 1, 1) (0, 1,-1), (1,-1, 0), (1, 0,-1), (-1, 0, 1), (-1, 1, 0), (0,-1, 1)

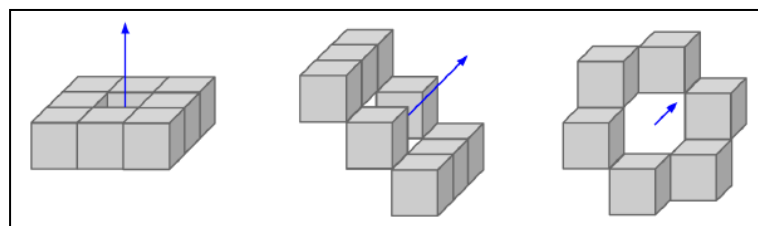


Figure 2.11 The example of neighbor patterns

[Source: Zhong. Y., Extracting Vessel Structure From 3D Image Data, p. 51]

2.2.4 Ray Casting Algorithm

Ray Casting Algorithm is one of the Direct Volume Rendering methods used to generate the three dimensions model image. The input of this algorithm comes from the medical image, for example MRI and CT scan image. The images are stacked to generate the point cloud. This algorithm uses the light principle to generate the visualization. When the light shines through the object, its shadow is created on the screen. This algorithm uses the shadow to specify the shape of the 3D model. By adjusting the transfer function, the user can determine the transparency of the object, which allows the user to segment the organs in question. For example, given the dataset of legs of which bones are covered with skin and tissue, the user can specify the skin as highly-transparent objects and the bones as low-transparent objects. The result image will visualize the leg bones from the medical image data. After the shadow of the object is created, the shading technique is adopted. Shading technique can stimulate the refraction of light which causes a virtual image to look like a real object. Finally, the medical visualization image will appear on the output screen. There are five processes of Ray Casting algorithm [22], data preparation, classification, shading, sampling and composition.

Data preparation is the first process of Ray Casting algorithm. The medical images are usually kept in the standard medical format called “Dicom file”. Dicom is a standard for handling, storing, printing, and transmitting information in medical imaging [23]. It consists of two parts, header and medical image data. The header part records the information about the patient such as name, gender, age and the information of the medical image such as the kind of imaging technique, part of body, slice thickness and image dimension. The medical image data are stored in the second part. This process will extract the medical image data from the Dicom files and rearrange in the three dimensions array. We will refer to this array as the “point cloud” throughout the research. Each point in the point cloud can be accessed by using the Cartesian coordinate system. After the point cloud has been created, the user can adjust the brightness and contrast of the data by using Equation 2.26. Adjusting brightness and contrast allows the user to see data in the medical image clearly. After that, the prepared point cloud will be sent to the next process.

$$f'_v = cf_v + b$$

where f_v is the intensity value before the adjustment.

b is the brightness parameter.

c is the contrast parameter.

Equation 2.26 Brightness and contrast adjustment

Classification process is the organ segmentation process. After the point cloud has been prepared in data preparation process, each point in point cloud will be classified to determine the organ segment. Each pixel in the medical image represents the intensity value in Hounsfield unit (HUs) [24]. The range of value is between -1,000 to 3,000 so it is kept into two bytes. The intensity value also represents the density of element in the medical image. For example, water is mapped to 0 HUs, air is mapped -1000 HUs and bones are mapped to 45-3000 HUs. Equation 2.27 shows the relationship between the density of element and the intensity value in Hounsfield unit, where μ represents the density value of element and μ_{H_2O} represents the density value of water.

$$HU = \frac{\mu - \mu_{H_2O}}{\mu_{H_2O}} \times 1,000$$

where μ is the density value of an element.

Equation 2.27 Relation between density of element and intensity value in Hounsfield unit

Table 2.3 Hounsfield value for selected tissue types [24]

Tissue Type	Hounsfield Value Interval
Air	-1000
Lung tissue	-900 to -170
Fat tissue	-220 to -30
Water	0
Pancreas	10 to 40
Liver	20 to 60
Heart	20 to 50
Kidney	30 to 50
Bones	45 to 3000

The intensity value of each pixel in point cloud will be converted to an opacity value by using the transfer function. The range of opacity value is between 0 and 1 and it represents the transparency of the object. That means, the pixel with a high opacity value is an object with low transparency. Its shadows will be created on the screen where the light can shine through. On the other hand, the pixel with a low opacity value shows an object with high transparency. Equation 2.28 shows the transfer function which is called ‘‘isovalue contour surfaces’’

$$\alpha(x_i) = \alpha_v \begin{cases} 1 & \text{if } |\nabla f(x_i)| = 0 \text{ and } f(x_i) = f_v \\ 1 - \frac{1}{r} \frac{|f_v - f(x_i)|}{|\nabla f(x_i)|} & \text{if } |\nabla f(x_i)| > 0 \text{ and } f(x_i) - r|\nabla f(x_i)| \\ & \leq f_v \leq f(x_i) + r|\nabla f(x_i)| \\ 0 & \text{otherwise} \end{cases}$$

where α is the opacity value.

α_v is the upper bound of the opacity value.

$f(x_i)$ is the intensity value at location x_i .

x_i is the specific location.

f_v is the intensity value of selected organ.

r is the constant parameter.

Equation 2.28 Isovvalue contour surfaces transfer function

The notation $\alpha(x_i)$ is the opacity value of each pixel: α_v is the upper bound of opacity value; $f(x_i)$ is the intensity value of each pixel; f_v is the intensity value of selected organ and $|\nabla f(x_i)|$ is the gradient vector of each pixel in point cloud. From this equation, the pixel with an intensity value equal to f_v will have the highest opacity value and the pixel with an opacity value similar to f_v will have a lower opacity value depending on the difference between f_v and $f(x_i)$. So, the important parameter in order to segment the organ in question in the medical image is f_v . Since the intensity value of the organ varies according the dataset, the user should specify the appropriate f_v value to clearly segment the organ in question, which is quite complicated.

Shading process is the illusion of light. In order to visualize the medical image in 3D model image, the direction of light and the shadow of the object are very important. The effect of light can visualize the 2D image to look like the real object. By adopting Phong Shading technique [22][24], the shading process can stimulate the reflection of the light on each point in point cloud. The Phong Shading technique consists of three components. There are ambient light, diffuse reflection and specular reflection.

Ambient light part is the background color of the light. When the light occurs upon the surface of the object, it will reflect into the observer's eyes. The color of the light that the observer sees is the ambient light. This research assumes that the data point in point cloud is black so the color of the ambient light is equal to the color of the light. From Equation 2.29, C_{amb} is the reflected intensity, C_{light} is the intensity value of the light and K_a is the coefficient of the surface ambient reflectivity. The range of K_a value is between 0 and 1. It specifies how the object's surface can reflect the light.

$$C_{amb} = C_{light} \times K_a$$

where C_{amb} is the reflected intensity.

C_{light} is the intensity value of the light.

K_a is the coefficient of the surface ambient reflectivity.

Equation 2.29 The reflected intensity of ambient light

Diffuse reflection is the simulation of the reflected light. Reflection of the light is very important for medical image visualization because it presents the shape of the object which makes the virtual image look like the real object. From Equation 2.30 C_{diff} is the reflected intensity, K_d is the coefficient of surface diffuse reflectivity, $N(x_i)$ is the normalized vector of surface at location x_i and L is the normalized vector of light. This equation shows that if the normalized vector of surface and normalized vector of light are parallel, the intensity of point will be brightened. On the other hand, if the normalized vector of surface is perpendicular or opposite the normalized vector of light, this point will be darkened, as shown in Figure 2.12. The range of K_d value is between 0 and 1. It specifies how the object's surface can reflect the light.

$$C_{diff} = C_{light} \times K_d \times (N(x_i) \cdot L)$$

where C_{diff} is the reflected intensity.

C_{light} is the intensity value of the light.

K_d is the coefficient of surface diffuse reflectivity.

$N(x_i)$ is the normalized vector of surface at location x_i .

x_i is the specific location.

L is the normalized vector of light.

Equation 2.30 The reflected intensity of diffuse reflection

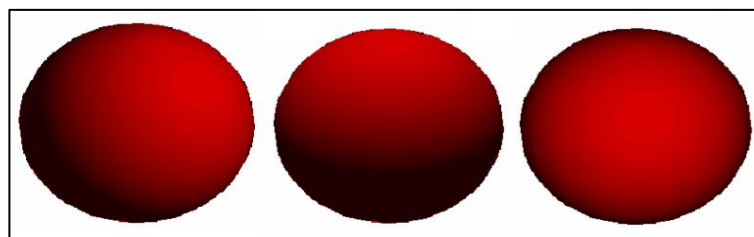


Figure 2.12 Diffuse reflection from different light directions [25]

Specular reflection is the model of shiny and glossy surface. It simulates the characteristics of the surface. The shiny surface can reflect the image of the light source. From Equation 2.31, C_{spec} is the reflected intensity, K_s is the coefficient of surface specular reflectivity, R is the normalized vector of reflected ray, V is the normalized vector in direction of the observer, and n is the exponent used to approximate the highlight. This equation shows that the point of the surface where the normalized vector of the reflected ray parallel with the normalized vector in direction of the observer will shine. The range of K_s value is between 0 and 1. It specifies how the object's surface can reflect the light.

$$C_{spec} = C_{light} \times K_s \times (R \cdot V)^n$$

where C_{spec} is the reflected intensity.

C_{light} is the intensity value of the light.

K_s is the coefficient of surface specular reflectivity.

R is the normalized vector of reflected ray.

V is the normalized vector in direction of the observer.

n is the exponent used to approximate the highlight.

Equation 2.31 The reflected intensity of specular reflection

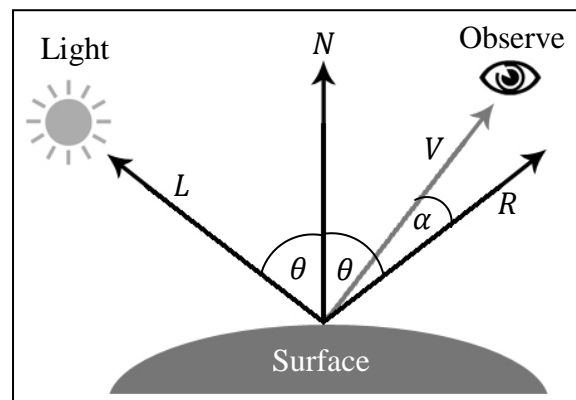


Figure 2.13 N normalized vector of surface, L normalized vector of light, V normalized vector in direction of the observer and R normalized vector of reflected ray. [25]

$$C(x_i) = C_{amb} + \frac{1}{distance} (C_{diff} + C_{spec})$$

$$C(x_i) = C_{amb} + \frac{C_{light}}{distance} [(K_d \times (N(x_i) \cdot L)) + (K_s \times (R \cdot V)^n)]$$

where $C(x_i)$ is the intensity value at location x_i .

C_{amb} is the reflected intensity.

C_{light} is the intensity value of the light.

C_{diff} is the reflected intensity.

C_{spec} is the reflected intensity.

K_d is the coefficient of surface diffuse reflectivity.

$N(x_i)$ is the normalized vector of surface at location x_i .

x_i is the specific location.

L is the normalized vector of light.

K_s is the coefficient of surface specular reflectivity.

R is the normalized vector of reflected ray.

V is the normalized vector in direction of the observer.
 n is the exponent used to approximate the highlight.

Equation 2.32 Phong Shading

The next process is sampling process. The Ray Casting algorithm uses the light principle to generate the visualization. Each pixel in the output screen is like a ray caster. The rays are cast and pass through the point cloud and they will sample the point along the way. The sampling step size is the important parameter in this process. It specifies the distance between the sampled points and also affects the quality of image visualization. High sampling step size can produce high quality output image but it requires more computation time. Very low sampling step size can introduce the artifacts on the output image. Normally, the straight ray does not pass through the exact point in point cloud. It will interpolate the sampled point by using the intensity value of nearby points. Figure 2.14 depicts the rays passing through the point cloud. The intensity value of sampling point is calculated by using Equation 2.33.

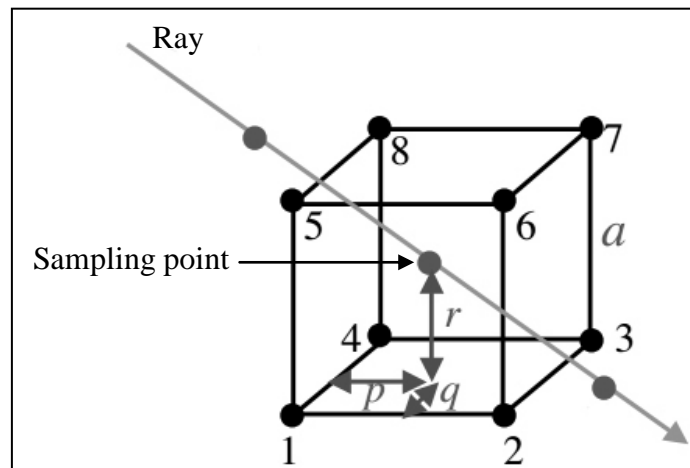


Figure 2.14 Interpolation the sampled point [22]

$$\begin{aligned}
 f_v = & f_1 \left(1 - \frac{p}{a}\right) \left(1 - \frac{q}{a}\right) \left(1 - \frac{r}{a}\right) + f_2 \left(\frac{p}{a}\right) \left(1 - \frac{q}{a}\right) \left(1 - \frac{r}{a}\right) \\
 & + f_3 \left(\frac{p}{a}\right) \left(\frac{q}{a}\right) \left(1 - \frac{r}{a}\right) + f_4 \left(1 - \frac{p}{a}\right) \left(\frac{q}{a}\right) \left(1 - \frac{r}{a}\right) \\
 & + f_5 \left(1 - \frac{p}{a}\right) \left(1 - \frac{q}{a}\right) \left(\frac{r}{a}\right) + f_6 \left(\frac{p}{a}\right) \left(1 - \frac{q}{a}\right) \left(\frac{r}{a}\right) \\
 & + f_7 \left(\frac{p}{a}\right) \left(\frac{q}{a}\right) \left(\frac{r}{a}\right) + f_8 \left(1 - \frac{p}{a}\right) \left(\frac{q}{a}\right) \left(\frac{r}{a}\right)
 \end{aligned}$$

where f_v is the intensity value of sampling point

f_i is the intensity value of point i

a is the distance between point 1 and point 5

p is the distance between point 1 and sampling point along the x-axis

q is the distance between point 1 and sampling point along the y-axis

r is the distance between point 1 and sampling point along the z-axis

Equation 2.33 Interpolation the sampled point [22]

Composition is the final process of Ray Casting algorithm. This process will calculate the intensity value of each pixel on 3D model image. The range of intensity value is between 0 and 255, where 0 represents black color and 255 is white color. After the sampling process has sampled the data through which each ray passes, the composition

process will combine the color and opacity of all the data in the ray and paint the result of intensity value on the output screen. Equation 2.34 shows change of the intensity value and the opacity value of the light after it passes through the sampling data.

$$C_{out}(u_i) = C_{in}(u_i)(1 - \alpha(x_i)) + c(x_i)\alpha(x_i)$$

where $C_{out}(u_i)$ is the intensity value of the light after it passes through the sampling point i

$C_{in}(u_i)$ is the intensity value of light before it passes through the sampling point i

$c(x_i)$ is the intensity value of the sampling point i

$\alpha(x_i)$ is the opacity value of sampling point i

Equation 2.34 Composition

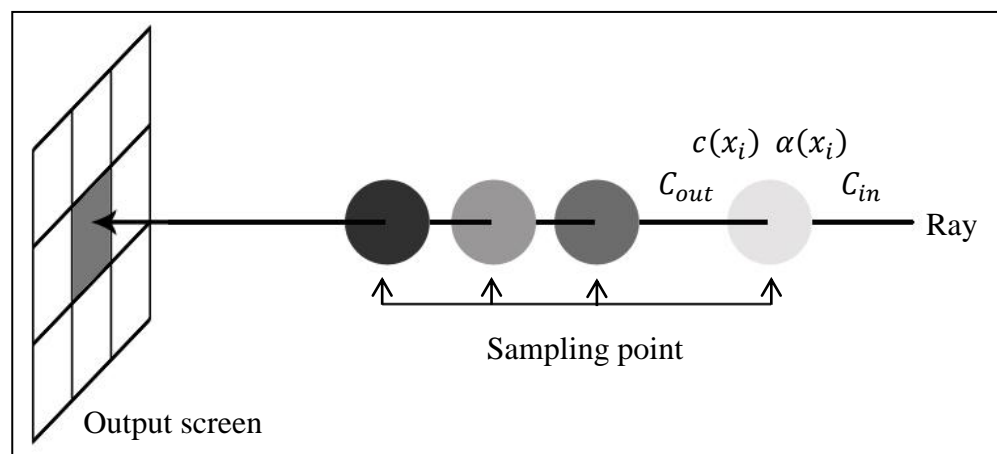


Figure 2.15 Composition the sampling point along the ray

The notation $c(x_i)$ is the intensity value of the sampling point i , $\alpha(x_i)$ is the opacity value of sampling point i and $C_{out}(u_i)$ is the intensity value of the light after it passes through the sampling point i , and $C_{in}(u_i)$ is the intensity value of light before it passes through the sampling point i . As can be seen in Figure 2.15, before the light passes through the sampling point, it has an initial intensity value and initial opacity value. After the light pass through the sampling point, its intensity value and opacity value is changed by the intensity value and opacity value of the sampling data which it passes through. The intensity value of light changes when it passes through the sampling point with the opacity value that is not equal to zero until it reaches the output screen. Lastly, the intensity value of the light will be painted on the screen.

CHAPTER 3 THE HYBRID LEVEL SET METHOD

In this chapter, we present our version of the Level Set Method (LSM). This modified version is called Hybrid LSM. The Hybrid LSM is the combination between edge based and region based active contour model. This modified version combines the advantage of both algorithm and it also improves the performance of LSM. In Section 3.1, we introduce the idea of our modified version of LSM. Next in Section 3.2, we describe the development of the Hybrid LSM. In the last section, we show the benchmark results in order to compare the Hybrid LSM with the edge based and region based active contour model. The discussion is also presented at the end of this chapter.

3.1 The Ideas of the Hybrid LSM

Level Set Method is used to identify the boundary of objects in the image and distinguish between foreground and background. In the classical active contour model, the evolution of model is based on the edge of the object which we categorize as the edge based active contour model. The edge of the object is used to prevent the contour from leaking out the desired boundary by terminating the evolution of level set function when the contour at zero level reaches the boundary of the desired object. The advantage of this technique is that it can be used to segment only the boundary of the desired object by setting the appropriate initial level set function despite several objects on the foreground image. However, this technique is not appropriate for segmenting objects which have no clear boundary or segmenting objects from noisy images.

After that, the region based active contour model is introduced to solve the weakness of the edge based active contour model. This technique still works well even though the edge of the segmented object is unclear and it is also robust to the noisy image. Moreover, we can set the initial contour at zero level everywhere on the image. Since the region based active contour model uses the information based on the region properties instead of boundary of the object, the model can find the suitable boundary of the object when the deviation of intensity value inside and outside the contour is minimized. However, this technique is not appropriate to segment the desired object in the image containing several objects in foreground. If we perform this model until it meets the convergence point, this method will segment all objects in the foreground, rather than segment only the desired object. There is no appropriate parameter used to terminate the evolution of level set function when the contour at zero level reaches the desired boundary.

Hybrid LSM is developed in order to segment the boundary of the organs in the medical images such as, segmenting the bone and specifying the boundary of the vessels in medical images. Due to the characteristic of the medical images, the organs usually have unclear boundary. So, applying the edge based active contour model directly does not work well with medical images. Moreover, the range of intensity value of the desired organs may overlap with the range of intensity value of the other organ. For example the range of intensity value of bones overlaps with that of vessel in CTA images. So, applying the region based contour model directly cannot distinguish the boundary of bones from the boundary of vessels.

So, the Hybrid LSM is the combination between edge based and region based active contour models. This modified version preserves the advantages of both original models while the performance of Level Set Method is improved. This model enables the Level

Set Method to segment the boundary of the object even though the edge of the segmented object is unclear and robust to the noisy image. In addition, this modify version can segment only the boundary of the desired object by setting the appropriate initial level set function. Moreover, the Hybrid LSM can reach the convergence point faster than both original models with more accuracy. Table 3.1 shows the overview comparison of characteristic between Hybrid LSM, edge based and region based active contour model.

Table 3.1 Characteristic of Hybrid LSM, edge based and region based active contour model

Characteristic	Hybrid LSM	Edge based	Region based
Depend on gradient of the image		✓	
Depend on region properties	✓		✓
Segment object which has unclear boundary	✓		✓
Segment only desired object	✓	✓	
Robust to noisy image	✓		✓

$$\frac{\partial \phi}{\partial t} = \rho \cdot \text{div}(d_p(|\nabla \phi|)\nabla \phi) + \delta_\varepsilon(\phi) \left[u \cdot \text{div}\left(\frac{\nabla \phi}{|\nabla \phi|}\right) - v - \lambda_1(I - c_1)^2 + \lambda_2(I - c_2)^2 \right] g$$

where $\rho, u, v, \varepsilon, \lambda_1$ and λ_2 are the constant parameters.

$\text{div}(d_p(|\nabla \phi|)\nabla \phi)$ is the regularization term.

δ_ε is the Dirac delta function.

ϕ is the level set function.

c_1 is the average intensity value inside the contour.

c_2 is the average intensity value outside the contour.

g is the stopping function.

I is the intensity value of input image.

Equation 3.1 First derivative of ϕ with respect to t for Hybrid LSM

The evolution of level set function over the time for Hybrid LSM is shown in Equation 3.1. The Hybrid LSM is developed by modifying four parts of conventional Level Set method. There are stopping function g , initial level set method, Dirac delta function $\delta_\varepsilon(\phi)$ and reinitializing function.

Stopping Function, g is replaced by our modified version which is based on the region properties of image instead of gradient of the image. The modified stopping function enables the Hybrid LSM to segment the object which has unclear boundary. The initial level set method and Dirac delta function, $\delta_\varepsilon(\phi)$ are chosen from the conventional edge based active contour model. They help to control the evolution of level set function and enable the Hybrid LSM to segment only the desired object. The modified stopping function g and Dirac delta function $\delta_\varepsilon(\phi)$ are applied in this equation by multiplying with the equation. Reinitializing function is replaced by regularization term, $\text{div}(d_p(|\nabla \phi|)\nabla \phi)$ as suggested by Li et al. [19] which helps prevent the level set function from becoming too flat and improve the speed of LSM. We embed the regularization term by adding it in front of the equation. Section 3.2 describes the development of Hybrid LSM in more details.

3.2 Development of Hybrid LSM

In this section, we describe the detail of Hybrid LSM. In order to explain how we develop this modified version, we separate the modified process into four parts. There are stopping function, initial level set function, Dirac delta function and reinitializing function. In each part, we compare and contrast the Hybrid LSM with both original models and show a pseudo code of the modified part. At the end of this section, we show the pseudo code of Hybrid LSM

3.2.1 Stopping Function

For the edge based active contour model, the stopping function is based on edge of the object. This technique applies the edge detector function, which depends on the gradient of the image, to find the boundary of the objects. Equation 2.1 shows the example of edge detector function. $g(x)$ becomes closer to zero at the boundary of the object where the intensity value of the object and intensity value of background are different and closer to one inside the object. Edge based active contour model applies the stopping function by multiplying the evolution of level set function over the time by $g(x)$, as shown in Equation 2.3. So, g is used to control the evolution of level set function and terminate the evolution process when the contour at zero level reaches the boundary of the desired object. Edge based active contour works well if the desired object has clear boundary. On the other hand, if the desired object has unclear boundary, the stopping function is not closer to zero on the edges of the object and it cannot be used to terminate the evolution of contour at zero level.

The region based active contour model does not have the explicit stopping function but this model aims to minimize the deviation of intensity value inside and outside the contour. The process is stopped when it reaches the convergence point where the boundaries of all objects in the foreground are segmented. With this property, this model can segment the boundary of the object in spite of the object's unclear boundary. However, we cannot use this model to segment only the desired object in the image containing many objects in the foreground, since there is no stopping function used to terminate the evolution of level set function when the contour at zero level reaches the desired boundary.

The Hybrid LSM is developed in order to combine the advantage of both edge based and region based active contour models. We introduce the modified stopping function which is based on the region properties of the image instead of gradient of the image. Equation 3.2 shows the definition of modified stopping function, where c_1 and c_2 are the average intensity value inside and outside the contour, respectively and where λ_1 and λ_2 are the constant parameters. $g(x)$ is equal to one inside the boundary of the object in the foreground and equal to zero at the background. The stopping function can be adjusted in each iteration depending on the value of c_1 and c_2 . This modified stopping function can solve the problem of the object's unclear boundary. The Hybrid LSM applies the modified stopping function by multiplying the evolution of level set function over the time by $g(x)$ as shown in Equation 3.1. So, the modified stopping function is used to control the evolution of level set function and stop the evolution process when the contour at zero level reaches the boundary of the desired object.

$$g(x) = \begin{cases} 1 & \text{if } -\lambda_1(I(x) - c_1)^2 + \lambda_2(I(x) - c_2)^2 > 0 \\ 0 & \text{otherwise} \end{cases}$$

where λ_1 and λ_2 are the constant parameters.

c_1 is the average intensity value inside the contour.

c_2 is the average intensity value outside the contour.

I is the intensity value of input image.

x is the specific location.

Equation 3.2 Modified stopping function

Table 3.2 Comparison of stopping function between Hybrid LSM, Edge Based and Region Based active contour model

Method	Equation
Edge Based	$g(x) = \frac{1}{1 + \nabla G(x, \sigma) * I(x) }$
Region Based	Implicit stopping function
Hybrid LSM	$g(x) = \begin{cases} 1 & \text{if } -\lambda_1(I(x) - c_1)^2 + \lambda_2(I(x) - c_2)^2 > 0 \\ 0 & \text{otherwise} \end{cases}$

3.2.2 Initial Level Set Function

Setting the initial level set function appropriately is an important step for the edge based active contour model. Since the model evolves the level set function until the contour at zero level reaches the boundary of the desired object, the initial contour of level set function should be inside or surround the boundary of the desired object. We can use this property to segment only the desired object in the image which has many objects on the foreground. Nevertheless, the region based active contour model does not strict the location of initial contour at zero level. The model allows us to set the initial contour everywhere on the image because at the end of the process, the model can specify the boundary of all objects in the foreground image. Chan-Vese [18] introduce the initial level set function as defined in Equation 2.15. Notice that there is no flat surface over the initial level set function which supports the region based active contour model to omit the reinitializing process.

In order to preserve the advantage that the model can segment only the desired object by setting the appropriate initial level set function, the Hybrid LSM chooses the way to initial level set function as the edge based active contour model.

Table 3.3 Comparison of initial level set function between Hybrid LSM, Edge Based and Region Based active contour model

Method	Equation
Edge Based	$\phi_0(x) = \begin{cases} c_0 & \text{if } x \text{ is the location of initial point} \\ -c_0 & \text{otherwise} \end{cases}$
Region Based	$\phi_0(x) = -\sqrt{(x - x_0)^2 + (y - y_0)^2} + r_0$
Hybrid LSM	$\phi_0(x) = \begin{cases} c_0 & \text{if } x \text{ is the location of initial point} \\ -c_0 & \text{otherwise} \end{cases}$

3.2.3 Dirac Delta Function

Dirac delta function is used to simulate the evolution response around the border of the contour at zero level. The conventional Dirac delta function allows only the border of the contour at zero level to evolve in each iteration, as shown in Equation 2.13. This Dirac delta function supports the stopping function. As the evolution occurs only around the border of the contour, the stopping function can terminate the evolution process when the border of the contour reaches the boundary of the desired object where $g(x)$ becomes closer to zero.

Chan-Vese [18] introduce the Dirac delta function which allows the surface inside and outside the contour to take the evolution effect. He suggests that the modified Dirac delta function allow the model to obtain the global minimizer. However, this Dirac delta function does not support the stopping function. Since this Dirac delta function allows the surface outside the contour to evolve in each iteration, the contour at zero level can leak out of the boundary of the desired object.

The Hybrid LSM chooses to apply the conventional Dirac delta function. This is to preserve the advantage that this function can support the stopping function to prevent the contour at zero level from leaking out of the boundary of the desired object.

Table 3.4 Comparison of Dirac delta function between Hybrid LSM, Edge Based and Region Based active contour model

Method	Equation
Edge Based	$\delta_\varepsilon(\chi) = \begin{cases} \frac{1}{2\varepsilon} [1 + \cos(\frac{\pi\chi}{\varepsilon})] & \chi \leq \varepsilon \\ 0 & \chi > \varepsilon \end{cases}$
Region Based	$\delta_\varepsilon(\chi) = \frac{1}{\pi\varepsilon} \left(\frac{1}{1 + (\frac{\chi}{\varepsilon})^2} \right)$
Hybrid LSM	$\delta_\varepsilon(\chi) = \begin{cases} \frac{1}{2\varepsilon} [1 + \cos(\frac{\pi\chi}{\varepsilon})] & \chi \leq \varepsilon \\ 0 & \chi > \varepsilon \end{cases}$

3.2.4 Reinitializing Function

$$\psi_\tau = \text{sign}(\phi(t))(1 - |\nabla\psi_i|)$$

where ψ_τ is the reinitializing function at the steady state.

ψ_i is the reinitializing function at iteration i .

i is the iteration.

$\phi(t)$ is the level set function at time t .

$\text{sign}(\cdot)$ is the sign function.

Equation 3.3 Conventional reinitializing function

The reinitializing process is used to prevent the level set function from becoming too flat by applying this process periodically during the evolution process. The level set function is flat when most of the surface does not belong to the zero level. The flat level set function causes premature convergence, especially for models which apply the Dirac delta function. The conventional reinitializing function is defined by Equation 3.3. The conventional reinitializing process starts by setting the ψ_0 equal to $\phi(t)$ and then

performs Equation 3.3 until it converges to the steady state. After that, the new level set function after the application of the reinitializing process becomes ψ_τ where ψ_τ represent the reinitializing function at steady state, ψ_i represent the reinitializing function at iteration i , $\phi(t)$ represent the level set function at time t and $\text{sign}(\cdot)$ is the sign function. Ideally, the reinitializing process should not change the contour at zero level and $|\nabla\psi_\tau|$ at steady state should converge to 1. Li et al. introduce the reinitializing function called Distance Regularized function [19], as shown in Equation 3.4. When this equation converges to the steady state, the surface inside the band around zero level becomes the sign distance function and the surface outside the band is flat. This band is called a signed distance band (SDB). The width of the band is approximately $2c_0$. The $|\nabla\phi|$ across the band is equal to 1 and values inside the band vary from $-c_0$ to c_0 . Figure 3.1 left depicts the Distance Regularized function at steady state and Figure 3.1 right depicts the signed distance band (SDB) around the zero level. However, the reinitializing process does not only take the computation time, it also reduces the performance of the model. It obstructs the model from obtaining the global minima. Li et al. introduce to embed the Distance Regularized function to the evolution of level set function over the time. This term is called regularization term. It helps to prevent the level set function from becoming too flat and also helps maintain the signed distance property $|\nabla\phi| = 1$ around the zero level. In addition, embedding the regularization term can replace the reinitializing process which helps to improve the speed of LSM. Chan-Vese [18] introduce the region based active contour model for which the reinitializing process is optional. They suggest the initial level set function where there is no flat surface, rather than using the reinitializing function to prevent the flat surface.

As the Hybrid LSM applies the Dirac delta function to contour the evolution of contour at zero level and the initial level set function is a flat surface, the reinitializing process is essential. We decide to embed the regularization term as suggested by Li et al. instead of performing the conventional reinitializing process.

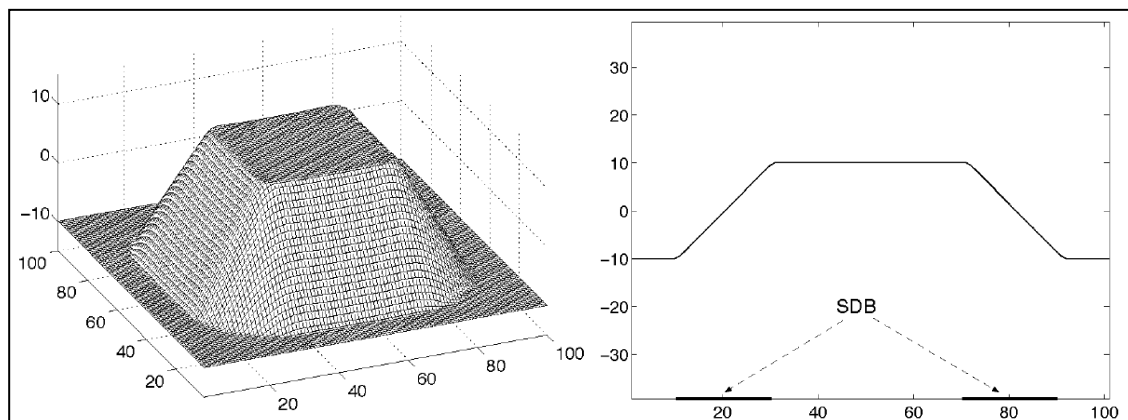


Figure 3.1 Distance Regularized function at steady state (left) and signed distance band (right) [19]

$$P(s) = \begin{cases} \frac{1}{2\pi} \sin(2\pi s) & \text{if } s \leq 1 \\ s - 1 & \text{if } s > 1 \end{cases}$$

$$d_p(s) = \frac{P(s)}{s}$$

$$\psi_\tau = \psi_i + \text{div}(d_p(|\nabla\psi_i|)\nabla\psi_i)$$

where ψ_τ is the reinitializing function at the steady state.

ψ_i is the reinitializing function at iteration i .

i is the iteration.

s is the input value.

Equation 3.4 Distance Regularized function

$$R(x) = \text{div}(d_p(|\nabla\phi(x)|)\nabla\phi(x))$$

where ϕ is the level set function.

x is the specific location.

Equation 3.5 Regularization Term

Table 3.5 Comparison of reinitializing function between Hybrid LSM, Edge Based and Region Based active contour model

Method	Equation
Edge Based	Conventional Reinitializing process $\psi_\tau = \text{sign}(\phi(t))(1 - \nabla\psi_i)$
Region Based	Reinitializing process is optional
Hybrid LSM	Embed the Regularization term $R(x) = \text{div}(d_p(\nabla\phi(x))\nabla\phi(x))$

In conclusion, the evolution of level set function over the time for Hybrid LSM is shown in Equation 3.1, where ρ , u , v , λ_1 and λ_2 are the constant parameters which control the sensitivity of each term. The modified stopping function g and Dirac delta function $\delta_\varepsilon(\phi)$ are applied in this equation by multiplying with this equation in order to control the evolution of the level set function. The regularization term is added in front of this equation in order to prevent the level set function from becoming too flat. The main force of the Hybrid LSM is the Propagation term which controls the characteristic of the model to minimize the deviation of intensity value inside and outside the contour at zero level, similar to the region based active contour model.

Figure 3.2 shows the pseudo code of Hybrid LSM. Notice that the line from 19 to 30 is the refinement process which is similar to the main process (line from 6 to 18). This process performs after the model reaches the first convergence point in order to refine the details of the object's boundary. In the main process, we use the regularization term to prevent the level set function from becoming too flat. However, the regularization term obstructs the model from obtaining the global minima. So, we perform the refinement process which does not use the regularization term. It enables the model to evolve again until it reaches the next convergence point which usually obtains a better result.

Level Set Method (LSM)

Input : input image, initial contour

Output: level set function ϕ_0

```

1    $\phi$  = level set function
2   I = input image
3    $\Omega$  = image domain
4    $\phi = \phi_0$  initial level set function
5   Setting the constant parameters  $\rho, u, v, \lambda_1$  and  $\lambda_2$ 
6   REPEAT
7       FOR EACH  $x \in \Omega$  DO
8           compute the reinitializing function,  $R(x)$  using Equation 3.5
9           compute  $g(x)$  using Equation 3.2
10          compute  $\delta_\varepsilon(\phi(x))$  using Equation 2.13
11           $E(x) = \text{div} \left( \frac{\nabla\phi(x)}{|\nabla\phi(x)|} \right)$ 
12          compute  $c_1$  using Equation 2.10
13          compute  $c_2$  using Equation 2.11
14           $P(x) = -\lambda_1(I(x) - c_1)^2 + \lambda_2(I(x) - c_2)^2$ 
15           $\frac{\partial\phi}{\partial t}(x) = \rho \cdot R(x) + \delta_\varepsilon(\phi(x))[u \cdot E(x) - v + P(x)]g(x)$ 
16           $\phi(x) = \phi(x) + \frac{d\phi}{dt}(x)$ 
17       END FOR
18   UNTIL the contour at zero level is stable
19   REPEAT
20       FOR EACH  $x \in \Omega$  DO
21           compute  $g(x)$  using Equation 3.2
22           compute  $\delta_\varepsilon(\phi(x))$  using Equation 2.13
23            $E(x) = \text{div} \left( \frac{\nabla\phi(x)}{|\nabla\phi(x)|} \right)$ 
24           compute  $c_1$  using Equation 2.10
25           compute  $c_2$  using Equation 2.11
26            $P(x) = -\lambda_1(I(x) - c_1)^2 + \lambda_2(I(x) - c_2)^2$ 
27            $\frac{\partial\phi}{\partial t}(x) = \delta_\varepsilon(\phi(x))[u \cdot E(x) - v + P(x)]g(x)$ 
28            $\phi(x) = \phi(x) + \frac{d\phi}{dt}(x)$ 
29       END FOR
30   UNTIL the contour at zero level is stable
31   RETURN  $\phi$ 

```

Figure 3.2 Sequence of Level Set Method

3.3 Method Comparison and Evaluation

In this section, the performance of the proposed method is compared with the original method, both edge based and region based active contour models by using various types of input images. The experimental methodology is presented and results are illustrated. In addition, we discuss the experimental results at the end of this section.

3.3.1 Experiment Methodology

This experiment aims to evaluate the performance of our proposed method by comparing it with both edge based and region based active contour models. We used the Distance Regularized Level Set Evolution model introduced by Li et al. [19] to

represent the edge based active contour model and Active Contours without Edges model introduced by Chan-Vese [18] to represent the region based active contour model. The evolution of level set function over the time of edge based and region based active contour model are defined in Equation 3.6 and 3.7, respectively.

$$\frac{\partial \phi}{\partial t} = \rho \cdot \text{div}(d_p(|\nabla \phi|)\nabla \phi) + \delta_\varepsilon(\phi) \left(u \cdot \text{div} \left(g \frac{\nabla \phi}{|\nabla \phi|} \right) - v \cdot g \right)$$

where ρ, v, ε and u are the constant parameters.

$\text{div}(d_p(|\nabla \phi|)\nabla \phi)$ is the regularization term.

δ_ε is the Dirac delta function.

ϕ is the level set function.

g is the stopping function.

Equation 3.6 First derivative of ϕ with respect to t for
Distance Regularized Level Set Evolution model

$$\frac{\partial \phi}{\partial t} = \delta_\varepsilon(\phi) \left[u \text{div} \left(\frac{\nabla \phi}{|\nabla \phi|} \right) - v - \lambda_1 (I - c_1)^2 + \lambda_2 (I - c_2)^2 \right]$$

where $u, v, \varepsilon, \lambda_1$ and λ_2 are the constant parameters.

δ_ε is the Dirac delta function.

ϕ is the level set function.

c_1 is the average intensity value inside the contour.

c_2 is the average intensity value outside the contour.

I is the intensity value of input image.

Equation 3.7 First derivative of ϕ with respect to t for
Active Contours without Edges model

The details of input images are shown in Table 3.7. There are various types of images, for example, the noisy image, clear and unclear boundary object image, multi gray-scale image and medical image. Table 3.7 also presents the initial contour at zero level, the referenced boundary of the desired object and the characteristics of input images in each experiment.

Table 3.6 The constant parameters

parameter	Edge based active contour model	Region based active contour model	Hybrid LSM
c_0	2	2	2
μ	1	1	1
λ_1, λ_2	-	1	1
v	-0.5	-0.5	-0.5
ρ	0.2	-	0.2
ε	1.5	$1.5/\pi$	1.5

For each experiment, the controlled variables were the constant parameters. We set the constant parameters as show in Table 3.6. The Dirac delta function of the edge based active contour model and our proposed method were conventional Dirac delta. We set the $\varepsilon=1.5$. While the Dirac delta function of the region based active contour model was Chan-Vese's Dirac delta, we set the $\varepsilon=1.5/\pi$. The response of both Dirac delta functions

were approximately equal, as illustrated in Figure 2.5. The input image, initial level set function and the referenced boundary of the desired object depended on the experiment.

In order to evaluate the performance of our proposed method, we used the Dice similarity, Sensitivity and Specification to measure the accuracy of the methods and used the number of iteration which the model used to reach the convergence point to measure the speed of the method.

$$D_{si} = \frac{2TP}{((TP + FN) + (FP + TP))}$$

where TP is the number of True Positive pixel.

FN is the number of False Negative pixel.

FP is the number of False Positive pixel.

Equation 3.8 The dice similarity

The definition of Dice similarity is shown in Equation 3.8. Where True Positive, TP is the number of pixels which is defined as desired object by reference and we can correctly detect them, False Positive, FP is the number of detected pixels which is not defined as desired object by reference and FN is the number of pixels which is defined as desired object but we cannot detect them. Dice similarity is used to measure the accuracy of our results by comparing them with the references.

$$\frac{\text{number of overlapped contours}}{\text{total number of contours in manual segmentation}} \times 100\%$$

Equation 3.9 Sensitivity

The definition of Sensitivity is shown in Equation 3.9. It is used to evaluate the performance of the proposed method. It measures how the model can detect the true positive pixels in the dataset. High sensitivity implies that the model can detect most of the pixels of the desired object.

$$\left(1 - \frac{\text{number of detected contours but not in reference}}{\text{total number of detected contour}}\right) \times 100\%$$

Equation 3.10 Specification

The definition of Specification is shown in Equation 3.10. It is used to evaluate the false positive detection. High specification implies that our results have low false positive detection.

Table 3.7 The input image, initial level set function and the referenced boundary of the desired object in each experiment.


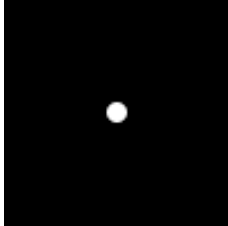
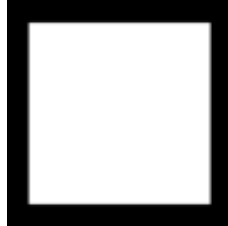
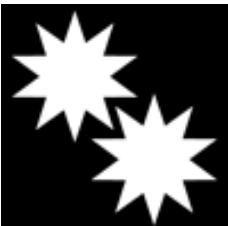
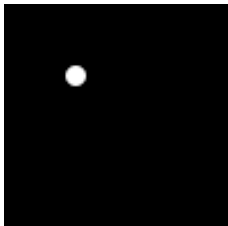
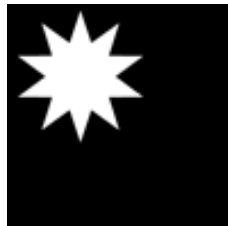
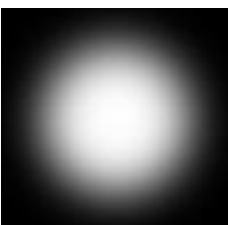
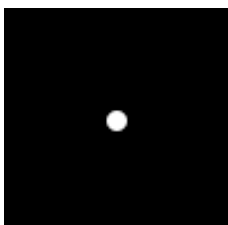
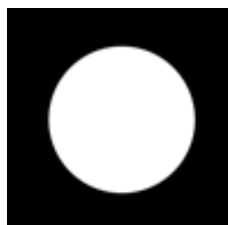
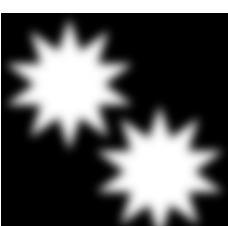
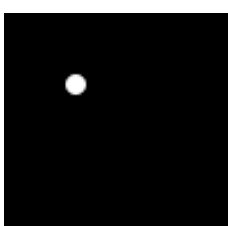
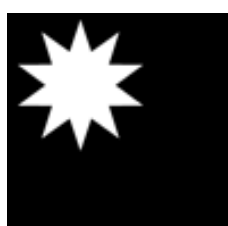
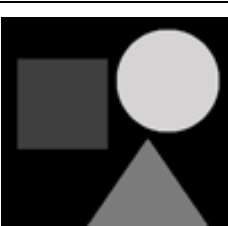
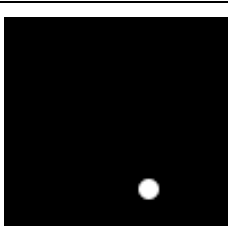
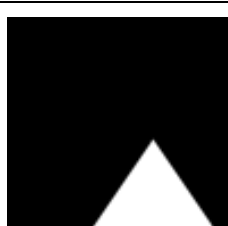

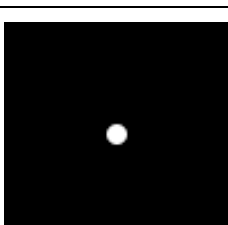
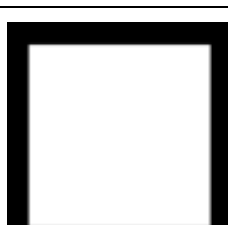
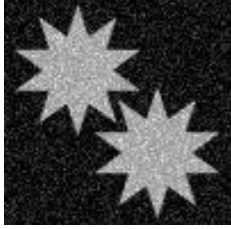
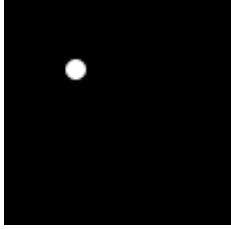
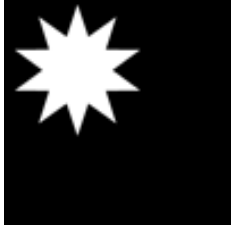

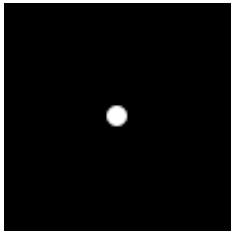
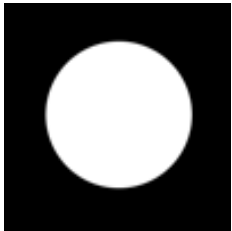
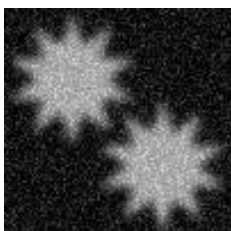
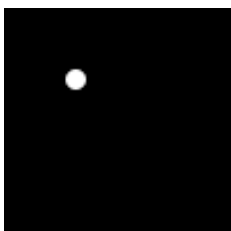
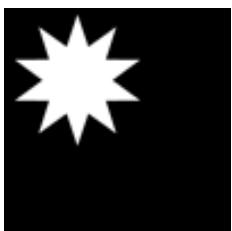
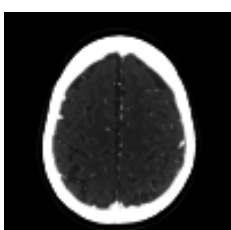
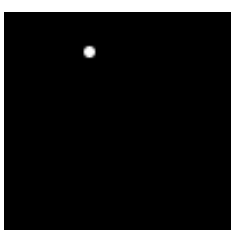

No.	Input image	Initial contour at zero level	Referenced boundary	Characteristics
1				Clear boundary image.
2				Clear boundary and multiple objects in foreground image.
3				Unclear boundary image.
4				Unclear boundary and multiple objects in foreground image.
5				Multi gray-scale and multiple objects in foreground image.
6				Clear boundary and noisy image.

Table 3.7 The input image, initial level set function and the referenced boundary of the desired object in each experiment. (Cont.)

No.	Input image	Initial contour at zero level	Referenced boundary	Characteristics
7				Clear boundary, multiple objects in foreground and noisy image.
8				Unclear boundary and noisy image.
9				Unclear boundary, multiple objects in foreground and noisy image.
10				Medical image.

3.3.2 Experimental Results and Discussion

The experimental results are shown in Table 3.8. We performed 10 experiments in order to compare our modified version Hybrid LSM with edge based and region based active contour models. In each experiment, we used different input images with different characteristics.

Column “No.” is the experiment number, column “Number of iteration” is the number of iteration which the model used to reach the convergence point, column “dice similarity”, “Sensitivity” and “Specification” show the evaluation of accuracy for each method. Those results were measured after the model reached the convergence point and terminated. The convergence point is defined as the point where the contour at zero level is stable for a period of time.

Figures 3.3-3.7 present graphs which show the change in dice similarity in each iteration. Solid line represents the results of the edge based active contour model, dot line represents the results of the region based active contour model and dash line represents the results of our modified version Hybrid LSM. Dice similarity in each

iteration measured by comparing the contour at zero level with the referenced boundary of the desired object. Those graph show the speed of each method to reach the convergence point and value of Dice similarity at the convergence point.

Table 3.9 shows the segmentation results in each experiment obtained by performing our modified version Hybrid LSM, edge based and region base active contour models. This table also displays the input images and the referenced boundary of the desired object.

In experiment 1, we aimed to segment the object which had clear boundary. We set the initial level set function center of the object. All methods could reach the boundary of the object; however, the region based active contour model had the lowest accuracy while our modified version Hybrid LSM had the highest accuracy. Notice that all the methods had the specification result equal to 100 percent which means the all detected pixels were inside the boundary of the object. As the reinitializing process obstructs the edge based active contour from reaching the global minima, the model could not specify the boundary of the object in high detail, especially at the corner of the object. The Hybrid LSM also faced this problem but after the refinement process the contour at zero level evolved until it covered all details of the object. The region based active contour had the lowest dice similarity result because the evolution process was terminated before the model reached the boundary of the object. The sensitivity result shows that there were many pixels inside the boundary of the object but this model could not detect them. So, we can see that the segmentation results of regions based active contour were smaller than the referenced boundary. Nonetheless, this method could handle the corner of the object. For the speed of the models, the region based active contour was the fastest, the second was Hybrid LSM and the last was the edge based active contour. The Hybrid LSM and the edge based active contour model used more iterations to reach the convergence because after they reached the first convergence point, they performed the refinement process until the model reached the next convergence point.

In experiment 2, we aimed to segment one object from the image which had two objects in the foreground. We set the initial level set function center of the desired object. The Hybrid LSM had the highest accuracy while the region based active contour model had the lowest accuracy. Due to the characteristic of the region based active contour model, this model segmented all images in the foreground which increased the false positive. Figure 3.3 (right) shows that the accuracy of the model was high at the beginning and then it slowly decreased until the model reached the convergence. That means the accuracy was high when the contour at zero level reached the boundary of the desired object and it decreased when the contour leaked out of the desired boundary and reached the boundary of the second object. After that, the model was stable when the boundaries of both objects were specified. The Hybrid LSM and edge based active contour model could segment only the desired boundary. However, the Hybrid LSM had higher accuracy. The edge based active contour had lower accuracy because of the reinitializing function. Figure 3.3 (right) shows that even though the refinement process was performed, the accuracy improved a little. For the Hybrid LSM, the accuracy was closer to the accuracy of edge based when the model reached the first convergence point, but after the refinement process the accuracy improved more than the edge based active contour model. The segmentation results of three methods are shown in Table 3.9. Notice that the even though the Hybrid LSM and the edge based active contour model could segment only the boundary of the desired object, they could not segment

the desired boundary in high detail especially at the acute angle corner. On the contrary, the region based active contour could handle this problem. For the iteration which the models used to reach the convergence point, Hybrid LSM was the fastest while the edge based active contour was the slowest.

In experiment 3, we aimed to segment the object which had unclear boundary. We set the initial level set function center of the object. The Hybrid LSM had the highest accuracy while the edge based active contour model had the lowest accuracy. In this experiment, the edge based active contour model failed to segment the boundary of the desired object. As shown in Table 3.9, the contour at zero level evolved until it reached the border of the image. Due to the limitation of the edge based active contour model, it could not handle the object without clear boundary. The stopping function did not become closer to zero at the boundary of the object because the boundary was unclear, so it could not be used to terminate the evolution process when the contour reached the desired boundary. Even though the contour at zero level leaked out of the boundary, it continued to grow until it reached the border of the input image. The model tended to remain stable when the contour covered the whole image as shown in Table 3.9. Figure 3.4 (left) shows that the accuracy of this model slowly increased at the beginning; however, after the contour leaked out of the desired boundary, the accuracy slowly decreased. Finally, the accuracy had a tendency to remain stable when the contour reached the border of the input image. So, the specification of this model was very low. The accuracy of the Hybrid LSM and region based active contour model was high because they could handle the object without clear boundary and because there is only one object in the foreground. However, the accuracy of Hybrid LSM was higher than the region based active contour model. The region based active contour had a lower dice similarity result because the evolution process was terminated before the model reached the boundary of the object. The sensitivity result shows that many pixels were inside the boundary of the object but this model could not detect them. So, we can see that the segmentation result of regions based active contour was smaller than the referenced boundary. For the Hybrid LSM, the refinement process helped to improve the accuracy of the model and this model also used the lowest iteration to convergence.

In experiment 4, the input image was similar to the input image in the experiment 2, but the boundaries of the objects were unclear. The Hybrid LSM had the highest accuracy while the region based active contour model had the lowest accuracy. In this experiment, the edge based active contour model failed to segment the boundary of the desired object. As shown in Table 3.9, the shape of segmentation result was not similar to the referenced boundary. Due to the limitation of the edge based active contour model, it could not handle the object which had unclear boundary. The stopping function based on the edge of the object terminated the evolution of the contour at zero level before it grew over the object. So, the sensitivity of this model was small. The performance of region based active contour in this experiment was similar to the performance in experiment 2 although the boundaries of the objects were unclear. For the Hybrid LSM, this model could segment the boundary of the desired object even though the boundary was unclear. Nonetheless, the accuracy was low because the shape of the input image was distorted by blurring. The model could not reconstruct the original shape of the object so, there were a lot of false positive and false negative when compared with the referenced boundary. For the iteration which the models used to reach the convergence point, Hybrid LSM was the fastest while the edge based active contour was the slowest.

In experiment 5, we used the multi gray-scale and multiple objects in the foreground for the input image. We aimed to segment only the rectangle object which had a lower intensity value than the intensity value of a circle object but higher than the intensity value of a square object. All the objects in the input image had clear boundaries. We set the initial level set function at the center of the desired object. The Hybrid LSM had the highest accuracy while the region based active contour model had the lowest accuracy. The accurate result of the region based active contour was low because this model attempted to segment all the objects in the foreground. Notice that the square object had been omitted. The intensity value of the square object was lower than the intensity value of the desired object, so the model decided that the square object was likely to belong to the background rather than the foreground. For the Hybrid LSM and edge based active contour model, the accuracy of both methods were similar; however, the Hybrid LSM had higher accuracy and the sensitivity result and specification result were also higher. That means our modified version of LSM could segment the boundary of the desired object in higher detail. However, there was some error at the acute angle corner of the object. For the iteration to convergence, the Hybrid LSM used the lowest iteration to reach the convergence point.

In experiment 6 to experiment 9, the input images were similar to the experiment 1 to experiment 4 but noises were added to the images. We aimed to evaluate how robust the models were to the noisy images. The edge based active contour model failed to segment the boundary of the object from the noisy images. The results of using this model are shown in Table 3.9. The contour at zero level could not reach the boundary of the desired object. Moreover the evolution process was terminated in the early stage because the stopping function based on the gradient of the image was disturbed by noises and became closer to zero inside the boundary of the desired object. The effect of noise can be alleviated by applying the Gaussian filter on the input image before performing the edge based active contour model but the Gaussian filter also blurred the boundary of the object. The Hybrid LSM and region based active contour model could resist to the noisy image. In Experiment 7 and Experiment 9, the region based active contour model was able to segment the boundary of the desired object and handle the detail of acute angle corner but the accuracy was low because the model was not terminated at the boundary of the desired object. While the Hybrid LSM had a higher accuracy result, it could segment the boundary of the desired object but there was some error at acute angle corner of the object. In Experiment 6, the segmentation results of both region based active contour model and Hybrid LSM were similar. However, the accuracy of Hybrid LSM was higher because the sensitivity result was higher. In Experiment 8, the accuracy of the region based active contour model was higher than the Hybrid LSM method because the sensitivity result was higher. Notice that the segmentation results of the region based active contour model was scattered because the model determined the noise as the objects in the foreground and it attempted to segment the boundary of all objects in the foreground image. For the Hybrid LSM, even though the accuracy was lower, the segmentation result was not scattered because the model determined that there was one object in the foreground and tried to segment the boundary of the object surrounding the initial contour. For the iteration to convergence, the Hybrid LSM used the lowest iteration to reach the convergence point.

In Experiment 10, the input image was the real medical image. This image was the cross section of the head so it presented a part of the brain and the skull. We aimed to segment the boundary of the skull. The initial level set function was set inside the

boundary of the skull. Although the segmentation results of three methods were similar, the Hybrid LSM had the highest accuracy because the sensitivity value was high and also used the lowest iteration to reach the convergence point.

According to the results from ten experiments, we can conclude that our modified version Hybrid LSM can segment the boundary of the desired object in high accuracy and resist the noisy image. This method continues to work well despite the objects' unclear boundary and can segment only desired objects from the image containing multiple objects in the foreground. Moreover, this method can reach the convergence faster than the conventional methods. However, some errors can occur when our proposed method is applied on the object which has acute angle corners. For the edge based active contour model, this method works well when it is applied on the object with clear boundary and used to segment only desired objects from images having multiple objects in the foreground. Nevertheless, this method cannot handle noisy images and also has some errors at acute angle corners. For the region based active contour, this method is good for the segmentation of objects, even those without clear boundaries. This method is resistant to noisy images and can handle the acute angle corner. However, this method usually generates a lot of false positive when it is applied on images which have multiple objects in the foreground. In some experiments, for instance, Experiments 1 and 3, the segmentation results from this method were a bit smaller than the referenced boundary because the evolution process was terminated before the model could reach the boundary of the object.

Dice similarity is used to evaluate the accuracy of the model by comparing the segmentation result with the referenced boundary. High dice similarity implies that the segmentation result is similar to the referenced one. The Hybrid LSM has higher dice similarity results than the edge based and region based active contour models except in Experiment 8. In that experiment, the accuracy of the region based active contour model was higher than the Hybrid LSM method. However, the segmentation results of the region based active contour model were scattered while the segmentation result of Hybrid LSM was not scattered and looked similar to the reference in spite of the lower dice similarity result.

Sensitivity is used to evaluate how the model can detect the true positive pixels in the dataset. High sensitivity implies that the model can detect most of the pixels of desired objects. The Hybrid LSM has high sensitivity results except in the case that the desired object has the acute angle corner, for instance Experiments 2, 4, 5, 7 and 9. In those experiments, the sensitivity results of the region based active contour model were higher than those of the Hybrid LSM because this model could handle the detail at the acute angle corner.

Specification is used to evaluate how the false positive detection. High specification implies that our results have low false positive detection and the contour at zero level of the model cannot leak out of the boundary of the desired object. The Hybrid LSM has higher specification results than the edge based and region based active contour models. However, in Experiments 6 to 9, the specification results of the edge based active contour model were equal to 100 percent because all detected pixels were desired objects. The evolution process of the model was nevertheless terminated in the early stage and the model failed to segment the boundary of the object from the noisy image.

Table 3.8 The experimental results

No.	Edge based active contour model				Region based active contour model				Hybrid LSM			
	Number of iteration	dice similarity	Sensitivity	Specification	Number of iteration	dice similarity	Sensitivity	Specification	Number of iteration	dice similarity	Sensitivity	Specification
1	1,763	99.92	99.84	100.00	682	99.37	98.75	100.00	924	100.00	100.00	100.00
2	3,845	97.67	96.96	98.38	2,953	66.72	99.93	50.07	1,425	98.27	96.60	100.00
3	14,921	49.46	100.00	32.86	4,636	97.32	94.77	100.00	718	99.32	98.65	100.00
4	17,344	82.05	74.80	90.85	10,060	58.02	86.46	43.66	974	85.88	82.33	89.74
5	1,748	98.74	99.81	97.70	3,775	57.00	100.00	39.86	906	99.16	99.53	98.79
6	5,762	4.25	2.17	100.00	5,144	99.63	99.38	99.89	1,012	99.69	99.63	99.75
7	5,566	16.48	8.98	100.00	7,218	66.30	99.20	49.78	836	96.98	94.28	99.85
8	7,109	8.93	4.67	100.00	14,918	91.51	89.67	93.43	1,209	86.41	76.08	100.00
9	5,566	16.48	8.98	100.00	14,404	58.29	85.81	44.13	1,013	84.03	76.97	92.52
10	8,746	96.20	99.60	93.02	4,821	98.16	99.92	96.47	1,250	98.44	100.00	96.92

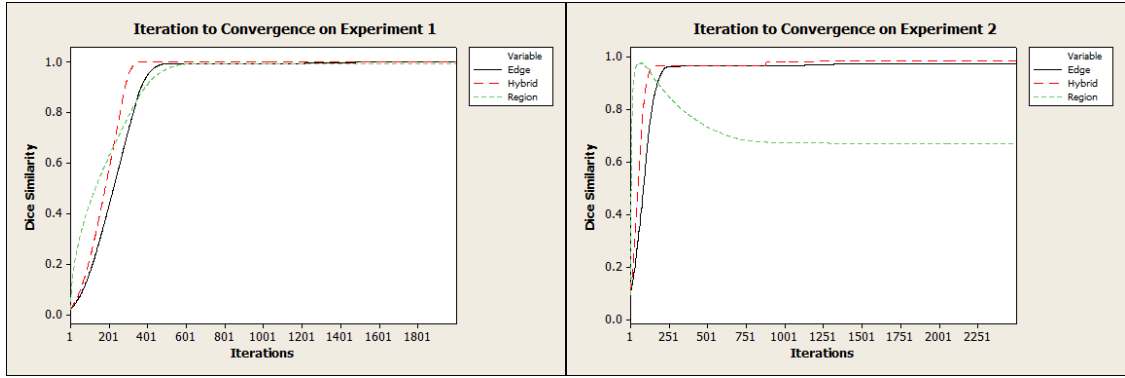


Figure 3.3 Iteration to convergence on experiment 1 and 2

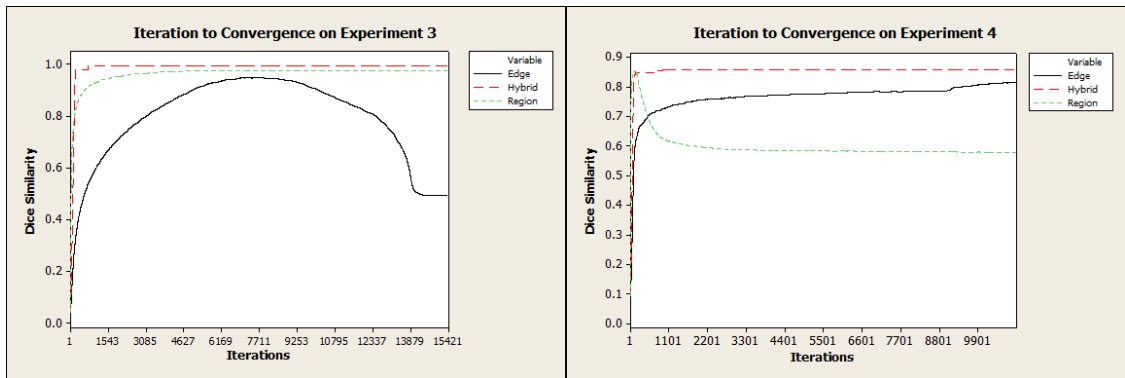


Figure 3.4 Iteration to convergence on experiment 3 and 4

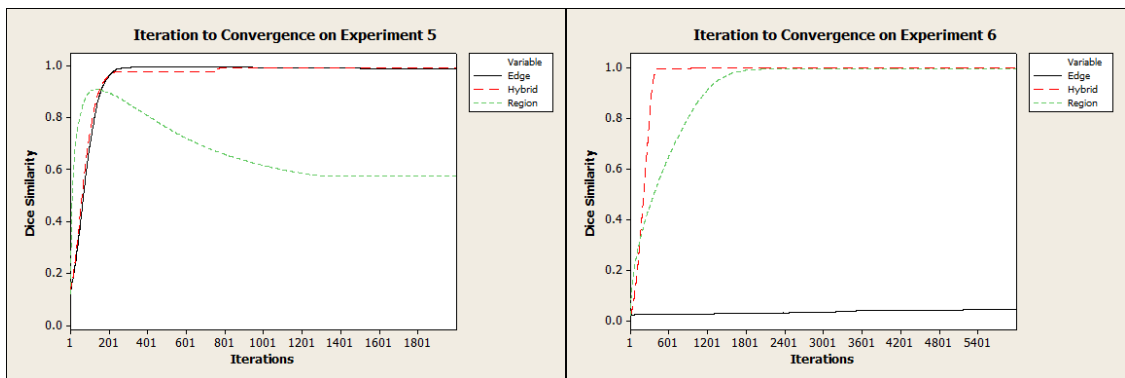


Figure 3.5 Iteration to convergence on experiment 5 and 6



Figure 3.6 Iteration to convergence on experiment 7 and 8

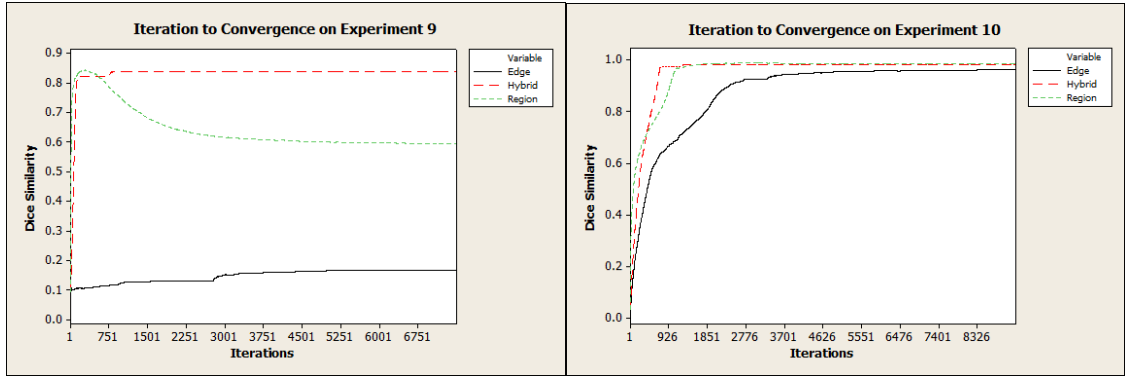
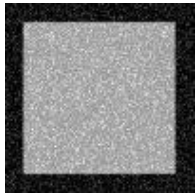
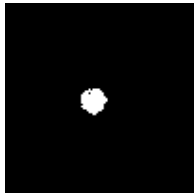


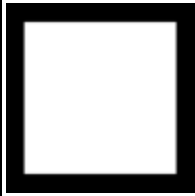
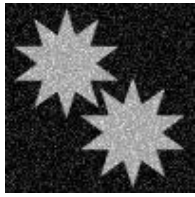


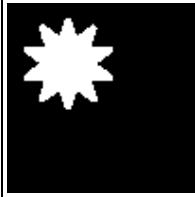
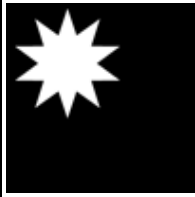

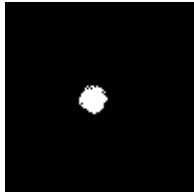

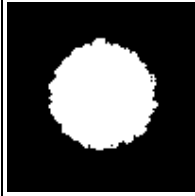
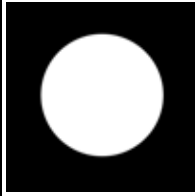
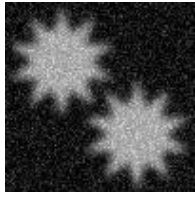

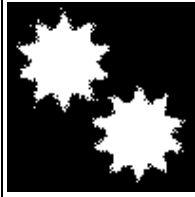

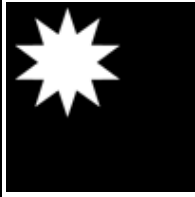
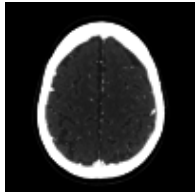






Figure 3.7 Iteration to convergence on experiment 9 and 10

Table 3.9 The segmentation results image in each experiment

No.	Input image	Edge based active contour model	Region based active contour model	Hybrid LSM	Referenced boundary
1					
2					
3					
4					
5					

Table 3.9 The segmentation results image in each experiment (Cont.)

No.	Input image	Edge based active contour model	Region based active contour model	Hybrid LSM	Referenced boundary
6					
7					
8					
9					
10					

CHAPTER 4 A FRAMEWORK FOR BLOOD VESSEL IMAGE SEGMENTATION

In this chapter, we discuss the Hybrid LSM in a framework which we used in order to segment the blood vessels from medical images. This framework is designed particularly for use with the CTA lumen segmentation of the carotid artery. The overview of our framework is illustrated in Figure 4.1.

Input data come from Carotid Lumen Segmentation and Stenosis Grading Challenge (CLS). Those are the set of CTA images which are the pixel data in 16-bits unsigned integers (Hounsfield units plus 1024). More details about the characteristic of our input data and the prior knowledge from training dataset are described in Section 4.1.

Bone segmentation is the preprocessing step. It aims to discriminate the bone structure from the vessel structure. Since the range of intensity value of bone overlaps with the range of intensity value of vessels, it causes errors in the vascular filtering process. In Section 4.2, we explain the bone segmentation process and the adaptation of Hybrid LSM in more details.

Histogram analysis process aims to specify the region of vessels more closely and eliminate other soft tissues which contaminate the vessels. This process adopts the Otsu's auto thresholding method to analyze the histogram which we describe in Section 4.3 in more details.

Vessel filtering is one of the important processes for carotid lumen segmentation. This process aims to specify the location of carotid artery by employing the vesselness filter to evaluate how likely it is for the voxel to belong to the vessels structure. Not only is the vesselness filter used to discriminate the vascular structure from the other, but it is also used to filter the size of the vessel. More details about the vessel filtering process are described in Section 4.4.

Vessel reconstruction process aims to reconstruct the vessels structure from the medial images dataset by using the information from the vessel filtering process. More details about vessels reconstruction process and adaptation of the center line extraction method and Hybrid LSM are described in Section 4.5.

For the visualization purpose, the last process is Ray Casting algorithm. We describe details of our framework to integrate the segmentation output with Ray Casting algorithm in Section 4.6. Lastly, in order to evaluate our achievement, the evaluation methods are described in Section 4.7.

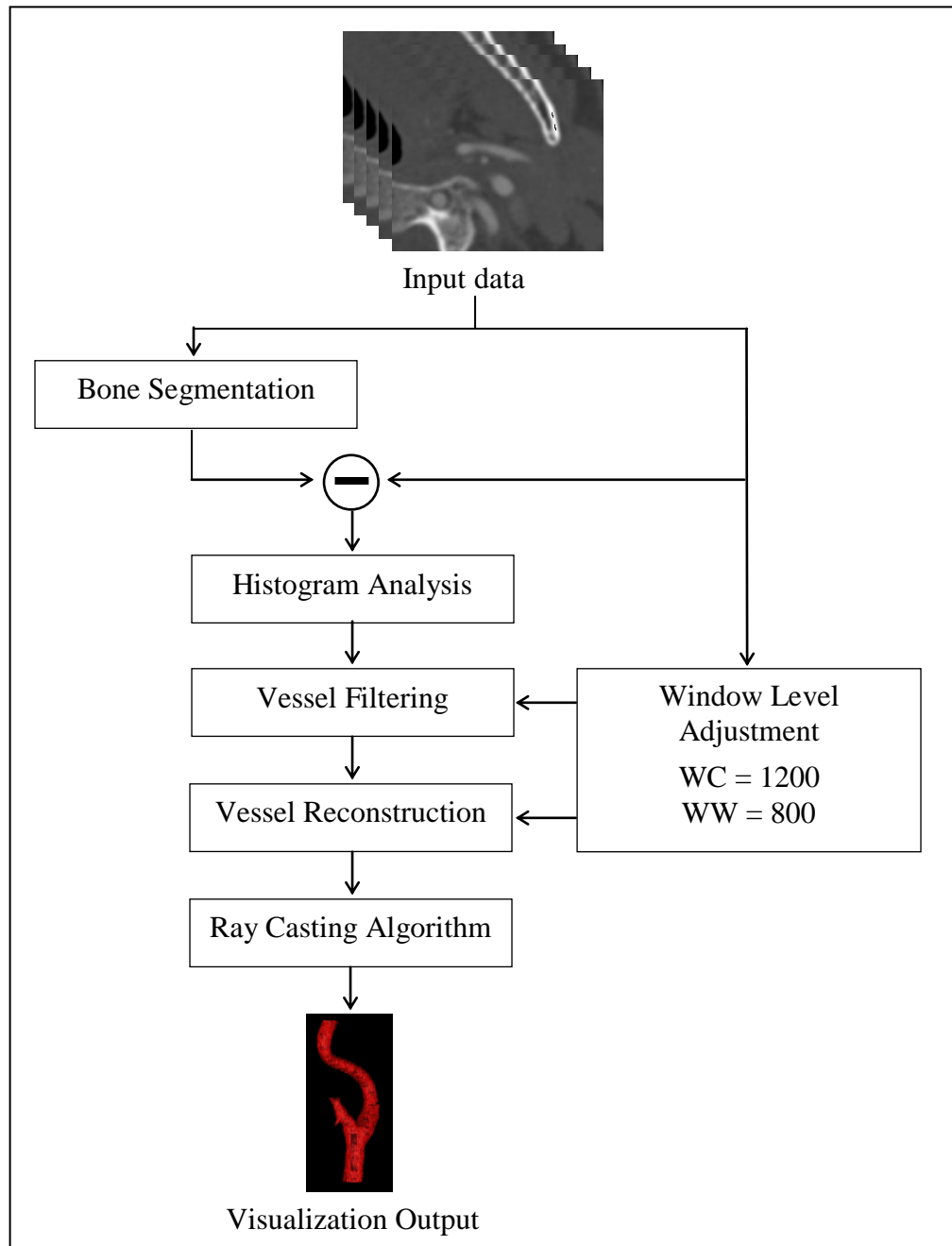


Figure 4.1 The summarization of our methodology work flow

4.1 Input data

Our input datasets come from Carotid Lumen Segmentation and Stenosis Grading Challenge (CLS). This challenge intends to compare the results of lumen segmentation and stenosis grading from different authors and different methods. They provide the datasets of the Carotid Bifurcation in a Computed Tomography Angiography (CTA). The Carotid Bifurcation consists of three main arteries: common carotid artery (CCA), internal carotid artery (ICA) and external carotid artery (ECA), as we have mentioned in Chapter 1.

The medical images in datasets are the images around the bifurcation where the CCA starts branching out into the lumen of the ICA and the lumen of ECA. The region of

interest covers at least 20 mm of the CCA below the bifurcation, at least 40 mm of ICA above the bifurcation and between 10 and 20 mm of ECA above the bifurcation, as depicted in Figure 4.2.

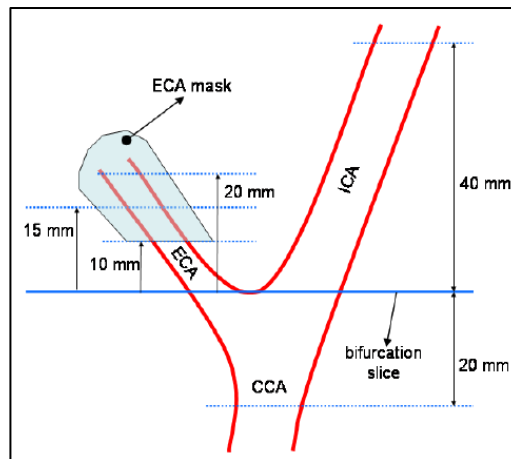


Figure 4.2 Schematic of the region of interest [26]

The provided datasets consist of 3 important sets of information as follows:

- Raw data of medical images: this data are the pixel data in 16-bits unsigned integers (Hounsfield units plus 1024).
- Region of interest: this data contain the minimum voxel index and the maximum voxel index which define the region to be segmented.
- The reference partial volume segmentation: this data is the partial volume of carotid lumen segmented by proficient radiologists manually. This data are in floating point value between 0 and 1.

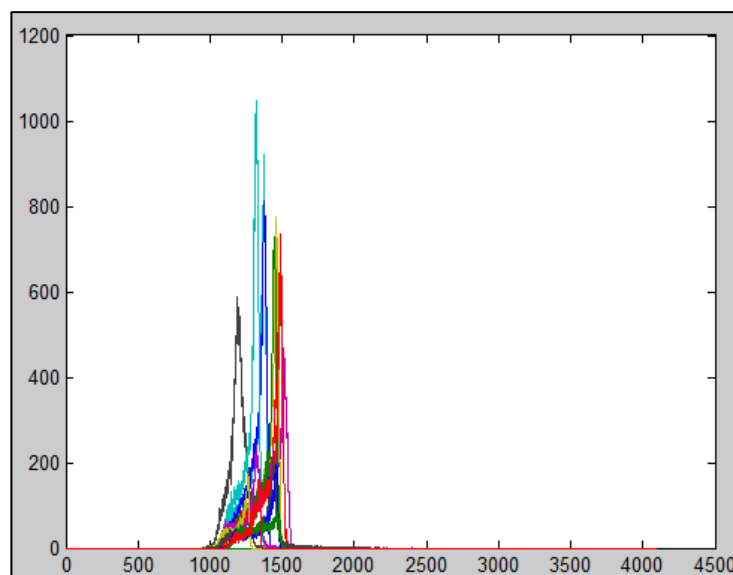


Figure 4.3 Histogram of the partial volume of carotid lumen from the training datasets

The histogram of the partial volume of carotid lumen from the training datasets is plot in Figure 4.3. Notice that the intensity values of vessels lie in the same range; the lowest value is 930 and the highest value is 2400. Notice that 99.24 percent of vessels lies in the range between 1060 and 1545. According to this information, we can conclude that

the range of vessels intensity value is between 1060 and 1545. The example image slices of the raw data are shown in Figure 4.4.

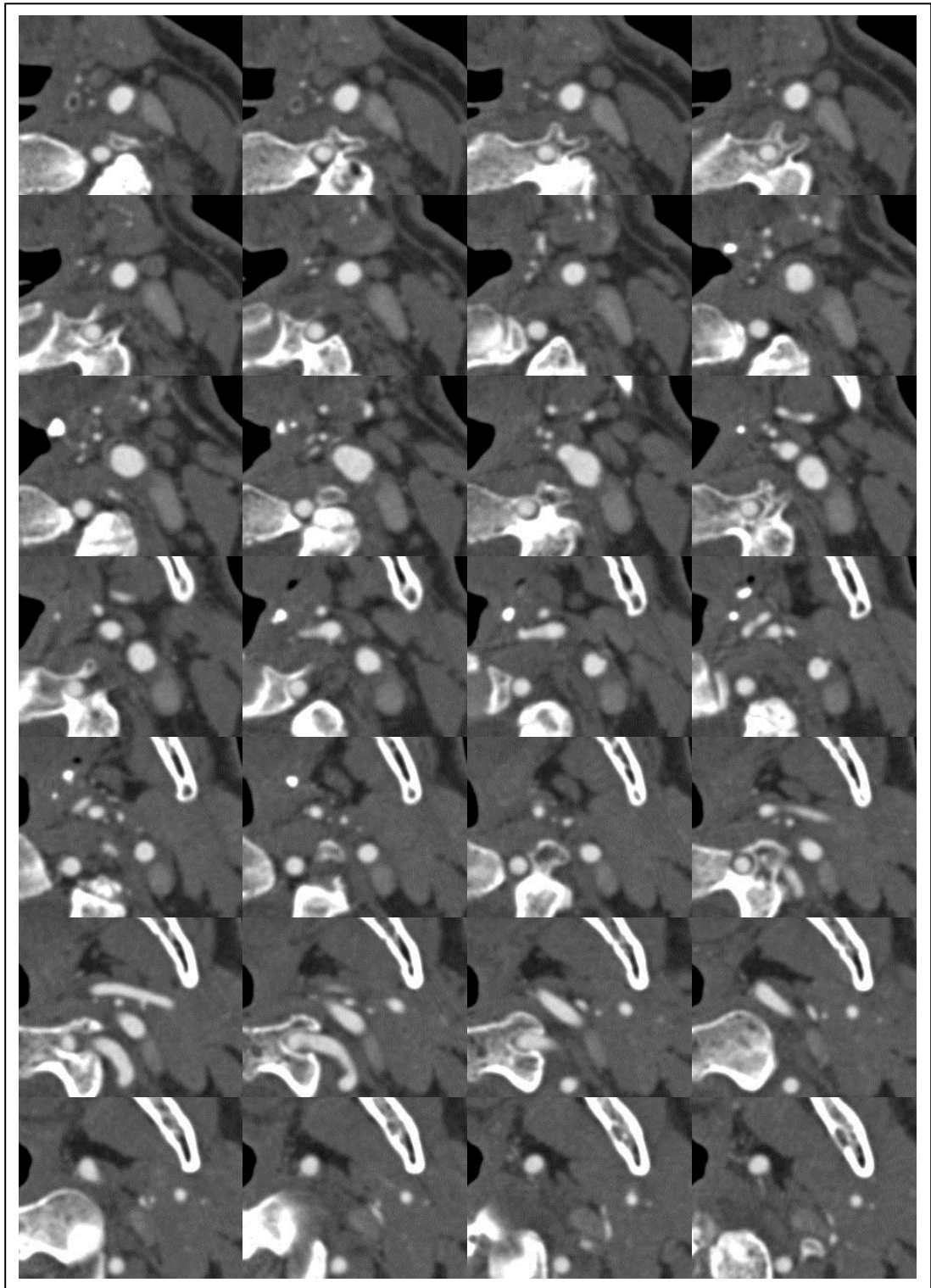


Figure 4.4 The example image slices of the raw data

4.2 Bone Segmentation

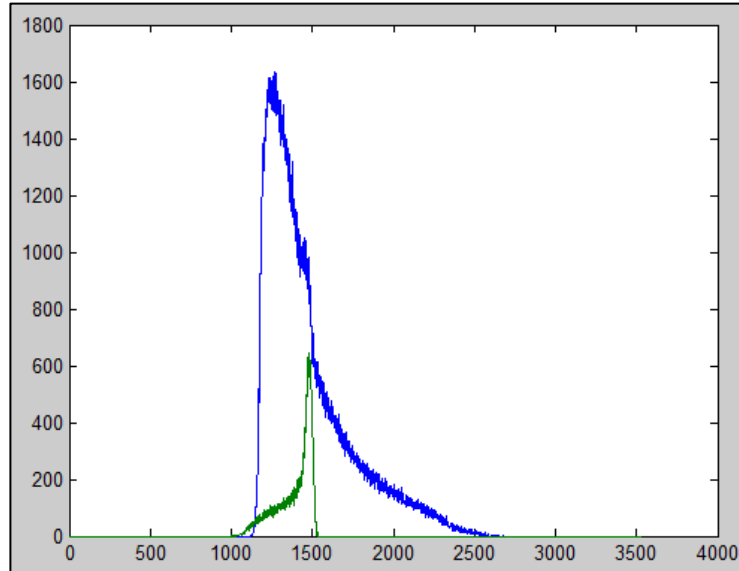


Figure 4.5 Histogram of bone (blue line) and carotid lumen (green line)

The bone segmentation is the preprocessing step for carotid lumen segmentation. As the range of intensity value of bone overlaps with that of carotid lumen, as shown in Figure 4.5, it causes a large number of errors in the vascular filtering process. The goal of this step is to define the locations of bones in the medical images and eliminate them before applying the Hessian-based vessel filtering to the medical images in the next step.

In order to define the boundary of the bones, we employed the Hybrid LSM. This technique is used to identify the boundary of the objects in the image and distinguish between foreground and background. It is also appropriate for the organ segmentation in medical images of which the boundaries are unclear. For bone segmentation, we started by generating the initialization of level set function and then created input images by adjusting the input data with the window level technique. Finally, we performed the Hybrid LSM and obtained the output. The example of an original medical image slice is shown in Figure 4.6. The final result of this process was the set of medical images where the intensity value of the bone voxels was set to be zero.

4.2.1 Initiating the Level Set Function

In this step, we attempted to specify the seed points which would ensure that all the seed points belong to bones and used them as the initial level set function for the Hybrid LSM. We performed this step for all the images in the dataset by applying the thresholding method and simple morphological operation.

First, we created the mask of vessels and bones by applying a thresholding to the original image. We then obtained Figure 4.7(a). The threshold value was the minimum intensity value of the vessel pixel which we obtained from the training datasets, as mentioned in Section 4.1. However, the result in Figure 4.7(a) did not cover all of the bone area. The intensity value of pixel inside the bone may be lower than the threshold value. Since the characteristics of bones consist of the strong osseous tissue on the surface (cortical bones) and the soft osseous tissue in the interior of bones (cancellous bones), the intensity value of pixel at the border of the bones is higher than the intensity

value of pixel inside the bones [15]. We applied a morphological operation “hole filling” to Figure 4.7(a) and obtained the mask of vessels and bones, as shown in Figure 4.7(b), in which “white” represents the location of bones and vessels and “black” represents the others. Despite our naming of “mask of vessels and bones”, this mask also covers parts of the other tissues.

The next step is creating the mask of vessels. We applied a double thresholding to the original image and obtained Figure 4.8(a). The threshold values are the minimum and maximum intensity value of the vessel pixel. Similar to the mask of vessels and bones, some pixels exist at the center of blood vessels which have a higher intensity value than that of the threshold. Then, we applied a morphological operation “hole filling” and obtained Figure 4.8(b). Even though we assign this mask as “mask of vessels”, it still covers parts of bones and the other tissues.

After that, we subtracted Figure 4.7(b) by Figure 4.8(b). The difference between them was the seed points which ensured that all the seed points belong to the bone, as shown in Figure 4.9. “White” represents the pixel which we defined as the initial level set function. An initial surface for this example is shown in Figure 4.10.

$$\phi_0(x) = \begin{cases} c & \text{if } I(x) \text{ is white} \\ -c & \text{otherwise} \end{cases}$$

where ϕ_0 is the initial level set function.

x is the specific location.

$I(x)$ is the intensity value of segmentation result at a location x .

c is the constant parameter.

Equation 4.1 Relationship between initial level set function and subtraction result.

Equation 4.1 defines the relationship between the initial level set function and the subtraction result. Where $\phi_0(x)$ is the value of initial level set function at location (x) , $I(x)$ is the intensity value of segmentation result and c is the constant value.

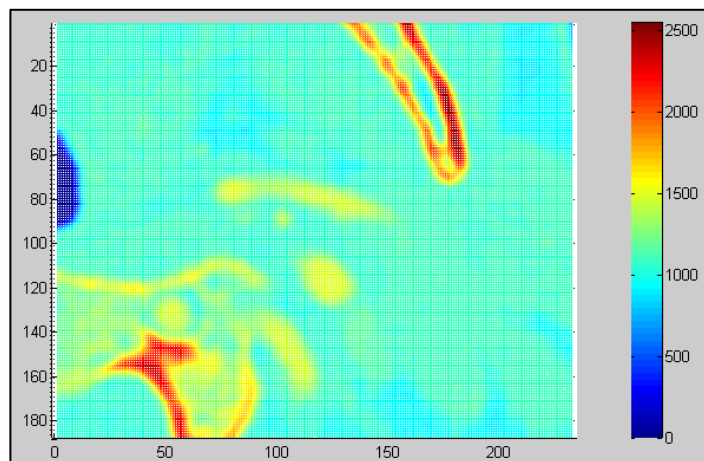


Figure 4.6 The example of slice of medical image

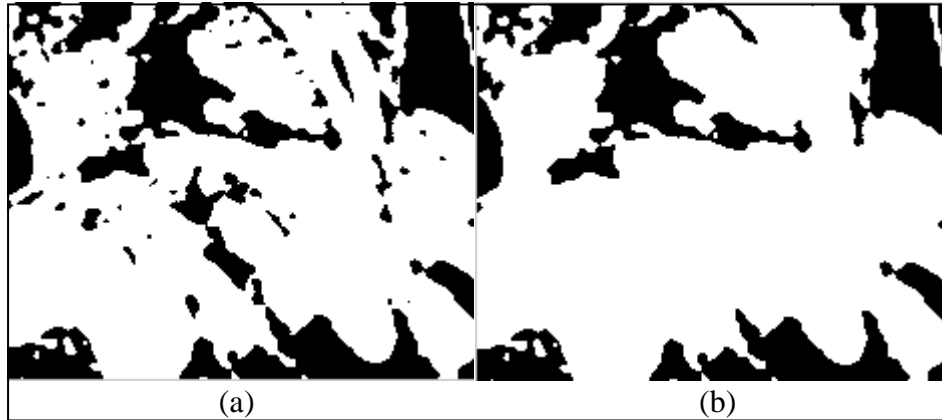


Figure 4.7 The example of slice after threshold by 1060 (a), and after fill the hole (b)

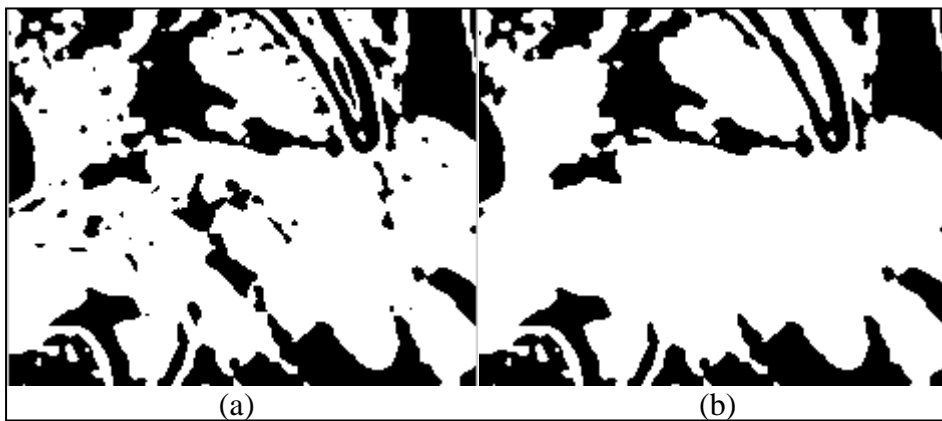


Figure 4.8 The example of slice after threshold by 1060 and 1545 (a), and after fill the hold (b)



Figure 4.9 The result of subtraction between Figure 4.7(b) and 4.8(b)

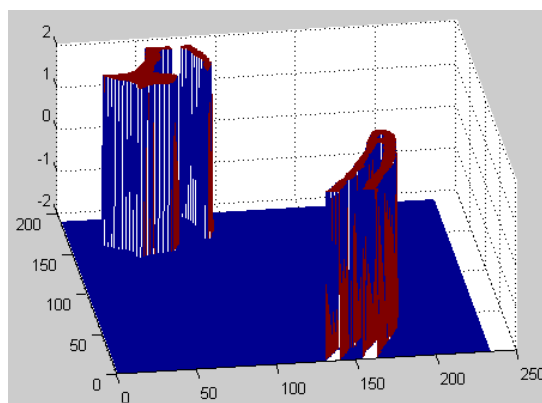


Figure 4.10 The example of slice of initial surface for level set function

4.2.2 Adjusting the Input Data

As the data from the original dataset are the pixel data in 16-bits unsigned integers which have the intensity value in Hounsfield unit (plus 1024), we were required to convert them to a normal gray scale image which has the intensity value between 0 and 255 before using them as the input of Level Set method. We opted for the window level technique. This technique changes the range of data which is presented on the gray scale image, as depicted in Figure 4.11. It is a common practice to visualize medical images in the gray scale image. By adjusting two parameter window center WC and window width WW , doctors can explore the details of organs in medical images. The equation of window level technique is defined in Equation 4.2. Figure 4.12 shows the comparison of two medical images which come from the same data but the window level is different.

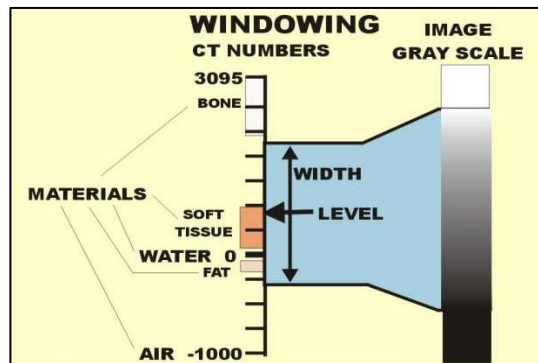


Figure 4.11 Depiction of window level

[Source: <http://www.sprawls.org/resources/CTIMG/module.htm>]

$$I'(x) = \begin{cases} 0 & \text{if } (I(x) \leq WC - 0.5 - \frac{(WW - 1)}{2}) \\ 1 & \text{if } (I(x) > WC - 0.5 + \frac{(WW - 1)}{2}) \\ \frac{I(x) - (WC - 0.5)}{(WW - 1) + 0.5} & \text{otherwise} \end{cases}$$

where I' is the intensity value after adjustment.

I is the intensity value of the input data.

x is the specific location.

WC is the window width.

WW is the window center.

Equation 4.2 Window level

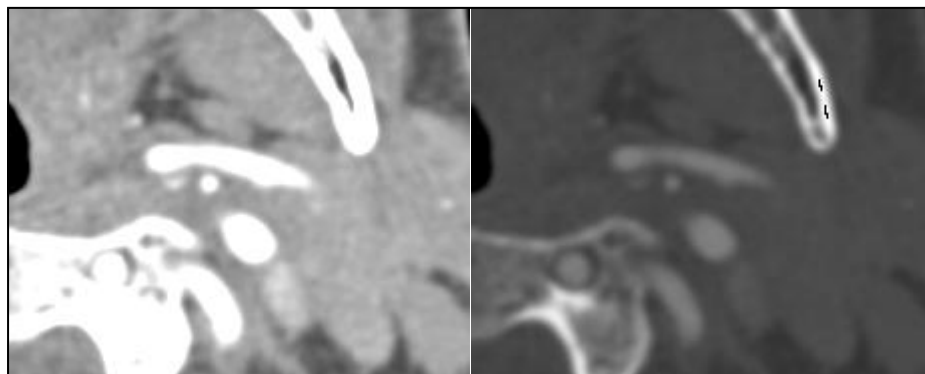


Figure 4.12 Example of window level adjustment. Left image $WC=1000$ $WW=500$
Right image $WC=1500$ $WW=2000$

According to our empirical study, the input data were adjusted by setting the window center to 2000 and window width to 2000 for the bone segmentation process. The range of window was between 1000 and 3000 which covers most details in the dataset. So, we obtained the images in which the bones and vessels were brighter than the other tissues, as shown in Figure 4.13.

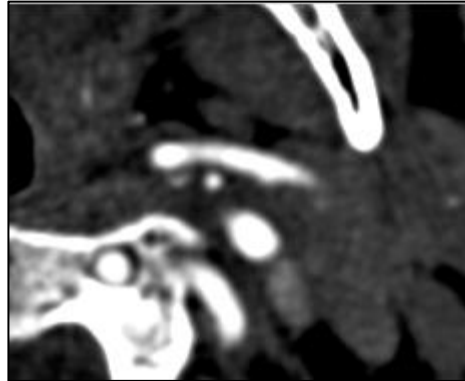


Figure 4.13 Example of slice of input data by setting WC=2000 WW=2000

4.2.3 Applying the Hybrid LSM to Bone Segmentation

Table 4.1 The constant parameters for Hybrid LSM in bone segmentation process

parameter	Hybrid LSM
c_0	2
μ	1
λ_1, λ_2	5
v	-1.5
ρ	0.2
ε	1.5

After initial level set function and input images were prepared, we performed the region growth by using the level set method. We adopted the Hybrid LSM model to evolve the level set function which could grow in a 3D space. We set the constant parameters as show in Table 4.1. Figures 4.13(a) – 4.13(c) show an example of the contour of the surface at zero level in iterations 25, 50 and 100, respectively. After the model reached the convergence point, the Hybrid LSM was terminated. We obtained the complete surface, and then we obtained the bone segmentation result by using Equation 4.3, where $I_s(x)$ represents the intensity value of segmentation result at position x in the 3D space and $\phi(x)$ represents the value of level set function at position x . Figure 4.15 shows an example of bone segmentation in which “white” represents the location of bones and “black” represents the others.

$$I_s(x) = \begin{cases} 1 & \phi(x) \geq 0 \\ 0 & \phi(x) < 0 \end{cases}$$

where I_s is the intensity value of segmentation result.

ϕ is the level set function.

x is the specific location.

Equation 4.3 Relationship between segmentation result ant level set function.

In the final step, we eliminated the bone structure in the original data by setting the intensity value of bone voxels to be zero, as shown in Figure 4.16. Equation 4.4 shows the relationship between the segmentation result I_s , original data I and the data after bones are eliminated I' , where x is the specific location in the 3D space. The whole steps in bone segmentation process are summarized and illustrated in Figure 4.17.

$$I'(x) = \begin{cases} I(x) & I_s(x) = 1 \\ 0 & I_s(x) = 0 \end{cases}$$

where I' is the intensity value after bones are eliminated.

I_s is the intensity value of a segmentation result.

I is the intensity value of the input data.

x is the specific location.

Equation 4.4 Bone elimination.

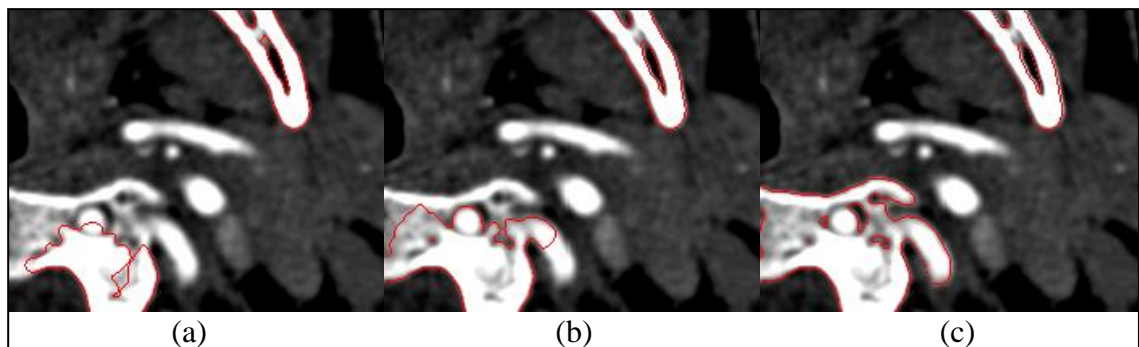


Figure 4.14 The example of slice of contour at zero level after iteration 25 (a), iteration 50 (b) and iteration 100 (c)



Figure 4.15 The example of slice of bone segmentation result

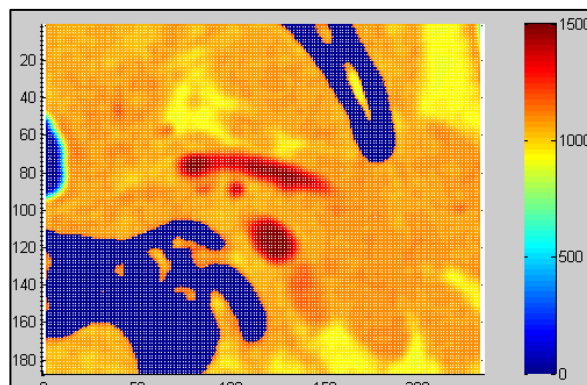


Figure 4.16 The example of slice of original data after bones are eliminated

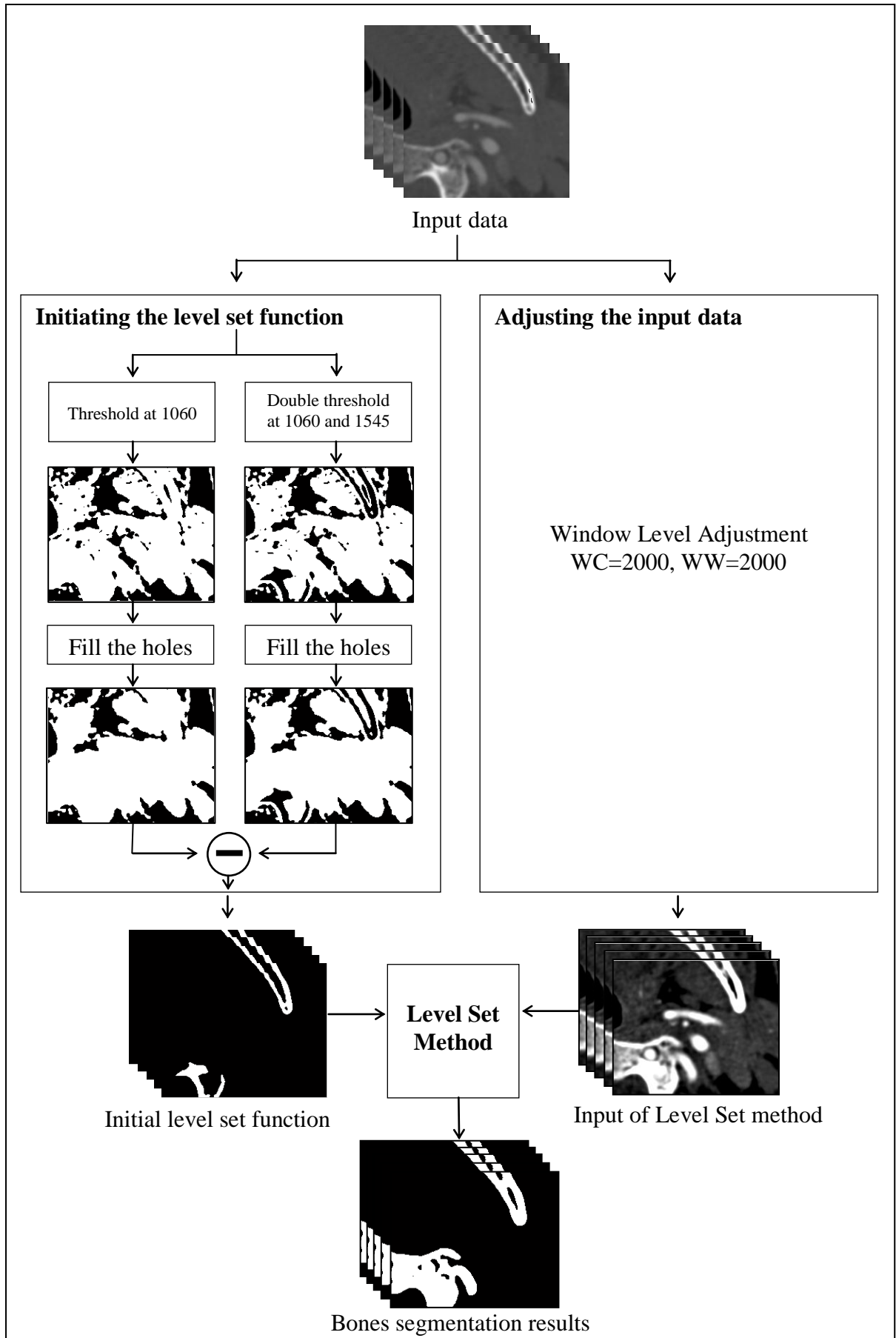


Figure 4.17 Summarization of bone segmentation process

4.3 Histogram Analysis

Even though bones had been eliminated from the data, other soft tissues still contaminated the vessels. In this process, we attempted to specify the region of vessels more closely. We deployed the prior knowledge from training datasets to eliminate voxels that were impossible to become vessels and used the histogram analysis technique to roughly discriminate the vessel and the other tissues.

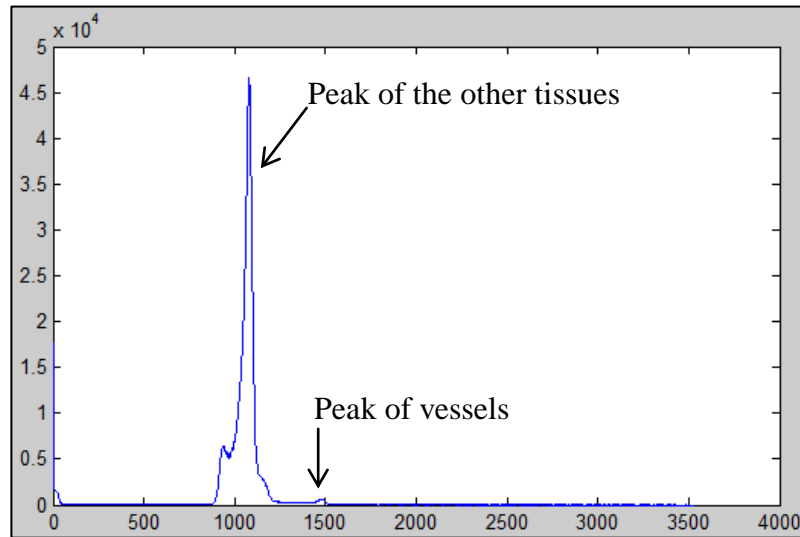


Figure 4.18 Histogram of data after bones are eliminated

Figure 4.18 shows the histogram of input data in which the bones are eliminated by the previous process. Notice that there are two peaks in the histogram; one is the peak of vessels and the other is that of the other tissues. In order to roughly specify the region of vessels, first we applied the double thresholding method to the input data. The range of threshold is the minimum and maximum intensity value of vessels which we obtained from the aforementioned training dataset in Section 4.1. The voxels which have the intensity value out of the threshold range had their intensity value set to be zero. This signifies that voxels impossible to be vessels would be eliminated. We then obtained the histogram of input data as shown in Figure 4.19.

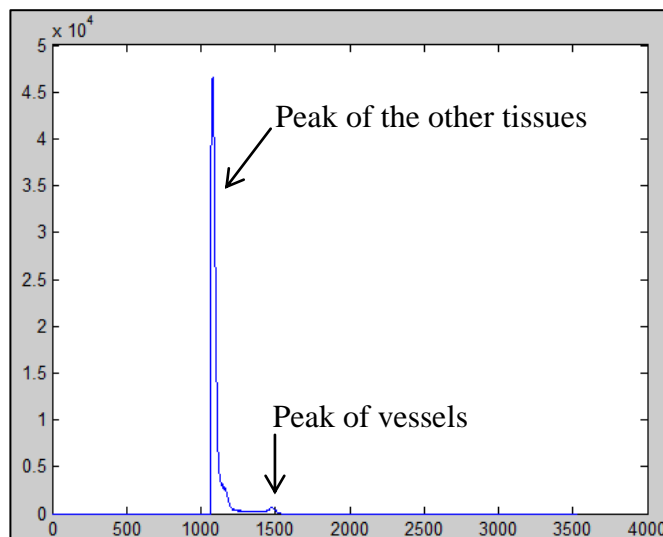


Figure 4.19 Histogram of data after filter by $1060 \leq \text{intensity value}$ and $\text{intensity value} \leq 1545$

Next, we opted for the Otsu's auto thresholding method to separate the peak of vessels from the peak of other tissues. The Otsu's method is used for converting the greyscale image to monochrome by using the appropriate threshold value. This technique assumes that the image consists of two classes of pixels: foreground and background. It tries to find the threshold value in which the summation of pixel spreads in the foreground and background is minimized. The within-class variance $\sigma_w^2(t)$ is the weighted sum of the variances of each cluster, as shown in Equation 4.5. Where $\omega_i(t)$ represents the probabilities of the class, $\sigma_i^2(t)$ represents the variances of the class and t represents the threshold value. This technique will calculate the within-class variance for all possible t value and choose the threshold value which minimizes the within-class variance.

$$\sigma_w^2(t) = \omega_1(t)\sigma_1^2(t) + \omega_2(t)\sigma_2^2(t)$$

where $\omega_i(t)$ is the probabilities of the class.

$\sigma_i^2(t)$ is the variances of the class.

t is the threshold value.

Equation 4.5 within-class variance

For the fast approach of Otsu's method, we can calculate the between-class variance and choose the threshold value which maximizes the between-class variance. The definition of between-class variance $\sigma_b^2(t)$ is shown in Equation 4.6 where σ^2 represents the variance of the global and $\mu_i(t)$ represents the mean of the class.

$$\sigma_b^2(t) = \sigma^2 - \sigma_w^2(t) = \omega_1(t)\omega_2(t)[\mu_1(t) - \mu_2(t)]^2$$

where σ^2 is the global variance.

$\sigma_w^2(t)$ is the within-class variance.

$\mu_i(t)$ is the mean of the class.

$\omega_i(t)$ is the probabilities of the class.

t is the threshold value.

Equation 4.6 Between-class variance

After this process finished, we obtained the vessels mask. It is the binary dataset in which the voxel of vessels was represented by 1 and the other tissues by 0. However, some vessel voxels which have the lower intensity value than the threshold value were also eliminated in this process. The sample output slice form this process is shown in Figure 4.20.



Figure 4.20 The example of slice of vessels mask

The advantages of this process are that not only can it roughly specify the region of vessels, but it can also reduce the computation time of Hessian-based vesselness filter in the next process. As the vesselness filter is a voxel-based approach, it consumes plenty

of computation time which can be significantly reduced by selecting the appropriate region of interest.

4.4 Vessel Filtering

Vascular filtering is one of the important processes for carotid lumen segmentation. In this process, we aimed to specify the location of carotid artery consisting of CCA, ICA and ECA. We employed the vesselness filter to evaluate the likeliness of the voxel belonging to the vessels. The vesselness filter was used to both discriminate the vascular structure from the other structure and to filter the size of vessels. In addition, the output of Hessian matrix was used in the center line extracting process. Therefore, in this process, we started with vesselness measurement, and then we filtered the small capillaries out while the main arteries were preserved. After this process, we obtained the vesselness response, corresponding eigenvector and eigenvalue for each voxel in the dataset.

First, after we obtained the vessel mask from the previous process, we used it as the region of interest where the vesselness of voxel inside this region was measured while the vesselness of voxel outside this region was set to be zero. The relationship between vesselness response and the region of interest is defined by Equation 4.7 where $v(x)$ represents the vesselness response and $v'(x)$ represents the vesselness response after filter by the region of interest.

$$v'(x) = \begin{cases} v(x) & \text{if } x \text{ inside region of interest} \\ 0 & \text{otherwise} \end{cases}$$

where $v'(x)$ is the vesselness response after filtering by the region of interest.

$v(x)$ is the vesselness response.

x is the specific location.

Equation 4.7 Relationship between vesselness response, input data and vessel mask.

It can be observed that we did not directly apply the vesselness filter on the vessel mask but we applied it on the input data. Our input data for this process were the data adjusted with the window level method. Since the vesselness filter works on the assumption that the colors of the vessels are brighter than those of the background, we set the window center as 1200 and the window width as 800. Those values were standard window level setting [26]. The example of slice of adjusted data is shown in Figure 4.21.

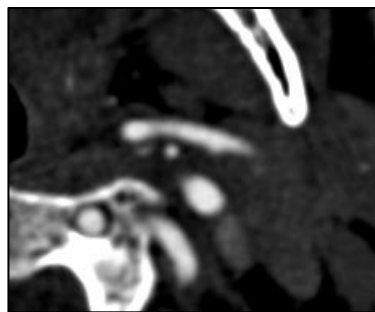


Figure 4.21 Example of slice of input data by setting WC=1200 WW=800

The vesselness measurement model is shown in Equation 2.24. We set the constant parameters α and β to be 0.5 and the parameter c is the half of the maximum value of S [20]. As this process performed in a multi-scaled vessel filtering fashion, the scale

factor σ enabled us to filter only the main arteries. So, we selected the value of scale factor which could cover all parts of the three main vessels based on our empirical study. We set the σ equal to 10 and 15. The multi-scaled vesselness response was computed by using Equation 2.25. The summarization of constant parameters setting is shown in Table 4.2.

Table 4.2 The constant parameters for vesselness measurement model

parameter	Vesselness model
α	0.5
β	0.5
c	the half of the maximum value of S
σ	10 and 15

4.5 Vessels Reconstruction

In this process, we aimed to reconstruct the vessels structure from the medical images by using the information which we obtained from the previous process: the vesselness response and corresponding eigenvector for each voxel. The reconstruction process started from interpreting the vesselness response, and then we extracted the center line of the vessels by using the eigenvector. Finally, we specified the boundary of the vessels by adopting the Hybrid LSM.

In terms of vesselness response interpretation, we defined that a voxel with the vesselness response higher than 30 percent of the maximum vesselness response is a voxel of vessels. The others were justified as noises. Figure 4.22 shows the scatter plot of data which are justified as vessels. Notice that some parts of the voxel are missing. Therefore, the reconstruction of the whole vessel structure must be done by using Hybrid LSM.

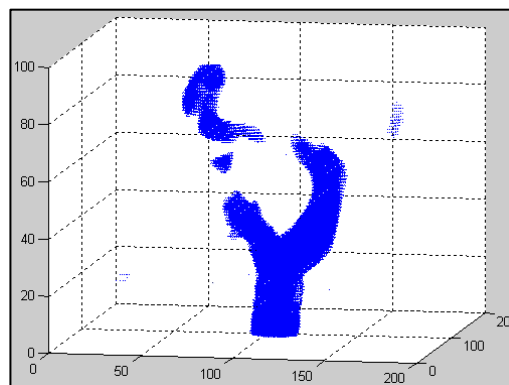


Figure 4.22 Scatter plot of data which are justified as vessels

Next, we applied the center line extraction technique on the voxel of the vessels. As we have mentioned in Section 2.2.3, the main idea of center line extraction is based on the fact that the voxels at the center of vessel have the highest vesselness response among their surrounding neighbors. This method is used to suppress the voxels of vessels which have lower vesselness response than that of any neighbor around it in the vessel orientation while the voxels at the center line are preserved. However, there are some failures of the method. It cannot suppress the noises which have no surrounding neighbors. As a result, we added the rule that the voxels which belong to the center line must have at least 8 surrounding neighbors. After this step, we received the set of voxels

which lie at the center of vessels corresponding to the vessel direction, as shown in Figure 4.23. Notice that there are some disconnected points in the center line. However, those points are sufficient to be used as the initial level set function.

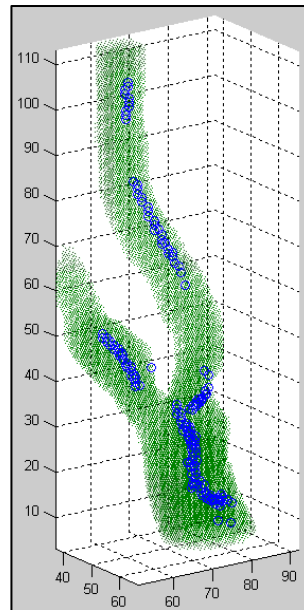


Figure 4.23 Scatter plot of center line

Table 4.3 The constant parameters for Hybrid LSM in vessel reconstruction process

parameter	Hybrid LSM
c_0	2
μ	1
λ_1, λ_2	5
ν	-1.5
ρ	0.2
ε	1.5

After the initial level set function was prepared, we were able to perform the Hybrid LSM. The input data for this step were the data adjusted with the window level method. We set the window center as 1200 and the window width as 800. This input data were the same data as the input of vesselness filter process. The constant parameters were set as show in Table 4.3. After the model reached the convergence point, the Hybrid LSM was terminated. We obtained the complete level set function, and then we received the vessel structure by using Equation 2.5. The vessels segmentation process is consequently finished. The example of slice of vessels segmentation result is shown in Figure 4.24. The scatter plot of segmentation result is shown in Figure 4.25.



Figure 4.24 Example of slice of segmentation output

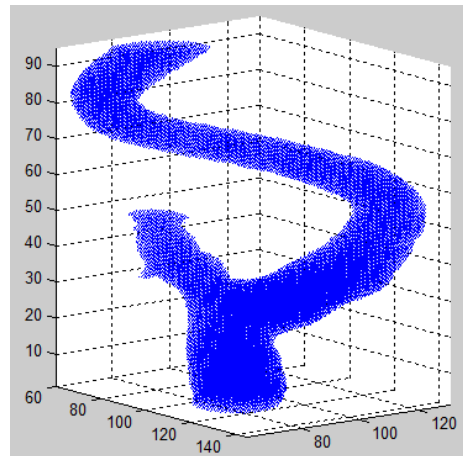


Figure 4.25 Scatter plot of segmentation result

4.6 Integration with Ray Casting Algorithm

For the visualization purpose, we integrated the segmentation result with the Ray Casting algorithm. This algorithm is one of the Direct Volume Rendering techniques (DVR). It is adopted to visualize medical images in a 3D model.

In conventional Ray Casting algorithm, the classification process relies on the transfer function to convert the intensity value of each voxel in dataset to the opacity value. f_v is the intensity value of the selected organ which is the important parameter of transfer function. The result image will present the organ segment which has intensity value similar to f_v . By adjusting the value of f_v , the Ray Casting can present the visualization of the organs which have a different intensity range. However, this technique cannot clearly present the visualization of the blood vessels where the range of intensity overlaps with the range of intensity value of bones and other tissues.

Instead of the conventional transfer function, the segmentation output of vessels was adopted. We can use the segmentation output to assign the appropriate opacity value to the voxel of vessels. Hence, the voxel which was determined to belong to the vessels by segmentation output is assigned high opacity value. On the other hand, the voxel which was not determined to belong to the vessels is assigned low opacity value, as shown in Figure 4.26(top).

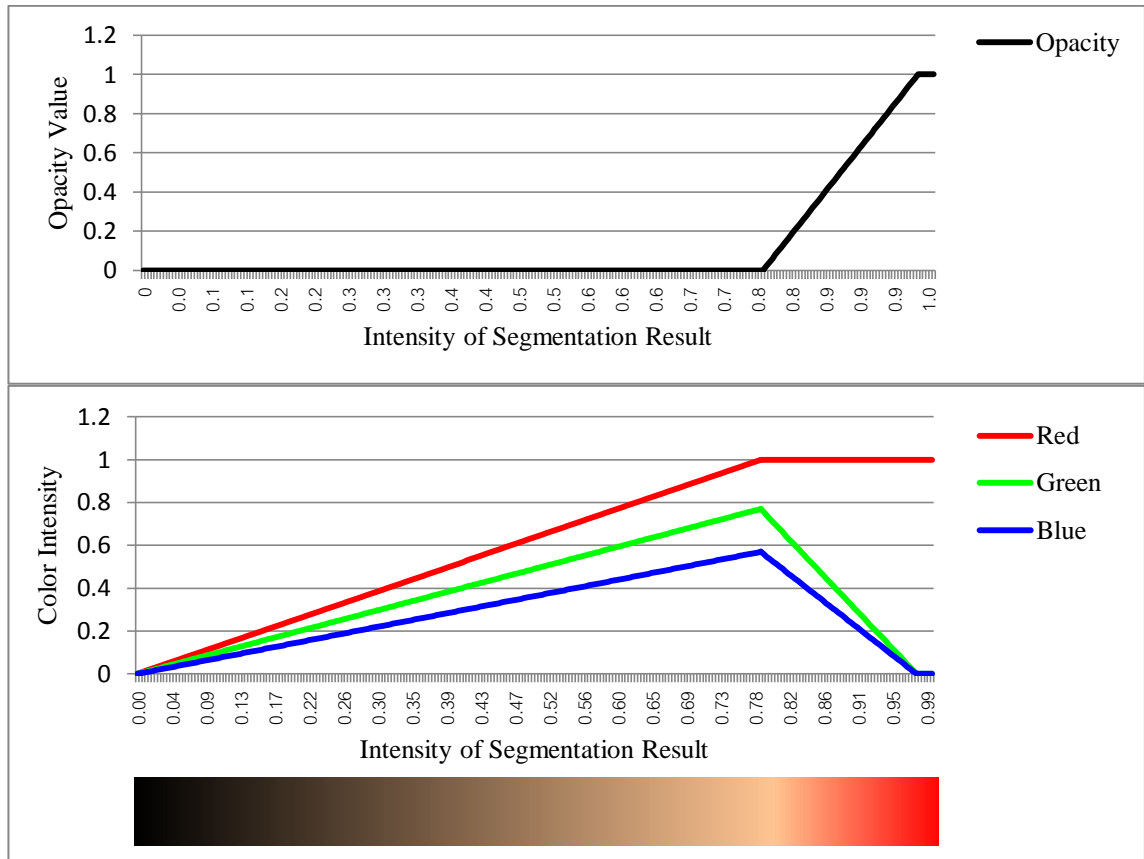


Figure 4.26 Graph of transfer function and opacity mapping (top) and graph of color transfer function (bottom)

Figure 4.26 (top) shows the graph of transfer function and opacity mapping. This graph is used to map the intensity value of vessels segmentation results to the opacity value. Notice that the opacity value is highest when the segmentation result is equal to one. Figure 4.26 (bottom) shows the graph of color transfer function. This graph is used in the shading process of Ray Casting. It maps the intensity of segmentation result to the color which is presented in the 3D visualization of images. For this example, the voxels of segmentation result which have high intensity are mapped in red which represents the artery blood vessels.

Finally, the opacity value and color of each voxel in the dataset were kept in another three dimensions array. This array would be sent to the composition process where the opacity value would be blended with the assigned color to generate the visualization of three dimensions model image as show in Figure 4.27.



Figure 4.27 Example of visualization result

4.7 Evaluation Method

In order to evaluate the achievement of our research, we compared our segmentation results with the manual segmentation based on the evaluation framework defined by the Carotid Lumen Segmentation and Stenosis Grading Challenge (CLS). There are three performance measurements.

4.7.1 The Dice Similarity D_{si}

$$D_{si} = \frac{2TP}{((TP + FN) + (FP + TP))}$$

where TP is the number of True Positive voxel.

FN is the number of False Negative voxel.

FP is the number of False Positive voxel.

Equation 4.8 The dice similarity

Where True Positive, TP , is the number of voxels defined as vessels by reference and we can correctly detect them, False Positive, FP is the number of detected voxels not defined as vessels by reference and FN is the number of voxels defined as vessels but we cannot detect them.

Dice similarity is used to measure the accuracy of our results by comparing them with the references.

4.7.2 Sensitivity

$$\frac{\text{number of overlapped contours}}{\text{total number of contours in manual segmentation}} \times 100\%$$

Equation 4.9 Sensitivity

The sensitivity is used to evaluate the performance of the proposed method. It measures how our method can detect the voxel of vessel in the dataset. High sensitivity implies the good performance.

4.7.3 Specification

$$\left(1 - \frac{\text{number of detected contours but not in reference}}{\text{total number of detected contour}}\right) \times 100\%$$

Equation 4.10 Specification

The specification is used to evaluate the false positive detection. High specification implies that our results have low false positive detection.

CHAPTER 5 RESULTS AND DISCUSSTION

In this chapter, we present our experimental results. In Section 5.1, we present the information about input datasets. Next, in Section 5.2, we present the visualization in 3D model for each dataset. Next in Section 5.3, we present the evaluation results by using the evaluation methods as we proposed in Section 4.7. Finally, at the end of this chapter, we discuss about our results and our method.

5.1 Experimental Inputs

The experiment was tested on four datasets of CTA medical images. The information about each dataset is shown in Table 5.1. The same set of constant parameters was used for all the experiments. Before the start of each dataset test, the experiment was setup by loading data to the memory and cropping only the region to be segmented. All experiments were run on CPU Intel Xeon E5620 2.40GHz, 16 GB memory and the source code was implemented by Matlab R2011a.

Table 5.1 Information about input datasets.

Data No.	Size	Side	Size of ROI	Position of ROI		Pixel Spacing
				min	max	
1	512x512x516	Left	233x188x95	203,19,269	435,206,363	0.26x0.26x0.6
2	512x512x501	Right	210x273x96	11,74,194	220,346,289	0.23x0.23x0.6
3	512x512x466	Left	210x254x97	284,49,203	493,302,299	0.23x0.23x0.6
4	512x512x497	Left	197x203x100	300,51,232	496,253,331	0.23x0.23x0.6
5	512x512x504	Left	218x179x95	247,161,244	464,339,338	0.26x0.26x0.6
6	512x512x524	Right	191x205x97	23,84,271	213,288,367	0.23x0.23x0.6
7	512x512x491	Left	210x220x98	286,146,227	495,365,324	0.23x0.23x0.6
8	512x512x564	Left	211x264x97	243,99,283	453,362,379	0.23x0.23x0.6
9	512x512x539	Left	185x245x96	277,93,262	461,337,357	0.25x0.25x0.6
10	512x512x786	Right	103x133x121	149,218,325	251,350,445	0.53x0.53x0.5
11	512x512x711	Left	84x103x121	273,270,265	356,372,385	0.59x0.59x0.5
12	512x512x570	Right	144x126x121	128,169,116	271,294,236	0.44x0.44x0.5
13	512x512x673	Left	80x86x134	267,203,302	346,288,435	0.58x0.58x0.45
14	512x512x666	Left	127x108x134	287,159,333	413,266,466	0.44x0.44x0.45
15	512x512x748	Right	83x100x134	130,181,248	212,280,381	0.45x0.45x0.45

Table 5.1 shows the information about input datasets. The Size column shows the dimension of each dataset and the Side column indicates which side of carotid artery is covered by region of interest. “Left” stands for the left carotid artery and “Right” stands for right carotid artery. The size of ROI shows the dimension of region of interest; the position of ROI indicates the minimum and maximum voxel of ROI in world coordinates of each dataset and pixel spacing indicates the ratio of one voxel in millimeter.

5.2 Visualization of Segmentation Results

After the vessels segmentation finished, we visualized the segmentation results and reference dataset by using Ray Casting algorithm. The visualization results are displayed in Table 5.2. Column “No.” is the dataset number; column “Visualization results” displays the visualization of our segmentation results and column “References” displays the visualization of references generated from the reference dataset provided by CLS.

Table 5.2 Visualization of segmentation results and references




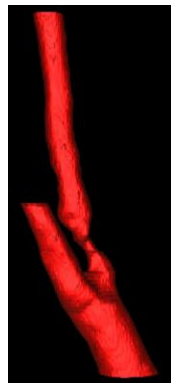
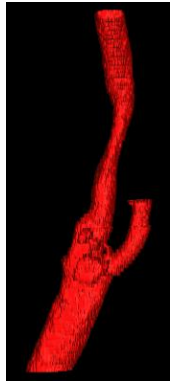
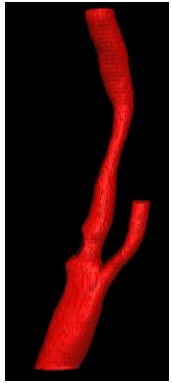
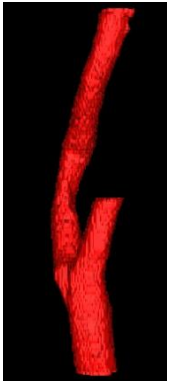
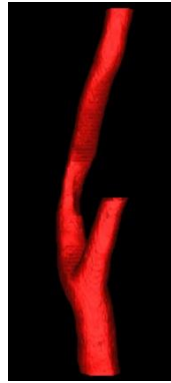
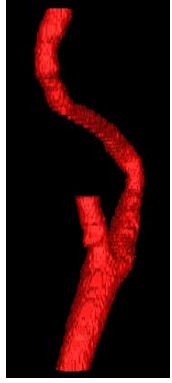
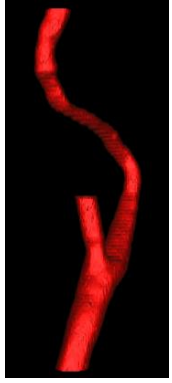
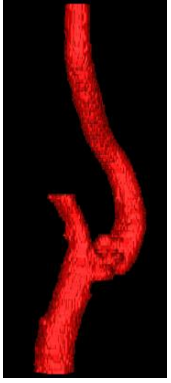
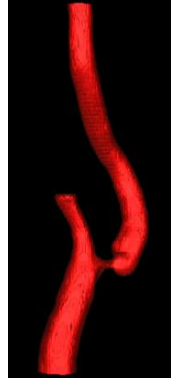



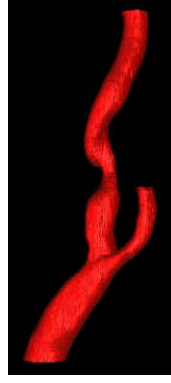

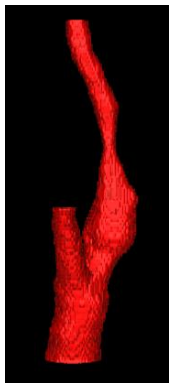

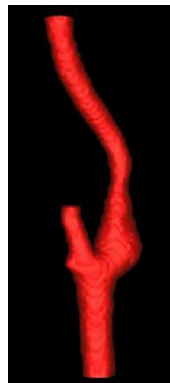
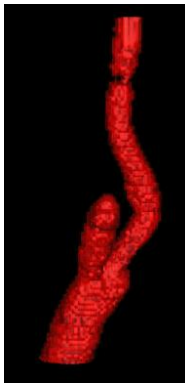


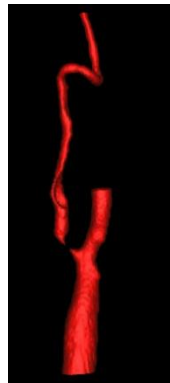


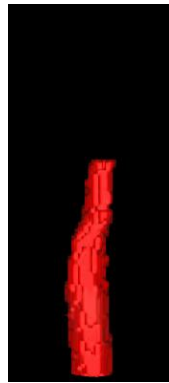

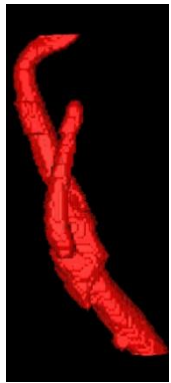

No.	Visualization results	References	No.	Visualization results	References
1			5		
2			6		
3			7		
4			8		

Table 5.2 Visualization of segmentation results and references (Cont.)

No.	Visualization results	References	No.	Visualization results	References
9			13		
10			14		
11			15		
12					

5.3 Evaluation Results

We used three measurement matrixes to evaluate our proposed method: dice similarity, sensitivity and specification. The evaluation results for each dataset are shown in Table 5.3 and the average of the evaluation results in Table 5.4.

Table 5.3 Evaluation results

Data No.	D_{si}	Sensitivity	Specification
1	90.99	89.96	92.03
2	88.23	86.72	89.79
3	95.48	96.40	94.57
4	86.36	88.81	84.04
5	84.65	91.56	78.72
6	90.69	88.38	93.13
7	84.53	88.21	81.14
8	83.07	89.50	77.51
9	88.89	95.67	83.01
10	90.02	81.85	98.37
11	86.68	84.61	88.85
12	72.04	85.47	62.26
13	87.25	81.70	93.60
14	92.78	92.58	92.99
15	84.71	80.46	89.43

Table 5.4 Average of evaluation results

Average D_{si}	Average Sensitivity	Average Specification
87.09	88.13	86.63

5.4 Discussion

According to the evaluation results shown in Table 5.3 and the visualization results shown in Table 5.4, we can conclude that our segmentation results were similar to the references [26] which were segmented by the manual method. We obtained the average Dice Similarity at 87.09, the average sensitivity at 88.13 and the average specification at 86.63. Notice that the average sensitivity result was high which indicates that our framework could detect most of the voxels defined as vessels. Moreover, the average Dice Similarity was high. This measurement indicates the accuracy of our framework. The average specification was not equal to 100% which indicates that there were some false positive errors.

As reported in the results, the dataset which Hybrid LSM performed poorly were datasets 5, 7, 8, and 12. These datasets have a common characteristic as there was the stenosis point around the bifurcation of the arteries. These points obstructed the model to grow over the vessel. Those errors were presented as the swollen surface around the bifurcation in the visualization results. Since those errors were false positive, the specification results were low. The voxels with high intensity around the bifurcation did exist but specialists did not regard them as vessels, as shown in Figure 5.1 left. Those voxels were the two juxtaposed vessels, so the model was confused and defined those voxels as a part of vessels, as shown in Figure 5.1 right. The red area represents the

segmentation result and the yellow area represents the overlapping area between segmentation result and reference.

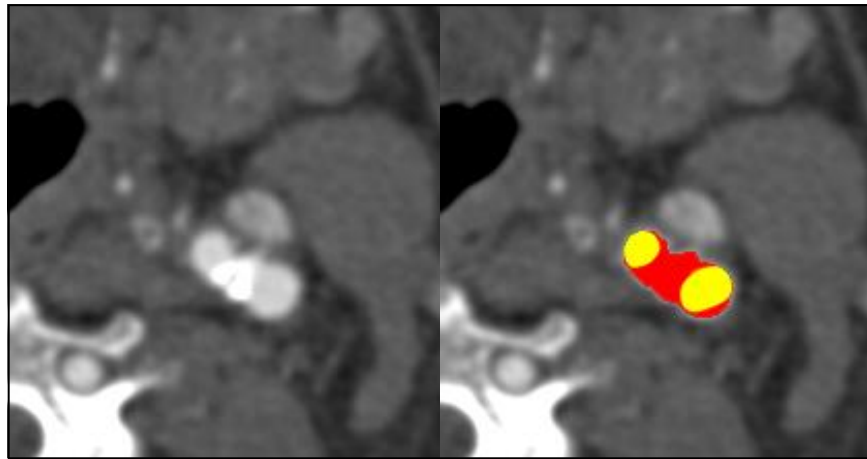


Figure 5.1 An example of medical image slide around the bifurcation.

Another cause which affects the evaluation results is that the segmentation results do not completely overlap with the reference even though the visualization of segmentation results and the reference are similar. As the references were segmented by the manual method, there were some voxels which the specialists may have missed and therefore was not defined as vessels. Nonetheless, our framework could detect them. On the other hand, some voxels having low intensity value were defined as vessels but our framework could not detect them. Figure 5.2 presents a cross section example of the overlaying segmentation result and reference on the medical image slide: the red area represents the segmentation result the green area the reference and the yellow area the overlapping area. Notice that even though both segmentation result and reference have a similar shape and area, they do not completely overlap with each other.

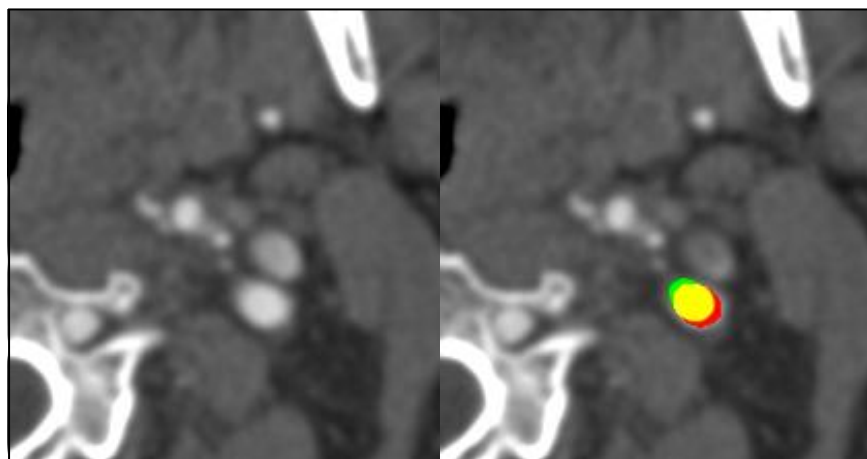


Figure 5.2 An example of cross section of the segmentation result and reference overlay on the medical image slide

According to the visualization results, notice that the surface of references is smoother than the surface of segmentation results. Due to the intensity value range of reference between 0 and 1, the intensity value of voxels at the surface can vary between 0 and 1 which makes the smooth surface. While the intensity value of segmentation results is binary, 0 and 1 make the rough surface.

CHAPTER 6 CONCLUSION

In order to segment the carotid artery from the CTA medical images and visualize the segmentation results in 3-Dimension model images, the Level Set method (LSM), vesselness filter and Ray Casting algorithm were adapted. The Level Set method was used to identify the boundary of the objects. We adapted this technique to specify the boundary of the bones and reconstruct the vessels structure. The vesselness filter was used to evaluate the likeliness of the voxels belonging to the vessels structure. Lastly, for the visualization purpose, we integrated the segmentation result with the Ray Casting algorithm.

The Level Set Method was used to identify the boundary of the objects in the image and distinguish the foreground and the background. This method can be categorized into two main types: edge based and region based active contour models. The edge based active contour model applied the edge detector function, which depended on the gradient of image as the stopping function. The edge of the object was used to prevent the contour to leak out of the desired boundary by terminating the evolution of level set function when the contour at zero level reached the boundary of the desired object. This method is appropriate for objects which have clear boundaries. In addition, it can segment only desired objects by setting the appropriate initial level set function even though there are many objects on the foreground image. The region based active contour model is based on the region properties of the objects instead of the boundary of the object. This model can find the suitable boundary of the object by minimizing the deviation of intensity value inside and outside the contour. Thus, this model still works well even though the edge of the segmented object is unclear and it also resists noisy images. Moreover, we can set the initial contour at zero level everywhere on the image.

Due to the characteristics of medical images, the organs usually have unclear boundary. So, applying the edge based active contour model directly does not work well with medical images. Moreover, the range of intensity value of the desired organs may overlap with the range of intensity value of the other organs. So, applying the region based contour model directly cannot distinguish the boundary of the bones and the boundary of the vessels. We therefore proposed the Hybrid LSM which is the combination between edge based and region based active contour models. This modified version preserves the advantages of both original models while the performance of Level Set Method is improved. This model enables the Level Set Method to segment the boundary of the object even though the edge of the segmented object is unclear and the model can also resist noisy images. In addition, this modified version can segment only the boundary of the desired object by setting the appropriate initial level set function. Moreover, the Hybrid LSM can reach the convergence point faster than both original models with more accuracy.

In order to evaluate the performance of our proposed method, we performed the experiments by comparing our modified version Hybrid LSM with the edge based and region based active contour models. We used the Distance Regularized Level Set Evolution model introduced by Li et al. [19] to represent the edge based active contour model and Active Contours without Edges model introduced by Chan-Vese [18] to represent the region based active contour model. Ten experiments were performed by using various types of input images. For example, noisy images, clear and unclear boundary objects images, multi gray-scale images and medical images. The accuracy of

each model was evaluated by Dice similarity, Sensitivity and Specification. In addition, the number of iterations which the models used to reach the convergence point was used to measure the speed of the method.

Our proposed method, Hybrid LSM can segment the boundary of the desired object in high accuracy and resist noisy images. This method still works well even though the objects have unclear boundary and can segment only desired objects from images which have multiple objects in the foreground. Moreover, this method can reach the convergence point faster than the conventional methods. However, there are some errors when our proposed method is applied on objects with acute angle corners. While the edge based active contour model works well with objects having clear boundary and can be used to segment only desired objects from images containing multiple objects in the foreground, this method cannot handle noisy images and also has some errors at acute angle corners. As for the region based active contour, it is good for the segmentation of objects even those with unclear boundaries. Although this method can work well with noisy images and acute angle corners, it usually generates several false positive when it is applied on images which have multiple objects in the foreground.

As the ultimate purpose of this research which aimed to segment the carotid artery from the CTA medical images, we proposed the segmentation framework in which our modified version, Hybrid LSM was used to specify the boundary of the bones and reconstruct the vessels structure. This framework started from eliminating the bone structure from medical images because the range of intensity value of bones overlapped with the range of intensity value of vessels. First, we created the initial level set function by specifying seed points which belong to the bone structure. Next, we made those seed points evolve to reach the boundary of the bones. After the location of bones was located, we eliminated them from the original data. Next, we adopted the histogram analysis to specify rough vessel masks. Next, we employed the vesselness filter to measure the vesselness of each voxel in datasets. We extracted the center line of vessels from the set of voxel which had high vesselness value. Finally, we reconstructed the vessel structure by using the center line as an initial level set function and evolve it by using the Level Set method. In addition, for the visualization purpose, the transfer function of Ray Casting algorithm was adapted to support our segmentation results.

To evaluate our achievement, we followed the evaluation framework provided by Carotid Lumen Segmentation and Stenosis Grading Challenge (CLS). There are three measurements: dice similarity, sensitivity and specification. We performed the experiments by using fifteen CTA datasets. All the datasets were carotid artery images. We obtained the average dice similarity at 87.09%, average sensitivity at 88.13% and average specification at 86.63% which shows that our method was efficient for vessel segmentation. However, some false positive occurred in the vessel reconstruction process. The error usually occurred around the bifurcation of artery. The voxels with high intensity around the bifurcation did exist but specialists did not regard them as vessels. They were close to the vessels, so the model was confused. Another cause of error is that the segmentation results do not completely overlap with the reference even though the visualization of segmentation results and reference are similar.

REFERENCES

- [1] ปาณิธร การปลื้มจิตต์, โรคหลอดเลือดแดงส่วนปลายอุดตัน [Online], Available: http://www.phyathai.com/medicalcenterdetail_article/1/44/PYT2/th [2013, April 19].
- [2] ปิยะนุช รักพาณิชย์, มาออกกำลังกายเพื่อป้องกันโรคหลอดเลือดหัวใจอุดตันกันเถอะ [Online], Available: http://www.thairunning.com/running_for_heart.htm [2013, April 19].
- [3] Manbachi, A., Hoi, Y., Wasserman, B. A. Lakatta, E.G. and Steinman, D.A. “On the Shape of the Common Carotid Artery with Implications for Blood Velocity profiles.”, **Physiol. Meas.** **32**, Vol. 12, pp. 1885–1897.
- [4] Sobieszczyk, P. and Beckman, J. **Carotid Artery Disease** [Online], Available: <http://circ.ahajournals.org/content/114/7/e244.full> [2014, June 23].
- [5] Society for vascular surgery. **Carotid Artery Disease, Stroke, Transient Ischemic Attacks (TIAs)** [Online], Available: [http://www.vascularweb.org/vascularhealth/Pages/carotid-artery-disease-,stroke-,transient-ischemic-attacks-\(tias\)-.aspx](http://www.vascularweb.org/vascularhealth/Pages/carotid-artery-disease-,stroke-,transient-ischemic-attacks-(tias)-.aspx) [2014, June 23]
- [6] Smelyanskiy, M., Holmes, D., Chhugani, J., Larson, A., Carmean, D.M., Hanson, D., Dubey, P., Augustine, K., Kim, D., Kyker, A., Lee, V.W., Nguyen, A.D., Seiler, L. and Robb, R., 2009, “Mapping High-Fidelity Volume Rendering for Medical Imaging to CPU GPU and Many-Core Architectures”, **IEEE Transactions on Visualization and Computer Graphics**, Vol. 15, No. 6, pp. 1563-1570.
- [7] Lee, B., Yun, J., Seo, J., Shim, B., Shin, Y.G. and Kim, B., 2010, “Fast High-Quality Volume Ray-Casting with Virtual Samplings”, **IEEE Transactions on Visualization and Computer Graphics**, Vol. 16, No. 6, pp. 1525-1532.
- [8] Knoll, A., Hijazi, Y., Westerteiger, R., Schott, M., Hansen, C. and Hagen, H., 2009, “Volume Ray Casting with Peak Finding and Differential Sampling”, **IEEE Transactions on Visualization and Computer Graphics**, Vol. 15, No. 6, pp. 1571-1577.
- [9] Wang, Q. and JaJa, J., 2008, “Interactive High-Resolution Isosurface Ray Casting on Multicore Processors”, **IEEE Transactions on Visualization and Computer Graphics**, Vol. 14, No. 3, pp. 603-614.
- [10] Marchesin, S., Dischler, J.M. and Mongenet, C., 2010, “Per-Pixel Opacity Modulation for Feature Enhancement in Volume Rendering”, **IEEE Transactions on Visualization and Computer Graphics**, Vol. 16, No. 4, pp. 560-570.
- [11] Marin, D., Aquino, A., Gegundez-Arias, M.E. and Bravo, J.M., 2011, “A New Supervised Method for Blood Vessel Segmentation in Retinal Images by Using Gray-Level and Moment Invariants-Based Features”, **IEEE Transactions on Medical Imaging**, Vol. 30, NO. 1, pp. 146-158.

- [12] Shang, Y., Deklerck, R., Nyssen, E., Markova, A., de Mey, J., Yang, X., and Sun, K., 2011, "Vascular Active Contour for Vessel Tree Segmentation", **IEEE Transactions on Biomedical Engineering**, Vol. 58, No. 4, pp. 1023-1032.
- [13] Zou, P., Chan, P., and Rockett, P., 2009, "A Model-Based Consecutive Scanline Tracking Method for Extracting Vascular Networks From 2-D Digital Subtraction Angiograms", **IEEE Transactions on Medical Imaging**, Vol. 28, No. 2, pp. 241-249.
- [14] Osher, S., and Sethian, J.A., 1998, "Fronts Propagating with Curvature-dependent Speed: Algorithm Based on Hamilton-Jacobi Formulations", **Journal of Computational Physics**, Vol 79, pp. 12-49.
- [15] Calder, J., Tahmasebi, A.M. and Mansouri, A., 2011, "A variational Approach to Bone Segmentation in CT images", **Medical Imaging 2011: Image Processing**, Vol 7962, pp. 15.
- [16] G.W. Jiji., 2013, "Segmentation of Blood Vessels and 3D Representation of CMR Image", **The Institution of Engineers (India)**, Vol 94(2), pp. 115-121
- [17] Oliveira, D.A.B., Feitosa, R.Q. and Correia, M.M., 2010, "Genetic Adaptation of Level Sets Parameters for Medical Imaging Segmentation", **Medical Information Science Reference**, Vol. 1, pp. 150-165.
- [18] Chan, T.F. and Vese, L.A., 2001, "Active Contours Without Edges", **IEEE Transactions on Image Processing**, Vol. 10, pp. 266-277.
- [19] Li, C., Xu, C., Gui, C. and Fox, M.D., 2010, "Distance Regularized Level Set Evolution and Its Application to Image Segmentation", **IEEE Transactions on Image Processing**, Vol. 19, pp. 3243-3254.
- [20] Frangi, A.F., Niessen W.J., Vincken, K.L. and Viergever, M.A., 1998, "Multiscale Vessel Enhancement Filtering", **Medical Image Computing and Computer-Assisted Intervention**, Vol. 1496, pp. 130-137.
- [21] Zhong. Y., 2014, "Extracting Vessel Structure From 3D Image Data", **University of Western Ontario - Electronic Thesis and Dissertation Repositor**.
- [22] Levoy, M., 1988, "Volume Rendering", **IEEE Computer Graphics & Applications**, pp. 29-37.
- [23] **Digital Imaging and Communications in Medicine (DICOM) Part 1**, National Electrical Manufacturers Association, USA.
- [24] Preim, B. and Bartz, D., 2007, **Visualization in Medicine**, Morgan Kaufmann Publishers, USA.
- [25] Foley & Van, D., **Illumination Models and Shading** [Online], Available: http://www.cs.brandeis.edu/~cs155/Lecture_16.pdf [2013, April 19].
- [26] Hameeteman, K., Zuluaga, M.A., Freiman, M., Joskowicz, L., Cuisenaire, O., Valencia, L.F., Gülsün, M.A., Krissian, K., Mille, J., Wong, W.C., Orkisz, M.,

Tek, H., Hoyos, M.H., Benmansour, F., Chung, A.C., Rozie, S., van Gils, M.J., van den Borne, L., Sosna, J., Berman, P., Cohen, N., Douek, P.C., Sánchez, I., Aissat, M., Schaap, M., Metz, C.T., Krestin, G.P., van der Lugt, A., Niessen, W.J. and van Walsum, T., 2011, "Evaluation Framework for Carotid Bifurcation Lumen Segmentation and Stenosis Grading", **Medical Image Analysis**, Vol. 15, pp. 477-488.

CURRICULUM VITAE

NAME	Mr. Tananan Pattanangkur
DATE OF BIRTH	6 July 1990
EDUCATIONAL RECORD	
HIGH SCHOOL	High School Graduation (Science-Mathematics) Taweethapisek School, 2007
BACHELOR'S DEGREE	Bachelor of Engineering (Computer Engineering) King Mongkut's University of Technology Thonburi, 2011
MASTER'S DEGREE	Master of Engineering (Computer Engineering) King Mongkut's University of Technology Thonburi, 2014
SCHOLARSHIP/ RESEARCH GRANT	King Mongkut's Diamond Scholarships for Excellent Academic Performance, 2012-2013 National Research University (NRU), 2012-2013
EMPLOYMENT RECORD	Developer Novitat Co. Ltd., 2012-2014
PUBLICATION	Pattanangkur, T., Tanupabrunson, S., Areekijseere, K. and Pumma, S., 2013, "The Design of SkyPACS: a High Performance Mobile Medical Imaging Solution", The Symposium on GPU Computing and Applications 2013 , Singapore.