3-23-2012

A Novel 3-D Segmentation Algorithm for Anatomic Liver and Tumor Volume Calculations for Liver Cancer Treatment Planning

Mohammed Goryawala
Florida International University, mgory001@fiu.edu

DOI: 10.25148/etd.FI12050223

FLORIDA INTERNATIONAL UNIVERSITY

Miami, Florida

A NOVEL 3-D SEGMENTATION ALGORITHM FOR ANATOMIC LIVER AND TUMOR VOLUME CALCULATIONS FOR LIVER CANCER TREATMENT PLANNING

A dissertation submitted in partial fulfillment of the

requirements for the degree of

DOCTOR OF PHILOSOPHY

in

BIOMEDICAL ENGINEERING

by

Mohammed Z. Goryawala

2012

To: Dean Amir Mirmiran
College of Engineering and Computing

This dissertation, written by Mohammed Z. Goryawala, and entitled A Novel 3-D Segmentation Algorithm for Anatomic Liver and Tumor Volume Calculations for Liver Cancer Treatment Planning, having been approved in respect to style and intellectual content, is referred to you for judgment.

We have read this dissertation and recommend that it be approved.

Armando Barreto

Anthony McGoron

Wei-Chiang Lin

Malek Adjouadi, Major Professor

Date of Defense: March 23, 2012

The dissertation of Mohammed Z. Goryawala is approved.

Dean Amir Mirmiran
College of Engineering and Computing

Dean Lakshmi N. Reddi
University Graduate School

Florida International University, 2012

ACKNOWLEDGMENTS

First and foremost I would like to express my deepest gratitude to my advisor, Dr. Malek Adjouadi, for his infinite patience, hard work, advice, and support. I thank him for giving me the opportunity to work with him, which I have cherished more than anything during my graduate studies. He has not only supported and helped me in my course work but also in every facet of my life for becoming a better individual.

I extend my appreciation to my committee members, Dr. Anthony McGoron, Dr. Armando Barreto, Dr. Wei-Chiang Lin and Dr. Seza Gulec. I thank you for all the help and support that you have provided throughout my career at Florida International University.

I would like to thank my colleagues Dr. Mercedes Cabrerizo, Ana Guzman, Ruchir Bhatt, the team at JNMC, and all my friends for their help and support throughout the project.

I would also like to thank the National Science Fondation under grants CNS-1042341, CNC-0959985, and HRD-0833093. My thanks also go the University Graduate School at Florida International University for providing me with the Dissertation Year Fellowship.

Finally and most dearly I would like to thank my entire family for their patience and undying support through the period of my Ph.D.

ABSTRACT OF THE DISSERTATION

A NOVEL ALGORITHM FOR ANATOMIC LIVER AND TUMOR VOLUME DETERMINATION FOR LIVER CANCER SIRT PLANNING

by

Mohammed Z. Goryawala

Florida International University, 2012

Miami, Florida

Professor Malek Adjouadi, Major Professor

Three-Dimensional (3-D) imaging is vital in computer-assisted surgical planning including minimal invasive surgery, targeted drug delivery, and tumor resection. Selective Internal Radiation Therapy (SIRT) is a liver directed radiation therapy for the treatment of liver cancer. Accurate calculation of anatomical liver and tumor volumes are essential for the determination of the tumor to normal liver ratio and for the calculation of the dose of Y-90 microspheres that will result in high concentration of the radiation in the tumor region as compared to nearby healthy tissue. Present manual techniques for segmentation of the liver from Computed Tomography (CT) tend to be tedious and greatly dependent on the skill of the technician/doctor performing the task.

This dissertation presents the development and implementation of a fully integrated algorithm for 3-D liver and tumor segmentation from tri-phase CT that yield highly accurate estimations of the respective volumes of the liver and tumor(s). The algorithm as designed requires minimal human intervention without compromising the accuracy of the segmentation results. Embedded within this algorithm is an effective method for extracting blood vessels that feed the tumor(s) in order to plan effectively the appropriate treatment.

Segmentation of the liver led to an accuracy in excess of 95% in estimating liver volumes in 20 datasets in comparison to the manual gold standard volumes. In a similar comparison, tumor segmentation exhibited an accuracy of 86% in estimating tumor(s) volume(s). Qualitative results of the blood vessel segmentation algorithm demonstrated the effectiveness of the algorithm in extracting and rendering the vasculature structure of the liver. Results of the parallel computing process, using a single workstation, showed a 78% gain. Also, statistical analysis carried out to determine if the manual initialization has any impact on the accuracy showed user initialization independence in the results.

The dissertation thus provides a complete 3-D solution towards liver cancer treatment planning with the opportunity to extract, visualize and quantify the needed statistics for liver cancer treatment. Since SIRT requires highly accurate calculation of the liver and tumor volumes, this new method provides an effective and computationally efficient process required of such challenging clinical requirements.

TABLE OF CONTENTS

CHAPTER PAGE

I. Introduction 1

1.1. Treatment Options for CRLC 1

1.1.1. Surgical Strategies 1

1.1.2. Radiation Therapy 2

1.1.2.1. Intensity Modulated Radiation Therapy (IMRT) 3

1.1.2.2. Three-dimensional (3-D) conformal radiation therapy (3DCRT) 3

1.1.2.3. External Beam Radiation Therapy (EBRT) 3

1.1.2.4. TomoTherapy® Highly Integrated Adaptive Radiotherapy....................4

1.1.3. Chemotherapy 4

1.1.4. Immunotherapy 6

1.1.5. Brachytherapy 6

1.1.6. Cryotherapy and Radiofrequency Ablation....................7

1.2. Selective Internal Radiation Treatment (SIRT) 7

1.2.1. Selectiveness of SIRT 7

1.2.2. Development of SIRT 8

1.2.3. Dose calculations for SIRT 9

1.3. Objectives of the study....................9

1.4. Outline of the dissertation 10

II. Imaging in CRLM 12

2.1. Imaging Modalities 12

2.1.1. Contrast-enhanced ultrasound (CEUS) 12

2.1.2. Computed Tomography (CT) 13

2.1.3. Magnetic Resonance Imaging (MRI) 14

2.1.4. Positron emission tomography/computed tomography (PET/CT)..........15

2.2. Instrumentation and Image Acquisition....................16

2.3. Dataset Details 17

III. Liver Segmentation 21

3.1. Review of Liver Segmentation Techniques....................23

3.2. Liver segmentation based on multiple equidistant slice initialization process (Ver. 1.0) 26

3.2.1. Modified K-means algorithm and its suitability to liver segmentation...28

3.2.2. Initialization of the Localized Contouring Method *Ver. 1.0* 33

3.2.3. Localized-region-based active contouring algorithm....................35

3.3. Algorithm (Ver. 2.0) with information-based slice selection for contour initialization 41

3.3.1. Changes to the modified K-means algorithm....................41

3.3.2. Initialization of the Contouring-Based Segmentation 42

3.4. Liver segmentation algorithm using single slice contour initialization (Ver. 3.0) 47

3.4.1. Contour Initialization 48
3.4.1.1 Anisotropic Diffusion Filter 48
3.4.1.2. Intensity based Region Growing 49
3.4.1.3. Volume of Interest Based Correction or Three View Correction.........51
3.4.1.4. Single Slice Selection 51
3.5. Three-Dimensional Image Rendering...54

IV. Tumor and Blood Vessel Segmentation 55
4.1. Review of automatic/semi-automatic tumor segmentation techniques 56
4.2. Tumor Segmentation Method 58
4.2.1. Liver Extraction...59
4.2.2. Image De-noising 60
4.2.3. Intensity-based Region-Growing 61
4.2.4. Localized Contouring Algorithm 62
4.2.5. Rendering and Volume Calculations..62
4.3. Review of hepatic automatic/semi-automatic blood vessel segmentation techniques 63
4.4. Hepatic Blood Vessel Segmentation Method 66
4.4.1. Liver Extraction for blood vessel segmentation.....................................66
4.4.2. Vascular Edge Enhancement...67
4.2.3. Intensity-based Region-Growing 68

V. Parallel Processing for 3D Segmentation and Volume Rendering 70
5.1. Review of Parallel Processing Techniques...71
5.2. Aims for development of the parallel computing infrastructure........................74
5.3. Setup of the parallel computing platform 74
5.4. Parallel Computing Results 80
5.4.1. Benchmarking with FFT based process 80
5.4.2. Platform Comparison 89
5.4.3 Specific Application: Liver Segmentation 97

VI. Results...107
6.1. Accuracy Analysis of the liver segmentation approach..................................107
6.1.1. Results for Liver Segmentation algorithm *Ver. 1.0*..............................107
6.1.2 Results for Liver Segmentation algorithm *Ver. 2.0*...............................110
6.1.3. Results for Liver Segmentation algorithm *Ver. 3.0*..............................111
6.1.4. 3-D Rendering of the Segmented Livers...113
6.2. Comparative and Statistical Analysis 115
6.2.1. Comparative Analysis 115
6.2.2. Statistical Analysis 118
6.3. Tumor and blood vessel segmentation...121
6.3.1. Tumor segmentation...121
6.3.2. Blood vessel segmentation 126
6.4. Extending the algorithm's extent towards other applications.........................131

VII. Discussion 133
7.1. Liver Segmentation Algorithm 133
7.2. Parallel Computing 137
7.3. Tumor and Blood Vessel Segmentation 142
7.4. Contributions of the study ………………………..………………...………. 145

VIII. Conclusions and Future Work..147

References ...………………………………………………………………………...150

VITA ...………………………..…………………………………………….........................162

LIST OF FIGURES

FIGURE PAGE

Figure 2.1: Box-plot showing variation in number of slices for the 20 datasets 20

Figure 2.2: Box-plot showing variation in slice thickness for the 20 datasets 20

Figure 3.1: Overview of the three versions of the segmentation algorithm......................22

Figure 3.2: General algorithm for 3D liver segmentation and rendering - *Ver. 1.0* 27

Figure 3.3: The 5 reference regions, M1 through M5, needed to apply the K-means based segmentation method on a liver CT dataset. 29

Figure 3.4: Selection of seed-points for K-means based image segmentation step...........31

Figure 3.5: Comparison of the Modified K-means method and traditional K-means method. 32

Figure 3.6: Contour Based Liver Segmentation Initialization Process.............................34

Figure 3.7: Contour Based Segmentation (in progress)...36

Figure 3.8: Liver Segmentation example (a) User initialization shown in red (b) Final output of the algorithm after localized contouring step..37

Figure 3.9: Extracted Liver 38

Figure 3.10: Extracted Liver with Contour Displayed...38

Figure 3.11: Original Image...39

Figure 3.12: Result of the Algorithm 39

Figure 3.13: Extracted Liver with Contour Displayed...40

Figure 3.14: Extracted Liver 40

Figure 3.15: General algorithm for 3D liver segmentation and rendering for *Ver 2.0* algorithm..42

Figure 3.16: Division of liver CT image in blocks of sizes (a) 64x64 (b)32x32 (c) 16x16 (d) 8x8 43

Figure 3.17: Generation of dataset profile curve depicting the slices corresponding to the marked points on the dataset profile curve. 45

Figure 3.18: Selection of block size based on Ψ performance 46

Figure 3.19: Structure of the algorithm *Ver. 3.0* for 3D liver segmentation and rendering48

Figure 3.20: Different views showing the seed-line in a given dataset 50

Figure 3.21: Volume-of-Interest based correction results (a) Original slice (b) Coronal Mask (c) Sagittal Mask (d) Transverse Mask (e) Corrected Mask............................52

Figure 3.22: Automatic Large Slice Selection algorithm. A typical curve is shown with corresponding slices. The red vertical line shows the location of the largest slice which is then is used by the algorithm. 53

Figure 4.1: Working of the median filter showing the value 150 (marked in red) being replaced by 124 which is the median of the 8 neighborhood values..........................61

Figure 4.2: Outputs after every step of the tumor segmentation algorithm. The red overlay shows the tumor masks generated after the intensity based region growing and the localized contouring 63

Figure 4.3: Outputs after every step of the blood vessel segmentation algorithm. The red overlay shows the masks generated after the intensity based region growing algorithm..69

Figure 5.1: Parallel processing approach for the *Ver. 3.0* algorithm. The serial approach is outlined in the yellow box. 78

Figure 5.2: Details of the parallel processing approach for the *Ver 3.0* algorithm. The serial approach is outlined in the yellow box. 79

Figure 5.3: Computation Time vs. Number of *Workers* for processing 1000 images on System 1 81

Figure 5.4: Speed up vs. Number of *Workers* for System 1 82

Figure 5.5: Computation Time vs. Number of *Workers* for processing 1000 images on System 2. 82

Figure 5.6: Speed up vs. Number of *Workers* for System 2 83

Figure 5.7: Computational Time vs. Number of Images for system 1 84

Figure 5.8: Speed up vs. Number of Images for system 1 84

Figure 5.9: Computational Time vs. Number of Images for system 2 85

Figure 5.10: Speed up vs. Number of Images for system 2 85

Figure 5.11: Computational Time vs. Size of Image for system 1 87

Figure 5.12: Speed up vs. Size of Image for system 1 87

Figure 5.13: Computational Time vs. Size of Image for system 2 88

Figure 5.14: Computational Time vs. Size of Image for system 2 88

Figure 5.15: Computational Time vs. Number of *Workers*: A Comparison of Windows and Linux platforms conducted on System 1 89

Figure 5.16: Speed up vs. Number of *Workers*: A Comparison of Windows and Linux platforms conducted on System 1 90

Figure 5.17: Computational Time vs. Number of Images: A Comparison of Windows and Linux platforms conducted on System 1 91

Figure 5.18: Speed up vs. Number of Images: A Comparison of Windows and Linux platforms conducted on System 1 92

Figure 5.19: Speed up vs. Size of Images: A Comparison of Windows and Linux platforms conducted on System 1 92

Figure 5.20: Computational Time vs. Size of Images: A Comparison of Windows and Linux platforms conducted on System 1. 93

Figure 5.21: Computational Time vs. Number of *Workers*: A Comparison of Windows and Mac OS-X platforms conducted on System 2 94

Figure 5.22: Speed up vs. Number of *Workers*: A Comparison of Windows and Mac OS-X platforms conducted on System 2 94

Figure 5.23: Computational Time vs. Number of Images: A Comparison of Windows and Mac OS-X platforms conducted on System 2 95

Figure 5.24: Speed up vs. Number of Images: A Comparison of Windows and Mac OS-X platforms conducted on System 2 96

Figure 5.25: Computational Time vs. Size of Images: A Comparison of Windows and Mac OS-X platforms conducted on System 2 96

Figure 5.26: Speed up vs. Size of Images: A Comparison of Windows and Mac OS-X platforms conducted on System 2..97

Figure 5.27: Average computational time for segmentation of a single slice of the datasets as a function of the Number of *workers* employed for *Ver 1.0* algorithm. The Error bars show the standard deviation across the 20 datasets. 98

Figure 5.28: Average speed up for segmentation of the datasets as a function of the Number of *workers* employed for *Ver. 1.0* algorithm. The Error bars show the standard deviation across the 20 datasets 100

Figure 5.29: Average computational time for segmentation of a single slice of the datasets as a function of the Number of *workers* employed for *Ver 2.0* algorithm. The Error bars show the standard deviation across the 20 datasets 102

Figure 5.30: Average speed up for segmentation of the datasets as a function of the Number of *workers* employed for *Ver 2.0* algorithm. The Error bars show the standard deviation across the 20 datasets. 102

Figure 5.31: Average computational time for segmentation of a single slice of the datasets as a function of the Number of *workers* employed for *Ver 3.0* algorithm. The Error bars show the standard deviation across the 20 datasets. 103

Figure 5.32: Average speed up for segmentation of the datasets as a function of the Number of *workers* employed for *Ver. 3.0* algorithm. The Error bars show the standard deviation across the 20 datasets. 105

Figure 5.33: The time required for segmenting the entire dataset as a function of the number of slices in the dataset for both the serial and parallel approaches using *Ver 3.0* algorithm...106

Figure 6.1: Illustrative examples of segmentation results for 4 datasets. (a) Subject 19 (b) Subject 2 (c) Subject 15 (d) Subject 20 using the Ver. 1.0 liver s segmentation algorithm...109

Figure 6.2: Illustrative examples of segmentation results for 4 datasets. (a) Subject 1 (b) Subject 19 (c) Subject 7 (d) Subject 16 using the Ver. 2.0 liver segmentation algorithm...111

Figure 6.3: Illustrative examples of segmentation results for 4 datasets. (a) Subject 19 (b) Subject 8 (c) Subject 12 (d) Subject 18 using the Ver. 3.0 liver segmentation algorithm...113

Figure 6.4: Illustrative examples of 3D renderings for 4 datasets (a) Subject 19 (b) Subject 8 (c) Subject 12 (d) Subject 18. The 3-D rendering shown are obtained using algorithm Ver 3.0. Similar 3-D rendering can be generated for Ver 1.0 and Ver 2.0 algorithms also. 114

Figure 6.5: Illustrative examples of tumor segmentation results for 4 datasets. (a) Subject 5 (b) Subject 6 (c) Subject 14 (d) Subject 17 123

Figure 6.6: Illustrative examples of tumor segmentation results for 4 datasets. (a) Subject 5 (b) Subject 8 (c) Subject 14 (d) Subject 17 125

Figure 6.7: Illustrative examples of blood vessels segmentation results for Subject 10 .126

Figure 6.8: Illustrative examples of blood vessel segmentation results for 3 datasets. (a) Subject 10 (b) Subject 4 (c) Subject 17 127

Figure 6.9: Generating the 3-D fused solution containing the liver, tumor(s) and blood vessels for enhanced visualization.......129

Figure 6.10: Examples of 3-D fused solution. (a) Subject 4 (b) Subject 17 130

LIST OF TABLES

TABLE PAGE

Table 2.1: Clinical Characteristics of the 20 datasets 17

Table 2.2: Voxel Dimensions for the 20 liver CT datasets 19

Table 5.1: Computational Time for *Ver. 1.0* Algorithm 99

Table 5.2: Computational Time for *Ver. 2.0* Algorithm 101

Table 5.3: Computational Time for *Ver. 3.0* Algorithm 104

Table 6.1: Comparative results between the proposed algorithm and manual gold standard for *Ver. 1.0* algorithm 108

Table 6.2: Comparative results: proposed algorithm *Ver. 2.0* vs. manual gold standard 110

Table 6.3: Comparative results: proposed algorithm *Ver. 3.0* vs. manual gold standard 112

Table 6.4: Comparison of proposed algorithm with other algorithms found in the literature 115

Table 6.5: Absolute volume errors for the 20 datasets for each version of the algorithm 117

Table 6.6: ANOVA Results for various Users 117

Table 6.7: Absolute error for various user initializations for *Ver. 1.0* algorithm 118

Table 6.8: ANOVA Results for various Users for *Ver. 1.0* algorithm 118

Table 6.9: Absolute error for various user initializations for *Ver. 2.0* algorithm 119

Table 6.10: ANOVA Results for various Users for *Ver. 2.0* algorithm 120

Table 6.11: Absolute error for various user initializations for *Ver. 3.0* algorithm 120

Table 6.12: ANOVA Results for various Users for *Ver. 3.0* algorithm 121

Table 6.13: Comparative results between the proposed algorithm and manual gold standard for Ver. 1.0 algorithm 122

Table 7.1: ANOVA Test results for experiments on System 1....................................... 138

Table 7.2: ANOVA Test result for experiments in system 2.. 141

CHAPTER I

INTRODUCTION

The American Cancer Society estimates that nearly 18,000 people are diagnosed with liver cancer annually. This unfortunately makes 1 out of 183 individuals born today susceptible to liver cancer at some point during their lifespan. Liver cancers and colorectal cancers amount up to 140,000 of the estimated new cancer cases identified in the Unites States only (Siegel et al., 2011). The number of new cases of liver and colorectal liver metastases (CRLM) amount to more than a million cases throughout the world. Also, liver cancer and CRLM are responsible for about 79000 deaths in males and 60000 deaths in females.

1.1. Treatment Options for CRLC

The most common conventional treatment options for treating liver cancers and CRLM include surgery, radiation therapy and chemotherapy (Scheele et al., 1997). The choice of the treatment is determined mainly by the extent of the disease and the health of the individual. Below we describe some of the common conventional treatment options along with some new and more recent development in the field of treatment of liver cancer and CRLM.

1.1.1. Surgical Strategies

Surgery is the one of the oldest method for treating liver cancers. Although surgery has been around for a long time, advances in the past three decades have resulted in increasing significantly the chances for patient survival (Hughes et al., 1986). Resection of the tumor is done by the removal of the affected region or the organ. A partial hepatectomy is the result of the removal of a part of the liver which is affected whereas a total hepatectomy is where the entire liver is removed to be substituted by a donor liver. With the advent of segment-based resection, the resection of multiple and bilateral liver tumors has now become possible (Ballantyne and Quin, 1993). Techniques like vascular occlusion, total vascular exclusion and caval clamping have been developed in modern era to reduce the blood loss during surgery (Lau et al., 2010; Wang et al., 2011). Also new research in prosthetic grafts and perfusion techniques has provided surgeons with added potential for surgical success (Hardwigsen et al., 2001; Lau et al., 2010).

1.1.2. Radiation Therapy

Radiation therapy is a one of most widely used form of treatment for liver cancer (Geoghegan and Scheele, 1999). Radiation therapy utilizes high energy rays such as x-rays or gamma rays for the destruction of the unhealthy tumor cells. The radiation may completely destroy the tumor or lessen the size of the tumor in order to reduce pain or make the patient eligible for surgery.

Radiation therapy has been adopted as a standalone liver cancer treatment or in conjunction with chemotherapy. Various new techniques have been developed for radiation therapy, all with the main goal of delivering higher radiation doses to the tumor while minimizing the damage to healthy liver tissue.

The radiation techniques commonly used in the clinical environment include Intensity Modulated Radiation Therapy (IMRT), 3-D Conformal Radiation, and External Beam Radiation Therapy (EBRT).

1.1.2.1. Intensity Modulated Radiation Therapy (IMRT)

Intensity Modulated Radiation Therapy (IMRT) is one of the widely used technique in radiation therapy treatment (Swanson et al., 2012). IMRT uses Computed Tomography (CT) and based on the patient's tumor size, shape and location a plan is set for the delivery of the radiation dose in three dimensions. IMRT utilizes a modulated beam of radiation to deliver radiation at a pin-point location with extremely high accuracy. IMRT has the advantage over conventional radiation therapy to deliver the dose more accurately and enable higher doses to be delivered to the tumor region albeit at a higher cost.

1.1.2.2. Three-dimensional (3-D) conformal radiation therapy (3DCRT)

Three-dimensional (3-D) conformal radiation therapy (3DCRT) is an extension of the IMRT technique which uses multiple radiation beams to deliver higher doses of radiation to the tumor region. 3DCRT offers a higher does more accurately by focusing the beams of radiation to conform to the contour of the tumor. It also prevents excessive damage to surrounding tissue by focusing the beams to the tumor region only and deliver higher doses per tissue weight to the tumor (Swanson et al., 2012).

1.1.2.3. External Beam Radiation Therapy (EBRT)

External Beam Radiation Therapy (EBRT) is the most commonly used radiation treatment procedures adopted in the U.S.A. since the treatment is mainly an outpatient procedure with reduced risks than surgery. EBRT reduces surgical risks such as bleeding, post-operative pain and risk of heart attacks and stroke. EBRT uses a high energy external radiation beam to deliver radiation to the tumors. The energy of the radiation beam is mainly governed by the depth and position of the tumor. Liver cancer treatment uses Megavoltage X-rays to deliver radiation to the diseased regions of the liver.

1.1.2.4. TomoTherapy® Highly Integrated Adaptive Radiotherapy

TomoTherapy® Highly Integrated Adaptive Radiotherapy (HI-ART) is an advanced radiation therapy treatment option usually used in situations where an individual has reached their maximum tolerated radiation dose or when the tumor is located in a very hard-to-reach place. HI-ART uses enhanced Computed Tomography (CT) to track and locate the tumor's exact shape, size and position and deliver the radiation from 360° position all around the patient as contrasted to IMRT and EBRT. TomoTherapy allows the physician to deliver maximum radiation dose to the liver while reducing the amount of radiation delivered to the surrounding tissue.

1.1.3. Chemotherapy

Only 20-25 percent of the patients affected by liver cancer are estimated to have resectable disease. Generally, as a rule of thumb the maximum resectable size of the tumor is considered to be 5 cm in diameter (Cai and Cai, 2012). For the non-resectable cases neoadjuvant treatments are believed to be providing the means necessary to reduce the size of tumor to make the tumors resectable. Various studies have reported the use of chemotherapy drugs such as 5-fluorouracil (5-FU), 5-FU-α-interferon, 5-FU-cisplatin-doxorubicin, 5-FU-pirubicin, 5-FU-leucovorin-oxaliplatin and others, all for down-staging the resectability for patients with colorectal liver metastases (Scheele et al., 1997). Studies carried out by Bismuth et al. have demonstrated a 5 year survival rate of around 40%(Bismuth et al., 1996).

Chemotherapy drugs are usually administered intravenously using injection or orally. In advanced cases chemotherapy drugs are sometimes also administered directly in to the artery that forms the primary blood supply to the tumor. Such a technique called intra-arterial chemotherapy provides higher doses of the drug to be delivered to the tumor as well as reducing the effect of the drugs on healthy tissue (Seront and Van den Eynde, 2012).

Chemoembolization is a combination of chemotherapy and embolization techniques. For chemoembolization a radiologist directs microspheres combined with chemotherapy drugs through a catheter into the artery feeding the tumor to block the supply of blood to the diseased tissue. Chemoembolization allows high doses of chemotherapy drugs to be delivered to the tumor as well as blocking the flow of blood to the tumor region. The reduction of blood flow to the tumor restricts the tumor from obtaining the necessary nutrients resulting in the reduction of the size and extent of tumors (Ho et al., 2012).

Chemotherapy however results in various side effects such as neurotoxicity, nausea, vomiting, hair loss and mouth sores among others. In order to reduce these side effects of chemotherapy, Fractionated Dose Chemotherapy has been utilized. Fractionated Dose Chemotherapy delivers the total dose of the anticancer drugs in smaller amounts over longer period of time. Fractionated Dose Chemotherapy not only offers the advantage of reducing the side effects but also increases the effectiveness of the therapy by working for a longer period of time (Frei and Canellos, 1980).

1.1.4. Immunotherapy

Immunotherapy is being used as an exciting treatment modality for patients with liver cancer and colorectal liver metastases. Immunotherapy is generally used as an adjuvant therapy in conjunction with either chemotherapy or surgery. Immunotherapy includes the development of vaccine, cytokines as well as monoclonal antibodies to develop tumor-specific treatment. Various studies have shown immunotherapy to be an effective adjuvant therapy in increasing the survival of patients following resection (Elias et al., 1995). Although immunotherapy has been deemed as tumor specific treatment with few side effects long term studies have shown immunotherapy to have various side effects. These side effects range from minor conditions such as skin depigmentation to severe conditions affecting vital organs such as liver, bowel, and lungs (Amos et al., 2011).

1.1.5. Brachytherapy

Brachytherapy involves the use of a short-ranged radiation source very close to the tumor site. The radiation dose is delivered to the tumor site either using capsules or a wire that allows the radiation to impact only the tumor while avoiding the healthy liver regions. A key feature of brachytherapy is that it allows for a very high dose of irradiation to be delivered to the tumor in a short interval of time since the irradiation is localized to only a small region. Various studies have shown that brachytherapy have demonstrated similar cancer survival rates as compared to surgery or EBRT, and in some cases higher survival rates when it is combined with other therapy options(Armstrong et al., 1994; Ricke and Wust, 2011).

1.1.6. Cryotherapy and Radiofrequency Ablation

Cryotherapy also known as cryosurgery or cryoablation therapy uses extreme cold to freeze and destroy the cancer cells. Cryotherapy which is a minimally invasive technique has not been established primarily as a curative technique for liver cancer treatment but has been widely used in conjunction with surgery for liver resection. Cryotherapy studies have shown local recurrence rates of around 12-39% with a mean 5 year survival rate of 17% (Adam et al., 1997; Pathak et al., 2011). Radiofrequency Ablation uses a high frequency alternating current to generate the heat necessary to destroy liver cancer cells. The radio frequency alternating current is delivered to the tumor region with the use of a needle like probe to increase the temperature of the surrounding region and destroy the tumor. Radio frequency ablation is widely used as an outpatient procedure. Radio frequency ablation studies have showed local recurrence rates of around 10-31% with a mean 5 year survival rate of 24%(Pathak et al., 2011).

1.2. Selective Internal Radiation Treatment (SIRT)

Selective Internal Radiation Treatment (SIRT) is a liver directed therapy which delivers the radiation for the treatment of the cancer via intrahepatic arterial administration. The treatment delivers microspheres laden with yttrium 90 (Y-90) which is a high energy beta particle-emitting radio-isotope.

1.2.1. Selectiveness of SIRT

The treatment is based on the notion of preferential deposition of microspheres in the tumor which yields a high concentration of the radiation in the tumor region as compared to nearby healthy tissue. The primary reason for the selectiveness is the transfemoral injection of the microspheres in the hepatic artery from which the tumor blood supply is derived. Since the radio-isotope β-radiation has a mean range of only 2.5 mm the primary effect of the radiation is seen only in regions having the tumor or the near vicinity (Gulec and Fong, 2007). This provides the selectiveness which renders very little radiation to be delivered to the healthy liver tissue. Thus SIRT has been determined as a liver cancer treatment which delivers maximum radiation dose to the tumor and minimum toxicity to the surrounding healthy tissue which has been one of the major challenges in clinical practice.

1.2.2. Development of SIRT

Treatment with Y-90 microspheres was first demonstrated by Ariel et al for a single patient with colorectal cancer liver metastases (Ariel and Padula, 1978). Later, further studies published by the same group used SIRT with conventional chemotherapy to extend the life of the patients by about 28 months (Ariel and Padula, 1982). Thereafter, Gray et al. demonstrated that the optimal size for the microspheres is around 30-35 μm with 4000 microspheres per gram of the liver tissue for the administration of the effective dosage for the treatment of the cancer without inducing toxicity (Gray et al., 1990). Various studies have described clinical trials using SIRT with Y-90 microspheres for treatment of CRLM. The major studies published include a phase 3 study using hepatic artery chemotherapy (HAC) with floxuiridine, a phase 2 trial using chemotherapy with the without SIRT, a phase 1 study with oxaliplatin, a phase 1 study with irinotecan hydrochloride and a pre-operative SIRT treatment for tumor downsizing (Gray et al., 2001; Gulec et al., 2010; Sharma et al., 2007; Van Hazel et al., 2004).

1.2.3. Dose calculations for SIRT

In the first studies published by Ariel et al. a dose of 3.7-5.5 Gbq of Y-90 resin microspheres is suggested (Ariel and Padula, 1978). The calculation of the dose of the Y-90 microspheres is based on the Committee on Medical Internal Radiation Dose (MIRD) approach. It is estimated that out of the total dose delivered to the patient about 120-180 Gy is delivered to the liver (Gulec et al., 2006). For the calculation of this dose it is required to calculate the fractional liver uptake, fractional tumor uptake and the lung uptake (Gulec et al., 2006). To determine the fractional liver and tumor uptake it is necessary to obtain the mass of the liver and tumor. Since mass can be directly related to the volume, volume calculation of the liver and tumor from anatomic information is of prime importance in calculation of the fractional uptake of the tumor and liver. Volumes of various organs under investigation can be obtained either through anthropomorphic phantoms, modeling using Monte Carlo simulation or through imaging techniques such Computed Tomography (CT) or Magnetic Resonance Imaging (MRI) (Barot et al., 2010; Gulec et al., 2010).

1.3. Objectives of the study

Present techniques for the determination of the anatomic volume of the liver and tumor involve manual segmentation of the liver from the CT scans. The number of slices in a CT dataset vary from 200-400, which makes the task of computing the volume of the liver manually excessively time consuming. Also, the task is greatly dependent on the skill and proficiency of the technician/doctor, which contributes to human error affecting the results.

To overcome this major drawback, this dissertation core objective was to develop a new and effective semi-automatic algorithm for the segmentation of the liver and tumor from Computed Tomography (CT) scans which results in accurate calculations of the volume of the liver for SIRT treatment. The study aimed at incorporating a semi-automatic approach to segment the blood vessels in the liver.

The successful implementation of the algorithm serves to ensure a better outcome for the SIRT based liver cancer treatment, and provides the following advantages:

i. Requirement of less user interaction resulting in both time and labor saving.
ii. Reduction in the user to user variability and error in volume calculation.

Furthermore, a primary objective of the study was to enable the rendering of the liver in 3-D for the end-user to visualize the location as well as extent of the liver in the body cavity. Moreover, since the 3-D rendering aimed to provide a complete solution of the underlying disease the study incorporates the tumor and blood vasculature of the liver embedded within the liver. The rendering thus aimed to provide the end-user the needed information about the position, size, and extent of the tumor along with the blood vasculature feeding it.

1.4. Outline of the dissertation

Chapter 2 introduces the various imaging techniques that are used in the treatment of CRLM. It also provides the details of the instrumentation and the datasets used for the study.

Chapter 3 describes the development of a semi-automatic algorithm for segmenting the liver from 3-D liver CT scans. The chapter describes the progression of the various versions of the algorithm in order to develop an optimal liver segmentation process to significantly reduce human interaction while maintaining high accuracy in the results.

Chapter 4 provides a summary of the advances in the field of tumor and blood vessel segmentation followed by a description of the developed in-house semi-automatic methods for the segmentation of tumor(s) and blood vessels from CT scans.

Chapter 5 describes the development of the parallel processing architecture to reduce the computational burden of the liver segmentation process It also provides the results of the performance on testing the developed MatLab based technique for various benchmarking processes.

Chapter 6 provides the results obtained using the liver segmentation process. Volumes obtained by the proposed segmentation approach are compared to the manually obtained volumes. The chapter also provides the results obtained for the tumor and blood vessel segmentation algorithms.

Finally, chapter 7 highlights the novelties in the features of the developed algorithms. Chapter 7 provides some insight into the merits and limitations of the liver segmentation, parallel computing, tumor segmentation and the contextual blood vessel extraction with respect to the tumor location.

CHAPTER II

IMAGING IN CRLM

Various imaging modalities have been developed in recent times for anatomical and functional imaging of cancer. This chapter introduces some of the commonly used modalities for imaging liver cancers and CRLM with the aim to provide a brief overview of the advantages and disadvantages of each modality. Also, this chapter introduces the imaging technique used in the study and the various datasets it is applied to for validation purposes.

2.1. Imaging Modalities

The commonly used imaging techniques, which have been adopted across the nation for the diagnosis of CRLM, include contrast-enhanced ultrasound (CEUS), Computed Tomography (CT), magnetic resonance imaging (MRI), liver specific contrast enhanced MRI, and positron emission tomography/ computed tomography (PET/CT) (Xu et al., 2011).

2.1.1. Contrast-enhanced ultrasound (CEUS)

Contrast-enhanced ultrasound (CEUS) is a powerful technique for imaging focal liver lesions. Perfusion mapping with the aid of contrast agents allows mapping the focal lesions for their eventual characterization. Bernatik and his colleagues have demonstrated that the diagnostic yield of CEUS as compared to CT was about 97% (Bernatik et al., 2001). CEUS is widely used in CRLM imaging for preoperative assessment of the disease. Further use of CEUS in staging and monitoring the disease is limited due to various shortcomings. These include but are not limited to operational issues requiring a highly skilled technician for its proper use, misleading diagnosis of patients with fatty liver due to chemotherapy, and difficulty in visualization of segmental distribution and three-dimensional (3-D) analysis (Hiraoka et al., 2010).

2.1.2. Computed Tomography (CT)

Computed Tomography (CT) has been established as the mainstay for follow-ups of patients with CRLM. The features of CT, which places it at the forefront of imaging CRLM, include sub-millimeter resolution, large coverage enabling imaging the entire abdomen in a single scan, and isotropic pixel sizing for 3-D visualization and volumetric analysis. Also, high-resolution scans with maximum intensity have enabled the development of 3-D rendering techniques for accurate localization and segmentation of the tumor region (Kamel et al., 2003). CT has also enabled imaging the hepatic arterial and portal venous anatomy as a substitute to conventional angiography for surgical liver cancer resections (Sahani et al., 2004).

One of the major questions in setting up an imaging protocol using CT is the number of CT scans a patient has to undergo to ensure accurate and complete determination of the extent of the disease. Portal-venous phase CT is the most commonly adopted practice in clinical environments. However, portal-venous phase CT fails to image the presence of calcified tumors, which are more easily and better seen in unenhanced CT scans. Also, small CRLM are seen as hyperattenuating during the hepatic arterial phase scans whereas large tumor show the presence of a hyperattenuating rim and a hypoattenuating center in similar scans (Miles et al., 1993). Large lesions are often seen as hypoattenuating lesions during the portal venous phase (Kopp et al., 2002). Often an equilibrium phase is added to the study to eradicate the drawbacks of the other phase CTs. The study under consideration here uses a multi-phase CT for better determination of the tumor regions.

Several studies have also been reported for assessing the optimal slice thickness for the CT scans. Studies have shown that 2.5 mm and 3.75 mm slice thickness were considerably superior than 5, 7.5 or 10 mm in imaging the disease (Weg et al., 1998). Also, studies have demonstrated that lowering the slice thickness to 1 mm added no value towards lesion detection due to the increase in image noise (Kulinna et al., 2001). Slice thickness of 2-4 mm is viewed as a recommended setting for axial viewing (Xu et al., 2011). Although CT has been established as the backbone imaging technique in every oncology program, it should be noted that up to 25% of the liver metastases are often missed in CT scans (Scott et al., 2001; Valls et al., 2001).

2.1.3. Magnetic Resonance Imaging (MRI)

Magnetic resonance imaging (MRI) is on the rise for the imaging of CRLM. MRI based studies generally include unenhanced T1- and T2- weighted and contrast enhanced pulse sequences. T1-weighted MR imaging enable the physician to assess the parenchyma for infiltration of fatty deposits as well as any diffused fatty infiltrates (Schima et al., 1997). T2-weighted images provide images with suppressed fatty deposits, which make it difficult to differentiate them from tumor masses. Also, heavily T2-weighted images with echo time of 160-180 milliseconds (msec) offer the possibility to differentiate solid and no-solid lesions (Bennett et al., 2000). Following the acquisition of unenhanced images, contrast enhanced pulse sequences are always obtained. Generally, a set of nonspecific gadolinium chelates based contrast image and a liver specific contrast based images are obtained.

Blyth et al. have suggested adopting MRI as the “gold standard” for pre-operative imaging in CRLM studies due to the high sensitivity in tumor detection (Blyth et al., 2008). MRI however cannot be employed in cases where pacemakers, implantable defibrillators or other metallic foreign bodies are present. Also, MR imaging is limited in anatomic coverage due to increased time and cost of acquisition of whole body scans. Newer multichannel MR scanners are planned to provide fast and increased coverage for scanning the abdomen and chest regions. MR imaging provides better soft tissue contrast making it ideal for liver imaging, however the adoption of MR images into clinical protocols is limited by the availability and cost of MR imaging.

2.1.4. Positron emission tomography/computed tomography (PET/CT)

PET/CT has been adopted recently at the forefront of imaging in CRLM (Gulec, 2007). PET/CT provides the unique opportunity to image not only the anatomy of the disease but the function extent of the disease as well. PET/CT hybrid scanners provide the opportunity for seamlessly fusing high-resolution CT images with functional PET images showing the glucose uptake of the cells. Since glucose uptake is directly related to the metabolism of the cells and since tumor cells have a higher uptake, PET images reveal these highly active tumor cells. Functional imaging coupled with high-resolution anatomic imaging provides the advantage of clearly demarking the boundaries of the tumor, which is essential for the complete removal of the cancer. Studies by Kinkel et al. performed showed that for equivalent specificities, PET/CT showed higher sensitivities than CEUS, CT or MRI in the detection of hepatic metastases (Kinkel et al., 2002) Also, other studies have shown that PET/CT displays a better trend than other previously mentioned modalities for the identification of CRLM (Bipat et al., 2005; Mainenti et al., 2010).

Although PET/CT shows great promise towards detection as well as staging of the tumor, its efficacy is not yet fully established in the field. Its main drawbacks remain the limited availability, high costs of imaging studies, and exposure to radiation. PET/CT is however highly recommended in patients where the diagnosis is still uncertain with the use of other conventional diagnostic modalities. Even with the advent of various imaging modalities studies have shown the failure of one or more techniques in detecting small tumors in the liver (Elizondo et al., 1991; Ward et al., 2003). As a consequence of these drawbacks, diagnosis of liver cancer and colorectal cancer is often established as a multi-modality imaging study to correctly image the boundaries of the cancer for total resection (Sica et al., 2000). It is well established that improper and partial resection of liver metastases generally results in recurrence of the disease.

2.2. Instrumentation and Image Acquisition

The CT images acquired for the study in this dissertation are obtained using a GE Discovery LS whole-body PET/CT scanner developed by GE Healthcare (Milwaukee, US). The GE Discovery LS provides registered and fused anatomical CT and functional PET images. The scanner equipped with a crystal size of 4 x 8 x 30 mm, an axial field of view of 152 mm allowing imaging of the entire abdomen area in one or two bed positions, and is capable of capturing 2-D images, 3-D images and 4-D gated images. The images acquired for the study are part of combined CT-angiogram (CTA) protocol, which involves a three-phase CT liver scan, and FDG PET/CT, carried out in a single study (Barot et al., 2010). The protocol is structured for dosimetric measurements, which involve calculation of the anatomical liver volume, anatomical tumor volume and functional tumor volume. The acquisition parameters for the CT datasets were 140 kVp, 80 mA, 0.5 rotation time, and a 512x512 pixel matrix. The images provided by the scanner are in Digital Imaging and Communication in Medicine (DICOM) format, which is considered the standard in medical imaging.

2.3. Dataset Details

Twenty datasets were used for testing the developed algorithm. The clinical characteristics of the datasets used for the study are given in Table 2.1, summarizing the extent of the disease, the location of the tumors if any, and the number of CT slices in the dataset that encompass the liver completely. It is seen that for most of the patients the disease is limited to the hepatic compartment with multiple metastatic tumors found.

Table 2.1: Clinical Characteristics of the 20 datasets[1]

Dataset	Extent of disease	Location of Tumor	Number of CT Slices in Dataset
1	**NA**		163
2	**NA**		377
3	Hepatic+ extrahepatic (Left lateral neck)	Right lobe of liver-segment VI - multifocal (Primary site of tumor-kidney)	208
4	Hepatic	Right lobe of liver (Primary site of tumor-Colorectal)	295

[1] **NA denotes non-availability of medical information.**

5	Hepatic	One large lesion 3.3cm in left lobe small in rt lobe (primary colorectal)	247
6	Hepatic	Multiple metastatic tumors in liver	189
7	Hepatic	Right lobe of liver (Primary site of tumor-Tonsillar ca)	194
8	Hepatic+ extrahepatic (Left Adrenal)	Rt lobe of liver (primary gallbladder)	192
9	Hepatic	Multiple metastatic tumors in liver	168
10	Hepatic+ Extrahepatic	Multiple metastatic tumors in liver (Primary site of tumor-Cholangiocarcinoma)	169
11			170
12	Hepatic	left lobe of live (Hepatocellular ca)	364
13	**NA**		132
14	**NA**		365
15	Hepatic	Not specified (Primary Colorectal)	189
16	Hepatic (Lung)	Multiple (primary colon)	176
17	**NA**		230
18	Hepatic	Left lobe of live (Primary tumor site-Colon)	324
19	Hepatic	Multiple metastatic tumors in liver	251
20	**NA**		356

The number of slices in the datasets varies anywhere between 132 and 377. Also, the pixel size for the datasets varies between 0.54 mm to 0.97 mm and the slice thickness varies between 0.6 mm and 5 mm. Slice thickness in CT defines the spacing between two slices of the CT dataset, with lower slice thickness providing greater resolution for imaging. However, it should be noted that reducing the slice thickness could introduce noise in the datasets. Table 2.2 shows the information of the voxel dimensions of each of the 20 datasets.

Figure 2.1 and Figure 2.2 show the distribution for the twenty datasets with respect to the number of slices and the slice thickness. The distribution of the number of slices is important to observe the effect larger datasets have on the processing time of the algorithm. It is seen from Figure 2.1 that the maximum number of slices for any dataset was 377 while the minimum was around 132. Also, the median of the number of slices shown by the red line in Figure 2.1 is around 201 with the mean number of slices being around 237. The above mentioned statistics provide an idea of the vast diversity in size seen within the acquired datasets.

Table 2.2: Voxel Dimensions for the 20 liver CT datasets

Dataset No	Voxel Dimension		
	x (mm)	y (mm)	z - Slice Thickness (mm)
1	0.98	0.98	3.00
2	0.90	0.90	0.60
3	0.77	0.77	1.25
4	0.77	0.77	1.25
5	0.55	0.55	1.25
6	0.74	0.74	5.00
7	0.70	0.70	1.25
8	0.70	0.70	2.50
9	0.69	0.69	0.60
10	0.70	0.70	1.25
11	0.98	0.98	4.25
12	0.75	0.75	0.63
13	0.76	0.76	1.25
14	0.76	0.76	0.60
15	0.70	0.70	2.50
16	0.76	0.76	1.25
17	0.70	0.70	0.63
18	0.78	0.78	1.25
19	0.70	0.70	1.25
20	0.63	0.63	0.60
Minimum	0.55	0.55	0.60
Maximum	0.98	0.98	5.00
Mean	0.75	0.75	1.61
Median	0.75	0.75	1.25

It is seen in Figure 2.2 that the average slice thickness for the 20 datasets was around 1.6 while its median was around 1.25 as demonstrated by the red line. Also, some datasets had a much higher slice thickness as marked by the red 'plus' equaling to 4.25 and 5 mm. Such variations in the slice thickness have to be kept in mind in designing the algorithm and also for 3-D rendering and volume calculations.

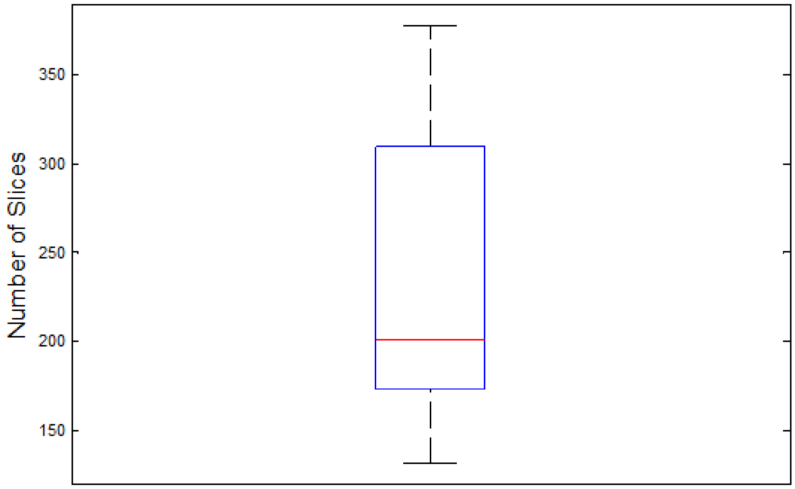


Figure 2.1: Box-plot showing variation in number of slices for the 20 datasets

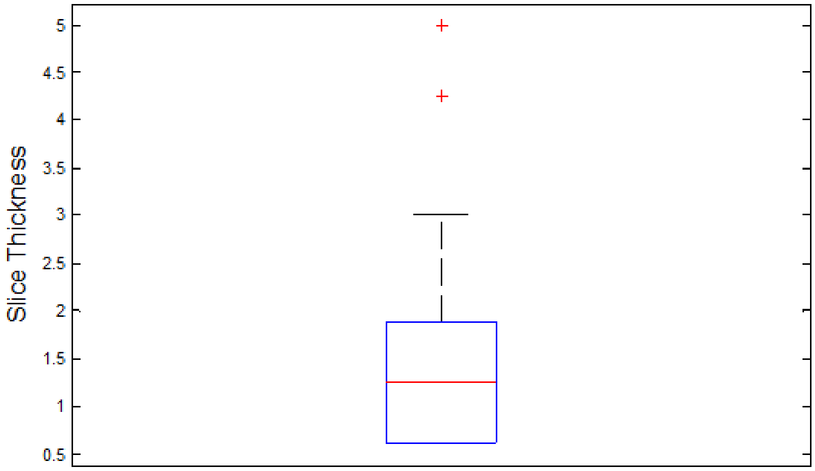


Figure 2.2: Box-plot showing variation in slice thickness for the 20 datasets

CHAPTER III

LIVER SEGMENTATION

Selective Internal Radiation Therapy (SIRT) with Yttrium-90 (Y-90) microspheres has emerged as an effective liver-directed therapy (Lau et al., 1998). SIRT dosimetry involves calculations of the dose to be delivered to the tumor as given by the fractional liver uptake and the fractional tumor uptake respectively. This involves the accurate calculation of anatomical and functional tumor volumes and anatomical volumes of liver for the determination of the tumor to normal liver ratio and consequently for the calculation of the dose of Y-90 microspheres (Murthy et al., 2005) to be administered.

Present techniques for the determination of the anatomic volume of the liver involve tedious manual segmentation of the liver from the Computerized Tomography (CT) scans. The number of slices in a CT dataset varies from 200-400, which makes the task of computing the volume of the liver manually excessively tedious and time consuming. Also, the task is greatly dependent on the skill and proficiency of the technician/doctor, which could contribute to human error and skew the final segmentation results and consequently the volume calculations.

This chapter describes the development of a semi-automatic algorithm for segmenting the liver from 3-D liver CT scans. The chapter is divided into sections that describe the progression of the various versions of the algorithm in seeking an optimal liver segmentation algorithm based on a single slice initialization process. The first version of the algorithm called version 1.0 (*Ver. 1.0*) achieved the segmentation using human initialization of multiple slices spaced equally throughout the liver dataset. Version 2.0

(*Ver. 2.0*) developed following this initial approach aimed to reduce the amount of slices initialized by the user while maintain the accuracy of the liver segmentation results. The last and final version of the algorithm version 3.0 (*Ver. 3.0*) was aimed at minimizing human intervention in the initialization phase to only a single slice. This last version of the algorithm is capable of segmenting the liver with high accuracy while only a single slice is to be initialized by the user. The segmentation processes along with the major steps and changes in the three versions are illustrated in Figure 3.1.

This chapter begins by summarizing some of the significant approaches used for liver segmentation found in the literature followed by the detailed description of the three versions of the liver segmentation algorithm. Since some of the portions of the segmentation algorithm are carried forward from one generation to the next, only those portions that undergo substantial changes are mentioned in the later versions.

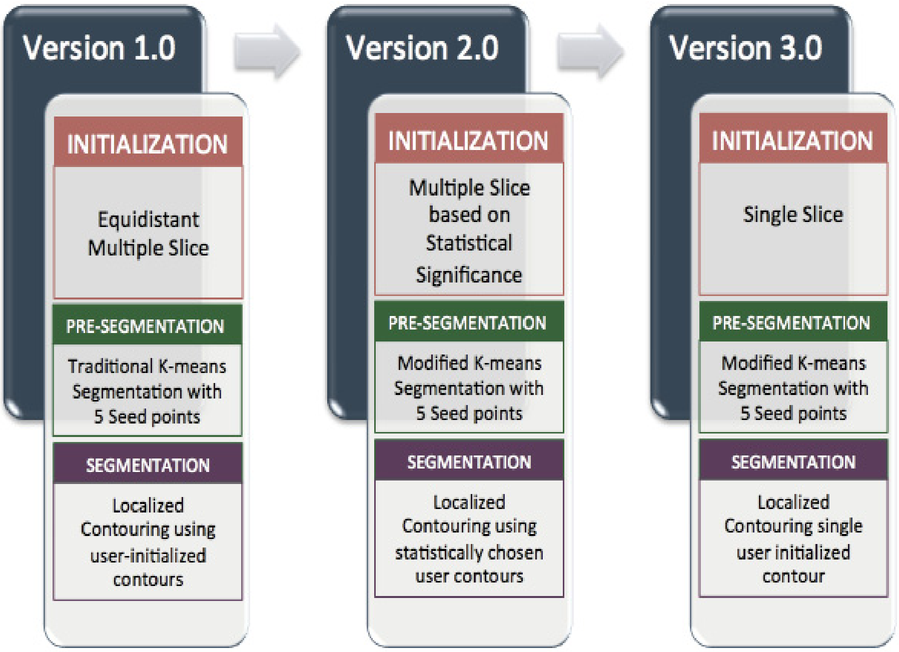

Version 1.0
INITIALIZATION
Equidistant Multiple Slice
PRE-SEGMENTATION
Traditional K-means Segmentation with 5 Seed points
SEGMENTATION
Localized Contouring using user-initialized contours
Version 2.0
INITIALIZATION
Multiple Slice based on Statistical Significance
PRE-SEGMENTATION
Modified K-means Segmentation with 5 Seed points
SEGMENTATION
Localized Contouring using statistically chosen user contours
Version 3.0
INITIALIZATION
Single Slice
PRE-SEGMENTATION
Modified K-means Segmentation with 5 Seed points
SEGMENTATION
Localized Contouring single user initialized contour

Figure 3.1: Overview of the three versions of the segmentation algorithm

3.1. Review of Liver Segmentation Techniques

Image segmentation is an important preprocessing step in many image processing applications including complex task such as brain segmentation from MR images (Shu et al., 2009), lung segmentation from CT images (Jaffar et al., 2011; Kobashi and Hata, 2010), and other medical image analysis (Deng et al., 2010; Tsai et al., 2010). Current automatic and semi-automatic procedures for liver segmentation are based on techniques that rely on (1) shape constrained segmentation using heuristic approaches (Kainmüller et al., 2007), local shape models (Seghers et al., 2007), atlas based techniques (Furukawa et al., 2007; Rikxoort et al., 2007) or nonlinear models (Heimann and Meinzer, 2009; Saddi et al., 2007); (2) rule based segmentation(Schmidt et al., 2007); (3) gradient vector flow (Chi et al., 2007; Liu et al., 2005; Massoptier and Casciaro, 2008); and (4) two or three dimensional region growing (Lim et al., 2005; Ruskó et al., 2007).

Kainmuller et al demonstrated a fully automatic liver segmentation algorithm based on a combination of a statistical deformable and a constrained free-form model (Kainmüller et al., 2007). The model is based on the typical intensity distribution of the liver boundary and intensity of neighboring anatomical structures. Seghers et al demonstrated a segmentation algorithm based on local shape models (Seghers et al., 2007) . The algorithm models the liver as a set of landmarks with local appearance models based on local shapes as contrasted to previous developed models based on global shape properties.

Atlas-based techniques have performed quite well in the segmentation of the liver. A method introduced by Furukawa et al. segments the liver based on maximum a posteriory probability (MAP) estimation using a probabilistic atlas of the liver (Furukawa et al., 2007). A novel application of the so-called level set method prevents the extraction of muscles as a part of the liver for accurate liver segmentation. A labeling approach that determines the probability of each voxel belonging to the liver based on various positional properties of each voxel has shown good performance (Rikxoort et al., 2007). A careful inspection of the results in the publication has nonetheless shown multiple local errors with inaccurate determination of liver boundaries, which is one of the major issues corrupting the performance of atlas-based techniques.

Heimann et al. demonstrated the development of a fully automatic evolutionary algorithm for liver segmentation based on statistical shape modeling (SSM) and a deformable mesh (Heimann and Meinzer, 2009). Training of the algorithm was based on 20 datasets to obtain the parameter of the SSM. Testing on 10 liver datasets showed performance comparable to results obtained by a human segmentation with an average score of 59 points based on a rating system developed at the MICCAI Grand Challenge workshop, 2007.

A two-stage algorithm developed by Saddi et al. performed the segmentation of the liver by first estimating the pose and global shape properties based on low dimensional space scanned training set and then performing a template matching algorithm to recover local deformations (Saddi et al., 2007). The method is semi-automatic requiring a single seed point in the initialization phase. A rule-based technique using Cognitive Network

Technology was also demonstrated for the segmentation of the liver (Schmidt et al., 2007). The technique based on classification of the pixels using a semantic knowledge base showed a mean overlap error of 16% in the segmentation of the liver.

Gradient vector flow (GVF) based algorithms have been very successful in the segmentation of the liver in recent studies. A method introduce by Chi et al. used a rotational template matching followed by local edge enhancement of the rib cage area with a GVF geometric snake (Chi et al., 2007). The algorithm trained using 20 datasets and tested on 10 datasets, provided good accuracy towards the segmentation of the liver. Another method based on GVF snake developed by Liu et al. demonstrated high accuracy of about 95% in estimating the liver volumes (Liu et al., 2005). The method is a combination of a GVF snake with canny edge detector and a modified concavity removal algorithm to extract the liver region. Massoptier et al. described a technique using a combination of a statistical model with a GVF active region-growing algorithm (Massoptier and Casciaro, 2008). The algorithm required no interaction between the user and analysis system and resulted in a volume overlap of more than 90%.

Deformable contours have found a prominent niche in the field of liver segmentation due to their high accuracy with no requirement for a training phase. An unsupervised deformable contour method for segmentation of the liver from CT images showed results comparable to those obtained by manual tracing using a combination of multilevel thresholds and multiscale morphological filters (Lim et al., 2005). However, the algorithm was dependent on the prior knowledge about the location of the liver. Based on region growing contours, another fully automated approach was able to segment the liver parenchyma in several cases (Ruskó et al., 2007). However, the presence of large tumors resulted in underestimated volume calculations.

Although these techniques offer highly accurate results, the algorithms need to accommodate varying protocols, data from different sources, artifacts, and the presence of pathological structures such as tumors (Campadelli et al., 2009; Heimann et al., 2009).

3.2. Liver segmentation based on multiple equidistant slice initialization process (*Ver. 1.0*)

A new method for liver segmentation has thus been developed by integrating a multiple seed point K-means clustering process and the active contours algorithm. The implementation strategy is illustrated in Figure 3.2. The novelty in this algorithm is in the manner the modified K-means based segmentation is used in combination with a localized contouring algorithm. This K-means segmentation approach requires the identification of five separate regions of the input CT images, which is a corollary contribution to this method. A key contribution however is the development of a new localized contouring algorithm, based on local regional thresholds defined in a small region around a point of interest. Pixels in an image obtained by CT scanning are displayed in terms of relative radio density rather than the traditional gray level intensity. The CT images are segmented keeping the pixel values in the Hounsfield units (HU). The HU value will be used to identify the liver tissues in the CT images. Typically the liver window is defined between -40 HU and 180 HU.

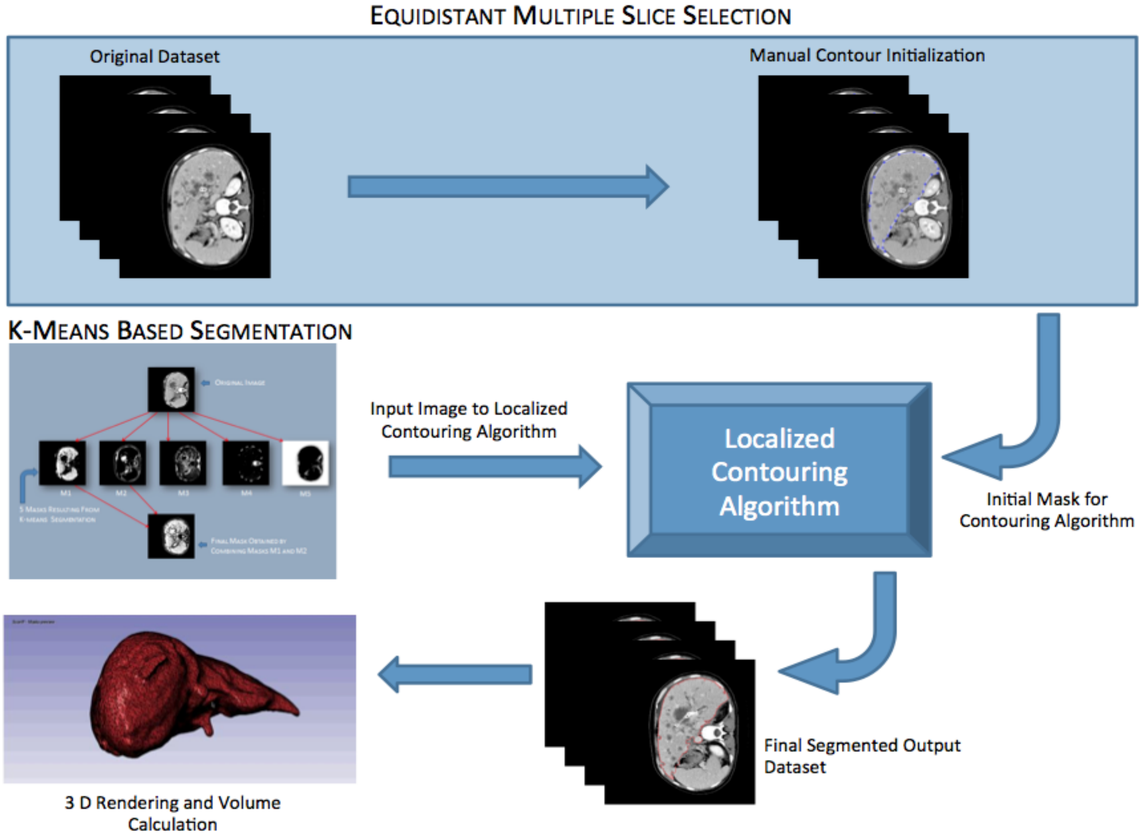

EQUIDISTANT MULTIPLE SLICE SELECTION
Original Dataset
Manual Contour Initialization
K-MEANS BASED SEGMENTATION
ORIGINAL IMAGE
M1
M2
M3
M4
M5
5 MASKS RESULTING FROM K-MEANS SEGMENTATION
FINAL MASK OBTAINED BY COMBINING MASKS M1 AND M2
Input Image to Localized Contouring Algorithm
Localized Contouring Algorithm
Initial Mask for Contouring Algorithm
Final Segmented Output Dataset
3 D Rendering and Volume Calculation

Figure 3.2: General algorithm for 3D liver segmentation and rendering - *Ver. 1.0*

Although reduced manual interaction is desired, the variations in the liver datasets seen in patients having tumor along with the need to obtain accurate volume determination for the SIRT treatment justify the use of the proposed semi-automatic method. For the version 1.0, the algorithm was designed to require the user to manually pick 5 points of varied intensity in the scan for the K-means segmentation as well as a rough outline of the liver in widely spaced slices throughout the given CT dataset. From this initialization process, the following automated steps of the modified K-means algorithm are thus considered.

3.2.1. Modified K-means algorithm and its suitability to liver segmentation

This is a traditional clustering technique which tries to partition the given dataset points into various clusters whose means are similar (Kanungo et al., 2002; Pappas, 1992). The K-means algorithm as used here aims to minimize the squared Euclidean distance for clustering the data points to their respective groups. In the case of the images to be segmented, the data points are pixels which are to be clustered around the mean intensity or the pixels a priori chosen by the user. The points are clustered based on the intensity of the selected points by minimizing the objective function *U* as defined in Eq. 1

$$\mathrm{U} = \sum_{i=1}^{k} \sum_{x_j \epsilon S_i} (x_j - \eta_i)^2 \tag{1}$$

In this equation, k represents the number of clusters S_i with i = 1, 2, … K, η_i is the intensity of the selected point, while x_j represents each point that belongs to cluster S_i ($x_j \in S_i$). In the case of liver segmentation, the CT slices are portioned into five regions whose mean intensity levels can be given either by the user or preset. The five regions manually identified on each CT slice are: (1) Liver, (2) Surrounding organs, (3) Peripheral Muscles, (4) Ribs/Spinal cord, and (5) Outside of the body. The segmentation algorithm using the K-means yields 5 masks namely M1 through M5 corresponding to the aforementioned 5 regions as depicted in Figure 3.3.

Selection of the seed points for the K-means algorithm is an important feature for the algorithm. A random selection of seeds for the clusters as shown in Figure 3.4 results in the generation of incorrect masks. This is due to the clusters being formed around those random seeds which range from -1024 HU to 3000 HU for a typical CT slice. Since the liver was typically seen in the -40 HU to 180HU range random selection of seeds provides very less probability to achieve the required feat. In case of uniformly selected points in the range of the CT acting as seeds, the centroids are centered around -1024 HU, -205HU, 614HU, 1433HU and 2252HU. These seeds fail to segment the regions containing the liver from the image as seen in Figure 3.4. An alternative would be to use uniformly selected seed points in the -40 HU to 180 HU liver range. Empirically however, such a selection of seeds yields centroids too close to one another for suitable segmentation. Also a large number of points below -40HU and above 180 HU were misclassified and marred the resultant masks.

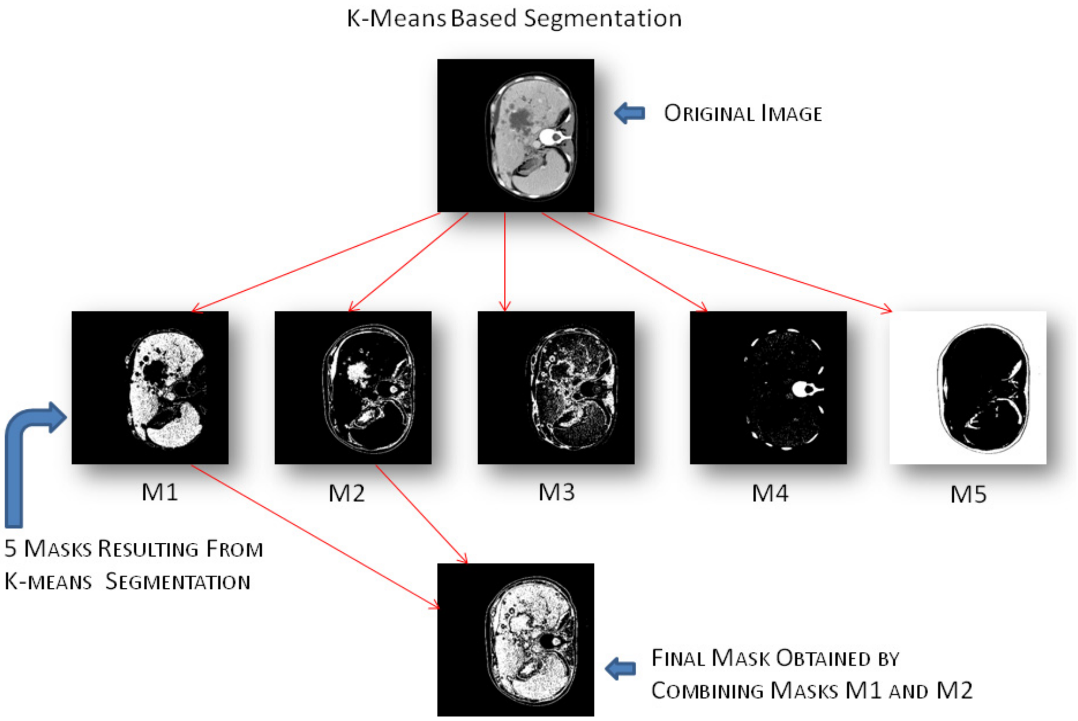


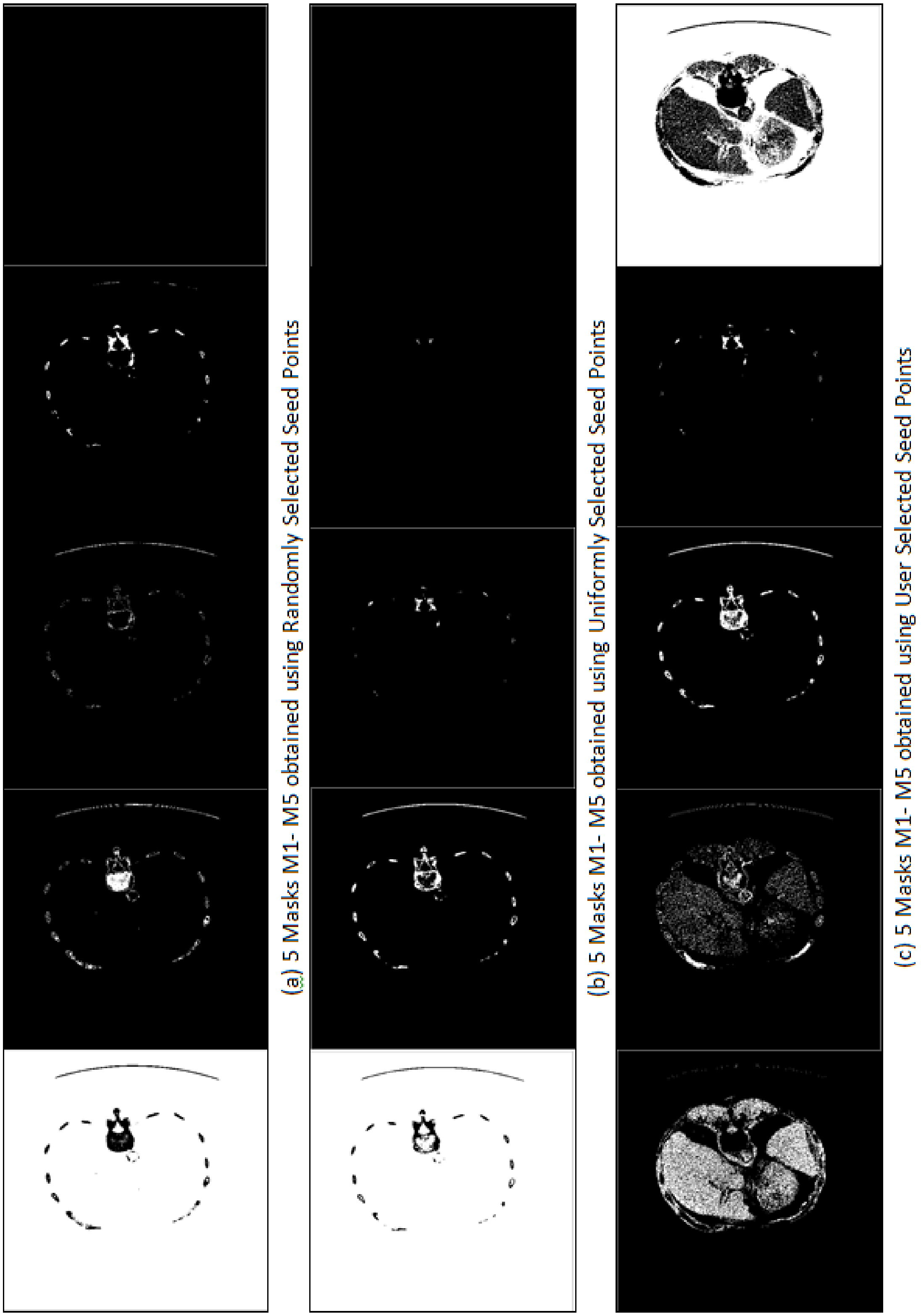
Figure 3.3: The 5 reference regions, M1 through M5, needed to apply the K-means based segmentation method on a liver CT dataset.

Hence, the modified K-means approach as deployed in this study segments effectively the different regions of the CT slice around the user selected points which act as seeds for each of the aforementioned five masks. The selection of the seeds, rather than a random selection or uniform selection of points in the entire intensity range of the image, yielded much better segmentation results as shown in Figure 3.4 for any given slice. These seed points clearly differentiate the various organs of interest as compared to the other two methods of seed selection where the liver region is not at all visible.

Another important finding of this study is in determining that the first two masks namely liver (M_1) and surrounding organs (M_2) can be utilized to effectively segment the image. The masks M_1 and M_2 are in this case ORed (+) together to obtain the final mask (M_{final}) as shown in Eq. 2.

$$M_{final} = M_1 + M_2 \tag{2}$$

This final mask is what is applied on the CT slice for segmenting the liver region. Based on empirical results, it was determined that the optimal mask would require a combination of the first two identified regions since in some cases the entire liver is not seen in mask M1 due to the inhomogeneous intensity distribution across the entire liver region. Combining masks M_1 and M_2 ensures that no part of the liver is missed out. In order to demonstrate the merits of the modified K-means approach, Figure 3.5 shows the results of applying the modified K-means as compared to the traditional K-means approach for segmenting the liver. Part (A) of Figure 3.5 demonstrates the differences obtained in the masks M_1 and M_2 and the mask M_{final} for the input CT slice shown. Figure 3.5 (a2-a4) shows the results obtained using the modified K-means, and Figure 3.5 (a5-a7) show the results obtained using the traditional K-means approach. It is observed that M_{final} obtained using the traditional K-means overestimates the liver region.

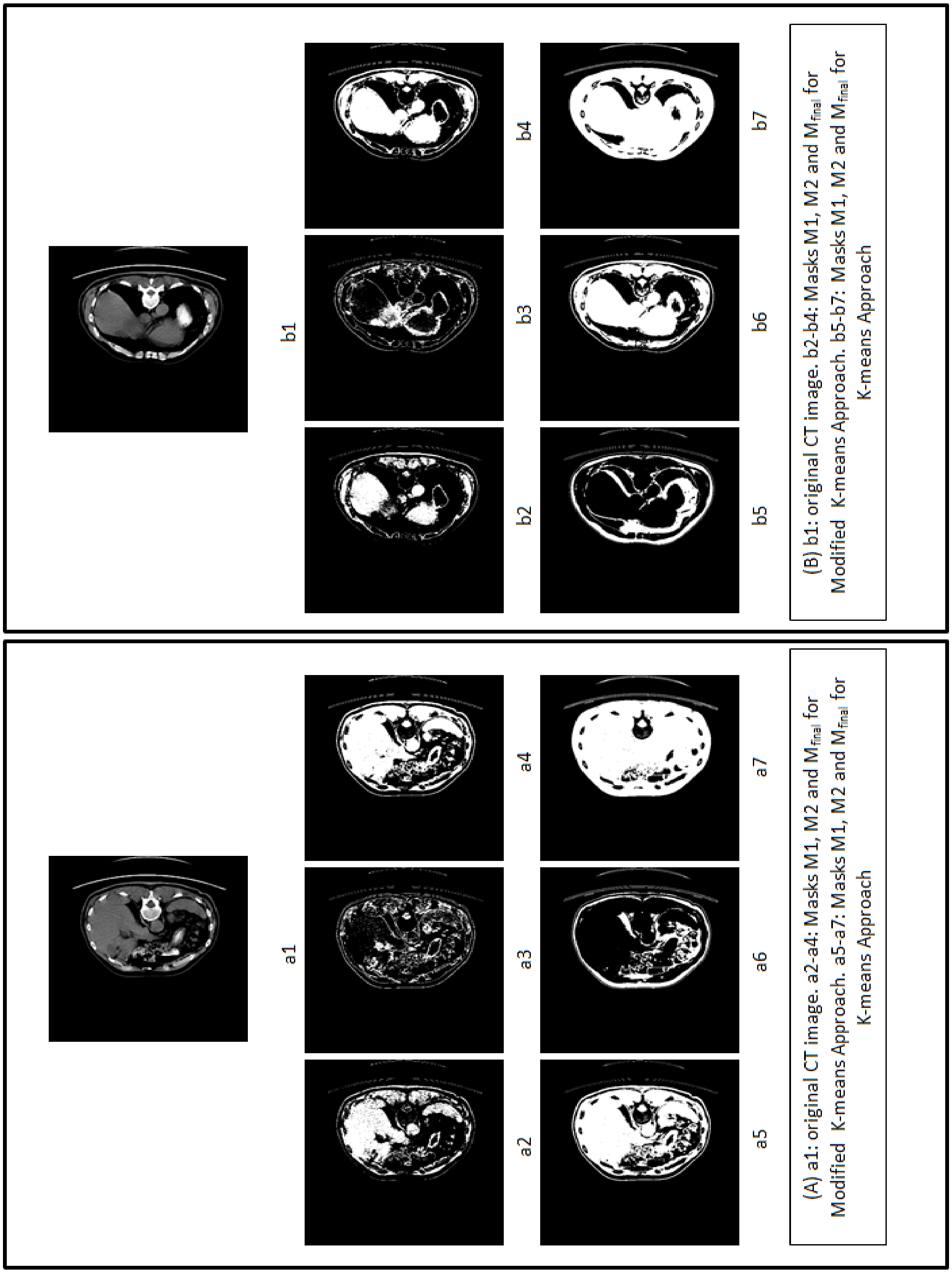


(a) 5 Masks M1- M5 obtained using Randomly Selected Seed Points

(b) 5 Masks M1- M5 obtained using Uniformly Selected Seed Points

(c) 5 Masks M1- M5 obtained using User Selected Seed Points

Figure 3.4: Selection of seed-points for K-means based image segmentation step.



(A) a1: original CT image. a2-a4: Masks M1, M2 and M_{final} for Modified K-means Approach. a5-a7: Masks M1, M2 and M_{final} for K-means Approach

(B) b1: original CT image. b2-b4: Masks M1, M2 and M_{final} for Modified K-means Approach. b5-b7: Masks M1, M2 and M_{final} for K-means Approach

Figure 3.5: Comparison of the Modified K-means method and traditional K-means method.

An observation from Figure 3.5 (A) would indicate that only using mask M_1 from the traditional K-means approach would yield the required liver region. In order to show that this is not actually the case for every slice, results for another slice of the dataset is shown in Part (B) of Figure 3.5. Figure 3.5 (B) demonstrates the need of using the mask M_{final} as obtained in Eq. 2. Note that by using only mask M_1 for either the modified or the traditional K-means would results in the underestimation of the liver region. Segmented image S1 after the modified K-means is obtained by applying Eq. 3 as follows:

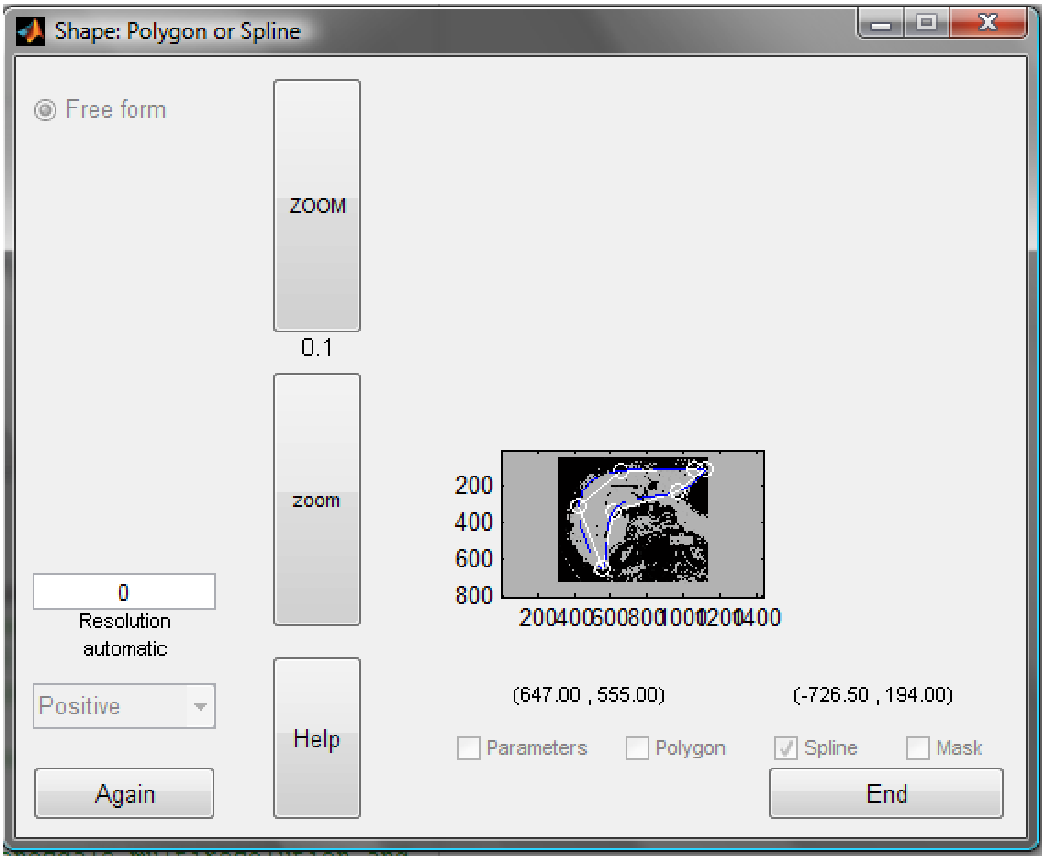
$$S_1 = M_{final} * I \tag{3}$$

S_1 is the image after the K-means clustering, and I is the CT image slice being analyzed.

3.2.2. Initialization of the Localized Contouring Method *Ver. 1.0*

The next step of the algorithm is the contour-based segmentation process. One of the major issues that concern contour-based segmentation is the initialization of the contour. This considers a user defined mask as input for the initialization. The user marks the approximate boundaries of the liver using a mouse pointer in slices that are widely apart from one another.

Figure 3.6 below shows the initialization step of the process for a particular slice of the dataset. As can be the user marks the approximate boundaries of the liver by using a mouse pointer. The algorithm is thus to a great extent independent of the choice of this initial guess. The number of points to be selected is also user defined. The expectation is that the higher the number of points considered the better the results would be; however, we did not see this effect in the images we processed as long as a sufficient number of points delineating the boundary of the liver were used. Once the initial guess is fed, the algorithm iteratively proceeds to determine the boundary of the liver without any more user interaction required.



● Free form

0
Resolution automatic

Positive

(647.00 , 555.00)

(-726.50 , 194.00)

□ Parameters □ Polygon ☑ Spline □ Mask

Figure 3.6: Contour Based Liver Segmentation Initialization Process

In this version of the algorithm (*Ver. 1.0)*, the user is presented with various slices that are equidistant apart in the dataset for initialization. Typically an inter-slice distance of five is selected for the analysis, i.e. the user initializes every fifth slice of the dataset. The determination of the inter-slice distance was based on an error analysis whereby the error in calculation of the volume of the liver was carried out with different inter-slice distances. It was seen that for up to an inter-slice distance of 5 the error was within the 10% clinical bounds. An increase in the inter-slice distance resulted in larger errors. A detailed description of this error analysis is presented in the results section.

3.2.3. Localized-region-based active contouring algorithm

Once the initialization for all the slices of the dataset is obtained interactively, a localized-region-based active contouring algorithm is deployed. Localized region growing algorithms are more robust than contouring algorithms not based on global energies for segmenting heterogeneous objects like the liver (Lankton and Tannenbaum, 2008). Let C denote the closed contour, such that C = {x|ф(x) =0} is a set of zero level for the signed distance function. The interior and the exterior of the contour are defined using the smoothened approximation of the Heaviside function in Eq. 4. The interior of the closed contour C is thus defined as follows:

$$H\Phi(x) = \begin{cases} 1, & \Phi(x) < -\varepsilon \\ 0, & \Phi(x) > \varepsilon \\ \frac{1}{2}\left\{1 + \frac{\Phi}{\varepsilon} + \frac{1}{\pi}\sin\frac{\pi\Phi(x)}{\varepsilon}\right\}, & otherwise \end{cases} \quad (4)$$

In reference to Eq. 4, the exterior of closed contour C can then be expressed as {1 – Hф(x)}. In order to calculate only the local energies, a masking parameter α is introduced in the algorithm. The parameter defines the radius of the circle around the contour which has to be masked in order to calculate the local energies. This localization radius is defined in terms of the size of the image and its value is calculated as shown in Eq. 5.

$$\alpha = \frac{1}{4}(P_r + P_c) \quad (5)$$

where P_r is the number of rows in the image and P_c is the number of columns in the image. The mask generated by using the above value of α is given by Eq. 6.

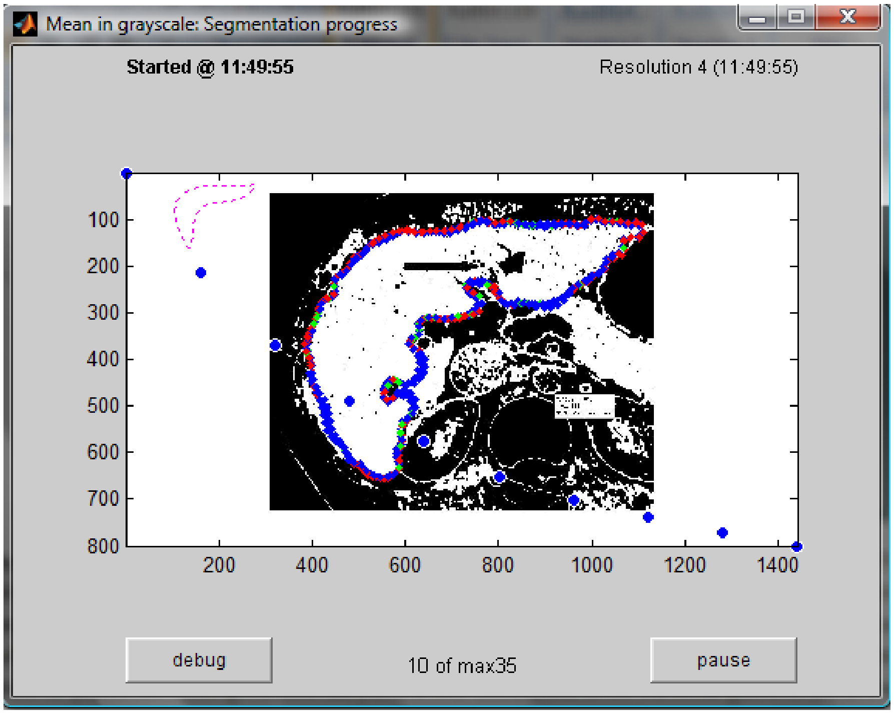
$$M_\alpha = \begin{cases} 1 & \left||x - y|\right| < \alpha \\ 0 & otherwise \end{cases} \quad (6)$$

The algorithm incorporates the well-known Chan–Vese energy paradigm to model the interior and the exterior of the contour for segmentation(Chan et al., 2000). The localized version of the Chan-Vese energy function is given by Eq. 7,

$$E_{CV} = \int H\phi(y)(S_1(y) - \mu_{int})^2 + (1 - H\phi(y))((S_1(y) - \mu_{ext})^2\, dy \qquad (7)$$

where μ_{int} μ_{int} and μ_{ext} are the mean intensities of the interior and exterior regions of the contour, respectively. The localized versions of the mean intensities are obtained by restricting the field of view only to the masked region given by masking the image with M_α M_α .

Figure 3.7 shows the iterative process of segmentation for a particular slice of the dataset. The red lines in the Figure 3.7 show the locations where the contour is trying to compress and the blue regions show the locations where the contour is trying to expand.



Mean in grayscale: Segmentation progress

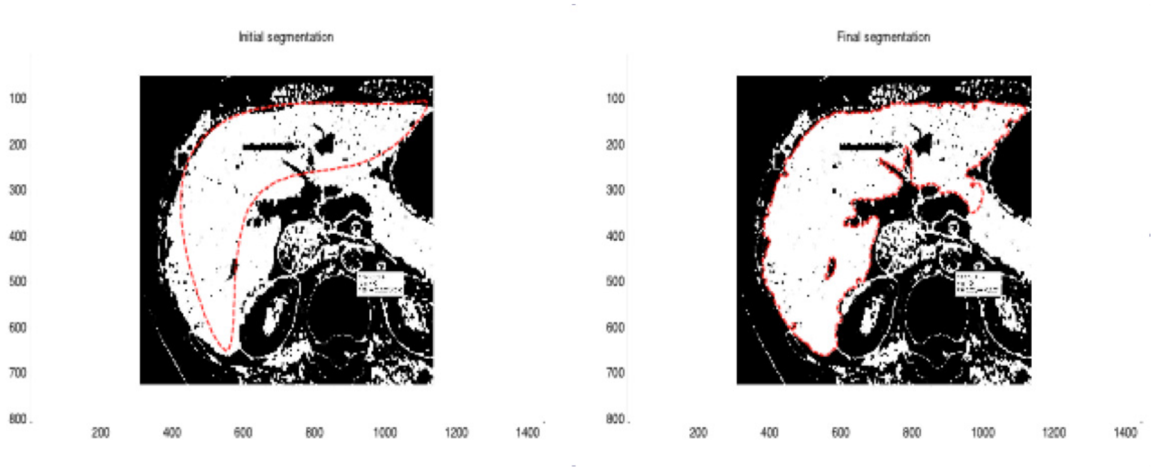
Started @ 11:49:55

Resolution 4 (11:49:55)

10 of max35

Figure 3.7: Contour Based Segmentation (in progress)

Figure 3.8 (a) shows the initial contour selected by the user, whereas Figure 3.8(b) shows the result of the completed localized contouring process.



(a) (b)

Figure 3.8: Liver Segmentation example (a) User initialization shown in red (b) Final output of the algorithm after localized contouring step

The extracted contour of the liver is as shown in Figure 3.9. These results show a very good segmentation of the liver as portrayed in Figure 3.13, where the contour is overlapped with the input slice. A key note to be added is here that since the first algorithm (K-means) works on the basis of discrepancies in the image intensities of the different organs and the since the second algorithm (contour based) works on the basis of extracting edge information the combined algorithm has the advantage of providing more detailed information and better segmentation results. Also since the algorithms provided in the document are independent of the type of image and feature to be segmented we can also use the same algorithm for delineating tumors from CT or PET data sets.

Another example of the process for a different dataset is shown in Figure 3.11 through Figure 3.14. Figure 3.11 shows the slice of the dataset to be segmented, Figure 3.12

shows the initial and final contours, Figure 3.13 shows an overlay of the final obtained contour on the liver slice and Figure 3.14 shows the boundaries of the liver segmentation.

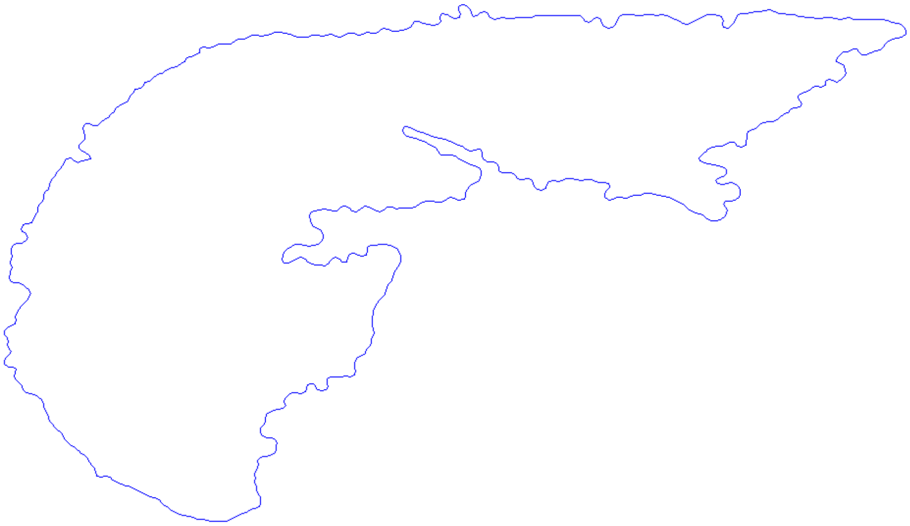


Figure 3.9: Extracted Liver

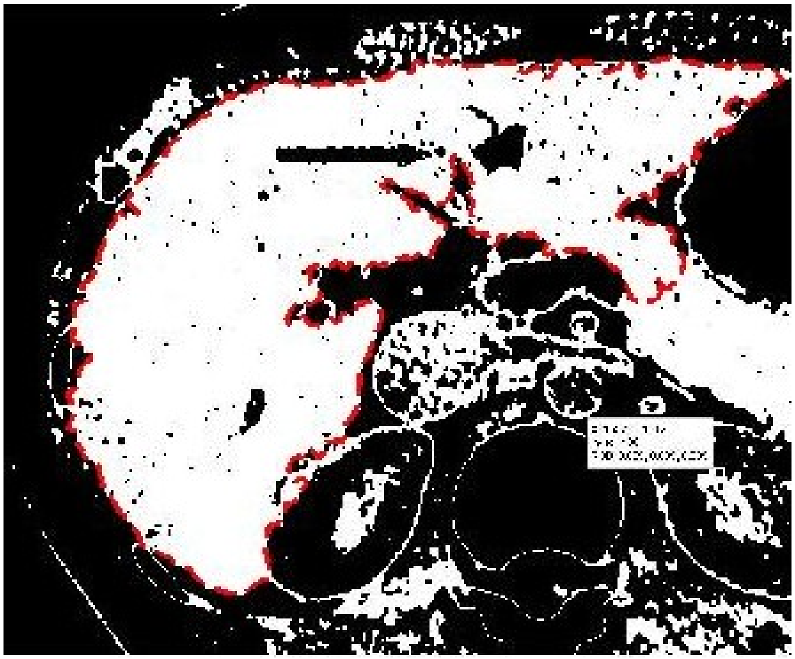


Figure 3.10: Extracted Liver with Contour Displayed

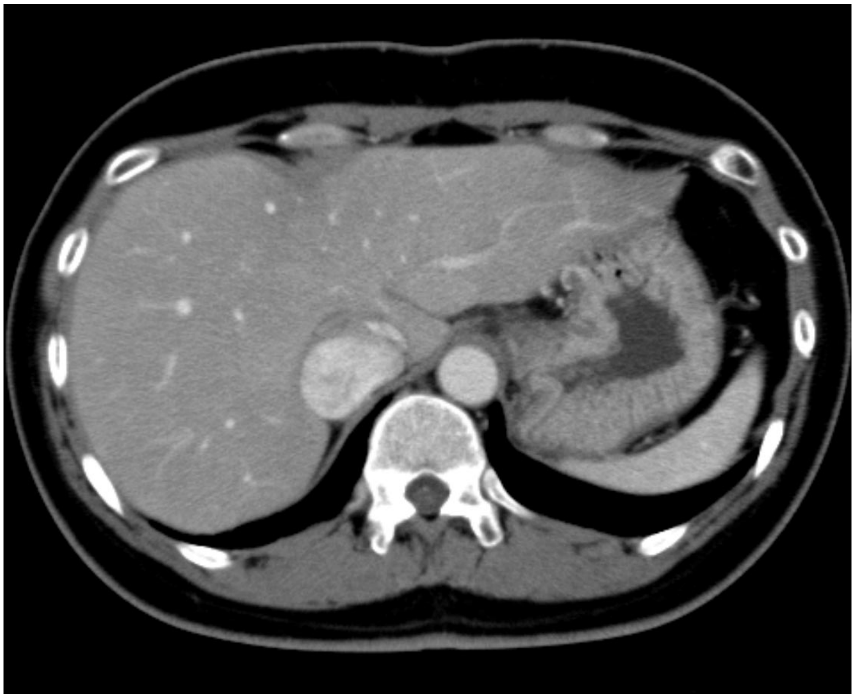


Figure 3.11: Original Image

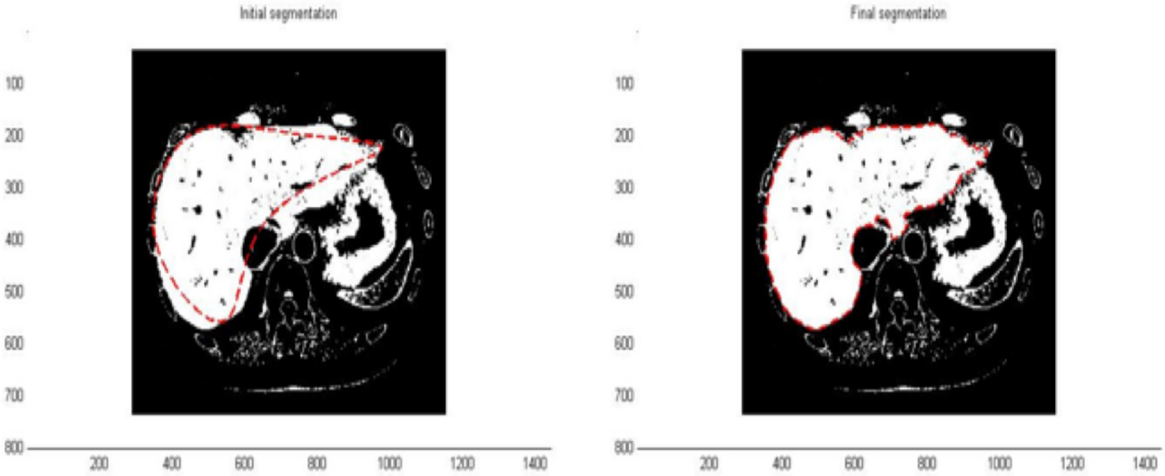


Figure 3.12: Result of the Algorithm

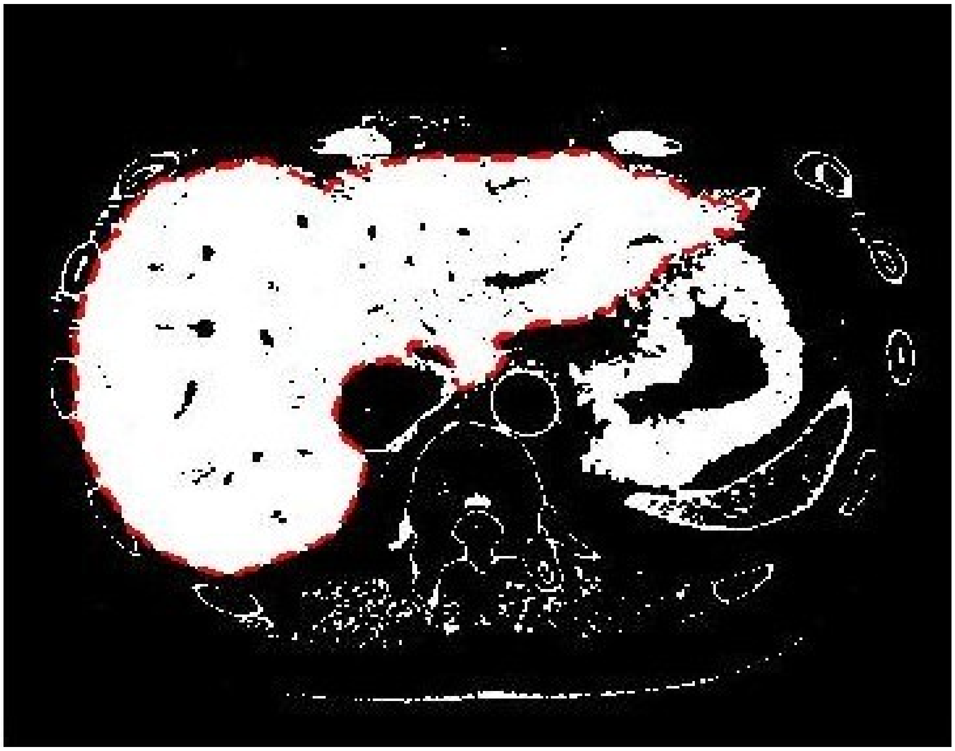


Figure 3.13: Extracted Liver with Contour Displayed

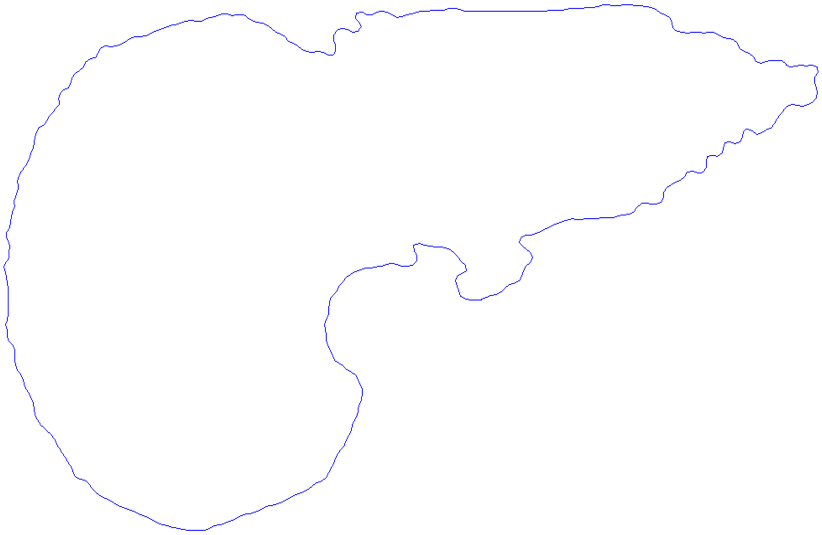


Figure 3.14: Extracted Liver

As mentioned earlier, *Ver. 1.0* of the algorithm requested a user to initialize the contour in equidistantly spaced slices. The major impetus for further development of the algorithm was to reduce the number of slices selected by the user while maintaining the high accuracy of the results. With this in mind, *Ver 2.0* was developed to reduce human intervention by using an information- based slice selector to enumerate the slices automatically for human initialization.

3.3. Algorithm (*Ver. 2.0*) with information-based slice selection for contour initialization

In *Ver. 2.0* of the algorithm, the major and most important change is the manner in which the slices that are to be marked by the user for initialization are automatically selected. In this section we first elucidate the changes that are made to the K-means based segmentation and then describe the changes made with regards to the contour initialization process. The general flow diagram of the *Ver. 2.0* algorithm is as shown in Figure 3.15.

3.3.1. Changes to the modified K-means algorithm

The K-means segmentation approach tries to partition the given dataset points into various clusters whose means are similar. Since the process is statistical in nature, in order to obtain more robust segmentation results, the algorithm for each image is replicated 3 times to achieve better results. The intersection of the results of the 3 segmentation runs for each image slice is taken as the final segmentation. Selection of the intersection of the 3 runs provided results which were free of misclassifications experienced using single runs. Lastly, the modified K-means is operated on a so called ‘online update mode’ where the sum of distances is calculated with the movement of every pixel to a different cluster. Although this step is slightly more time consuming than when using a batch update, higher accuracies are guaranteed since the local minima of the distance function can be calculated more accurately.

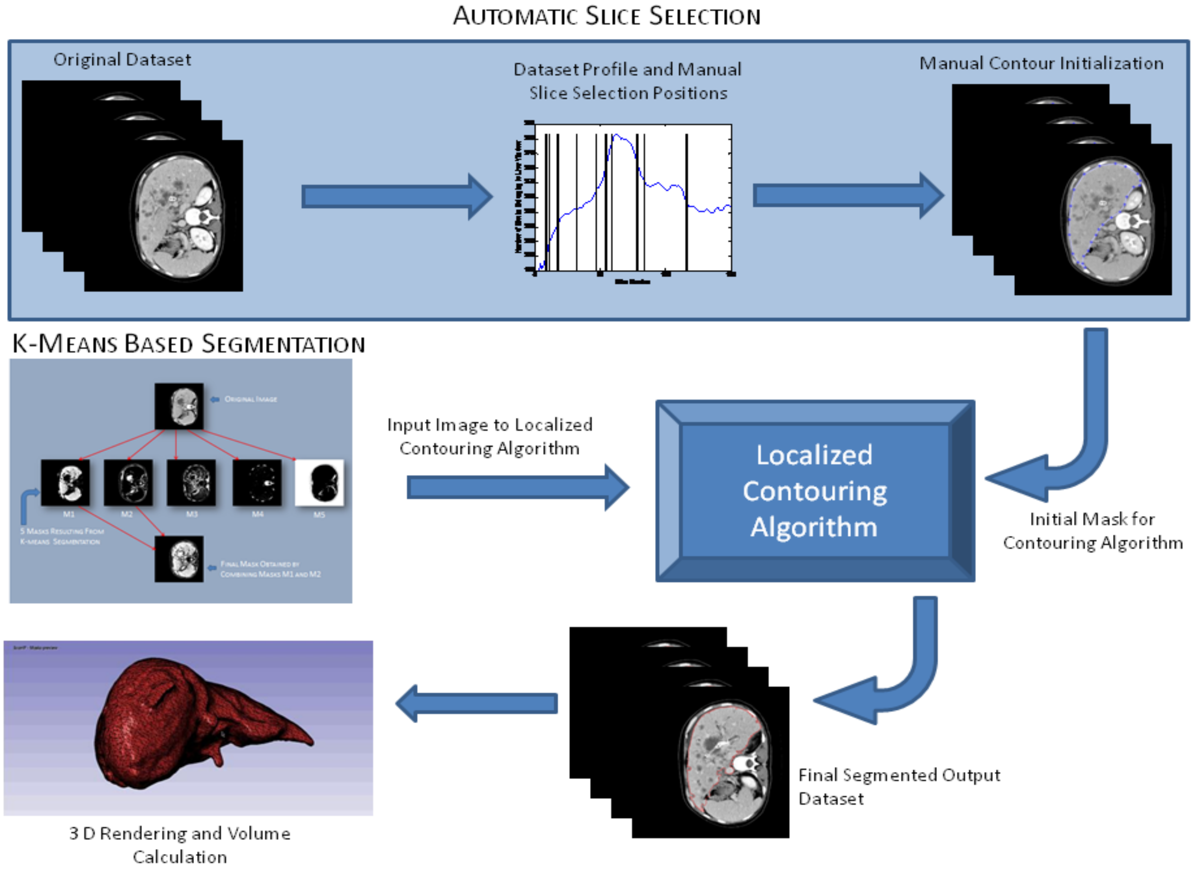

Automatic Slice Selection
Original Dataset
Dataset Profile and Manual Slice Selection Positions
Manual Contour Initialization
K-Means Based Segmentation
Original Image
M1
M2
M3
M4
M5
5 Masks Resulting From K-means Segmentation
Final Mask Obtained by Combining Masks M1 and M2
Input Image to Localized Contouring Algorithm
Localized Contouring Algorithm
Initial Mask for Contouring Algorithm
Final Segmented Output Dataset
3 D Rendering and Volume Calculation

Figure 3.15: General algorithm for 3D liver segmentation and rendering for *Ver 2.0* algorithm

3.3.2. Initialization of the Contouring-Based Segmentation

The next step of the algorithm is the contour-based segmentation process. One of the major issues that concern contour-based segmentation processes is the initialization of the contour. This contour-based segmentation algorithm takes a user-defined mask as input for the initialization algorithm. The user marks the approximate boundaries of the liver using a mouse pointer in slices that are widely apart from one another. Moreover, the selection of the slices in which the user picks the initial contour depends on the change in information (presence of liver) from slice to slice. This is obtained by first dividing the entire image S1 into blocks of 16x16 pixels as shown in Figure 3.16. The selection of blocks of 16x16 pixels was determined empirically to be the best solution.

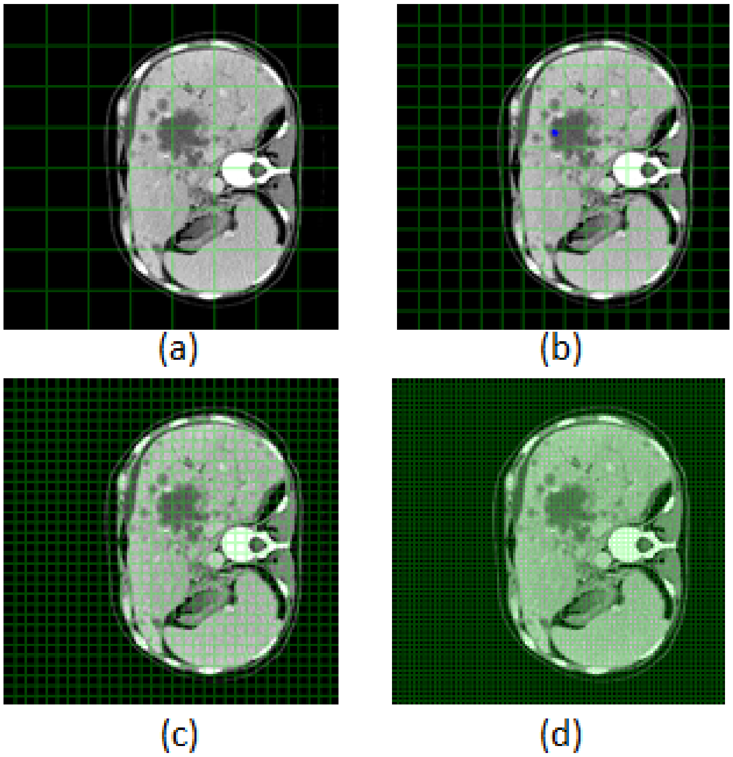


Figure 3.16: Division of liver CT image in blocks of sizes (a) 64x64 (b)32x32 (c) 16x16 (d) 8x8

Inside each block, the pixels that fall in the abdominal CT window of -40HU to 180HU were counted. A block was marked as being part of the liver region if at least more than half of the pixels were found in the abdominal CT window as described in Eq. 8. Thus,

$$block\ marked = \begin{cases} 1 & if\ {}^{n}/_{N} \geq 0.5 \\ 0 & otherwise \end{cases} \tag{8}$$

Elements n and N are the number of pixels in the abdominal window and the total number of pixel elements in the block, respectively. Recall that N = 16x16 was the window size selected for segmentation of the liver datasets presented in the results section.

Figure 3.17 shows a typical profile curve for dataset 1 which displays the number of blocks belonging to the liver window as a function of the slice number in a particular dataset, along with the slices used to calculate the value of the curve at some particular instant of time. Such a curve gives an estimate of the changes in the size as well as the potential extent of the liver region seen across slices, providing as a consequence the information needed to select those slices that most probably contain a large portion of the liver region.

The first slice of the CT dataset was always selected for manual contour initialization. In this algorithm, in order to minimize the number of slices considered for contour initialization, a subsequent slice would need to show a 5% change in the number of blocks marked as in Eq. 8 with respect to its predecessor. The intermediate slices that are not selected for manual initialization used the same initial contour as the slice closest to it in the dataset. This approach yielded around 4-5% of the entire dataset to be initialized by the user, rather than the time consuming slice by slice contouring of the entire dataset that is used in clinical practice. This led to good segmentation results and the volume measurements at were highly accurate. As an example, the slices chosen for manual initialization are marked by the black bars in Figure 3.17 for a particular dataset.

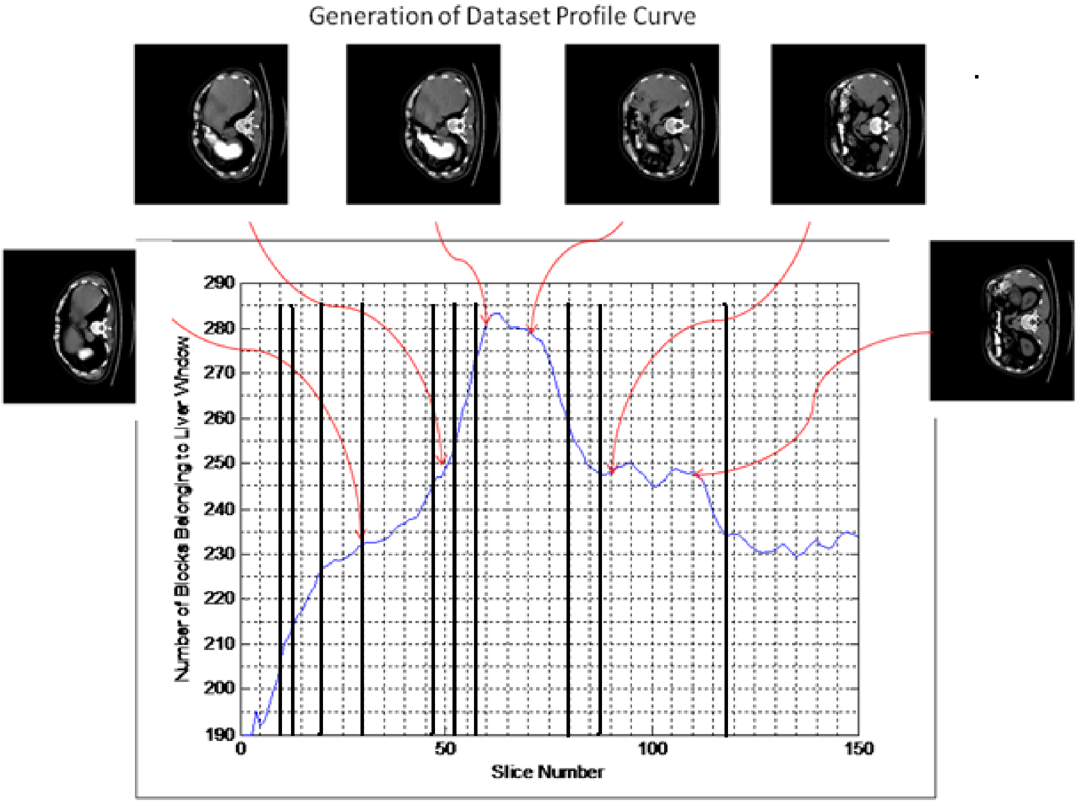


Figure 3.17: Generation of dataset profile curve depicting the slices corresponding to the marked points on the dataset profile curve.

The automatic slice selection algorithm aims at determining the slices that are to be manually initialized for the contouring algorithm. This algorithm calculates the change in the extent and structure of the liver shown across the slices by counting the number of blocks belonging to the liver as expressed by the range of -40 HU to 180 HU. This generates a profile curve for a particular dataset like the one shown earlier in Figure 3.17. Since block size is a parameter that gauges the accuracy of the algorithm, experiments were carried out using different block sizes. The performance of the algorithm is evaluated using a new parameter Ψ as defined in Eq. 9,

$$\Psi = |\, p - \bar{p}| \tag{9}$$

Element p represents the number of the slice manually selected, and $\overline{p}$ is the number of the slice picked by the algorithm. Figure 3.18 shows the parameter Ψ plotted as a function of the block size.

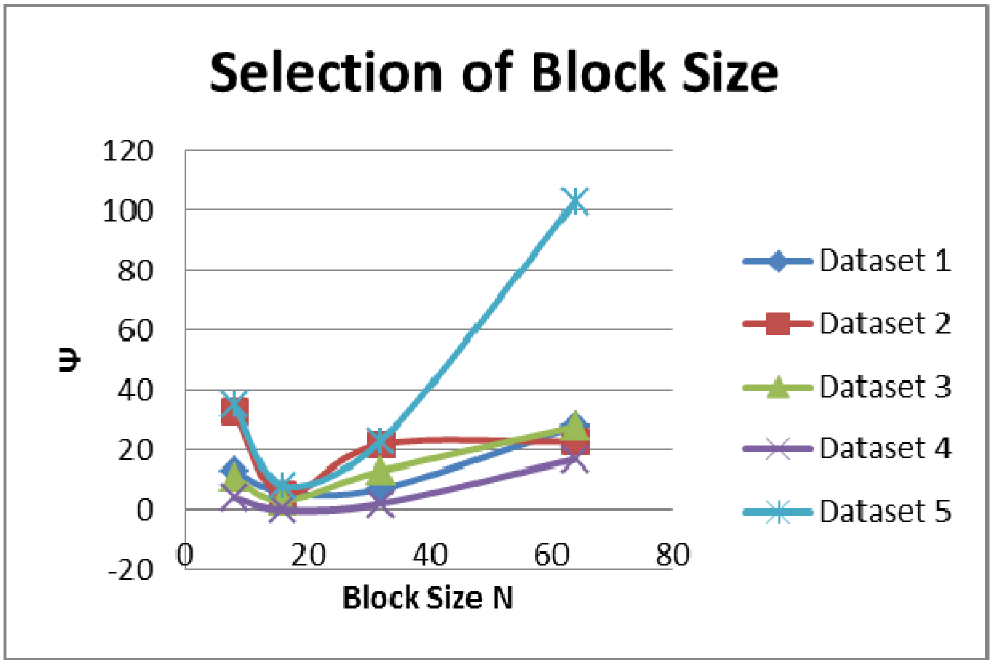


Figure 3.18: Selection of block size based on Ψ performance

Since Ψ represents the distance between the true largest slice obtained manually and the largest slice obtained by the algorithm, a smaller value of Ψ would indicate a better performance. Results in Figure 3.18 show that the lowest values of Ψ are consistently seen for block size 16x16. Also dataset 5 shows a very high value for block size of 64x64. This can be attributed to the presence of several smaller sized objects in the CT images such as duodenum whose intensity distribution is similar to that of the liver. Once the initialization for all the slices of the dataset is obtained interactively, the localized-region-based active contouring algorithm is employed as described in section 3.2.3.

Consequently, Ver. 2 of the algorithm relied on fewer slices that were automatically selected based reasoned information change from slice to slice.

With the challenging goal of reduce the number of slices used in the initialization phase to a single slice, *Ver. 3.0* was developed based on the selection of the manual contour of the slice with the largest extent of the liver in the dataset.

3.4. Liver segmentation algorithm using single slice contour initialization (Ver. 3.0)

The *Ver. 3.0* of the algorithm incorporates the K-means based segmentation algorithm form *Ver. 2.0*. Also, the localized contouring algorithm is carried forward in the new version of the algorithm with no changes to the core process. The major contribution of this version is in determining an initialization process that would rely on using only one CT slice and two seed points or end points to fulfill the segmentation process. In this section we describe the changes that were made towards this contour initialization approach for single slice initialization. The general flow diagram of *Ver. 3.0* algorithm is shown in Figure 3.19.

3.4.1. Contour Initialization

The most important issue that affects the performance of any contouring algorithm is the initialization of the contour, especially in situations like liver segmentation where various organs of similar intensity distributions are seen. This unique contour initialization process is achieved in a four-step process to reduce both human intervention and extent of error due to subtle variations in CT scans.

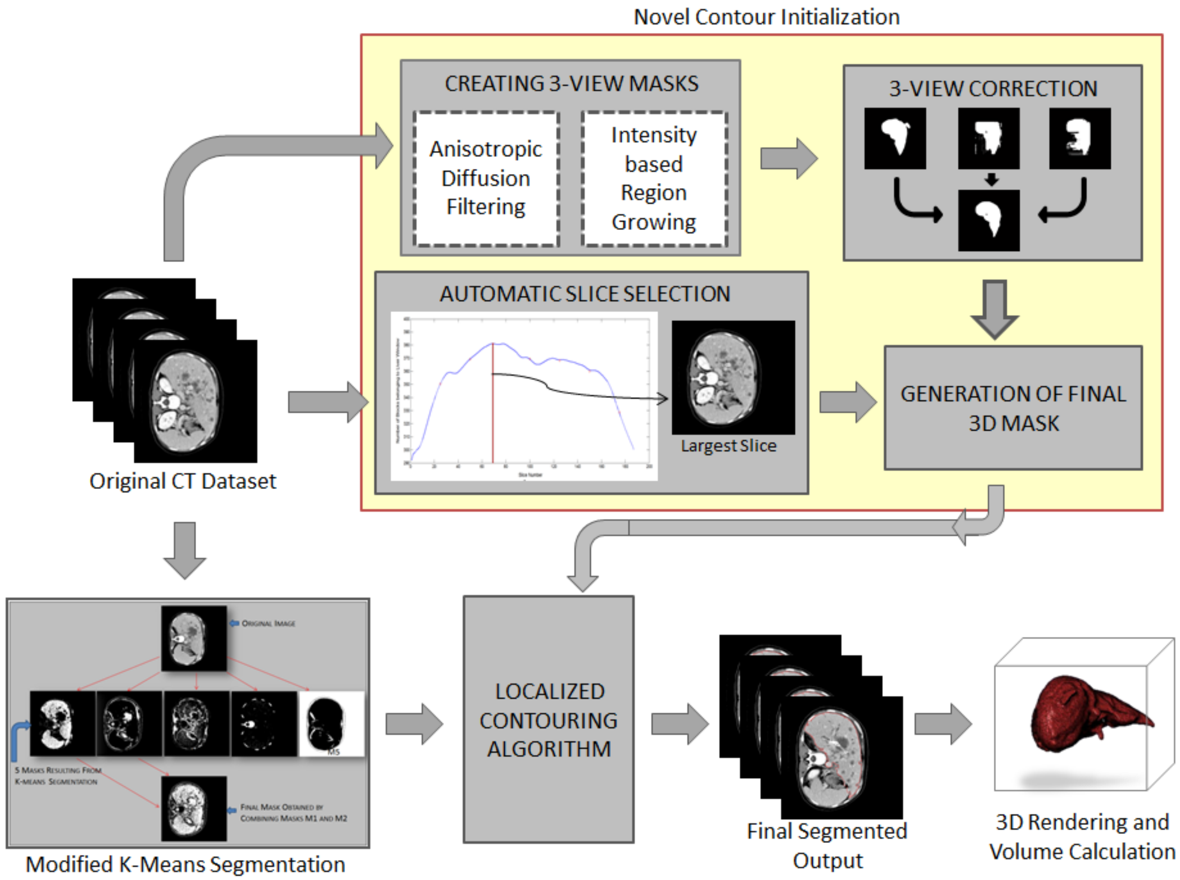

Novel Contour Initialization
CREATING 3-VIEW MASKS
Anisotropic Diffusion Filtering
Intensity based Region Growing
3-VIEW CORRECTION
AUTOMATIC SLICE SELECTION
Largest Slice
GENERATION OF FINAL 3D MASK
Original CT Dataset
Original Image
5 Masks Resulting From K-means Segmentation
M5
Final Mask Obtained by Combining Masks M1 and M2
Modified K-Means Segmentation
LOCALIZED CONTOURING ALGORITHM
Final Segmented Output
3D Rendering and Volume Calculation

Figure 3.19: Structure of the algorithm *Ver. 3.0* for 3D liver segmentation and rendering

3.4.1.1 Anisotropic Diffusion Filter

A Perona-Malik anisotropic diffusion filter is first applied to all the slices containing the liver in the dataset (Perona and Malik, 1990). The significance of the anisotropic diffusion filter is to reduce spurious and speckle noise effects seen in the liver CT images. For the diffusion filter, a 2D network structure of 8 neighboring nodes is considered for diffusion conduction. The considered neighbors are the north, south, east, and west, northeast, northwest, southeast and southwest directions. The conduction coefficient function used for the filter as applied on the liver slices aims to privilege wider regions over smaller ones based on the fact that the liver is the largest organ seen in the abdominal cavity. The conduction coefficient function is given by Eq. 10 where ∇I is calculated for the 8 directions and K is the gradient modulus threshold controlling the conduction.

$$\mathrm{C}(\mathrm{x},\mathrm{y},\mathrm{t}) = \frac{1}{1+\left(\frac{\|\nabla \mathrm{I}\|}{\mathrm{K}}\right)^2} \tag{10}$$

3.4.1.2. Intensity based Region Growing

Intensity based region growing process works by accepting a seed point and expanding the region of interest based on the intensity of the neighboring region. The region is iteratively grown by comparing all unallocated neighboring pixels to the region. The difference between a pixel's intensity value and the region's mean is used as a measure of similarity. The pixel with the smallest difference measured this way is allocated to the respective region. This process stops when the intensity difference between region mean and new pixel become larger than the selected threshold. The neighborhood for the algorithm is a region of 4 pixels around the point under consideration. Since the intensity based region growing algorithm works by accepting only those points in the neighborhood which fall in the intensity window, a fixed threshold will provide inaccurate results in case of liver scans where the intensity distribution differs from scan to scan. As a result, the threshold is devised in terms of the standard deviation of the image under consideration. Since the intensities considered for the algorithm are in Hounsfield units, a threshold of 5% of the standard deviation of the image is used.

The last and most important feature governing the operation of the intensity-based region growing algorithm is the selection of the seed points. Since the algorithm is designed to provide a solution with minimal user intervention, a linear interpolation method is used. The user is asked to select a single point belonging to the liver in the first and last slice in which the liver is seen. These two points act as the end points of a line which contain the seed points for the region growing algorithm. The intersection point of the line and the image slice is used as the seed point for the region growing algorithm for that slice.

The choice of linear interpolation was based on the fact that the structure of the liver does not change abruptly from slice to slice. In support of this last assertion, linear interpolation yielded the best results. Figure 3.20 shows the different views of the seed-line for a particular dataset, with arrows pointing to the position of the two selected seed points. The 3D rendered liver shown is the result after the entire segmentation process is completed.

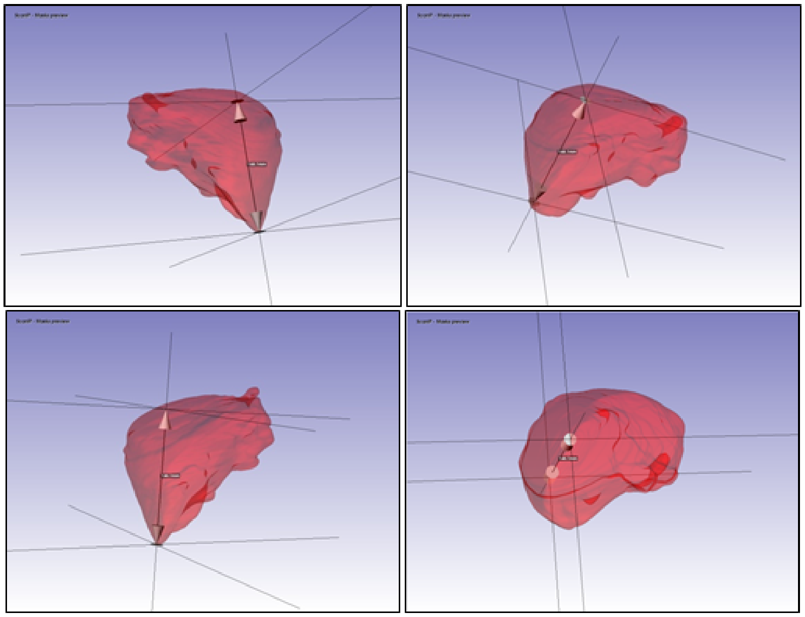


Figure 3.20: Different views showing the seed-line in a given dataset

3.4.1.3. Volume of Interest Based Correction or Three View Correction

Region growing algorithms based on intensity tend to incorporate some portions of neighboring regions like the spleen and the stomach. To overcome this challenge, a correction which utilizes information from all the different views namely coronal, transverse, and sagittal is taken into consideration. The correction algorithm is based on the idea that a voxel is considered to belong to the liver in the 3D space only if it is marked by the preceding steps of the algorithm in at least two of the three views.

The 3D masks generated (by assembling various 2D image masks) by the process are combined in a logical configuration as described by Eq. 11.

$$\mathrm{M}^{\mathrm{3D}}_{\mathrm{Correct}} = \mathrm{M}^{\mathrm{3D}}_{\mathrm{Cor}} \cdot \mathrm{M}^{\mathrm{3D}}_{\mathrm{Sag}} + \mathrm{M}^{\mathrm{3D}}_{\mathrm{Sag}} \cdot \mathrm{M}^{\mathrm{3D}}_{\mathrm{Tran}} + \mathrm{M}^{\mathrm{3D}}_{\mathrm{Tran}} \cdot \mathrm{M}^{\mathrm{3D}}_{\mathrm{Cor}} \qquad (11)$$

Where $\mathrm{M}^{\mathrm{3D}}_{\mathrm{Correct}}$ is the 3D corrected mask and $\mathrm{M}^{\mathrm{3D}}_{\mathrm{Cor}}$, $\mathrm{M}^{\mathrm{3D}}_{\mathrm{Sag}}$ and $\mathrm{M}^{\mathrm{3D}}_{\mathrm{Tran}}$are the 3D coronal, sagittal and transverse masks, respectively, as obtained by the process outlined in section 2.2.1.1 and 2.2.1.2. The '$\cdot$ ' and '+' operators are the pixel-by-pixel AND and OR operations, respectively. The results of this correction process are shown in Figure 3.21 where in part (e) the falsely detected voxels have been eliminated.

3.4.1.4. Single Slice Selection

The final step in the initialization process is the masking of the results obtained from the aforementioned region growing algorithm with the mask of the largest slice. In the context of the algorithm the largest slice is defined as the slice in which the liver is seen to be the largest. In order to extract automatically the largest slice, the Automated Large Slice Selection (ALSS) algorithm developed in house is utilized. The ALSS algorithm begins by first dividing each slice into blocks of 16x16 pixels. The 16x16 pixels size was empirically determined to be the best solution.

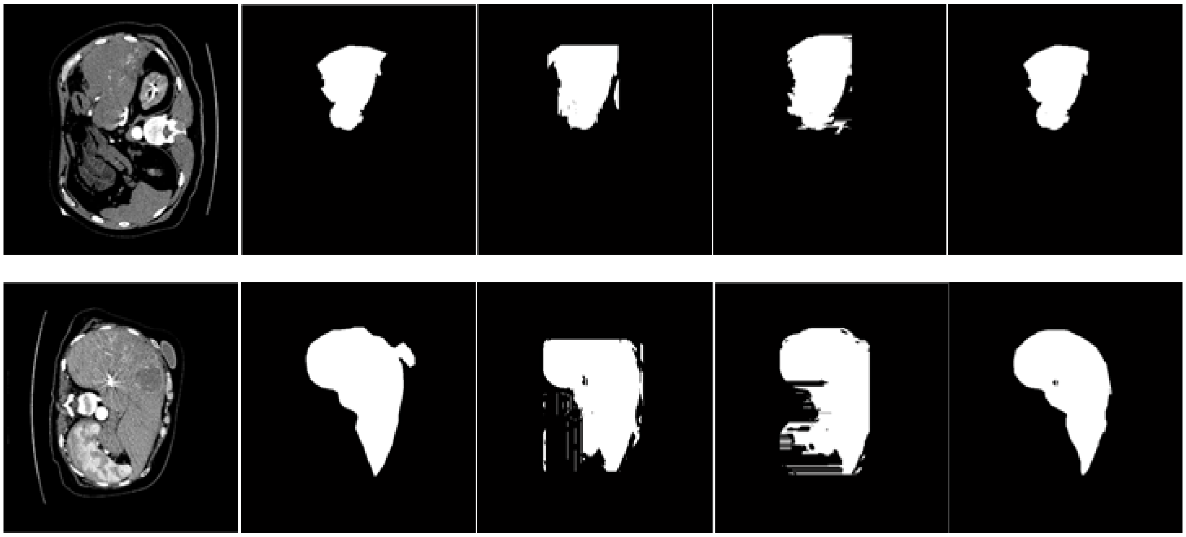


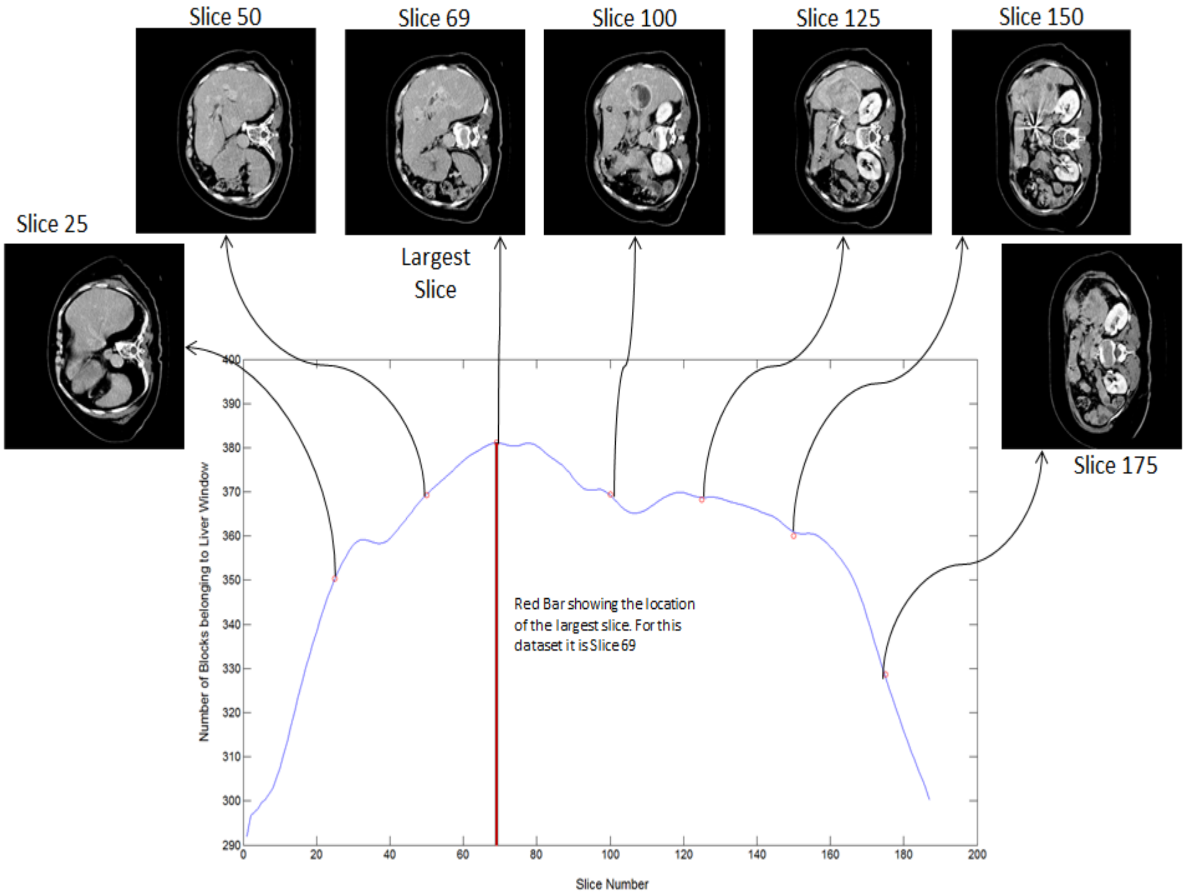
Figure 3.21: Volume-of-Interest based correction results (a) Original slice (b) Coronal Mask (c) Sagittal Mask (d) Transverse Mask (e) Corrected Mask

Inside each block, the pixels that fall in the abdominal CT window of -40HU to 180HU were counted (Ruskó et al., 2007). A block was marked as being part of the liver region if at least more than half of the pixels were found in the abdominal CT window as described by Eq. 12.

$$\text{block marked} = \begin{cases} 1 & \text{if } {}^{\mathrm{n}}\!/_{\mathrm{N}} \geq 0.5 \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

where n is the number of pixels in the abdominal window and N = 16x16 is the total number of pixel elements in the block.

Figure 3.22 shows a typical profile curve for dataset 1 which displays the number of blocks belonging to the liver window as a function of the slice number in a particular dataset and the slices used to calculate the value of the curve at some particular instant of time. Such a curve gives the estimate of the changes in the size as well as the potential extent of the liver region seen across slices, providing as a consequence the information needed to select those slices that most probably contain a large portion of the liver.



Red Bar showing the location of the largest slice. For this dataset it is Slice 69

Figure 3.22: Automatic Large Slice Selection algorithm. A typical curve is shown with corresponding slices. The red vertical line shows the location of the largest slice which is then is used by the algorithm.

The slice marked by the ALSS algorithm is presented to the user to demark the boundaries of the liver in that slice, which will make up the mask that delineates the liver. The mask is applied to all the slices of the dataset to exclude any portions that are not belonging to liver. This step is essential since the region growing algorithm sometimes over extracts the liver region by including some regions belonging to the duodenum, stomach and spleen. Once the initialization for all the slices of the dataset is obtained interactively, the localized-region-based active contouring algorithm is employed as described in section 3.2.3.

3.5. Three-Dimensional Image Rendering

In all the three versions of the algorithm, the 3D datasets are rendered using cost-effective third party software called ScanIP™ developed by Simpleware Ltd. based in the United Kingdom. The software renders the segmented dataset in 3D space and offers the possibility for the physician to view/edit/correct the rendered liver if necessary.

The software also calculates the volume of the liver by determining the number of voxels that are marked as being within the liver region by the segmentation algorithm. For the volume calculation, the software considers voxels marked by '1' as belonging to the liver and voxels marked by '0' as belonging to the surrounding. The developed algorithm differentiates between liver and background with no decision done by the software. The only inputs fed to the software are the segmented datasets and the original resolution of the CT datasets. The calculated volumes are in milliliters (ml).

CHAPTER IV

TUMOR AND BLOOD VESSEL SEGMENTATION

Segmentation of liver tumors is an important prerequisite task before any surgical intervention. A precise and accurate analysis of the lesions/tumors allows for accurate staging and evaluation of the available therapies that can be provided to the patient. It can help in deciding the best treatment approach as well as track the progress of the therapy over an interval of time. Also, tumor segmentation plays a vital role in the development of 3D surgical tools that can help and guide the surgeon for the complete removal of the tumor rendering the patient free of the underlying disease.

As emphasized in Chapter 1, fractional dose calculation of the liver and tumor depend on the volumes/masses of the liver and tumor. Hence in order to calculate the dose delivered to the tumor, it is essential to segment the tumor from the background and calculate the volume of the tumor region. Also, the ability to display the blood vessels along with the liver and the tumor/tumors provides an idea of the extent of the vasculature feeding the tumor as well as a map for the insertion point of the radio-isotope in cancer directed technologies like SIRT.

This chapter introduces some of the advances in the field of tumor and blood vessel segmentation followed by the description of in-house developed methods for the segmentation of the tumor and blood vessels from CT scans.

4.1. Review of automatic/semi-automatic tumor segmentation techniques

In order to improve the treatment and prognosis of CRLM, accurate and early detection of the tumor is a very important phase. Also, staging plays an important role towards the determination of treatment options and the assessing the progression/recession of the disease. Since CT is one the most commonly used imaging modalities in the prognosis and staging of liver tumors, tumor segmentation and volume calculations are essential. Various automatic/ semiautomatic techniques for liver tumor segmentation have been developed based on strategies which include entropy based segmentations, Bayesian approaches, multi-level thresholds, level set techniques and region growing techniques.

Choudhary et al. demonstrated a semi-automatic technique using an initial Watershed algorithm followed by minimum entropy-based region growing technique and level set smoothing (Choudhary et al., 2008). The technique was tested on 10 datasets rendering a mean volumetric error of 22.58%. Furthermore, Xu et al. demonstrated the use of a local entropy-based technique coupled with morphological filters towards liver tumor segmentation (Xu et al., 2010).

Yoav et al. demonstrated the development of a nearly automatic technique for the segmentation of liver tumors using a multi-resolution, multiclass Bayesian classification technique followed by morphological adjustment (Yoav et al., 2008). The method tested on 20 tumors in 9 patients showed promising results towards the determination of the tumor volume for clinical applications. Another semi-automatic Bayesian based approach for automatic segmentation of liver tumors was reported by Qi et al. (Qi et al., 2008). The technique segmented the tumor region by modeling the probability density functions of the tumor as a series of Gaussians followed by 3-D seeded region-growing algorithm. The seeds for the region growing algorithm are selected manually and updated during the region growing process.

Adaptive thresholding and multiple thresholding techniques have been widely used for the segmentation of liver tumors from CT images. Moltz et al. reported a technique for delineating the tumor boundaries using adaptive gray level thresholding combined with model-based morphological processing (Moltz et al., 2008). The technique was carried out on the segmentation of tumors from 10 CT datasets. Another study demonstrated the use of Isodata threshold towards the segmentation of the tumors from CT images (Abdel-massieh et al., 2010). The technique uses Gaussian smoothing flowed by a multilevel Isodata threshold and image binarization for the detection of the tumors.

Level-set techniques have been at the forefront of tumor segmentation techniques with a number of studies published. Smeets et al. demonstrated the development of a level-set algorithm based on statistical pixel classification with supervised learning for the segmentation of lesions (Smeets et al., 2008). The algorithm tested on 10 datasets provided an average volumetric difference of about 17.8%. Jimenez-Carretero et al. demonstrated the development of a multi-resolution 3D level set technique coupled with adaptive curvature technique for the classification of the pixels into tumor and background (Jimenez-Carretero et al., 2011). The technique demonstrated promising results for detection of elongated tumors avoiding internal leakages to close structures.

Region growing techniques are the most widely used technique for the segmentation of tumors in liver CT scans due to their high accuracy, excellent differentiation capabilities, and ease of use. Various studies have been demonstrated using variation of region-growing algorithms with either 2-D or 3-D region growing and with different energy functions. Wong et al. demonstrated a 2-D region growing technique based on knowledge-based constraints for the segmentation of the tumor in each slice (Wong et al., 2008). The technique demonstrated an average volume difference of 24.2% in the segmentation of the tumors. Ben-Dan and Shenhav demonstrated the development of active contour- based tumor segmentation using a weighted function of the probability of each pixel (Ben-Dan and Shenhav, 2008).

Various other techniques based on AdaBoost (Jingran et al., 2009; Yuanzhong et al., 2006), Support Vector Analysis (Xing et al., 2011) and CAD approaches (Ben et al., 2010) have been also developed for the segmentation of the tumors. Enhancing the contrast of the images before the application of other segmentation approaches has been suggested towards better segmentation of tumors in liver.

Even with the advent of multiple techniques for automatic tumor segmentation manual tumor segmentation form CT images by an expert has been regarded as the gold standard for all clinical purposes. This is due to the variability often seen in the different techniques with no single method being able to perform well in all situations. Nonetheless, studies have also shown that about a 17% variability is found in delineating the tumor manually by different experts (Moltz et al., 2011).

4.2. Tumor Segmentation Method

The tumor segmentation method developed for the study is a multi-stage segmentation technique comprised mainly of 5 stages.

1. **Liver Extraction**: first step in the segmentation is to extract the liver region.
2. **Image De-noising**: performed using a median filter to improve the contrast of the tumors in the liver and to reduce specular noise.
3. **Intensity based Region Growing**: a semi-automatic step requiring a single point per tumor as a seed. The region growing aims at encompassing regions of similar intensity to segment the tumor.
4. **Localized Contouring**: coarse tumor segmentation obtained in step 3 is improved upon using a localized contouring algorithm to improve the detection of the tumor region.
5. **Rendering and Volume Calculations**: tumor/tumors are rendered in 3-D space to display location, size and extent of the tumor within the liver. Volume calculations are carried out based on the number of voxels comprised within the tumor.

4.2.1. Liver Extraction

The first step of the process is to extract the 3-D liver from the CT dataset to extract all the tumors belonging to the liver region. For accomplishing this task, the 3-D liver segmentation approach detailed in Chapter 3 is utilized. The 3-D segmented liver volume region is applied as a pixel-by-pixel binary mask on the 3-D CT scan obtained in the venous phase. The utilization of the venous phase is based on literature review, which indicates that tumors are visualized best in the venous phase as compared to the arterial or equilibrium phase (Xu et al., 2011). The resultant output of the multiplication yields a

CT dataset containing only the liver region with the other parts of the CT scan substituted by zeros. Thus,

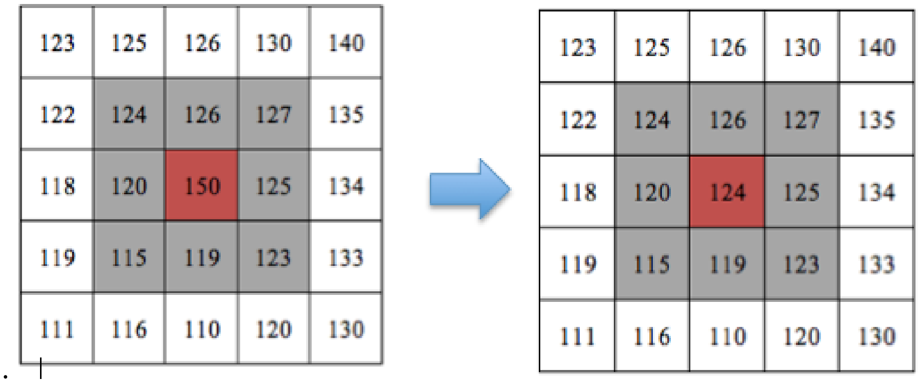
$$CT_{liver} = 3D_{liver} \,.\, CT_{venous} \tag{13}$$

$CT_{liver}, 3D_{liver}, and\ CT_{venous}$ are the output CT containing only the liver, the 3-D liver binary mask obtained through liver segmentation and the original venous phase CT input dataset.

4.2.2. Image De-noising

A median filter is first applied to all the slices of the CT scan containing the liver in the dataset. The significance of the median filter is to reduce spurious and speckle noise effects seen in the liver CT images and enhance the contrast of the tumor region (Arce and McLoughlin, 1987).

The filtering process is carried out in a slice-by-slice fashion for each slice of the dataset containing the liver. For the filter, a 2D network structure of 8 neighboring nodes is again considered for filtering. The median filter considers each pixel in the image in turn and examines the surrounding 8 pixels. The filter then proceeds to replace the value under consideration by the median value of the neighborhood pixels. The median value is calculated by first sorting the neighborhood pixel values and then obtaining their middle value. In case of even neighborhood pixels, the average of the two middle values is used as the median value to replace the pixel being considered. For calculating the filter neighborhood values for the next pixel value the updated values are considered. This can be displayed pictographically as shown in Figure 4.1.



123	125	126	130	140
122	124	126	127	135
118	120	150	125	134
119	115	119	123	133
111	116	110	120	130

123	125	126	130	140
122	124	126	127	135
118	120	124	125	134
119	115	119	123	133
111	116	110	120	130

Figure 4.1: Working of the median filter showing the value 150 (marked in red) being replaced by 124 which is the median of the 8 neighborhood values

4.2.3. Intensity-based Region-Growing

An intensity-based region-growing step similar to the one used in the *Ver. 3.0* algorithm discussed in section 3.4 for liver segmentation was utilized in the tumor segmentation step. Two changes were made to accommodate the targeted region for segmentation, tumor instead of liver.

1. **Change in seed-point method**:

   As contrasted to the liver segmentation approach, the tumor segmentation method uses a single seed point per tumor as an initialization for the intensity based region growing. This seed point is derived interactively from the user. The user uses the mouse pointer to mark a single point in the 3-D tumor in any one slice of the dataset in which the tumor can be seen. Using this seed point as the initial condition the region-growing algorithm evolves to include all the surrounding points in the 3-D space that fall inside the threshold of the region-growing algorithm.

2. **Change of Threshold for Region Growing Method**

   Recall that the threshold for the intensity-based region-growing step for the liver segmentation algorithm was set at 5% of the standard deviation of the image due to the distribution of the Hounsfield intensities in the entire abdomen that vary between -1000 HU and 2000 HU. Since a single seed point is selected for tumor segmentation, it was not possible to determine a threshold in terms of statistical measures of the intensity of the chosen seed point. Experimental evaluations reveal that a threshold of 10% of the intensity was sufficient to include all the points belonging to the tumor with little excess. Lower thresholds resulted in regions of tumor being missed, which is to be avoided.

4.2.4. Localized Contouring Algorithm

After the coarse segmentation of the tumor by the region-growing step, we aim to refine the segmentation results using the localized contouring algorithm that has been widely used in the developmental process of this dissertation. Segmentation obtained using region growing acts as an initial contour for the localized contouring step. The localized contouring step is applied in a slice-by-slice manner to each slice that contained any region of a tumor.

4.2.5. Rendering and Volume Calculations

Once the segmentation process is completed, the tumor/tumors are rendered in 3-D space to provide context in terms of location, size and extent of the tumor within the liver. The segmented 3D tumors are again rendered using ScanIP™. Figure 4.2 shows the output for the tumor segmentation algorithm at each step of the process.

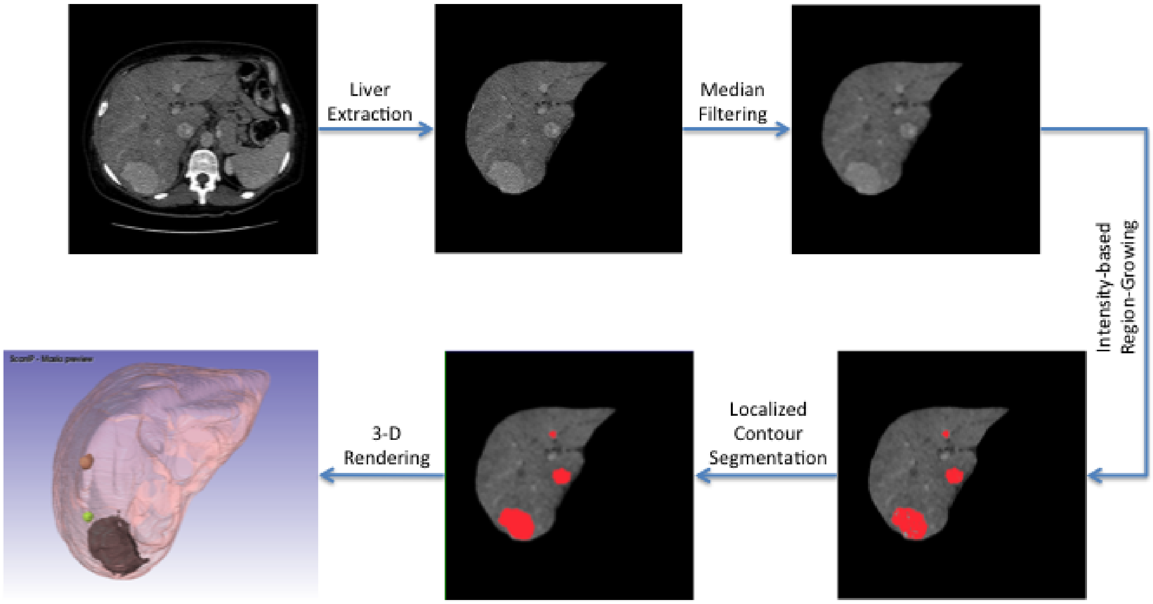

Liver Extraction
Median Filtering
Intensity-based Region-Growing
Localized Contour Segmentation
3-D Rendering

Figure 4.2: Outputs after every step of the tumor segmentation algorithm. The red overlay shows the tumor masks generated after the intensity based region growing and the localized contouring

4.3. Review of hepatic automatic/semi-automatic blood vessel segmentation techniques

Analysis of vasculature information from 3-D volumetric data has been investigated with great interest due to its importance towards a variety of medical applications. Accurate determination of the morphology and structure of blood vasculature enables an efficient, accurate and systematic diagnosis, surgical planning and resections in liver related diseases(Sonka et al., 2001). Assessment of liver vasculature is essential in cases related to living-related liver transplants to study the branching patterns of all hepatic vessel system and for oncology applications for studying and understanding the underlying vasculature before surgical intervention.

In case of SIRT, the ability to display the blood vessels along with the liver and the tumor/tumors in a 3-D space offers an idea of the extent of the vasculature feeding the tumor as well as provides a map for the insertion point of the radio-isotope(Gulec and Fong, 2007).

Various techniques for vessel or vessel-like tubular structures have been reported. The various techniques developed for vessel segmentation can be broadly classified into techniques based on pattern recognition approaches, model based techniques, tracking based techniques, neural network based techniques, and other algorithms for segmenting tubular structures(Kirbas and Quek, 2003).

Pattern recognition techniques are widely used for the extraction of blood vessels in medical image analysis. Multi-scale pattern recognition analysis studies have provided excellent results with fast computational times towards the extraction of vasculature from cardiac CT and MR images (Sarwal and Dhawan, 1997). Skeleton based approaches to extract and locate the centerline of the vessel tree have been used to extract blood vessel in CT. These techniques use one or more types of skeletonization approaches followed by thresholding to determine the vasculature structure (Kawata et al., 1996; Tozaki et al., 1994). Various ridge based approaches that treat the gray scale medical images as 3-D elevated maps have been established(Aylward et al., 1996). Region growing approaches that attempt to extract the 3-D vasculature from a single seed point are the most commonly used techniques for vasculature extraction. These approaches are based on intensity similarity and spatial proximity for the extraction of the blood vessels (Higgins et al., 1989; Selle et al., 2002). Although widely used they suffer the drawback of needing a seed-point for their initialization.

Parametric Deformable Model (PDM) based approaches that intend to find the vasculature by evolving snakes based on some internal and external forces have found increasing use vessel segmentation(Hui et al., 2000; Toledo et al., 2000). Level-set approaches for segmenting of blood vessel have been widely used for cardiac angiographs (Malladi et al., 1996; Malladi and Sethian, 1996). Level-set methods provide many advantages for vessel segmentation however have not been utilized very greatly towards hepatic vasculature extraction from CT(Selle et al., 2002).

Vessel tracking based approaches to determine the centerline of the vessels by analyzing the pixels in the orthogonal directions to the seed have been developed for vasculature segmentation in cardiac and retinal applications (Lecornu et al., 1992; Tolias and Panas, 1997). The use of these methods however is limited in case or hepatic vessel segmentation studies.

Neural network based approaches for blood vessel segmentation have been used for classification of pixels as vessel or non-vessel in CT Angiograms (CTA)(Nekovei and Ying, 1991). These methods do not aim to produce 3-D maps but only rely on classification of the pixels in to the different categories(Nekovei and Ying, 1995). Since CTA datasets are usually large it is very time consuming to train the neural network and hence these algorithms are used only on selected section of the CTA. Neural network based approaches have found little use in extraction of blood vasculature from hepatic CT.

4.4. Hepatic Blood Vessel Segmentation Method

The segmentation of the hepatic blood vessels from CT data is a 3-step process. The various steps of the process are outlined below:

1. **Liver Extraction**: The first step in the segmentation is to extract the liver region.
2. **Image De-noising**: Following liver extraction image de-noising is performed using an anisotropic diffusion filter aimed at enhancing small objects such as the blood vessels so that they can be extracted from the images
3. **Intensity based Region Growing**: The only semi-automatic step of the entire process is the region-growing algorithm, which needs a single point near the entrance of the aorta into the liver. The region growing aims at encompassing regions of similar intensity and spatial proximity to extract only the blood vessel.

4.4.1. Liver Extraction for blood vessel segmentation

For accomplishing this task, the 3-D liver segmentation approach detailed in Chapter 3 is utilized. The 3-D segmented liver volume region is applied as a pixel-by-pixel binary mask on the 3-D CT scan obtained in the arterial phase. The utilization of the arterial phase is based on the fact that the blood vessels are most clearly seen as bright objects in the arterial phase CT scan. The resultant output of the multiplication yields a CT dataset containing only the liver region with the other parts of the CT scan substituted by zeros. Thus,

$$CT_{liver} = 3D_{liver} \, . CT_{arterial} \tag{14}$$

$CT_{liver}, 3D_{liver}, and\ CT_{arterial}$ are the output CT containing only the liver, the 3-D liver binary mask obtained by liver segmentation, and the original arterial phase CT input dataset, respectively.

4.4.2. Vascular Edge Enhancement

For image processing and computer vision applications, anisotropic diffusion is a technique used to reduce the image noise without removing significant parts of the image content, typically edges, lines or other details that are important for the interpretation of the image(Perona and Malik, 1990).

The significance of the anisotropic diffusion filter in the particular application is to reduce spurious and speckle noise effects seen in the images and to enhance the edge information for extracting the vasculature. For the diffusion filter a 2D network structure of 8 neighboring nodes is considered for diffusion conduction. The considered neighbors are the north, south, east, west, northeast, northwest, southeast and southwest. The conduction coefficient function used for the filter applied on the liver slices aims to privilege edges over wider regions in order to enhance regions of high thermal activity associated with the vasculature. Thus, the conduction coefficient function used for the application as shown in Eq. 15 is given by

$$C(x, y, t) = e^{-\left(\frac{\| \nabla I \|}{K}\right)^2} \tag{15}$$

where ∇I is calculated for the 8 directions and K is the gradient modulus threshold that controls the conduction. The choice of the gradient modulus threshold $K = 5$ was empirical in nature with studies aimed to reduce spurious noise while at the same time maintaining the overall information in the image.

4.2.3. Intensity-based Region-Growing

A modified version of the intensity-based region-growing step similar to the one used in the tumor segmentation process discussed in section 4.2.3 was utilized for the extraction of the blood vessel network of the liver. The region growing method utilizes two key features seen for blood vessels in the CT scans, namely (1) the blood vessels in axial slices appear as extremely bright white spots with high HU in the CT slices; and (2) The blood vessels form a continuous network. These properties enable the extraction the blood vessels more accurately than using regular thresholding, since only connected points belonging to the blood vessels are segmented out.

The first important parameter for the region-growing algorithm is the selection of the seed point, which is the initial point from where the region-growing algorithm expands to include the rest of the vasculature. The seed point for the method is user selected using a mouse pointer by selecting a pixel belonging to the vasculature near the entrance of the aorta in to the liver. The second parameter of importance for the region-growing algorithm is the threshold for accommodating in the evolving vasculature. A threshold 10% of the intensity of the selected seed point provided good results in the study and has also been recommended in other studies using region-growing algorithms(Selle et al., 2002). Once the segmentation is completed, the blood vessels are displayed in the 3D space using third party software called ScanIP. The software enables the rendering, volume calculation and editing if necessary. Figure 4.3 shows the output for the blood vessel segmentation algorithm after every step of the processing.

Since the first step towards the segmentation of the tumor and blood vessels is to extract the liver, the results of the tumor and blood vessel segmentation algorithm rely on the accuracy of the liver segmentation method. Keeping this in mind, we have presented detailed description of the results for the tumor and blood vessel segmentation in section 6.3 after discussing the liver segmentation results. But before these results are detailed, Chapter 5 which follows addresses the heavy computational burden associated with 3-D segmentation in large volume medical imaging.

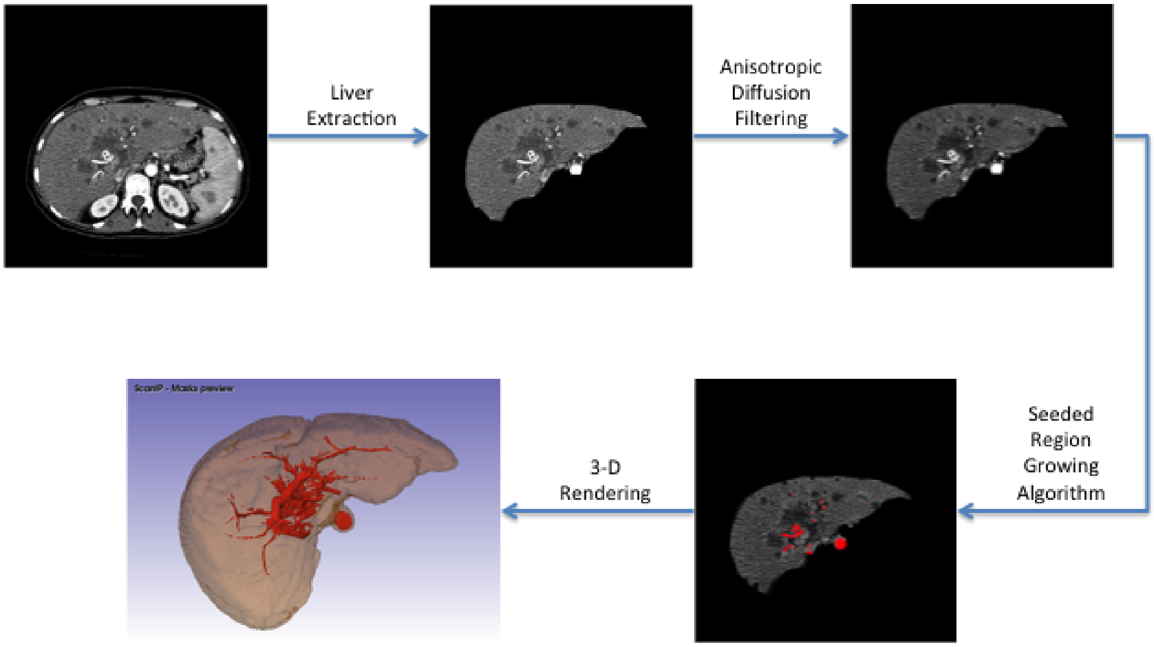

Liver Extraction
Anisotropic Diffusion Filtering
Seeded Region Growing Algorithm
3-D Rendering
ScanIP - Masks preview

Figure 4.3: Outputs after every step of the blood vessel segmentation algorithm. The red overlay shows the masks generated after the intensity based region growing algorithm.

CHAPTER V

PARALLEL PROCESSING FOR 3D SEGMENTATION AND VOLUME RENDERING

As in most 3D applications that include large datasets, as is the case of this study, there is an inordinately lengthy processing time requirement. The drawback is the consequence of the slice-by-slice segmentation approach that is required of such large datasets. The fact that each dataset is comprised of 200 to 400 slices demands a computationally-taxing process. The objective is thus to lower the processing time to offer a faster and effective solution for doctors and hospitals that rely on the processing of such 3D imaging tasks.

Parallel computing has been widely used for reducing the computational burden imposed by image segmentation tasks, particularly medical image segmentation (Maeda et al., 2011). Parallel processing makes it possible for multiple slices of the dataset to be processed simultaneously. For a cost-effective solution, the constraint imposed here is to be able to perform parallel computing within a single desktop computer. Parallel computing in this case is employed mainly to reduce the processing time required by the region growing algorithm and the localized region contouring algorithm, since they the most computationally demanding.

This chapter describes the performance results obtained on testing the MatLab parallel processing toolbox to execute a three-dimensional (3-D) liver reconstruction. The FFT algorithm was tested using the parallel processing toolbox, by changing system platform, number of *workers*, image size and number of images. A second set of experiments was conducted keeping the hardware fixed while changing the operating system to obtain unbiased results. The third set of experiments was performed to assess the effect of parallelization applied to a newly developed 3-D liver reconstruction algorithm.

5.1. Review of Parallel Processing Techniques

High Performance Computing (HPC) as a discipline is continually introducing new and enhanced tools to provide each time faster solutions. Moreover, the emphasis on high productivity metrics results in a reduction of the time-to-solution process (Bliss and Kepner, 2007; Krishnamurthy et al., 2007). High level languages are under higher scrutiny due to the potential high productivity gains, and high yield for those tasks that are computationally taxing. The MatLab scripting language combines the ease of a scientific programming language with an extremely intuitive environment for application development and visualization. Traditionally, MatLab was used as a prototyping algorithm environment and the final algorithm deployment was done using a traditional language such as C or PASCAL. However with enhancements introduced in the MatLab compiler and domain specific tuned libraries, the use of MatLab has become not only ubiquitous but also more appealing to address those problems that demand heavy computational requirements (Trefethen et al., 1996). There is also the convenience provided by the availability of many specialized toolboxes. However, these advantages are often downplayed by the perceived limitations imposed by the relatively low processing speed and inadequate memory space. Such limitations are accentuated when dealing with massive datasets such as in medical imaging (Bister et al., 2007).

The domain-specific library routines are precompiled and optimized to form specialized variants, which are substituted into the user script during user-script compilation. This approach results in a significant increment on scripts execution. In other words, script-based applications provide a high performance as if the user's application were available at library compile time (Choy et al., 2004). Thus, MatLab provides many alternatives to solve problems that involve intensive computational operations; these alternatives are based on parallelism and HPC techniques. Krishnamurthy et al. presented an exhaustive description of all the HPC techniques offered by MatLab (Krishnamurthy et al., 2007). The most popular techniques are MatLab-MPI(Choy and Edelman, 2005; Hudak et al., 2009), MatLab (Bliss and Kepner, 2007), Star-P(Edelman, 2007; Mirman, 2006), the MatLab Distributed Computing Toolbox (DCT) (Mathworks, 2009a) and the Parallel Computing Toolbox (PCT) (Smith, 2009).

In this chapter the focus is placed on using the MatLab parallel and distributed processing (PCT and DCT) toolboxes as they provide a standard and simple mechanism to avoid the need of writing the code required dealing with inter-processor communication minutiae. The motivation is to determine the impact of the operating system on the performance of parallelization of a given task by using PCT and DCT of MatLab. These tools provide users with access to high performance computing resources in terms of maximizing the multicore capabilities of local desktops, with the potential to extend such processes to clusters and grids (Mathworks, 2009c).

The infrastructure used for our experiments required the configuration of a job manager and client host machines. The job manager schedules and coordinates the execution of jobs and tasks evaluation. The client host machine can request the job manager a specific job and its related tasks, in which case a MatLab session, referred to as a client session is initiated. Moreover, a MatLab host machine through the MatLab DCT toolbox can be requested to execute a given task, in which case each MatLab session called a *worker*, would behave like a computing server. Since a given host machine can initiate several MatLab sessions or *workers*, the actual number of *workers* that can be utilized by a given host machine depends on the resources available to it (Chakravarti et al., 2008). The MatLab parallel computing toolbox is used for requesting distribution of the process among several MatLab sessions. The parallel computing toolbox provides up to 8 local *workers* on a single workstation. However, the numbers of usable *workers* that can be started for an application are dependent on the number of cores/processors present on the computer. For example, a dual core processor provides up to 4 *workers*, whereas a quad core computer can provide support for up to 8 *workers*, although the maximum number of *workers* does also depend on the version of MatLab used. The 'matlabpool' command in MatLab can be used to setup the number of desired *workers*. Moreover, a given MatLab coded algorithm executed under different operating systems and hardware configurations can thus be explored (Mathworks, 2009c) and evaluated on the basis of its computational merit.

In order to define a job manager and a worker session it is necessary to have the MatLab Distributed Computing Engine (MDCE) installed in the host computer (Mathworks, 2009b). This provides a flexible configuration since both client and server can be deployed on a single host computer. To further explore the benefits of parallelism, we assessed an algorithm that semi-automatically segments the liver region from 3-D CT scans. The computational burden imposed by such a task is what originally led to this investigation. The algorithm can calculate the volume of the extracted liver region and also render the segmented liver in 3-D view.

This chapter discusses the processing time required to complete such tasks and the gains that were obtained by utilizing MatLab-based parallel computing.

5.2. Aims for development of the parallel computing infrastructure

The current literature lacks the necessary detail for assessing the heavy computational requirements of medical applications that require multidimensional analysis. Furthermore, there is a need for comparative studies for evaluating system performance as a function of the number of *workers* available in the MatLab environment under different configurations. Another contribution is also to facilitate an environment to design and create a test bed to determine the metrics for effective parallelism as a function of the operating system and hardware configuration used.

5.3. Setup of the parallel computing platform

To evaluate the merits of parallelism under different configurations, three sets of experiments are conducted, with the intent to:

1. Evaluate benchmarking operations to measure performance improvements.

2. Perform platform comparisons between three different operating systems: Windows Vista, Linux Kubuntu version 9.04, and Mac OS version 10.5.

3. Assess the computational gains that were accomplished while performing 3-D liver segmentation followed by 3-D liver reconstruction.

First set of experiments: the built-in (2-D) FFT was used with MatLab R2009 for computing the fast Fourier transform of a given image. The MatLab script developed for this investigation contrasts the computation of the FFT in a single process and in parallel. The time required for the computation in both processes is assessed. The expected improvement in the computational time is referred here as the 'speed up' of the process, and is computed using Eq. 16 given below:

$$T_c = C_1/C_n \tag{16}$$

C_1 and C_n denote the computational time required to complete the given image processing task using a single worker and *n workers*, respectively.

Second set of experiments: Further more benchmarking experiments were conducted to show the dependence between the speed up factor and a series of variables, which include the number of operational *workers*, the number of images being processed, the size of the images, and the operating system used. The objective is to identify optimal parameter settings which can later be applied to optimize the time-to-solution process to problems involving multidimensional processing.

Third set of experiments: These experiments involved performing liver segmentation from CT images. Beyond the heavy computational requirement it demands, liver segmentation based on CT images is in itself considered a challenging task due to the presence of similar intensity objects in the abdomen with no clear delineation or boundary between them and the liver.

Two hardware platform configurations under system 1 and system 2 were used to perform benchmark experiments using FFT transform.

1. The configuration of system 1 made use of an HP DC6700 built around an Intel Core 2 Quad Processor operating at 2.67 GHz with 8 GB RAM. This system was used for comparing the performance of the parallelization process among the Windows and the Linux platforms.
2. The configuration of system 2 made use instead of a MacPro computer with two Intel Quad Core Xeon Processors operating at 2.26 GHz with 8.00 RAM. System 2 was necessary to compare the performance between the Windows Vista and the Mac OS-X platforms.

For these configurations, it is important to note that on one hand, by using different hardware configurations with the same operating systems, namely Windows Vista, we can observe the impact of the hardware on the performance of the parallelization process; and on the other hand by using different operating system on the same platform we can assess the effect of the operating system on the performance of the given application. The steps followed for the setup of the experiments were: (1) Define a job manager, through the MDCE service; (2) Create a job and respective tasks as a request of the client host via MatLab script; (3) Submit job to job queue for execution; (4) Assess job results; and (5) Record timing

The three aforementioned sets of experiments were conducted using the following steps:

1. Benchmarking using the FFT on images with resolution of 1024x1024 pixels by using 1, 2, 4 and 8 *workers*. Each process was replicated 3 times and the

average computational time was recorded. These experiments were carried out on both systems.

2. Platform performance comparisons between three different operating systems, which are Windows Vista, Linux Kubuntu version 9.04, and Mac OS version 10.5. This experiment was carried out in 3 stages: (i) Perform the FFT on an image changing the number of *workers* under a fixed operating system; (ii) Perform the FFT changing both the number of *workers* and the work load by increasing the number of images from 10 to 10,000; and (iii) Perform the FFT changing both the number of *workers* and the work load by increasing the image resolution from 128x128 to 1024x1024, with number of images fixed to 1000.
3. Parallel liver segmentation process is performed on CT images of size 512x512 pixels while assuming different variables. Each dataset is processed using an in-house developed algorithm for segmenting the liver from abdominal CT images. The steps of the segmentation process for both sequential and parallel approaches are provided in Figure 5.1 and Figure 5.2 for *Ver1.0/Ver. 2.0* and *Ver. 3.0*, respectively

Figure 5.1 which shows the parallel processing approach for *Ver. 1.0* and *Ver. 2.0* shows that the K-means segmentation and the localized contouring steps are parallelized. Figure 5.2 shows the parallel processing approach for *Ver. 3.0* which shows that in addition to the K-means and localized contouring step, the contour initialization process is also parallelized. Since *Ver. 1.0* and *Ver. 2.0* of the algorithm did not have an extensive contour initialization it was not incorporated in the parallelization.

Note that the number of slices processed at any given time depends on the number of *workers* selected for the parallel processing task. In this experiment all the datasets are processed using 1, 2, 4 or 8 *workers* to estimate each time the computational gains achieved using parallel computing.

The results were examined using analysis of variance (ANOVA) to identify the significance of each of the variables considered, including operating system, number of *workers*, number of images, and image size.

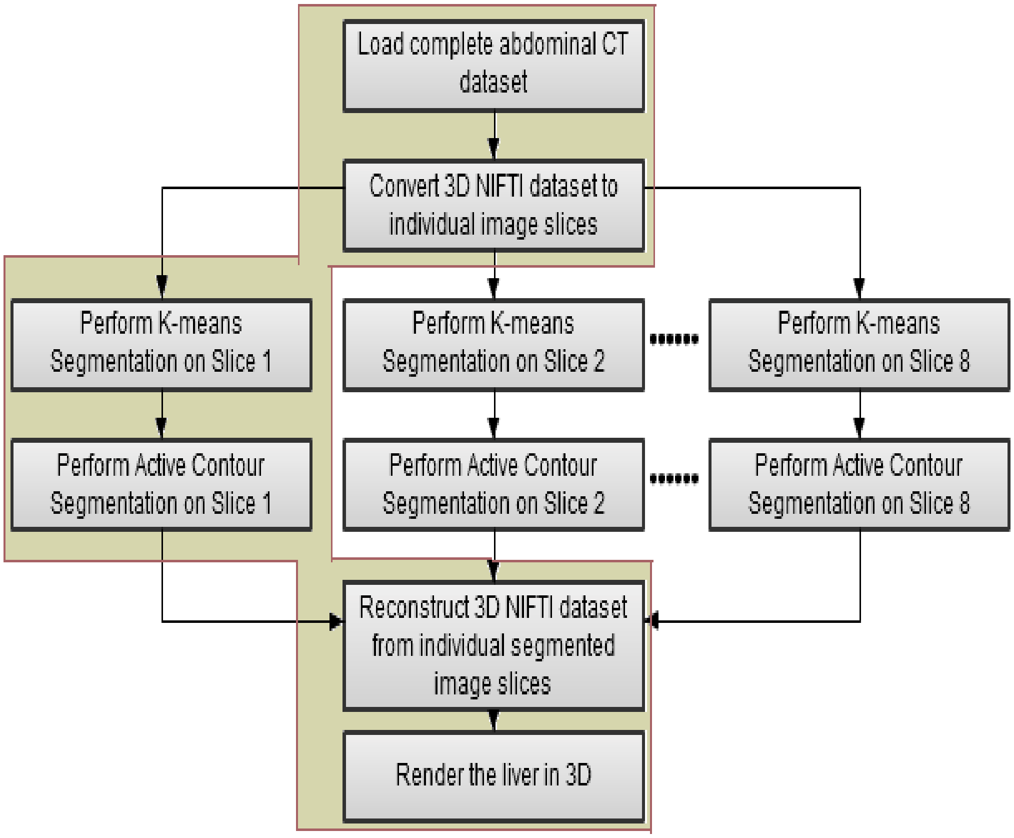
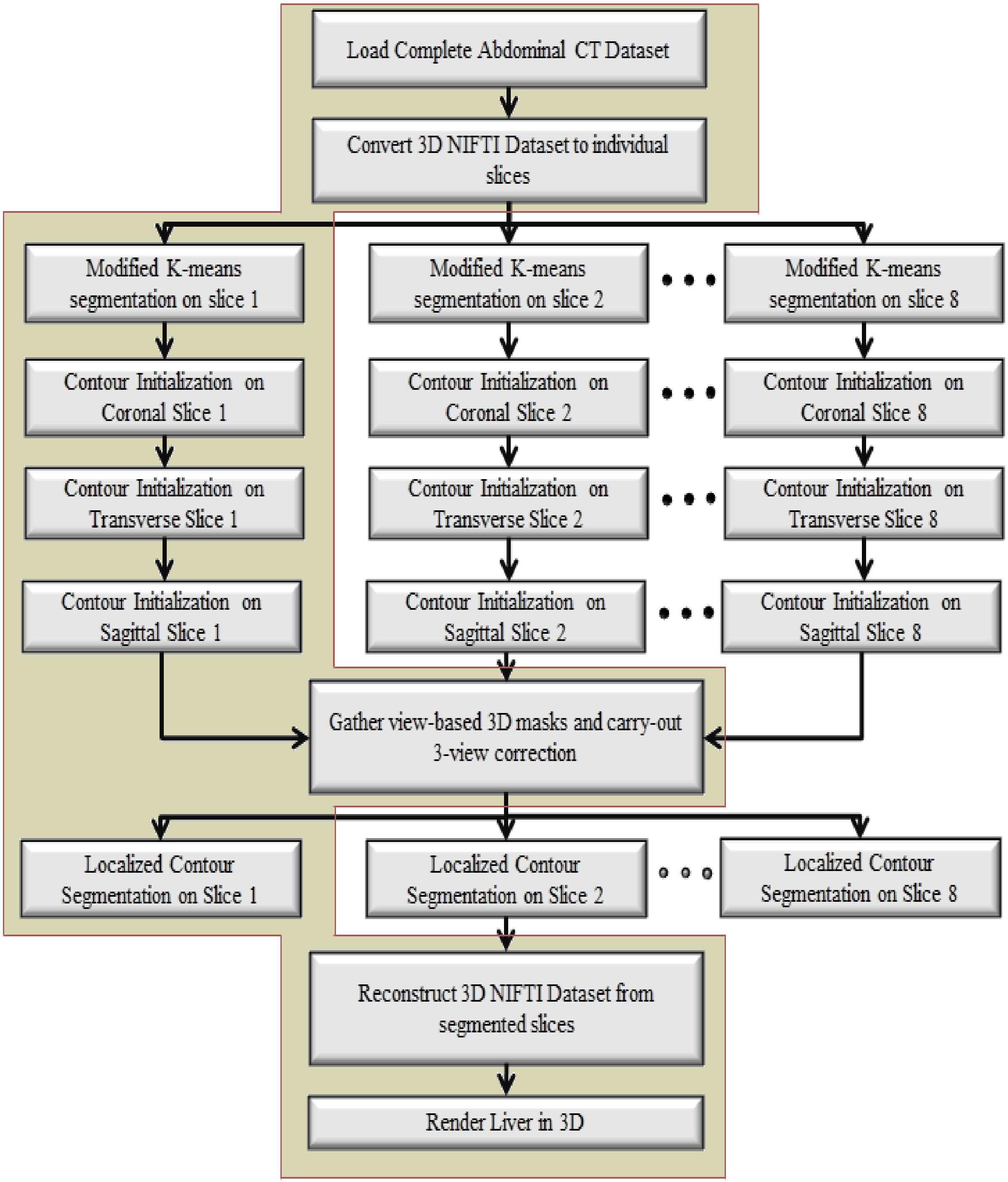

Load complete abdominal CT dataset
Convert 3D NIFTI dataset to individual image slices
Perform K-means Segmentation on Slice 1
Perform K-means Segmentation on Slice 2
Perform K-means Segmentation on Slice 8
Perform Active Contour Segmentation on Slice 1
Perform Active Contour Segmentation on Slice 2
Perform Active Contour Segmentation on Slice 8
Reconstruct 3D NIFTI dataset from individual segmented image slices
Render the liver in 3D

Figure 5.1: Parallel processing approach for the *Ver1.0/Ver. 2.0* algorithm. The serial approach is outlined in the yellow box.



Load Complete Abdominal CT Dataset
Convert 3D NIFTI Dataset to individual slices
Modified K-means segmentation on slice 1
Modified K-means segmentation on slice 2
Modified K-means segmentation on slice 8
Contour Initialization on Coronal Slice 1
Contour Initialization on Coronal Slice 2
Contour Initialization on Coronal Slice 8
Contour Initialization on Transverse Slice 1
Contour Initialization on Transverse Slice 2
Contour Initialization on Transverse Slice 8
Contour Initialization on Sagittal Slice 1
Contour Initialization on Sagittal Slice 2
Contour Initialization on Sagittal Slice 8
Gather view-based 3D masks and carry-out 3-view correction
Localized Contour Segmentation on Slice 1
Localized Contour Segmentation on Slice 2
Localized Contour Segmentation on Slice 8
Reconstruct 3D NIFTI Dataset from segmented slices
Render Liver in 3D

Figure 5.2: Parallel processing approach for the *Ver. 3.0* algorithm. The serial approach is outlined in the yellow box.

5.4. Parallel Computing Results

The parallel computing results are presented in context to (1) benchmarking using the FFT based process, (2) platform comparison results between the different operating systems, and (3) liver segmentation process for the twenty different clinical datasets available.

5.4.1. Benchmarking with FFT based process

For the benchmarking process three different experiments were run. The first set of experiments computed the time required for the processing of a thousand images with resolution of 1024x1024 pixels. The process was carried out with a single worker as a reference and later was repeated for 2, 4 and 8 *workers* running in parallel. Each of these experiments was run on both Windows based systems, with the process repeated 3 times in order to provide statistical measures. Figure 5.3 shows the computational time required for processing the images as a function of the number of *workers* for system 1. It was observed that the computational time is reduced from 415 to 245 seconds as the number of *workers* was increased from 1 to 8.

The results for the speed up vs. the number of *workers* for system 1 are shown in Figure 5.4. The speed up experiment had for reference the performance obtained while using a single worker. For experiment 1, system 1 had a speed up of 1.69 as a result of going from 1 to 8 *workers*. The results obtained in this set of experiments suggest a form of saturation after using more than 4 processors. As a consequence of this, no significant improvement in the speed up with increased number of *workers* was obtained.

On running similar experiments on system 2 as shown in Figure 5.5, we observe that the computational time was reduced from 406 to 75 seconds as the numbers of *workers* was increased from 1 to 8. In this case, a speed up from 3.25 with 4 *workers*, up to 5.37 with 8 *workers* was achieved. Note that the saturation effect observed in configuration 1 (Figure 5.4) is not present in configuration 2 as can be seen from Figure 5.6. Configuration 2 displays an almost linear trend in the speed up with the increase in the number of *workers*.

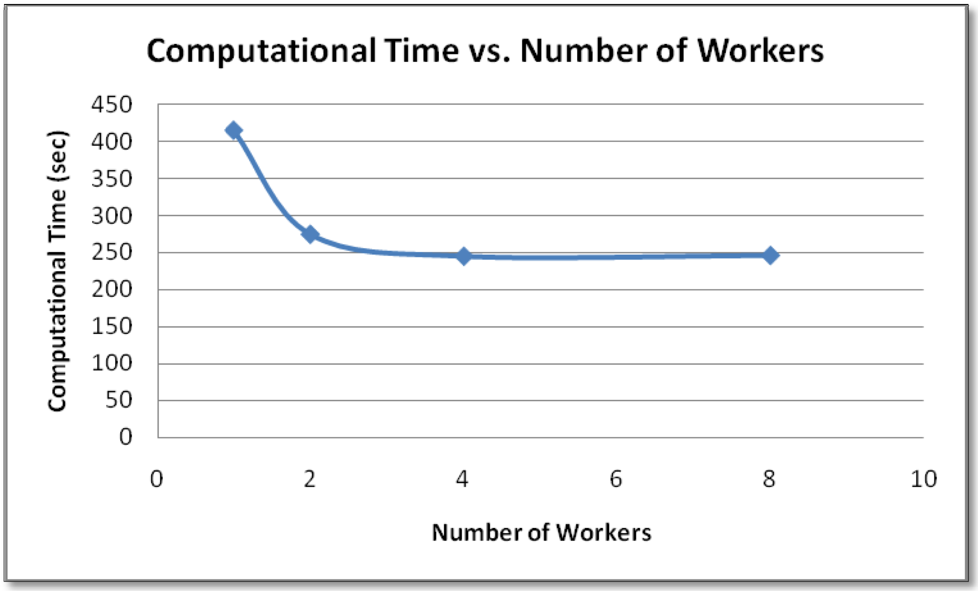


Figure 5.3: Computation Time vs. Number of *Workers* for processing 1000 images on System 1

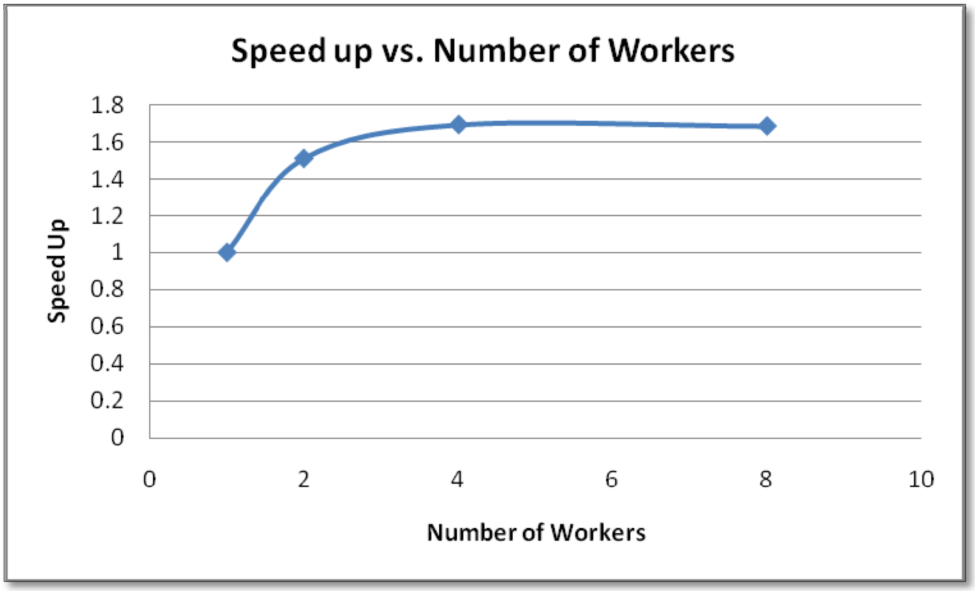


Figure 5.4: Speed up vs. Number of *Workers* for System 1

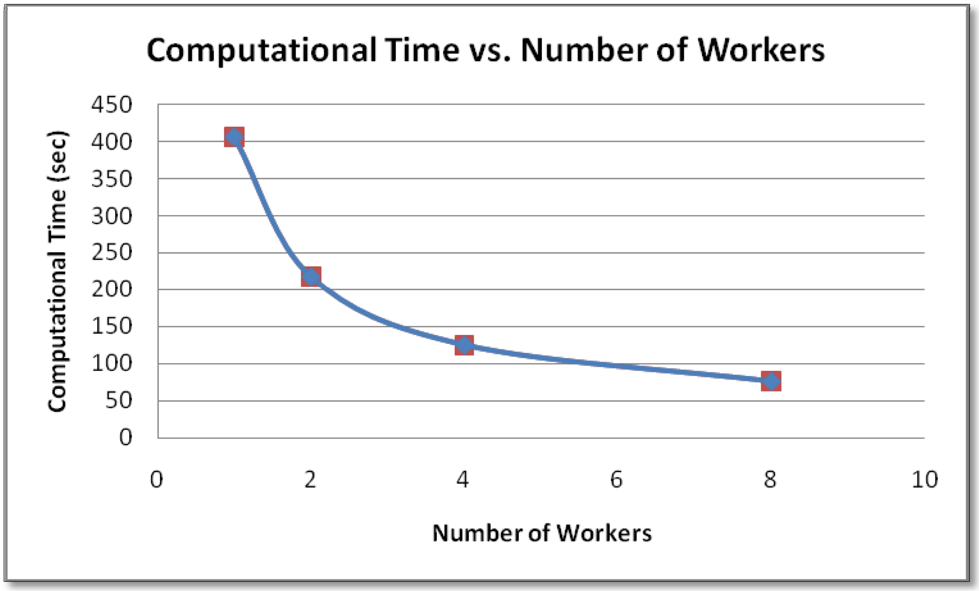


Figure 5.5: Computation Time vs. Number of *Workers* for processing 1000 images on System 2.

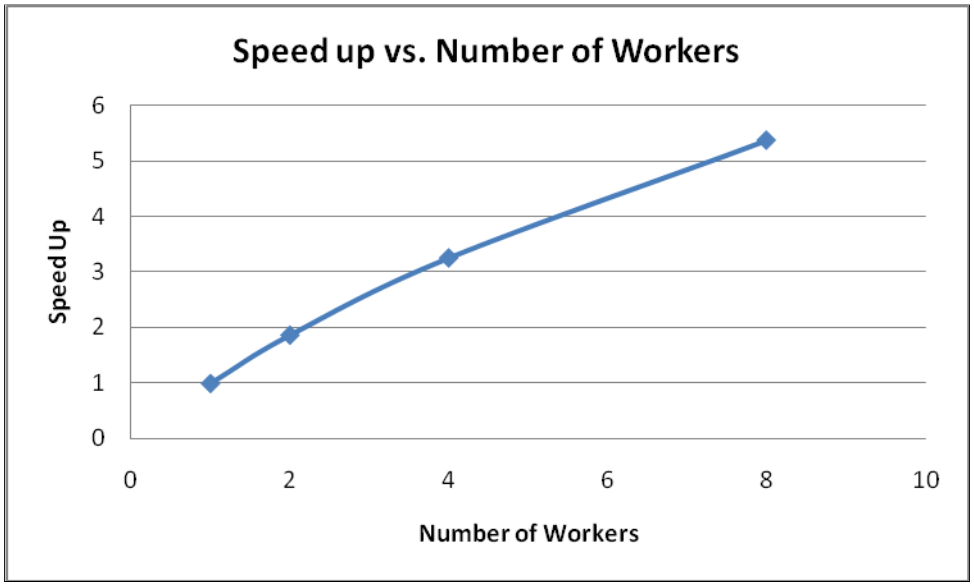


Figure 5.6: Speed up vs. Number of *Workers* for System 2

In the next set of experiments the performance of MatLab based parallelization is examined under the conditions of increased work load. In order to determine how both the computational time and speed up were affected with the increased number of images, experiments were conducted where the numbers of images were increased from 10 to 10,000. Figure 5.7 and Figure 5.8 display the results for system 1, while Figure 5.9 and Figure 5.10 display the counterpart results for system 2.

In configuration 1, the computational time was reduced significantly as the number of *workers* is increased from 1 to 4. However, a further increase to 8 *workers* did not improve the performance. Also, as can be seen from Figure 5.8, there is no improvement in the speed up when more than 100 images are used. We do observe however a rise in the speed up when the number of images are increased from 10 to 100, although for more than 100 images a plateau effect is experienced. For configuration 2, we observe the same kind of behavior as in configuration 1 in reference to the number of images being processed. This suggests that the outcome in the number of images used is independent of the system in use

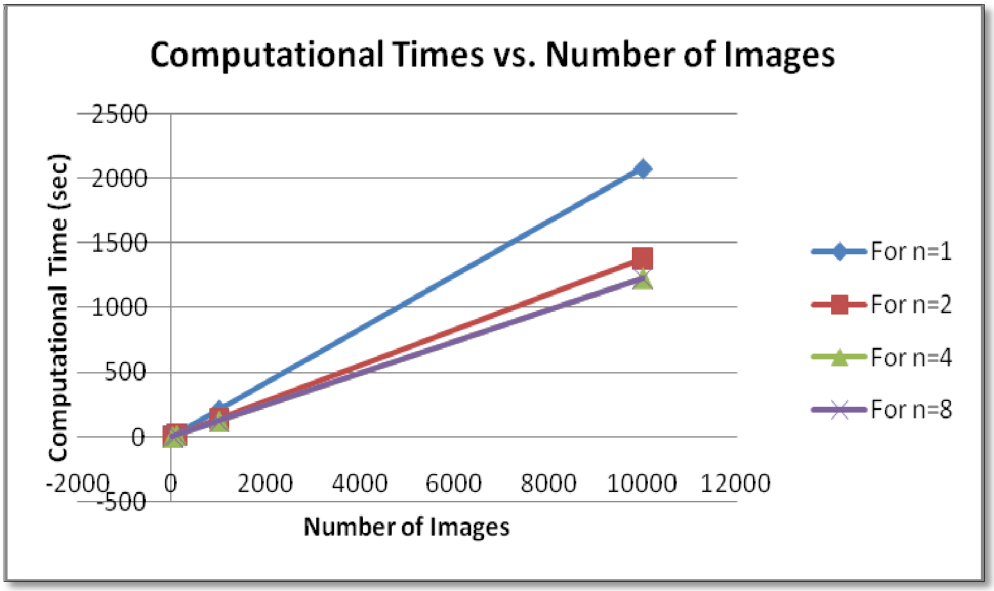


Figure 5.7: Computational Time vs. Number of Images for system 1

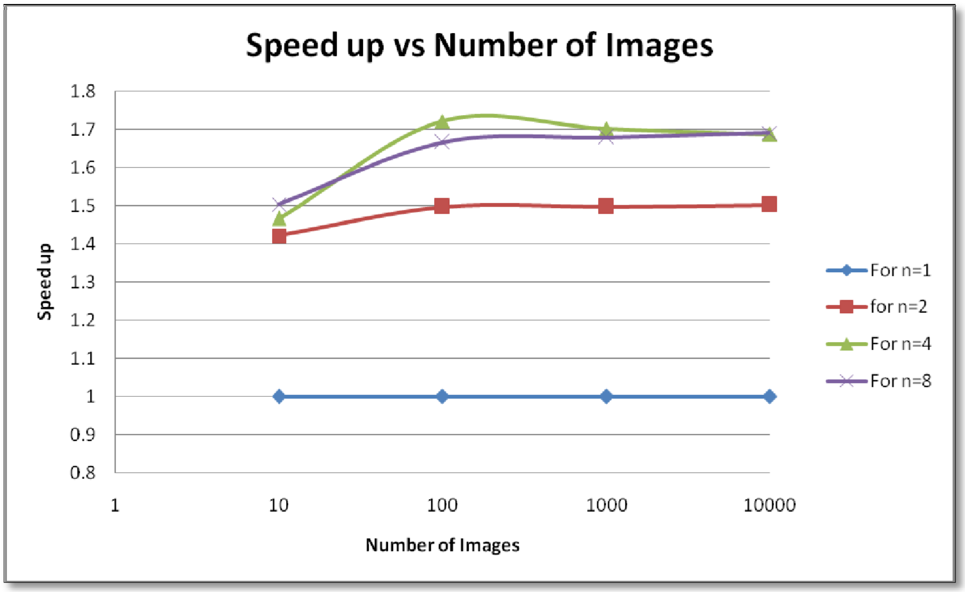


Figure 5.8: Speed up vs. Number of Images for system 1

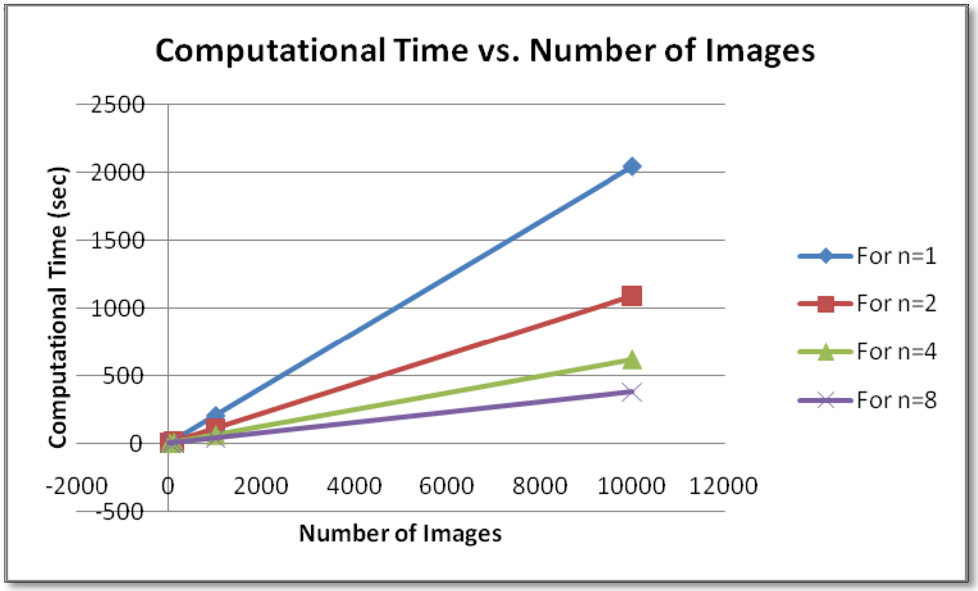


Figure 5.9: Computational Time vs. Number of Images for system 2

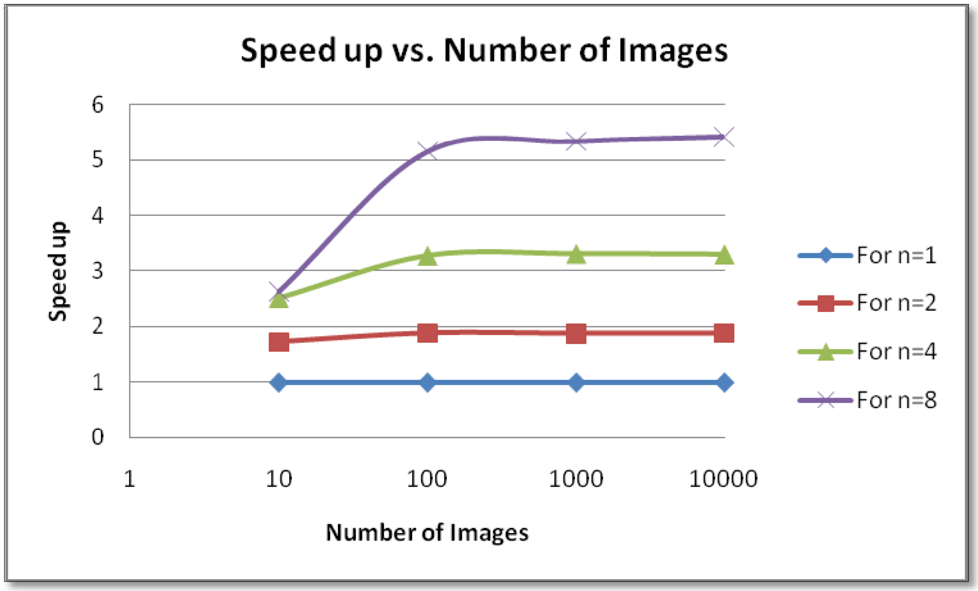


Figure 5.10: Speed up vs. Number of Images for system 2

The third set of experiments addressed the outcome of the parallel computing process as a function of the image size. For the experiments with 1000 images, different image sizes ranging from 128x128 to 1024x1024 were used, and the computational time required to perform the FFT of the images was recorded. The results of the experiment are shown in Figure 5.11 through Figure 5.14. Results in Figure 5.11 and Figure 5.12 were obtained using system 1, while results in Figure 5.13 and Figure 5.14 were obtained using system 2.

The computational time is seen to increase exponentially with the increase in size of the image. Also we did not observe a great improvement in the performance as the numbers of *workers* were increased from 4 to 8 as we had seen consistently for configuration 1.

Observe the fact that the best speed up is obtained for the image size of 256x256 which shows up as a hump in Figure 5.12. This phenomenon cannot be explained.

For system 2, the computational time increased exponentially with the image size as seen in Figure 5.11. However, the hump in Figure 5.12 which was seen with system 1 is not present for system 2. Also, the speed up for the system 2 was shown to be proportional to the increase in image size.

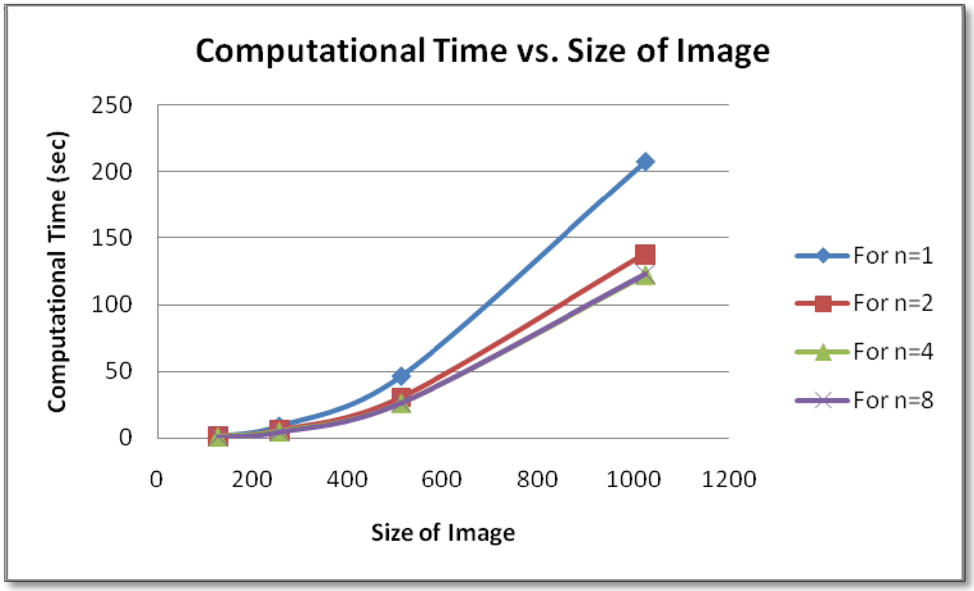


Figure 5.11: Computational Time vs. Size of Image for system 1

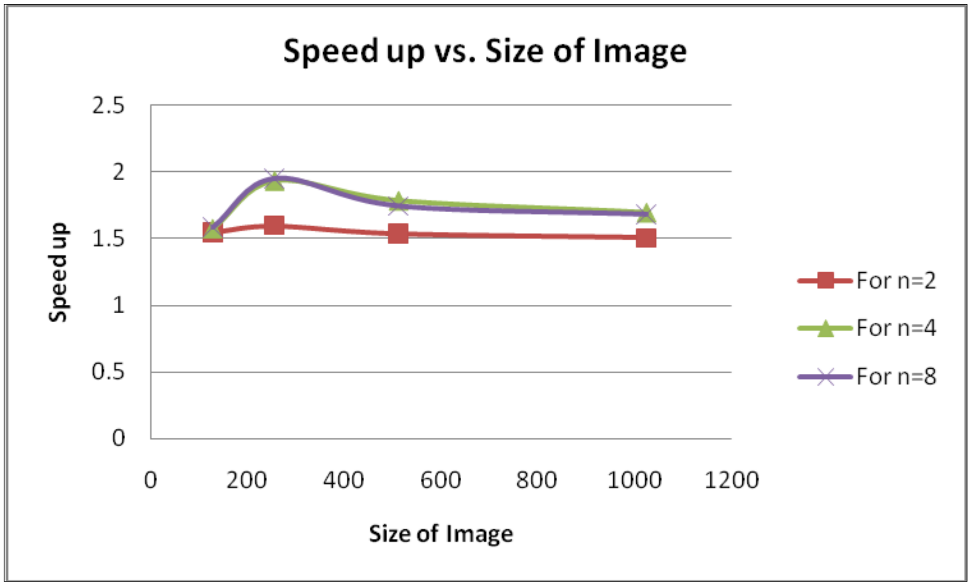


Figure 5.12: Speed up vs. Size of Image for system 1

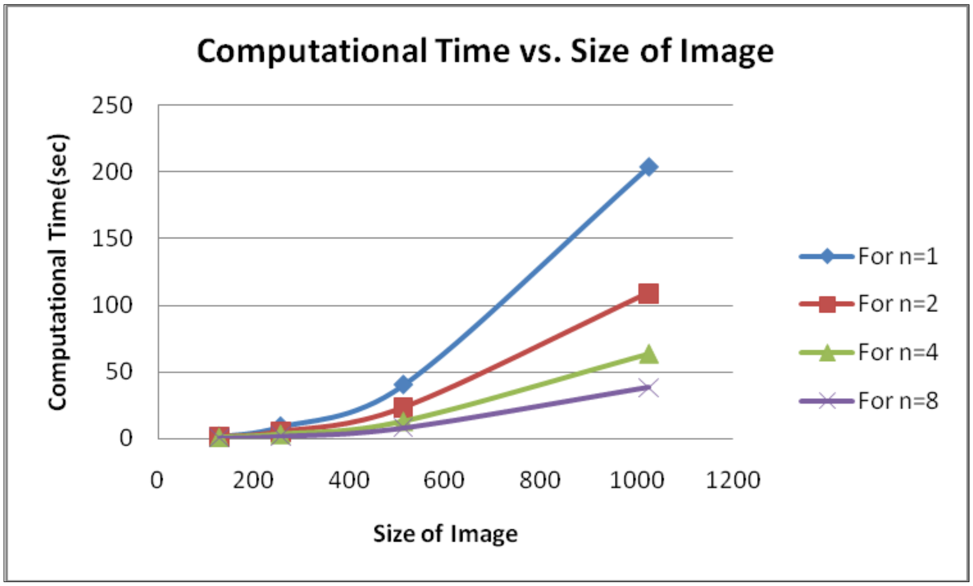


Figure 5.13: Computational Time vs. Size of Image for system 2

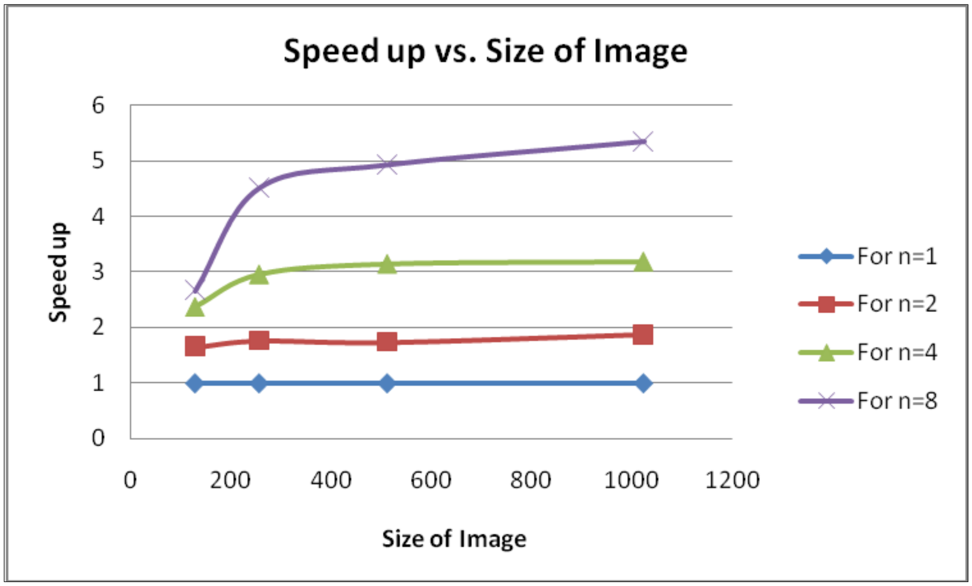
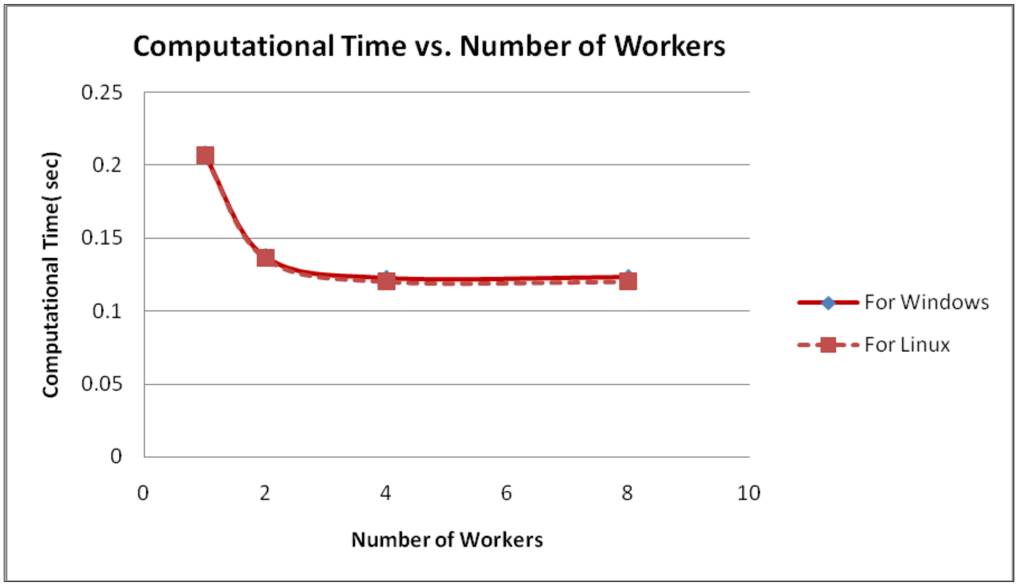


Figure 5.14: Computational Time vs. Size of Image for system 2

5.4.2. Platform Comparison

A system platform performance evaluation was carried out while varying the number of *workers*, the number of images, and the size of the images processed. Figure 5.15 through Figure 5.20 show the comparison of the performance of the parallelization process on Windows and Linux platforms, whereas Figure 5.21 through Figure 5.26 display the performance comparison of Windows and Mac OS-X platforms. In the graphs, results using Windows are displayed in solid lines, while the results using the Linux and Mac OS-X platforms are displayed in dotted lines.



\

Figure 5.15: Computational Time vs. Number of *Workers*: A Comparison of Windows and Linux platforms conducted on System 1

In Figure 5.15 and Figure 5.16 it is noted that both operating systems (Windows and Linux) behave similarly with respect to increasing the number of *workers*. The performances in both systems show saturation with no significant gains achieved on increasing the numbers of *workers* from 4 to 8. Although the Linux-based results do show an improvement as seen in Figure 5.16, however it is not significant.

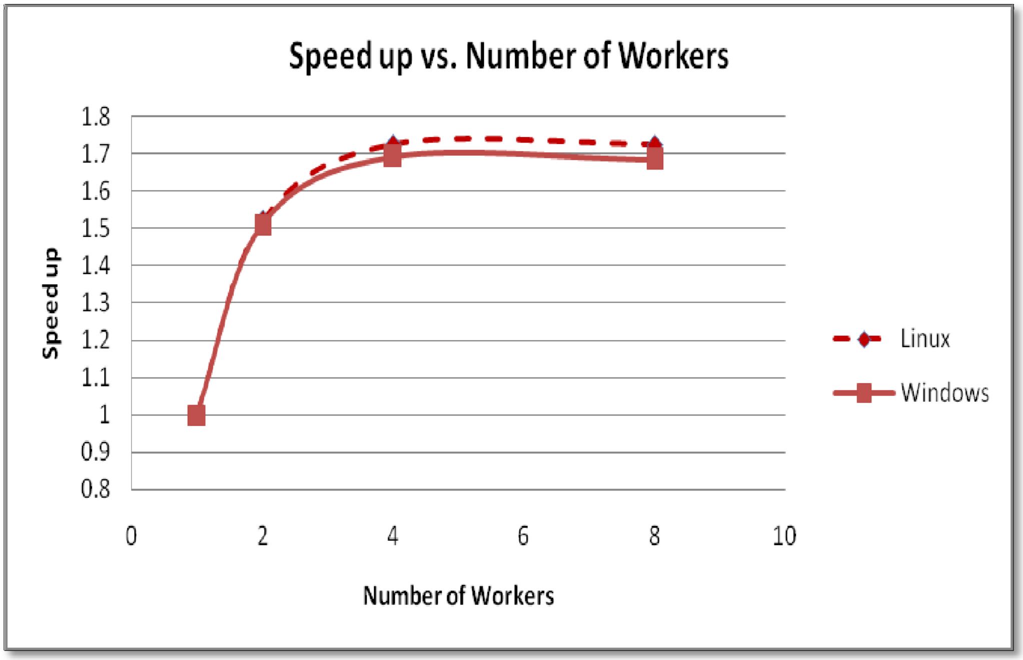


Figure 5.16: Speed up vs. Number of *Workers*: A Comparison of Windows and Linux platforms conducted on System 1.

Figure 5.17 and Figure 5.18 depict the performance reached in system 1 on using different numbers of *workers* (*n*) on Linux and Windows operating systems, respectively. The computational time with respect to the number of images shows that both systems perform equally well. However, in Figure 5.18 an interesting finding is that MatLab performs better for larger number of images on Linux, but performs worse for small number of images on the Windows system. Note that for 10 images the Linux based system performs poorly, but for more than 100 images it shows an improved performance that is actually better than the Windows-based system. Also, speed up for both Windows-based and Linux-based systems present similar trends. With a small number of images, the speed up increases almost linearly with the numbers of *workers*, and when the number of images exceeds 100, the speed up ratio reaches a plateau.

This platform comparison was also carried out with images of different sizes. The results for the computational time and the speed up are displayed in Figure 5.19 and Figure 5.20, respectively. In response to varying image size, both the Windows and the Linux platforms show comparable performances with no clear distinction between them. Both Windows and Linux platforms show an unexplainable hump at image size of 256x256, which was seen for system 1 earlier.

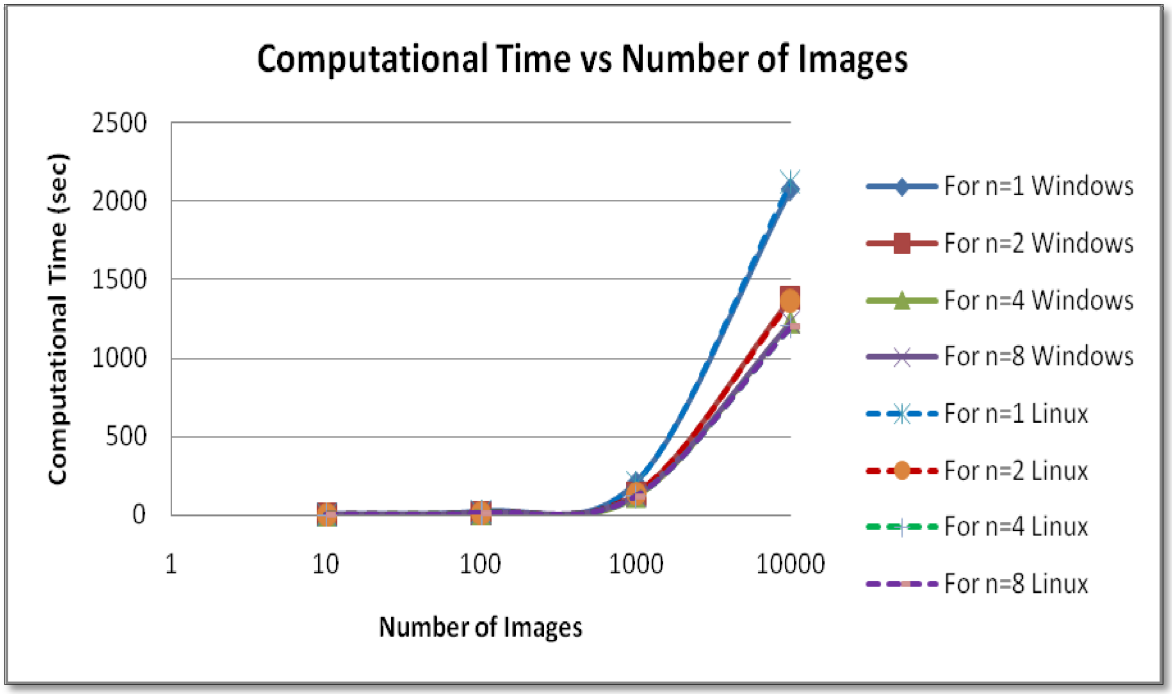


Figure 5.17: Computational Time vs. Number of Images: A Comparison of Windows and Linux platforms conducted on System 1

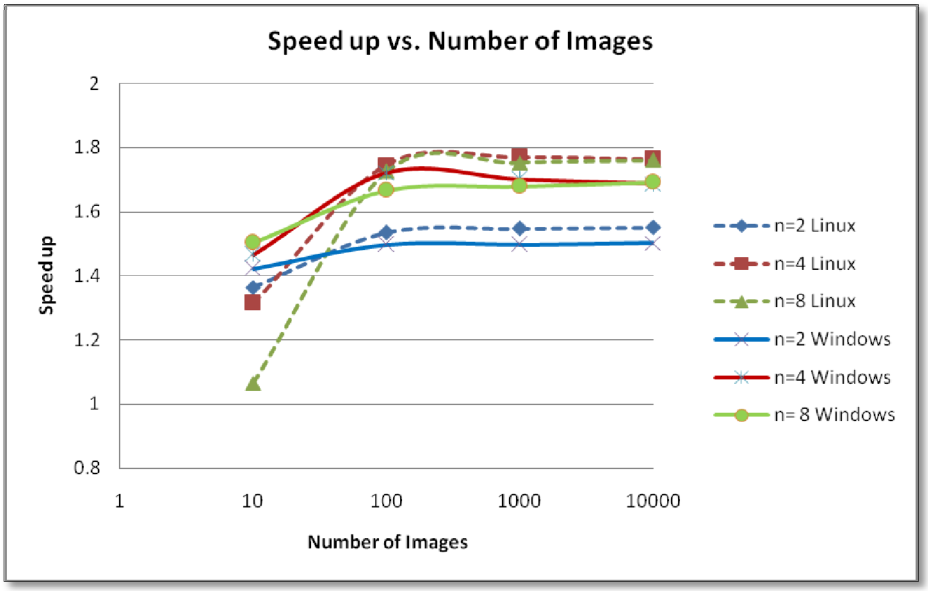


Figure 5.18: Speed up vs. Number of Images: A Comparison of Windows and Linux platforms conducted on System 1

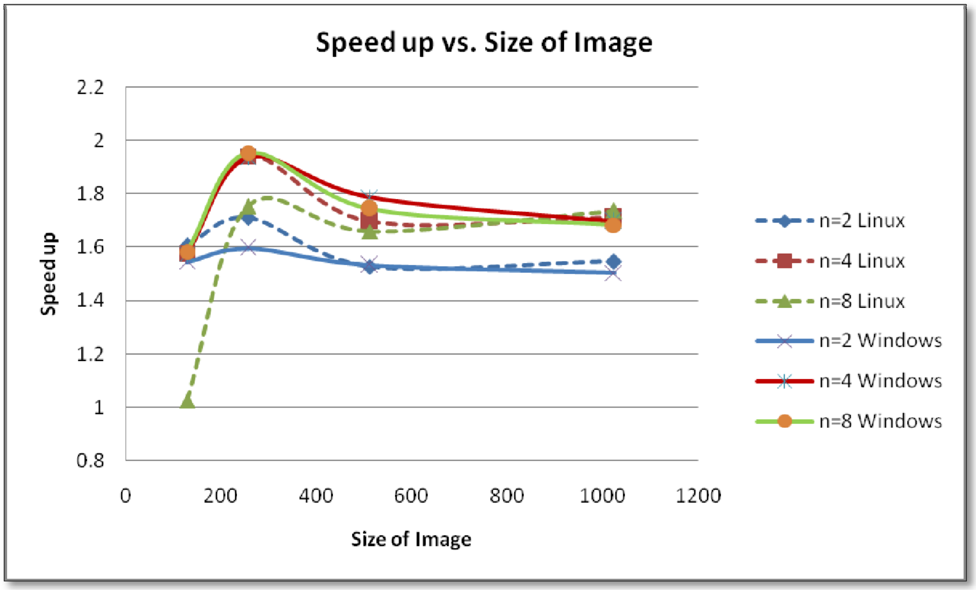


Figure 5.19: Speed up vs. Size of Images: A Comparison of Windows and Linux platforms conducted on System 1.

The computational time achieved by the Windows platform were lower than those obtained by the Mac OS-X platform by a factor of 2 as shown in Figure 5.21. Though the Windows platform yielded half the computational speed as compared to the Mac OS-X, the speed up as we go from a single worker to 8 *workers* is better for the Mac. Thus, although the Mac OS-X platform displays a better performance on the speed up, as shown in Figure 5.22, the Windows platform exhibits a much faster performance, as shown in Figure 5.21.

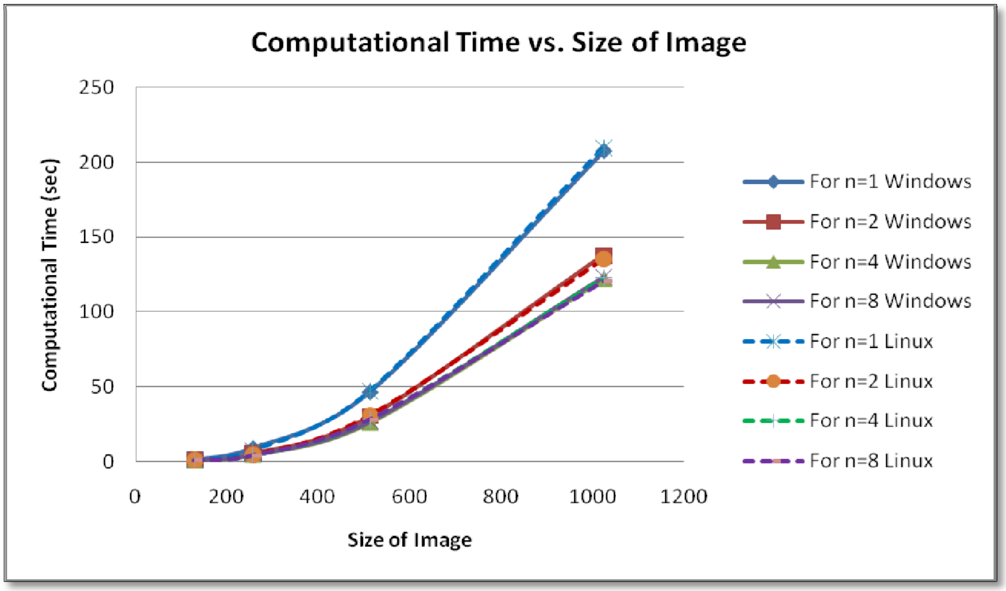


Figure 5.20: Computational Time vs. Size of Images: A Comparison of Windows and Linux platforms conducted on System 1.

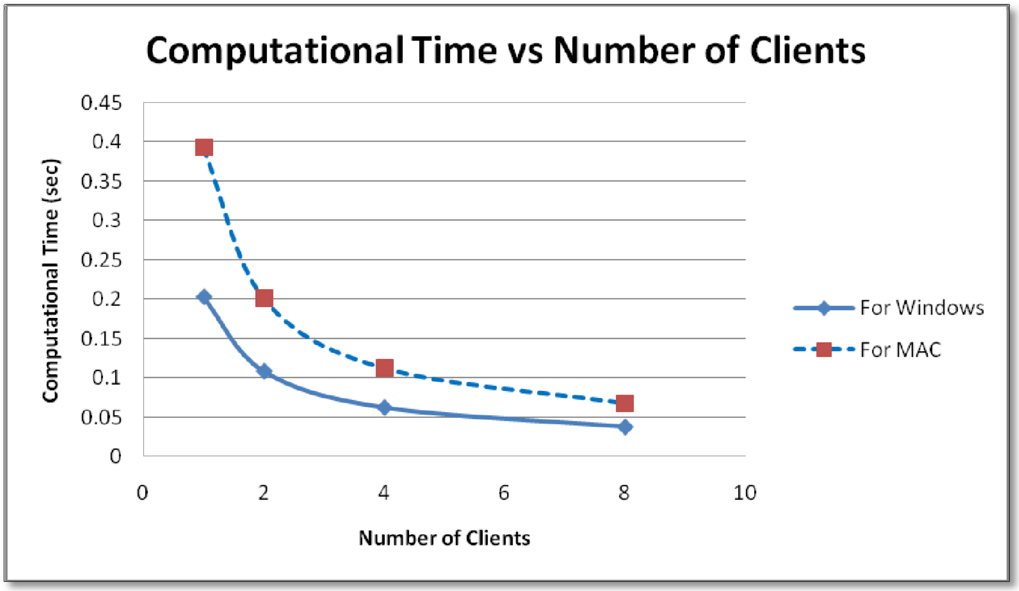


Figure 5.21: Computational Time vs. Number of *Workers*: A Comparison of Windows and Mac OS-X platforms conducted on System 2

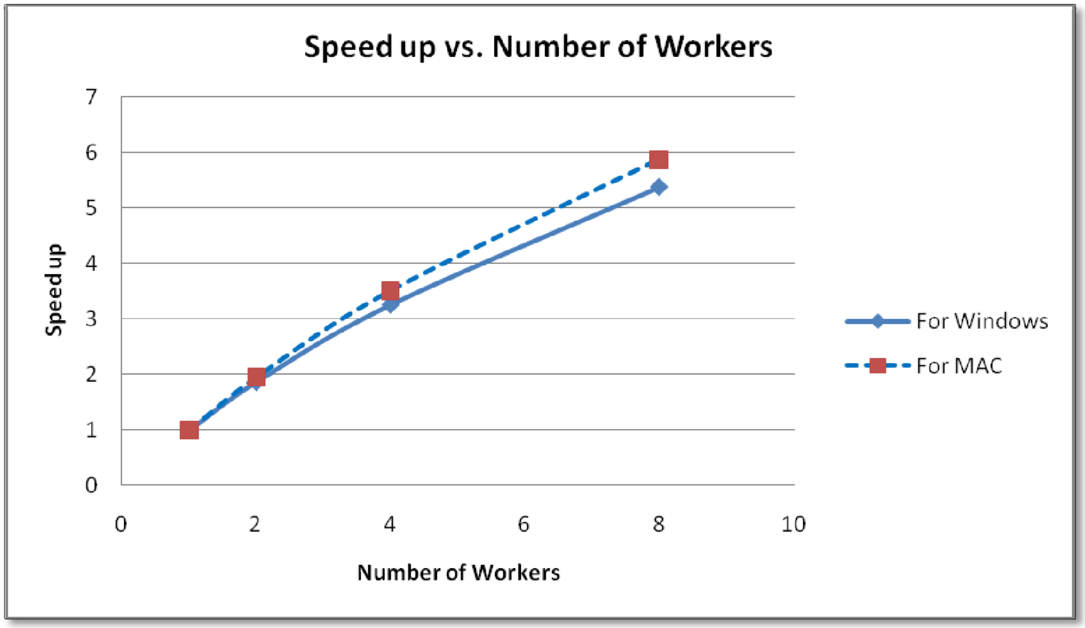


Figure 5.22: Speed up vs. Number of *Workers*: A Comparison of Windows and Mac OS-X platforms conducted on System 2

In terms of the different number of images used as illustrated in Figure 5.23, the results show that the computational time when using the Windows platform was half that of the Mac platform for any number of images when using the same number of *workers*. Note that the performance that was obtained using 2 *workers* on the Mac platform was obtained with only 1 worker on the windows platform. Once again, Mac OS-X exhibited higher speed up upon increasing the computing resources and the load, as shown in Figure 5.24.

Figure 5.23 displays a significant rise in the computational time as the size of the image was increased. We again observed the trend that the Mac platform took twice as long to finish a task as compared to the Windows platform with the same number of *workers*. Results shown in Figure 5.26 indicate that the Mac platform shows better speed up as compared to the Windows platform for larger image sizes.

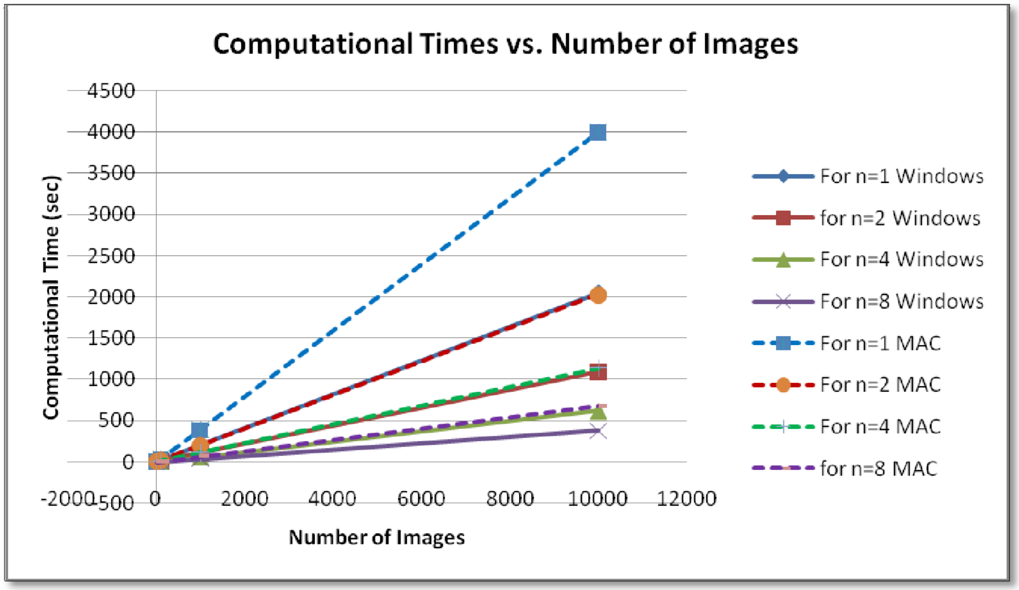


Figure 5.23: Computational Time vs. Number of Images: A Comparison of Windows and Mac OS-X platforms conducted on System 2

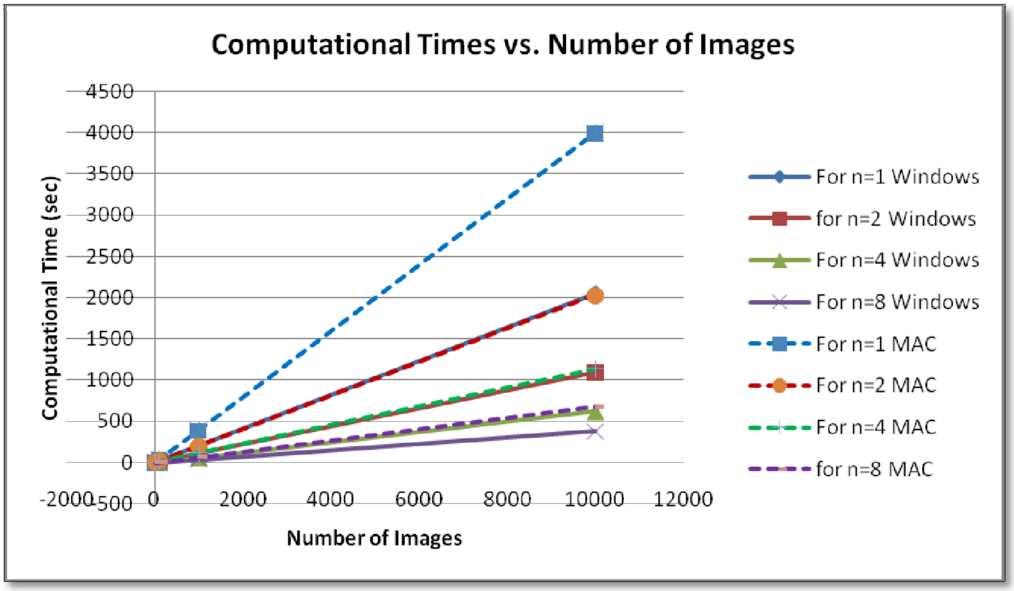


Figure 5.24: Speed up vs. Number of Images: A Comparison of Windows and Mac OS-X platforms conducted on System 2

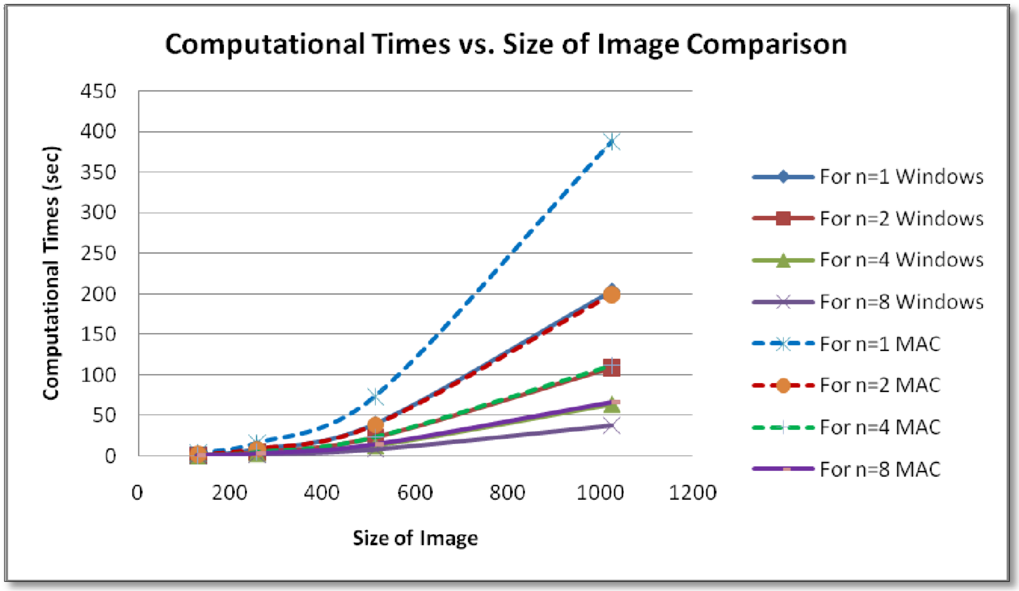


Figure 5.25: Computational Time vs. Size of Images: A Comparison of Windows and Mac OS-X platforms conducted on System 2

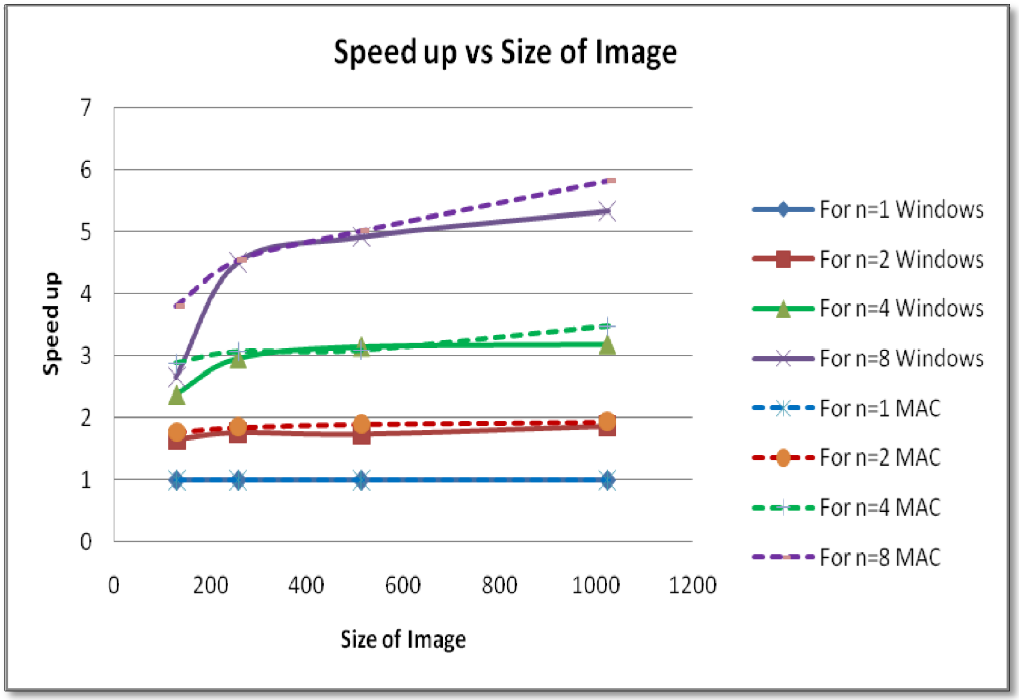


Figure 5.26: Speed up vs. Size of Images: A Comparison of Windows and Mac OS-X platforms conducted on System 2

5.4.3 Specific Application: Liver Segmentation

This section presents the results of the liver segmentation process implemented on system 2 with Windows Vista platform. Due to the computational demands of this algorithm and based on the performance obtained from the previous results, we concluded that it was unnecessary to perform this experiment on system 1 for both Windows and Linux platforms and on system 2 for Mac-OS-X. Taking into account the performance of the parallel computing algorithm with different variables, the segmentation was performed on images of size of 512x512. The number of slices for the 20 clinical datasets ranged from 132 to 377 as indicated in Chapter 1.

Table 5.1 presents the results of parallelizing *Ver. 1.0* of the liver segmentation algorithm as described in section 5.3. The table shows the time required to segment the dataset using the serial approach using 1 *worker* as compared to the parallel approach using 8 *workers*. The table shows the time required to segment the entire liver dataset as well as a single slice of the dataset. It is seen that the highest processing time using the serial approach was around 257 min which was reduced to 51 min using the parallel processing approach with 8 *workers*.

Figure 5.27 shows the average computational time for the twenty datasets as a function of the number of *workers* using the *Ver. 1.0* algorithm. It is seen from the graph that the computational time does not reduce linearly as the number of *workers* is increased.

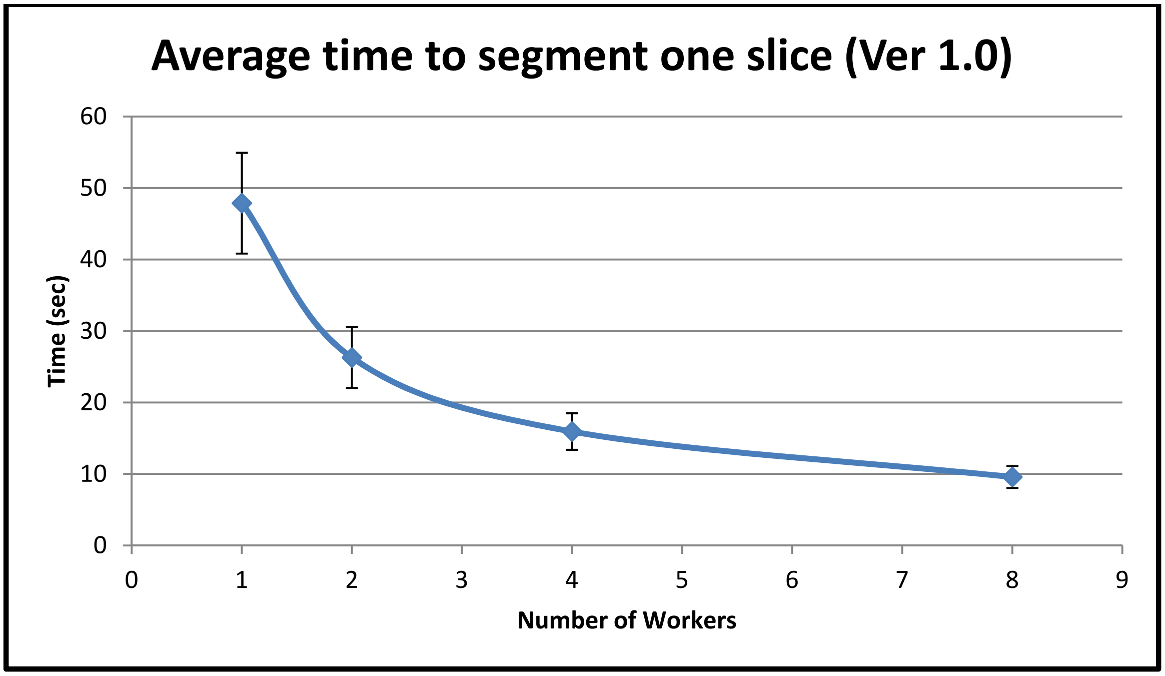


Figure 5.27: Average computational time for the segmentation of a single slice of the datasets as a function of the Number of *workers* employed for *Ver. 1.0* algorithm. The Error bars show the standard deviation across the 20 datasets.

Table 5.1: Computational Time for *Ver. 1.0* Algorithm

NO	Number of slices Containing Liver	Average time taken to segment entire dataset (min)		Average time taken to segment 1 slice (sec)	
		Serial Approach	Parallel Approach with 8 *workers*/cores	Serial Approach	Parallel Approach with 8 *workers*/cores
1	163	147.01	29.23	54.11	10.76
2	377	257.51	51.75	40.98	8.24
3	208	169.16	33.42	48.80	9.64
4	295	206.64	41.27	42.03	8.39
5	247	183.45	36.91	44.56	8.97
6	189	158.60	31.17	50.35	9.90
7	194	162.61	32.04	50.29	9.91
8	192	161.94	32.32	50.61	10.10
9	168	154.07	30.66	55.03	10.95
10	169	154.21	30.73	54.75	10.91
11	170	152.87	30.94	53.95	10.92
12	364	252.47	50.90	41.62	8.39
13	132	111.51	22.33	50.69	10.15
14	365	252.81	51.11	41.56	8.40
15	189	159.06	31.38	50.50	9.96
16	176	157.23	31.83	53.60	10.85
17	230	177.64	35.92	46.34	9.37
18	324	225.68	45.74	41.79	8.47
19	251	186.20	37.65	44.51	9.00
20	356	247.47	49.62	41.71	8.36
Average	237.95	183.91	36.85	47.89	9.58
Std Dev	79.79	42.10	8.61	5.14	1.00

Figure 5.28 shows the average speed up of the twenty datasets as a function of the number of *workers* engaged using *Ver. 1.0* algorithm. It is seen from Figure 5.28 that the increase in speed up is fairly linear as the number of *workers* is increased. However, the maximum speed up obtained with 8 *workers* is around 5. The ideal case speed up with 8 *workers* must be around 8 since 8 slices are processed at a time but this is not seen in the experimental results due to loss in overhead due to storing, parsing and sequencing of the data.

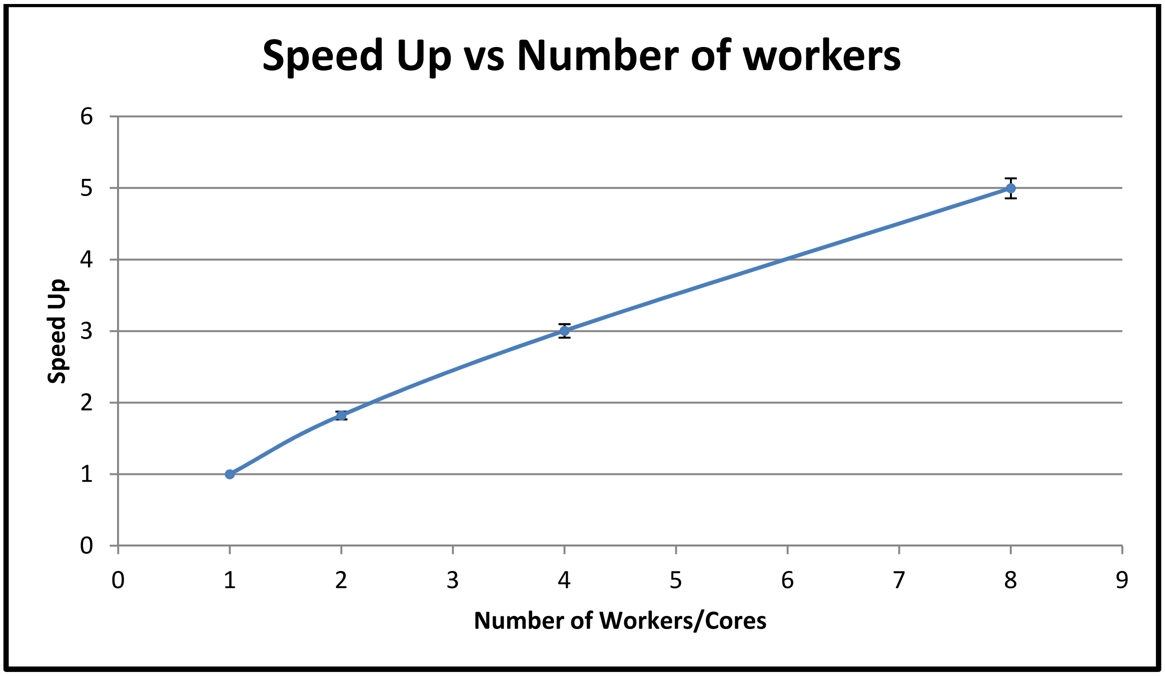


Figure 5.28: Average speed up for the segmentation of the datasets as a function of the Number of *workers* employed for *Ver. 1.0* algorithm. The Error bars show the standard deviation across the 20 datasets

Table 5.2 shows the computational time for the 20 datasets using the serial and parallel approaches for the *Ver. 2.0* algorithm. Figure 5.29 and Figure 5.30 show the average computational time and the average speed up as a function of the number of *workers* for the *Ver. 2.0* algorithm. The *Ver. 2.0* results show a similar trend as that of *Ver. 1.0* algorithm.

Table 5.2: Computational Time for *Ver. 2.0* Algorithm

Dataset No	**Number of slices Containing Liver**	Average time taken to segment entire dataset (min)		Average time taken to segment 1 slice (sec)	
		Serial Approach	Parallel Approach with 8 *workers*/cores	Serial Approach	Parallel Approach with 8 *workers*/cores
1	163	147.01	29.23	54.11	10.76
2	377	257.51	51.75	40.98	8.24
3	208	169.16	33.42	48.80	9.64
4	295	206.64	41.27	42.03	8.39
5	247	183.45	36.91	44.56	8.97
6	189	158.60	31.17	50.35	9.90
7	194	162.61	32.04	50.29	9.91
8	192	161.94	32.32	50.61	10.10
9	168	154.07	30.66	55.03	10.95
10	169	154.21	30.73	54.75	10.91
11	170	152.87	30.94	53.95	10.92
12	364	252.47	50.90	41.62	8.39
13	132	111.51	22.33	50.69	10.15
14	365	252.81	51.11	41.56	8.40
15	189	159.06	31.38	50.50	9.96
16	176	157.23	31.83	53.60	10.85
17	230	177.64	35.92	46.34	9.37
18	324	225.68	45.74	41.79	8.47
19	251	186.20	37.65	44.51	9.00
20	356	247.47	49.62	41.71	8.36
Average Std Dev	237.95 79.79	183.91 42.10	36.85 8.61	47.89 5.14	9.58 1.00

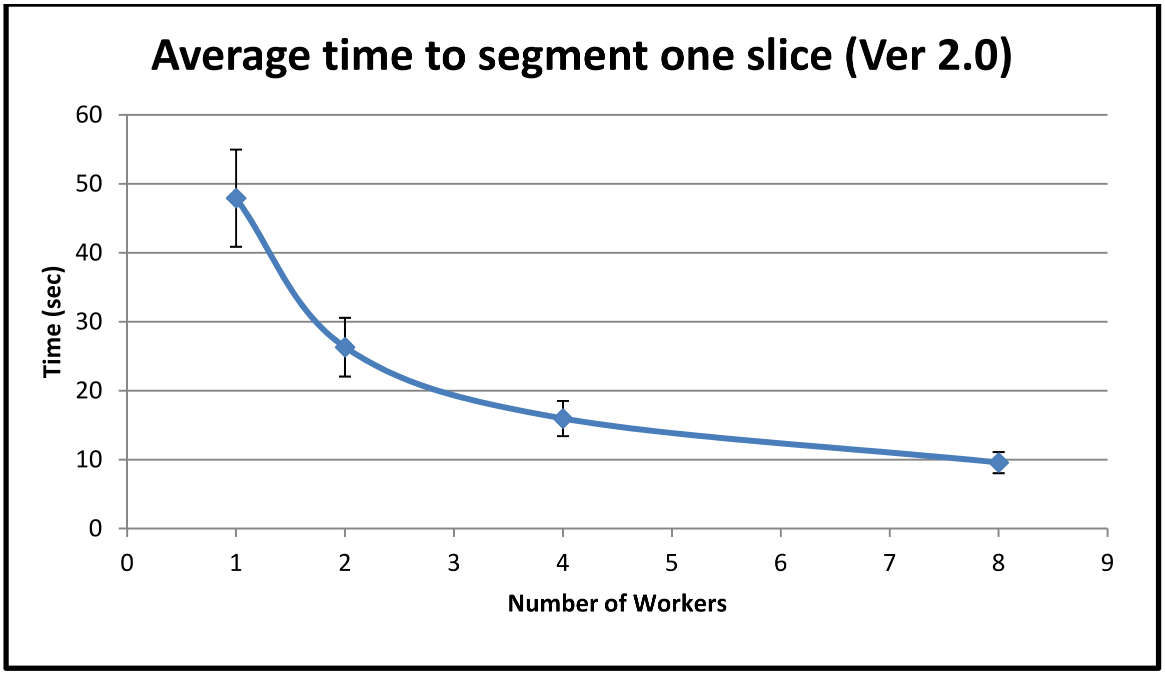


Figure 5.29: Average computational time for segmentation of a single slice of the datasets as a function of the Number of *workers* employed for *Ver. 2.0* algorithm. The Error bars show the standard deviation across the 20 datasets

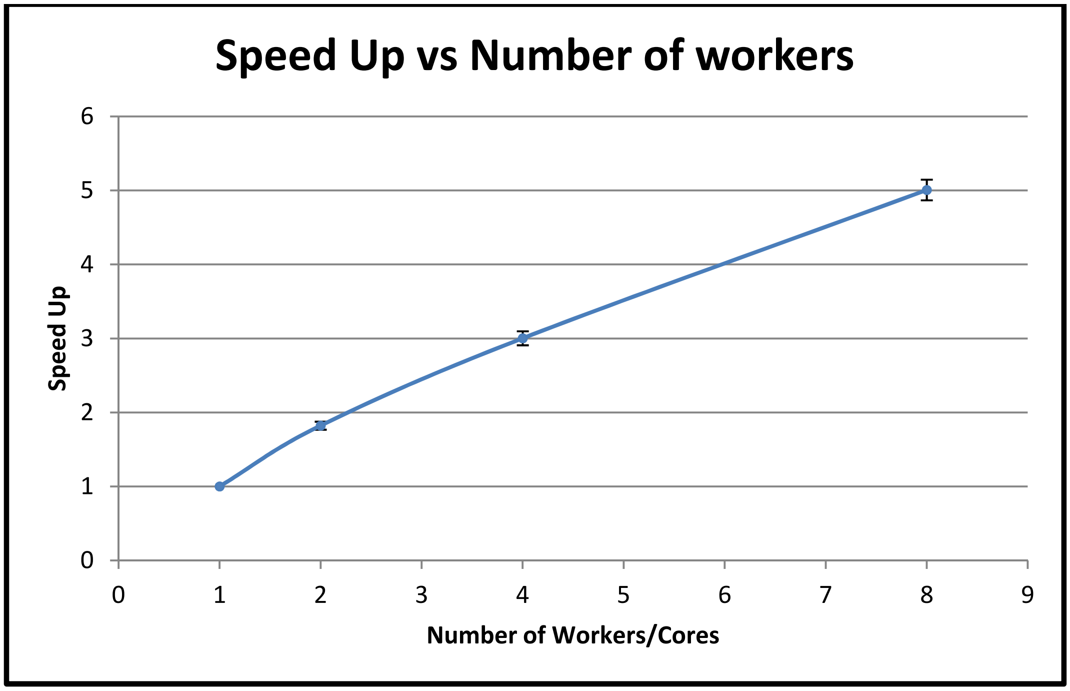


Figure 5.30: Average speed up for segmentation of the datasets as a function of the Number of *workers* employed for *Ver. 2.0* algorithm. The Error bars show the standard deviation across the 20 datasets.

Table 5.3 provides the computational time for the segmentation process using the *Ver. 3.0* algorithm. Figure 5.31 and Figure 5.32 demonstrate similar graphs as seen in Figure 5.29 and Figure 5.30 for the *Ver. 3.0* algorithm, respectively. Although, a similar trend is seen for the *Ver. 3.0* algorithm, with a non-linear decrease in computational time and a linear increase in speed up with increasing number of *workers*, the computational time for the *Ver. 3.0* algorithm are higher than that seen for the *Ver. 1.0* or the *Ver. 2.0* algorithms by about 5 min for segmenting the entire dataset. This is attributed to the extra time spent in the generation of the automated initial contours from the single initialized slice.

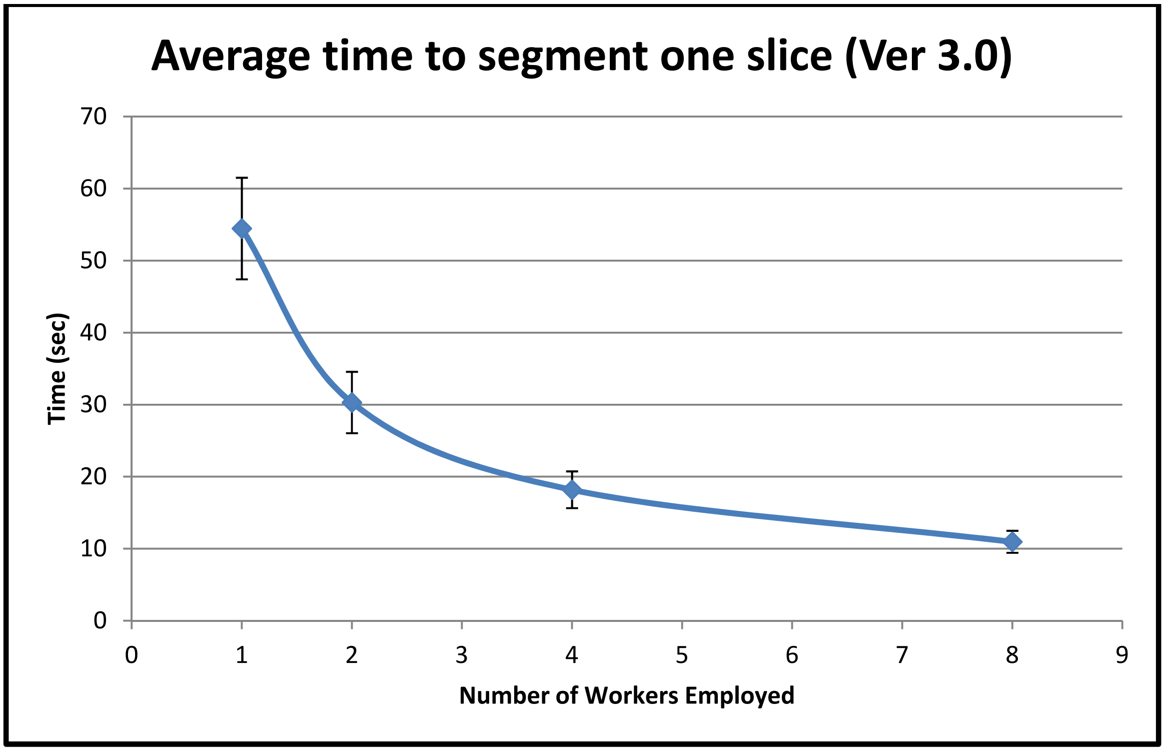


Figure 5.31: Average computational time for segmentation of a single slice of the datasets as a function of the Number of *workers* employed for *Ver. 3.0* algorithm. The Error bars show the standard deviation across the 20 datasets.

Table 5.3: Computational Time for *Ver. 3.0* Algorithm

NO	Number of slices Containing Liver	Average time taken to segment entire dataset (min)		Average time taken to segment 1 slice (sec)	
		Serial Approach	Parallel Approach with 8 *workers*/cores	Serial Approach	Parallel Approach with 8 *workers*/cores
1	163	171.26	35.32	63.04	13.00
2	377	280.28	55.58	44.61	8.85
3	208	192.68	38.64	55.58	11.15
4	295	230.74	47.16	46.93	9.59
5	247	208.12	42.13	50.56	10.23
6	189	183.48	38.41	58.25	12.19
7	194	186.25	35.70	57.60	11.04
8	192	184.36	36.99	57.61	11.56
9	168	176.52	36.54	63.04	13.05
10	169	176.98	35.30	62.83	12.53
11	170	177.39	36.34	62.61	12.83
12	364	275.23	53.71	45.37	8.85
13	132	135.95	27.95	61.80	12.70
14	365	275.54	53.37	45.29	8.77
15	189	183.85	38.39	58.37	12.19
16	176	180.28	34.46	61.46	11.75
17	230	200.23	38.81	52.23	10.13
18	324	248.43	50.19	46.01	9.29
19	251	210.05	42.30	50.21	10.11
20	356	270.89	55.49	45.66	9.35
Average	237.95	207.43	41.64	54.45	10.96
Std Dev	79.79	41.75	8.08	7.05	1.53

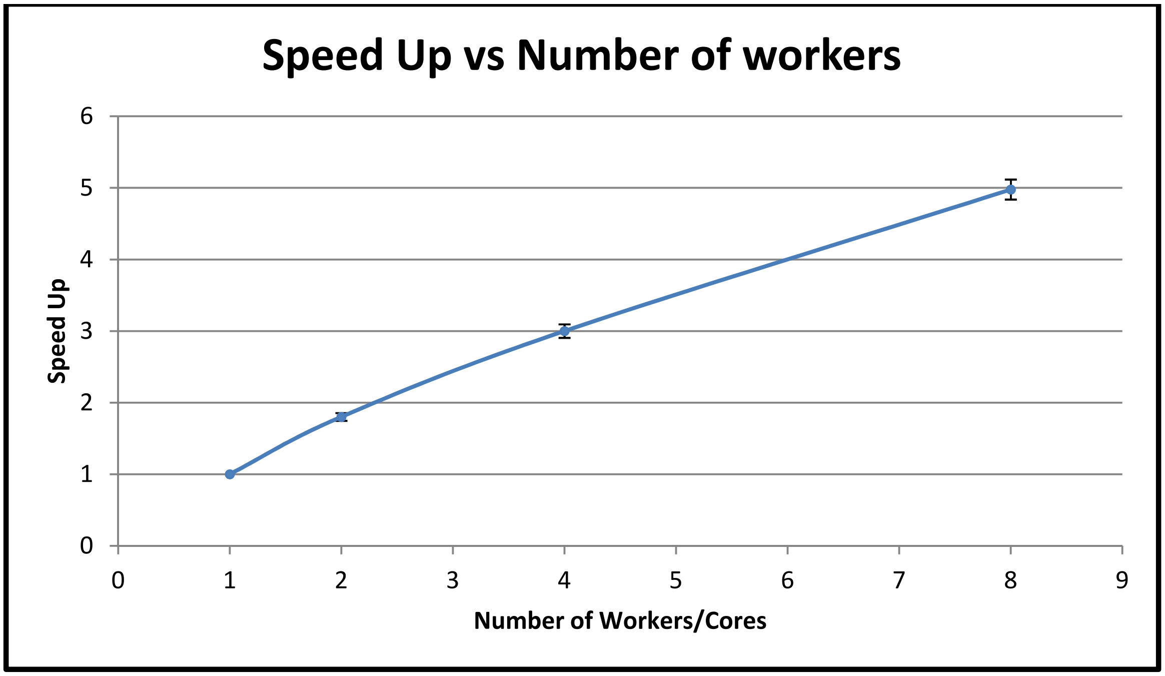


Figure 5.32: Average speed up for segmentation of the datasets as a function of the Number of *workers* employed for *Ver. 3.0* algorithm. The Error bars show the standard deviation across the 20 datasets.

Figure 5.33 provides the computational time as a function of the number of slices in the dataset for the *Ver. 3.0* algorithm. Figure 5.33 demonstrates that a linear relationship is observed between the number of slices in the dataset and the time required to segment the entire dataset. This is attributed to the fact that the designed algorithms proceed in a slice-by-slice manner. Such a trend can also be observed for *Ver. 1.0* and *Ver. 2.0* algorithms since these algorithms also operate in a slice-by-slice manner.

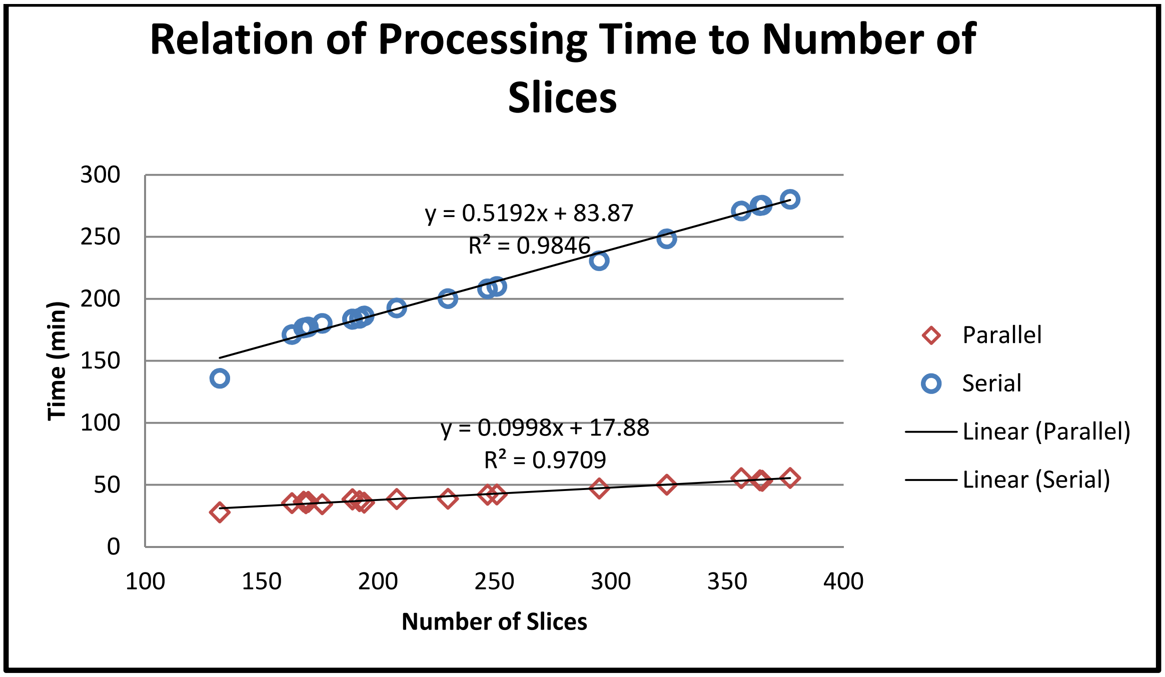


Figure 5.33: Time required for segmenting the entire dataset as a function of the number of slices in the dataset for both the serial and parallel approaches using *Ver. 3.0* algorithm

CHAPTER VI

RESULTS

This Chapter is divided into 4 main sections. The first section demonstrates the accuracy of the results obtained in estimating the volumes obtained by the proposed segmentation for the 20 liver datasets to the manually obtained volumes. The second section illustrates the comparative results and statistical analysis carried out to determine both consistency in the results and user independence. The third section of the results shows the improvement of the computational time due to the deployment of the parallel computing process. The final section demonstrates the results obtained for the tumor and blood vessel segmentation algorithms. The tumor segmentation algorithm is validated by comparing the obtained volumes with that of a manual expert. The blood vessel segmentation are qualitative in nature since quantitative validation of such results was not possible.

6.1. Accuracy Analysis of the liver segmentation approach

Twenty datasets are used for testing the segmentation algorithm. The characteristics of the datasets used are given in Chapter 2. In order to validate the segmentation results obtained by the algorithm, the volumes of the extracted livers were compared to manually calculated volumes. The manual calculation was done by an expert and is treated as the gold standard for the comparison which is routinely the suggested method (Heimann et al., 2009).

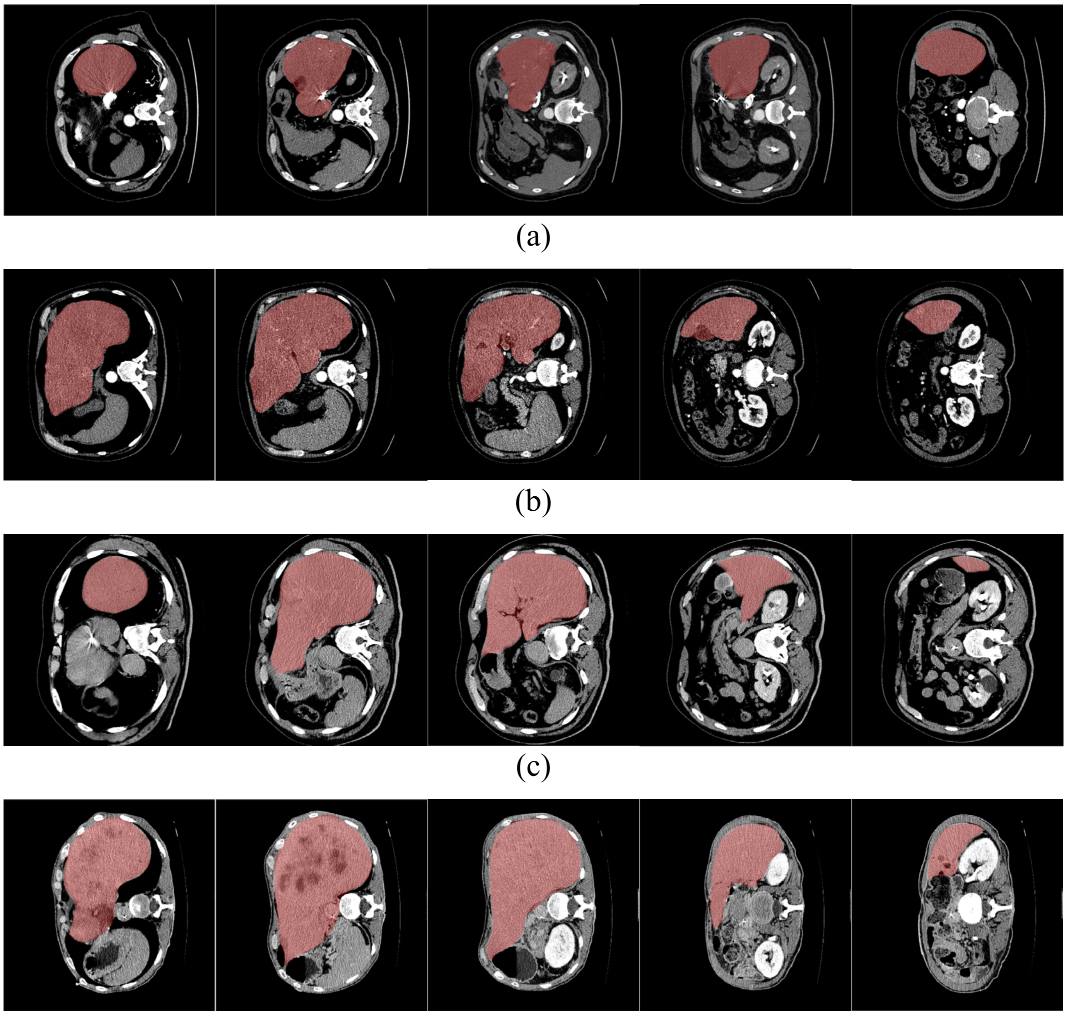
6.1.1. Results for Liver Segmentation algorithm *Ver. 1.0*

Table 6.1 provides the results of the comparison of the volumes obtained by *Ver. 1.0* of the algorithm to the manual volumes obtained by the segmentation carried out by the expert. The average calculated volumes are obtained by calculating the mean of the volumes obtained on performing the results three times. The absolute error is calculated as the absolute difference between the manual and average calculated volume and the percent error is the relative error in the volume calculation expressed in terms of per 100. A positive percent error denotes a calculated volume higher than the true volume and a negative percent error denotes an underestimation.

Table 6.1: Comparative results between the proposed algorithm and manual gold standard for *Ver. 1.0* algorithm

No	**Manual Segmentation Volumes (ml)**	**Average Calculated Volumes (ml)**	**Absolute Error (ml)**	**% Error**
1	1458.80	1418.74	40.06	-2.75
2	4079.50	4099.82	20.33	0.50
3	2419.18	2380.61	38.57	-1.59
4	1848.73	1876.78	28.05	1.52
5	1630.59	1624.38	6.21	-0.38
6	1568.82	1422.10	146.72	-9.35
7	1408.27	1336.97	71.30	-5.06
8	1331.81	1300.74	31.07	-2.33
9	2595.61	2647.43	51.83	2.00
10	2651.20	2693.55	42.36	1.60
11	1631.40	1663.90	32.50	1.99
12	2892.98	2600.33	292.65	-10.12
13	1904.25	1954.37	50.12	2.63
14	2704.79	2737.25	32.46	1.20
15	1602.26	1504.40	97.86	-6.11
16	1565.97	1672.89	106.92	6.83
17	2408.85	2372.54	36.31	-1.51
18	5336.28	5112.33	223.96	-4.20
19	1102.16	1144.50	42.33	3.84
20	2363.25	2411.08	47.83	2.02
Average	2225.24	2198.74	71.97	-1.6
Std Dev	1018.79	987.26	72.41	4.0

Figure 6.1 provides illustrative examples of the segmentation results for 4 different datasets. The results are displayed in the CT window of [-40 180] since it best displays the features in the abdominal slices. The figure shows results of the segmentation (shown in brown color) overlaid on the sparsely spaced CT slices of the datasets. The results are displayed to demonstrate the effectiveness of the algorithm in extracting the liver from surrounding organs.



(a)

(b)

(c)

(d)

Figure 6.1: Illustrative examples of segmentation results for 4 datasets. (a) Subject 19 (b) Subject 2 (c) Subject 15 (d) Subject 20 using the Ver. 1.0 of the liver segmentation algorithm

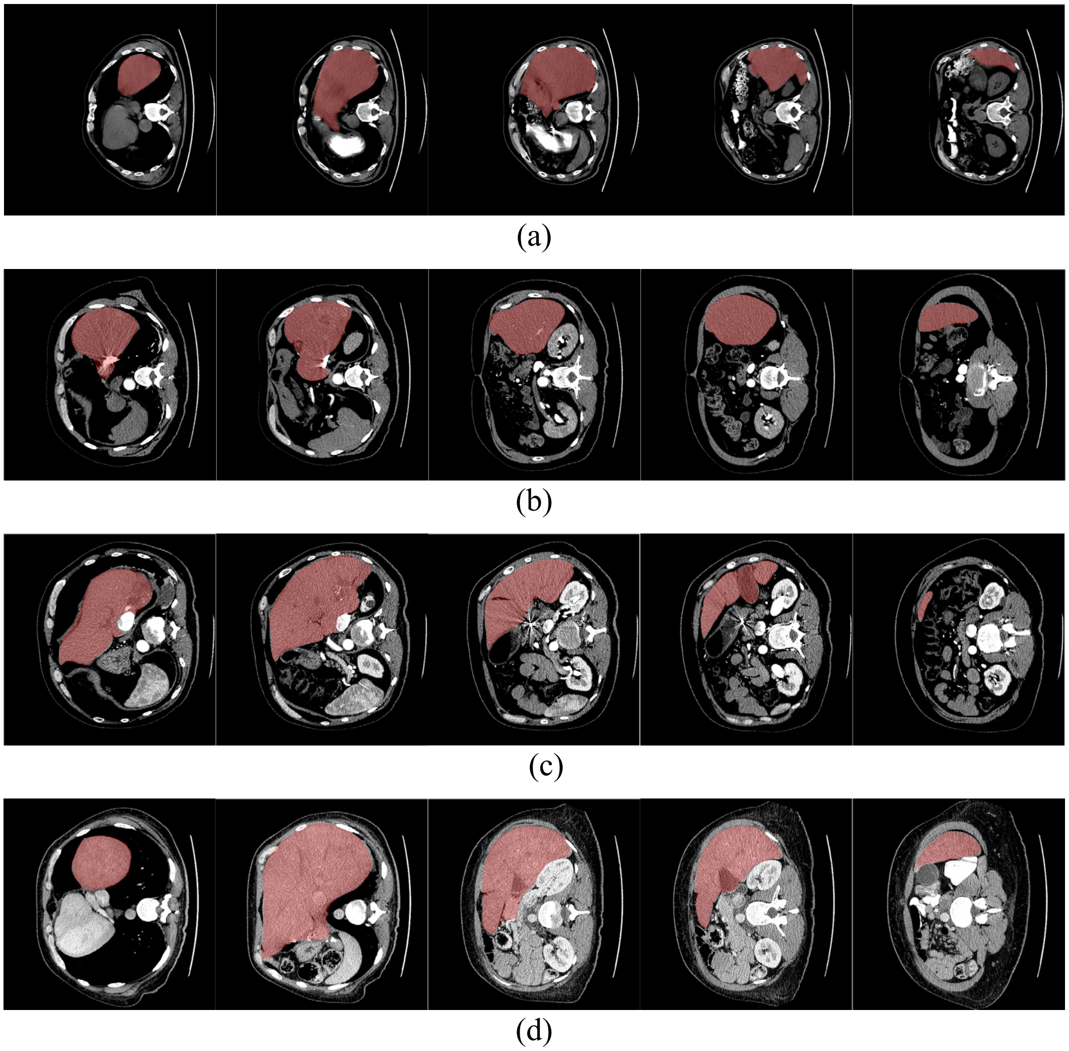
6.1.2 Results for Liver Segmentation algorithm *Ver. 2.0*

Table 6.2 provides comparative results for *Ver. 2.0* of the liver segmentation algorithm. The table provides the volumes calculated as the mean of three different runs as well as the absolute and percent error in the determination of the volume of the liver.

Table 6.2: Comparative results: proposed algorithm *Ver. 2.0* vs. manual gold standard

No	Manual Segmentation Volumes (ml)	Average Calculated Volumes (ml)	Absolute Error (ml)	% Error
1	1458.80	1395.55	63.25	-4.34
2	4079.50	4166.64	87.15	2.14
3	2419.18	2358.41	60.76	-2.51
4	1848.73	1843.13	5.60	-0.30
5	1630.59	1647.81	17.22	1.06
6	1568.82	1441.58	127.23	-8.11
7	1408.27	1341.06	67.20	-4.77
8	1331.81	1300.14	31.67	-2.38
9	2595.61	2577.37	18.24	-0.70
10	2651.20	2675.14	23.94	0.90
11	1631.40	1670.49	39.08	2.40
12	2892.98	2500.54	392.44	-13.57
13	1904.25	1939.50	35.26	1.85
14	2704.79	2827.38	122.59	4.53
15	1602.26	1546.04	56.22	-3.51
16	1565.97	1459.55	106.41	-6.80
17	2408.85	2326.45	82.41	-3.42
18	5336.28	5133.38	202.90	-3.80
19	1102.16	1095.41	6.76	-0.61
20	2363.25	2410.67	47.42	2.01
Average	2225.24	2182.81	79.69	-1.32
Std Dev	1018.79	1003.95	88.19	4.0

Again, as different illustrative purposes, Figure 6.2 shows the liver segmentation results obtained for 4 other datasets to demonstrate the results of the *Ver. 2.0* algorithm.


(a)

(b)

(c)

(d)

Figure 6.2: Illustrative examples of segmentation results for 4 datasets. (a) Subject 1 (b) Subject 19 (c) Subject 7 (d) Subject 16 using the Ver. 2.0 of the liver segmentation algorithm

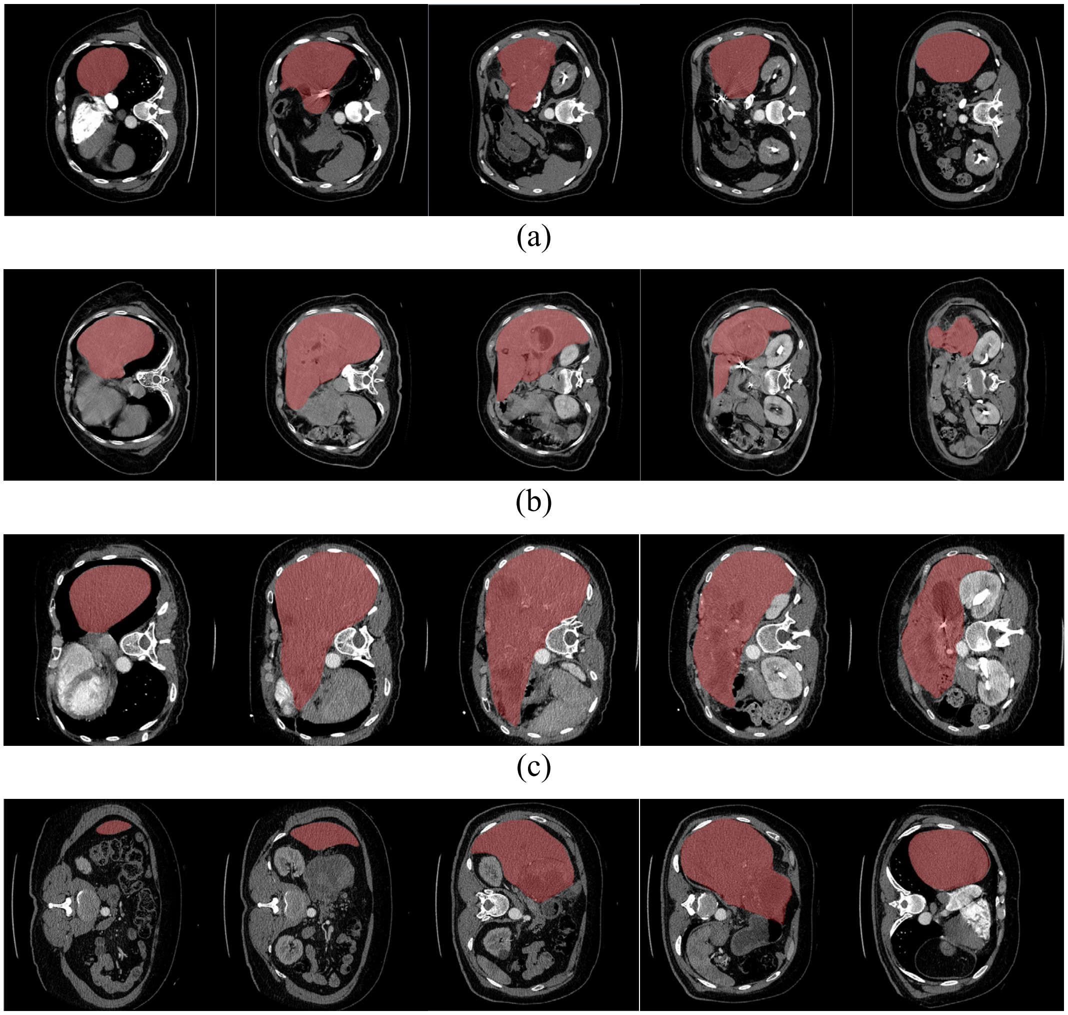
6.1.3. Results for Liver Segmentation algorithm *Ver. 3.0*

Comparative results of volumes obtained by the liver segmentation algorithm *Ver. 3.0* and the manually calculated volumes are provided in Table 6.3.

Table 6.3: Comparative results: proposed algorithm *Ver. 3.0* vs. manual gold standard

No	Manual Segmentation Volumes (ml)	Average Calculated Volumes (ml)	Absolute Error (ml)	% Error
1	1458.80	1426.00	32.80	-2.25
2	4079.50	4180.67	101.17	2.48
3	2419.18	2402.00	17.18	-0.71
4	1848.73	1855.33	6.60	0.36
5	1630.59	1649.33	18.74	1.15
6	1568.82	1448.67	120.15	-7.66
7	1408.27	1358.67	49.60	-3.52
8	1331.81	1318.00	13.81	-1.04
9	2595.61	2637.33	41.72	1.61
10	2651.20	2709.67	58.47	2.21
11	1631.40	1703.33	71.93	4.41
12	2892.98	2626.67	266.31	-9.21
13	1904.25	1966.00	61.75	3.24
14	2704.79	2832.67	127.87	4.73
15	1602.26	1545.67	56.59	-3.53
16	1565.97	1491.67	74.30	-4.74
17	2408.85	2376.00	32.85	-1.36
18	5336.28	5180.33	155.95	-2.92
19	1102.16	1139.33	37.17	3.37
20	2363.25	2466.00	102.75	4.35
Average	2225.24	2215.67	72.39	3.24
Std Dev	1018.79	1010.58	61.46	2.23

In the same fashion as in previous versions of the algorithm, Figure 6.3 shows the segmentation results obtained for 4 other different datasets. Subject 19 is maintained in all 3 versions as a case for comparison for the different versions of the algorithm. The point that can be made here for all three versions of the algorithm is that there is great variation in the intensity, structure, and position of the liver from dataset to dataset. Moreover, some of the datasets may reveal the presence of tumors.


(a)

(b)

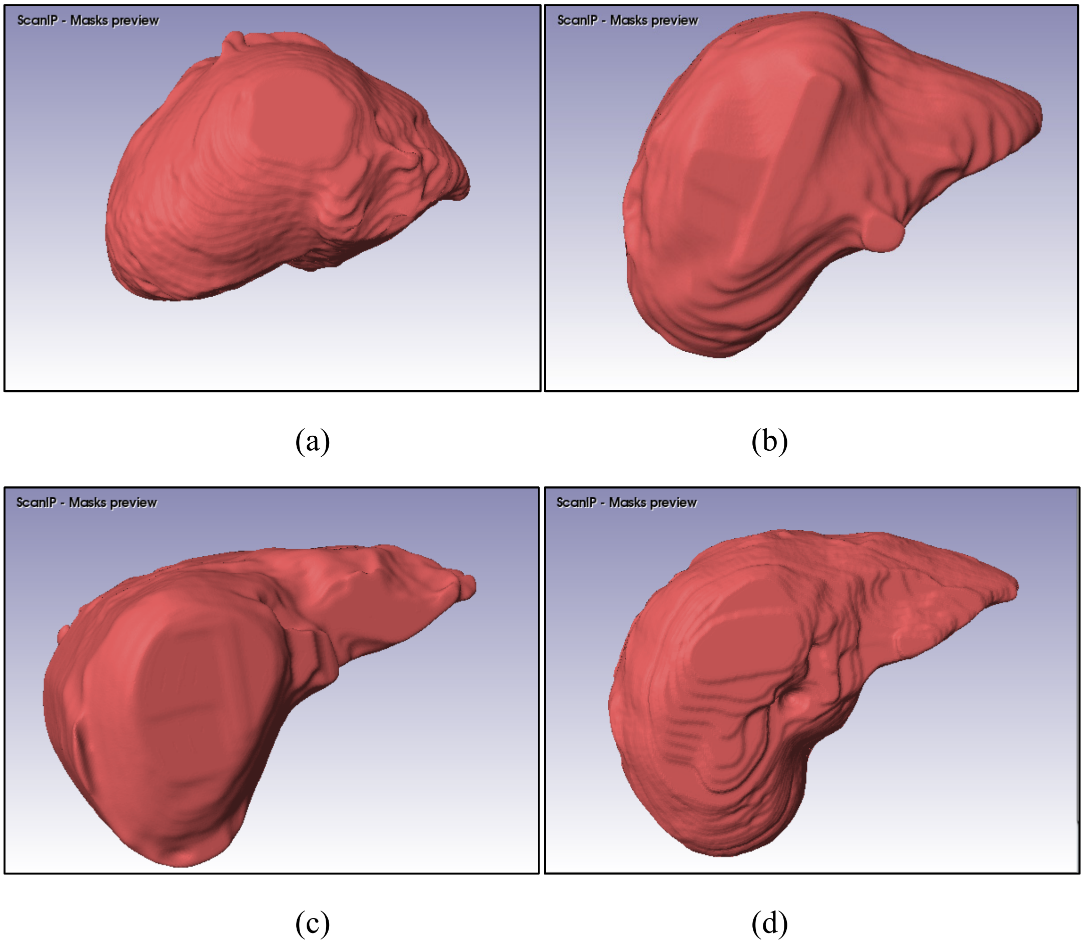
(c)

(d)

Figure 6.3: Illustrative examples of segmentation results for 4 datasets. (a) Subject 19 (b) Subject 8 (c) Subject 12 (d) Subject 18 using the Ver. 3.0 of the liver segmentation algorithm

6.1.4. 3-D Rendering of the Segmented Livers

Results of the 3D rendering process obtained from the ScanIP software are shown in Figure 6.4 for the different datasets. The 3D rendering displayed in Figure 6.4 are for 4 different datasets used in the study for illustrative purposes.



(a) (b)

(c) (d)

Figure 6.4: Illustrative examples of 3D renderings for 4 datasets (a) Subject 19 (b) Subject 8 (c) Subject 12 (d) Subject 18. The 3-D rendering shown are obtained using algorithm Ver 3.0. Similar 3-D rendering can be generated for Ver 1.0 and Ver 2.0 algorithms also.

The renderings shown here have solid surfaces and a mesh finish. However, translucent surfaces can be generated with varying opacities and colors if needed. The need to generate translucent surfaces would be useful to demonstrate the presence of a tumor inside the liver or to display the vasculature of the liver. As demonstrated later in the chapter the 3-D rendering can be displayed along with tumors and blood vessel to obtain a complete idea of the extent and location of the disease.

6.2. Comparative and Statistical Analysis

6.2.1. Comparative Analysis

Table 6.4 displays the comparison between some of the current techniques found in the literature and the proposed technique in terms of the accuracy of the segmentation process and the computational aspects of the proposed algorithm. For the comparison of accuracy, the average volume difference between the calculated and the manual volumes is presented along with the standard deviation for the particular study.

Table 6.4: Comparison of proposed algorithm with other algorithms found in literature

#	Method	Interaction Level	Volume Difference (% Error)	Processing Time per slice (sec)
1	Beck and Aurich(Beck and Aurich, 2007)	*high*	1.8±2.5	3.00
2	Beichel et al.(Beichel et al., 2007)	*high*	1.0±1.7	15.43
3	Chi et al.(Chi et al., 2007)	*none*	2.6±6.3	14.57
4	Dawant et al.(Dawant et al., 2007)	*medium*	2.5±2.3	8.57
5	Furukawa et al.(Furukawa et al., 2007)	*none*	-7.3±4.7	15.43
6	Heimann et al.(Heimann and Meinzer, 2009)	*none*	1.7±3.2	3.00
7	Kainmuller et al.(Kainmüller et al., 2007)	*none*	-2.9±2.9	6.43
8	Lee et al.(Lee et al., 2007)	*low*	1.3±2.9	3.00
9	Rusko et al.(Ruskó et al., 2007)	*none*	-3.8±6.4	0.21
10	Saddi et al.(Saddi et al., 2007)	*none*	1.2±4.4	2.36
11	Schmidt et al.(Schmidt et al., 2007)	*none*	-4.9±3.0	8.57
12	Seghers et al.(Seghers et al., 2007)	*none*	-6.8±2.3	12.86
13	Susomboon et al.(Susomboon et al., 2007)	*none*	-11.5±30	10.71
14	Algorithm *Ver. 1.0 (Goryawala et al., 2011)*	*high*	-1.6±4.0	7.30
15	Algorithm *Ver.2.0*	*medium*	-1.32 ± 4.0	7.50
16	Algorithm *Ver. 3.0*	*low*	3.24±2.23	10.96

Table 6.4 compares the average computational time required to segment a single slice of the dataset obtained for the proposed study to various studies found in the literature. Such a comparison is essential since different algorithms use datasets with different number of slices for the analysis which determines the time needed for processing the entire dataset.

The table also provides the interaction level of each algorithm where an interaction of less than 1 min is considered '*low*', 1 to 5 min is considered '*medium*' and greater than 5 min is considered '*high*'. An interaction level of '*none*' is displayed for automatic algorithms which are usually based on learning techniques and generally provide a lesser accuracy than interactive methods. The proposed algorithm falls in the *low* interaction category since it requires less than a minute to initialize the entire process.

Also, in order to compare the different versions of the algorithms, the percentage errors generated for the 20 different datasets for each version were compared using a 1-way ANOVA. The 1-way ANOVA analysis allows us to determine if a statistical difference is found in the accuracy for the three versions.

Table 6.5 shows the percentage errors obtained by the different versions of the algorithm for the 20 datasets.

The results of the 1-way ANOVA analysis are shown in Table 6.6. The results indicate that the version of the algorithm used in calculating the liver volume is not a significant factor (p =0.96). This suggests that the differences in the accuracy seen for the three versions in Table 6.4 are not statistically significant.

Table 6.5: Percentage volume errors for the 20 datasets for each version of the algorithm

Dataset No.	Percentage Error		
	Ver 1.0	*Ver 2.0*	*Ver 3.0*
1	2.75	4.34	2.25
2	0.50	2.14	2.48
3	1.59	2.51	0.71
4	1.52	0.30	0.36
5	0.38	1.06	1.15
6	9.35	8.11	7.66
7	5.06	4.77	3.52
8	2.33	2.38	1.04
9	2.00	0.70	1.61
10	1.60	0.90	2.21
11	1.99	2.40	4.41
12	10.12	13.57	9.21
13	2.63	1.85	3.24
14	1.20	4.53	4.73
15	6.11	3.51	3.53
16	6.25	6.83	4.74
17	1.51	3.42	1.36
18	4.20	3.80	2.92
19	3.84	0.61	3.37
20	2.02	2.01	4.35

Table 6.6: ANOVA Results for various Users

Source	**Sum of Squares**	**df**	**Mean Square**	**F Value**	**p-value Prob > F**
Version	0.60	2	0.30	0.04	0.96
Error	425.41	57	7.46		
Total	426.01	59			

6.2.2. Statistical Analysis

In order to determine if the user initialization of the segmentation has an impact on the segmentation results, 3 different users were asked to initialize the algorithm. 1-way ANOVA test was again carried out on the errors obtained after the segmentation process was initialized by the 3 users. The errors calculated as the percentage difference between the algorithm calculated volumes and the manual gold standard volumes are shown in Table 6.7 with the ANOVA results shown in Table 6.8.

Table 6.7: Percentage error for various user initializations for *Ver. 1.0* algorithm

Dataset No.	Percentage Error		
	User 1	User 2	User 3
1	2.03	1.83	8.44
2	2.35	2.01	1.15
3	7.76	0.23	3.21
4	4.42	3.50	3.63
5	2.62	1.69	2.07
6	14.74	8.06	5.26
7	3.89	4.73	6.57
8	2.27	1.80	2.93
9	2.51	3.63	4.87
10	3.51	1.38	2.66
11	0.20	6.96	1.19
12	4.12	16.23	10.00
13	2.06	2.21	8.04
14	3.46	3.47	3.61
15	5.44	6.25	6.64
16	6.89	2.00	9.87
17	3.75	2.29	1.51
18	0.29	3.99	8.32
19	5.82	6.71	1.01
20	1.83	0.90	5.14

Table 6.8: ANOVA Results for various Users for *Ver. 1.0* algorithm

Source	Sum of Squares	df	Mean Square	F Value	p-value Prob > F
User	8.8	2	4.4	0.41	0.67
Error	610.5	57	10.7		
Total	619.2	59			

It is seen from the ANOVA analysis that 'Users' is not a significant factor with a p-value of 0.67 (p>0.05). The results of the statistical analysis suggest the consistency of the segmentation algorithm in terms of the users.

Table 6.9 through Table 6.10 provide results of a similar analysis for the *Ver. 2.0* algorithm. As was the case of *Ver. 1.0* algorithm, the *Ver. 2.0* algorithm also demonstrates independence to user initialization (p = 0.46).

Table 6.9: Percentage error for various user initializations for *Ver. 2.0* algorithm

Dataset No.	Percentage Error		
	User 1	User 2	User 3
1	2.09	4.22	10.87
2	2.75	4.40	4.76
3	5.20	1.89	0.45
4	3.29	5.96	1.76
5	5.30	0.78	2.90
6	8.54	11.48	4.31
7	1.94	4.14	8.24
8	5.81	1.25	2.57
9	8.61	1.38	5.12
10	2.02	1.61	3.12
11	5.90	1.33	0.05
12	9.30	16.88	14.51
13	1.69	0.35	4.22

14	1.14	2.29	10.16
15	3.79	0.34	7.08
16	7.01	7.88	5.59
17	5.89	3.01	1.35
18	5.35	3.00	3.06
19	4.77	1.21	5.40
20	1.63	0.97	6.68

Table 6.10: ANOVA Results for various Users for *Ver. 2.0* algorithm

Source	Sum of Squares	df	Mean Square	F Value	p-value Prob > F
User	19.8	2	9.9	0.80	0.46
Error	708.1	57	12.4		
Total	728.0	59			

A similar analysis for *Ver. 3.0* algorithm shown in Table 6.11 through Table 6.12 demonstrate that like its predecessors the *Ver. 3.0* algorithm also demonstrates independence to user initialization (p =0.33).

Table 6.11: Percentage error for various user initializations for *Ver. 3.0* algorithm

Dataset No.	Percentage Error		
	User 1	User 2	User 3
1	1.32	1.43	6.64
2	0.99	2.88	3.57
3	3.81	1.16	2.84
4	2.99	4.96	3.04
5	3.21	1.45	1.68
6	10.89	8.34	3.75
7	0.87	4.92	4.78
8	1.11	0.46	2.46
9	4.15	4.06	4.91

10	1.73	2.52	2.37
11	4.27	7.02	1.94
12	5.81	12.03	9.78
13	0.35	0.88	8.49
14	5.15	1.26	7.77
15	6.01	2.01	2.58
16	3.00	2.74	8.49
17	1.74	2.15	0.20
18	1.67	1.24	5.85
19	5.25	4.98	0.11
20	0.12	4.05	8.88

Table 6.12: ANOVA Results for various Users for *Ver. 3.0* algorithm

	Sum of		Mean	F	p-value
Source	Squares	df	Square	Value	Prob > F
User	18.01	2	9.01	1.13	0.33
Error	455.54	57	7.99		
Total	473.56	59			

6.3. Tumor and blood vessel segmentation

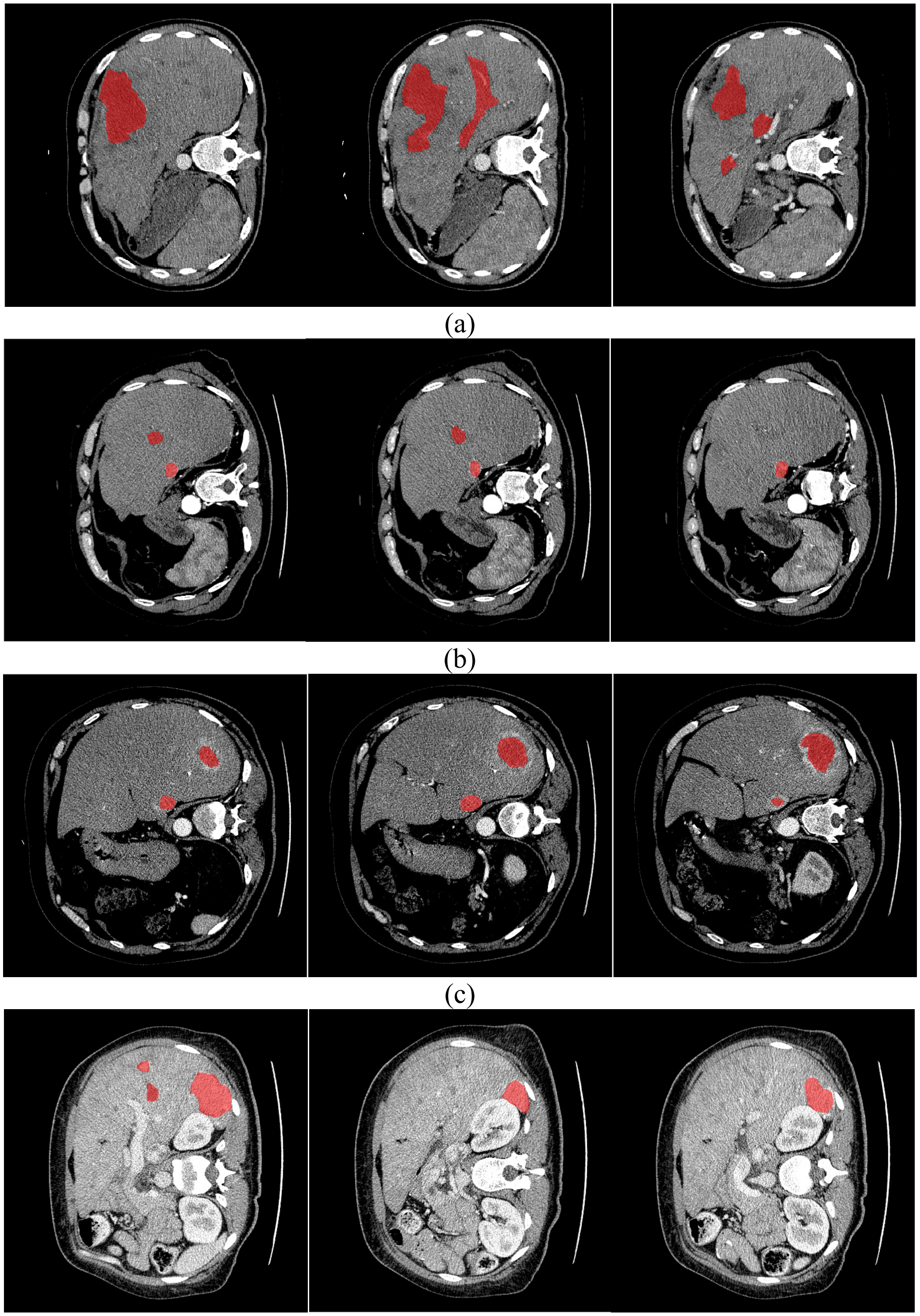
6.3.1. Tumor segmentation

Tumor segmentation was performed on 13 out of the 20 datasets. These 13 datasets are chosen since the manual gold standard tumor volumes calculated by an expert were only available for these 13 datasets. Table 6.13 provides the volumes obtained by the algorithm as compared to the ones provided by the physician.

Table 6.13: Comparative results between the proposed algorithm and manual gold standard for Ver. 1.0 algorithm

No	Manual Segmentation Volumes (ml)	Calculated Volumes (ml)	Absolute Error (ml)	% Error
1	NA	-		
2	NA	-		
3	219	191	28	12.79
4	130	95	35	26.92
5	565	476	89	15.75
6	73	58	15	20.55
7	NA	-		
8	373	252	121	32.44
9	712	523	189	26.54
10	195	173	22	11.28
11	NA	-		
12	381	305	76	19.95
13	NA	-		
14	225	186	39	17.33
15	NA	-		
16	188	154	34	18.09
17	243	201	42	17.28
18	3500	3100	400	11.43
19	7	10	3	-42.86
20	NA	-		

Figure 6.5 shows the results of the segmentation for four different datasets showing the regions of the image that are demarked as the tumor. Figure 6.5(a) – (d) shows the results for four different subjects. It can be seen from Figure 6.5 that the algorithm is capable of extracting tumors from the CT images with a good accuracy. The algorithm works efficiently with both calcified and non-calcified tumors as seen Figure 6.5 (a) and Figure 6.5(b).



(a)

(b)

(c)

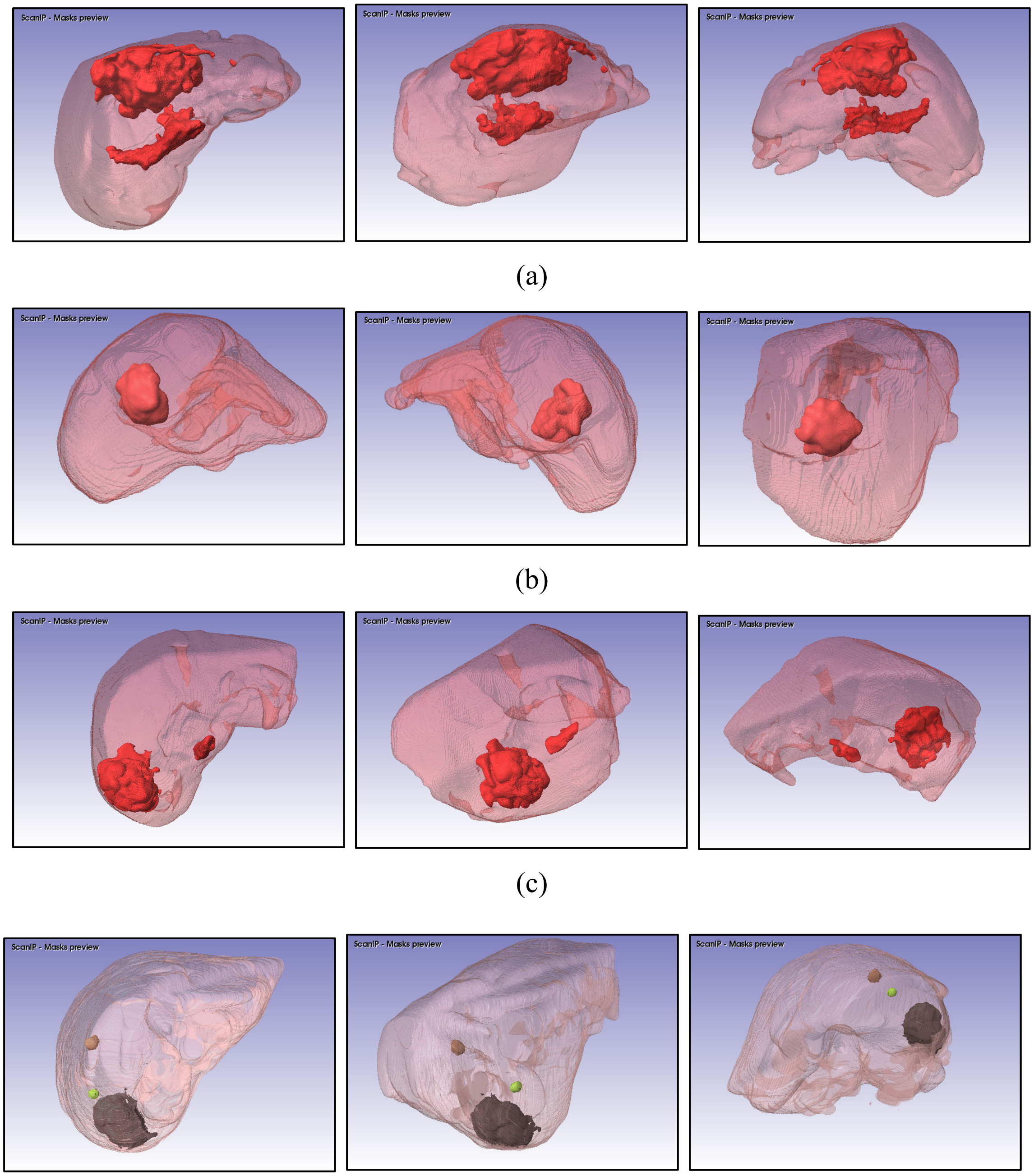
(d)

Figure 6.5: Illustrative examples of tumor segmentation results for 4 datasets. (a) Subject 5 (b) Subject 6 (c) Subject 14 (d) Subject 17

However, the algorithm fails to extract the entire tumor when a calcification is seen as a ring around a tumor mass as can be seen in Figure 6.5(c). The reason is that the algorithm's region growing step is an intensity-based step which excludes the calcified regions when the seed point is selected in the interior non-calcified region of the tumor. Such a difficulty is not seen when the tumor is entirely calcified or entirely non-calcified. The rejection of the calcified rings may justify the reason why the algorithm under-estimates the tumor volumes.

Once the tumors are segmented, ScanIP is used for the visualization of the tumors in 3-D with a translucent liver. Figure 6.6 shows the 3-D rendering results for 4 datasets obtained using the ScanIP software. Note that the software does not make a decision but just renders the segmented mask in a 3-D volume. This gives the opportunity to visualize the size, extent, and location with respect to the liver.

Figure 6.6 shows the different orientation for the 4 datasets to demonstrate the fact that the 3-D analysis makes is possible to better view and understand the geometry and extent of the disease. Also, as seen in Figure 6.6(d) it is possible to allocate different colored masks to each tumor for distinction and also for determination of individual volumes of each tumor.



(a)

(b)

(c)

(d)

Figure 6.6: Illustrative examples of tumor segmentation results for 4 datasets. (a) Subject 5 (b) Subject 8 (c) Subject 14 (d) Subject 17

6.3.2. Blood vessel segmentation

The results for the blood vessels segmentation are shown in Figure 6.7. This is to demonstrate the results of the segmentation process by overlaying a red mask on the CT slices in locations where the algorithm determines the presence of blood vessels.

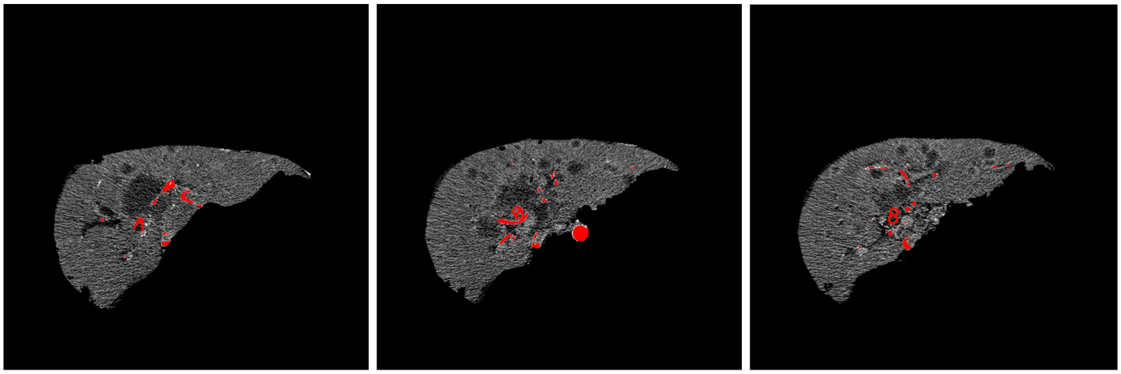
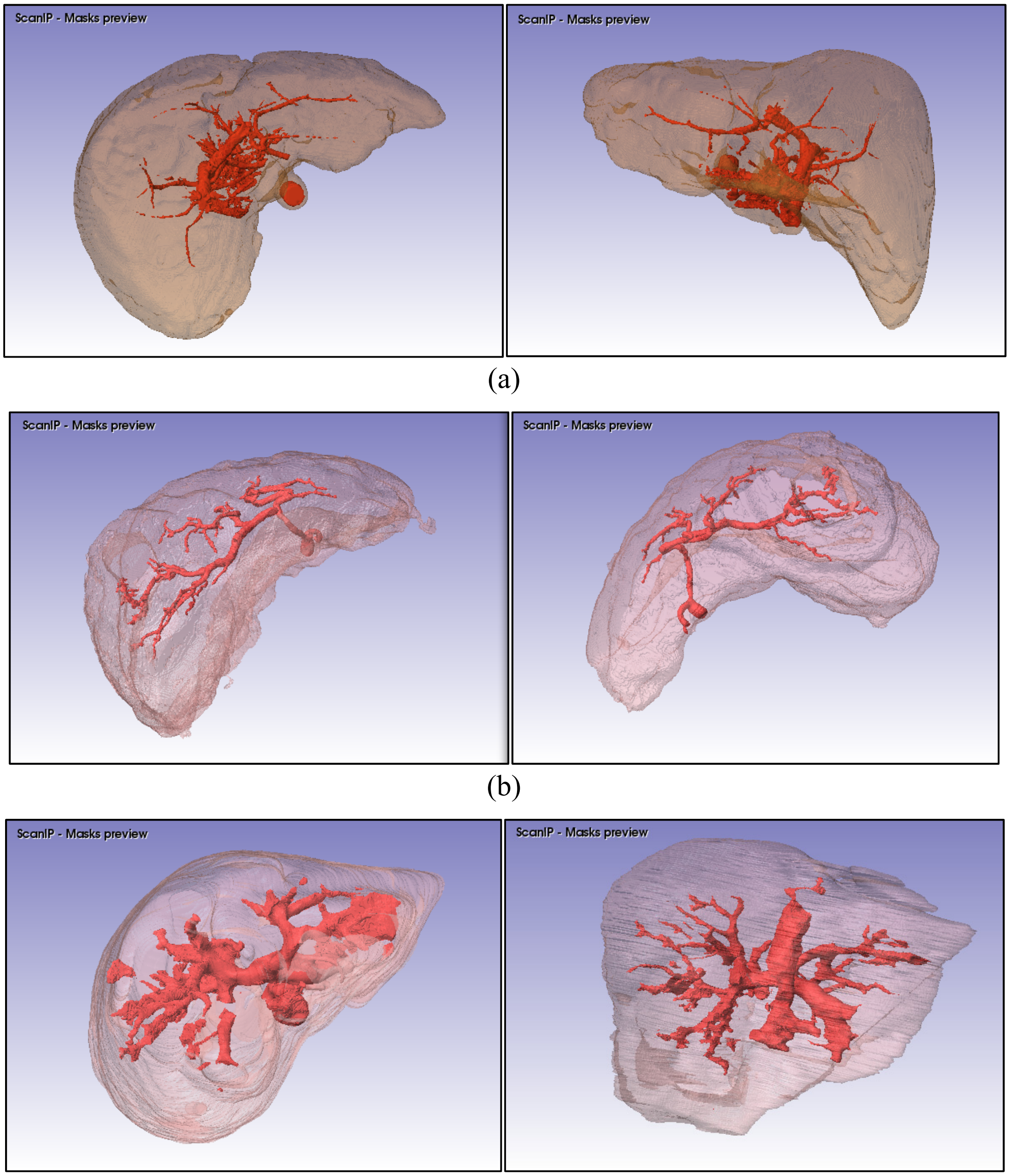


Figure 6.7: Illustrative examples of blood vessels segmentation results for Subject 10

It is seen from Figure 6.7, that it is hard to appreciate the findings of the blood vessel segmentation approach in images that shows spots and dots where the blood vessels are seen in the cross section axial slices. In order to truly appreciate the finding of the algorithm it is required to visualize the results in a 3-D environment in context with the liver. Such results are shown in Figure 6.8 for 2 different dataset in various orientations. It can be observed that the algorithm is capable of extracting even fine branches of the arterial system of the liver as well as the larger arteries. Validation results for vessel segmentation are very hard to achieve since it is generally obtained through corrosion casts that require expensive equipment and is very tedious to obtain (Selle et al., 2002). Another validation technique would be presumably to segment the blood vessels manually from the CT slices. This however would require a lot of time on the part of the expert and also would not yield very good results since blood vessels in cross sectional

CT slices are hard to visualize. The validation of blood vessels is not carried out in this dissertation and is deemed a challenging task.



(a)

(b)

(c)

Figure 6.8: Illustrative examples of blood vessel segmentation results for 3 datasets. (a) Subject 10 (b) Subject 4 (c) Subject 17

On the completion of all the modules of the segmentation process, the final task is to combine the results obtained by the liver, tumor and blood vessel segmentation approaches to produce a complete 3-D object detailing all the features of the liver. Three masks containing the results of the liver, tumor and blood vessel segmentation are fed to the software ScanIP for 3-D rendering. The fusing of the three masks to generate the final 3-D object is illustrated in Figure 6.9.

Figure 6.10 shows the results of the fused renderings of two illustrative datasets. The 3-D rendering are shown from different orientations to display the present tumor(s) and the blood vessels feeding the tumor(s). The knowledge of the anatomy and vessel architecture provides the tools necessary for the surgeon for liver resection surgery planning, living donor liver transplants and for the determination of the insertion point in treatments such as SIRT. The 3-D rendering provides an in-depth analysis aiding the surgeon locate all sources of blood feeding the tumor and any blood vessels which may not show up in 2-D angiograms due to their 3-D location.

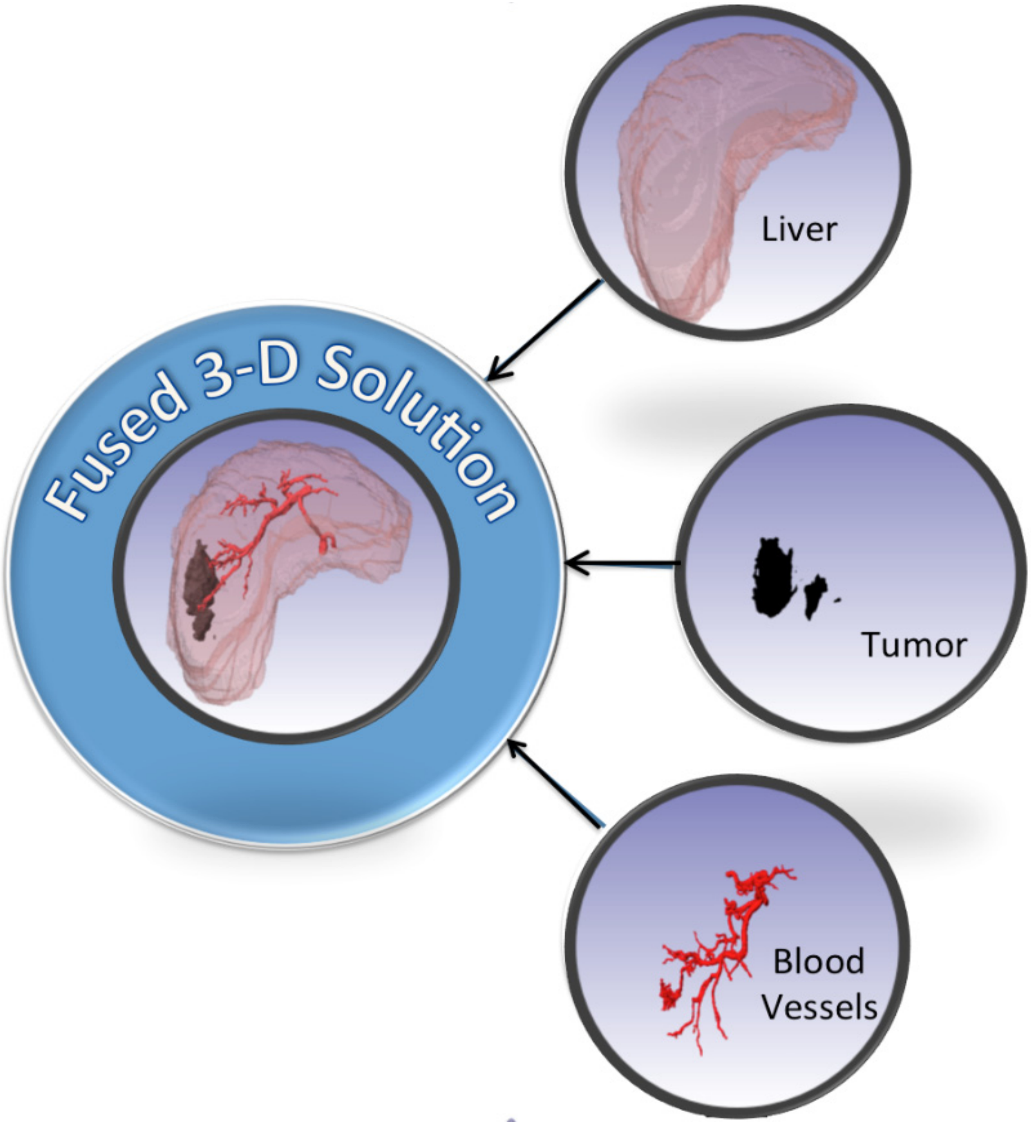
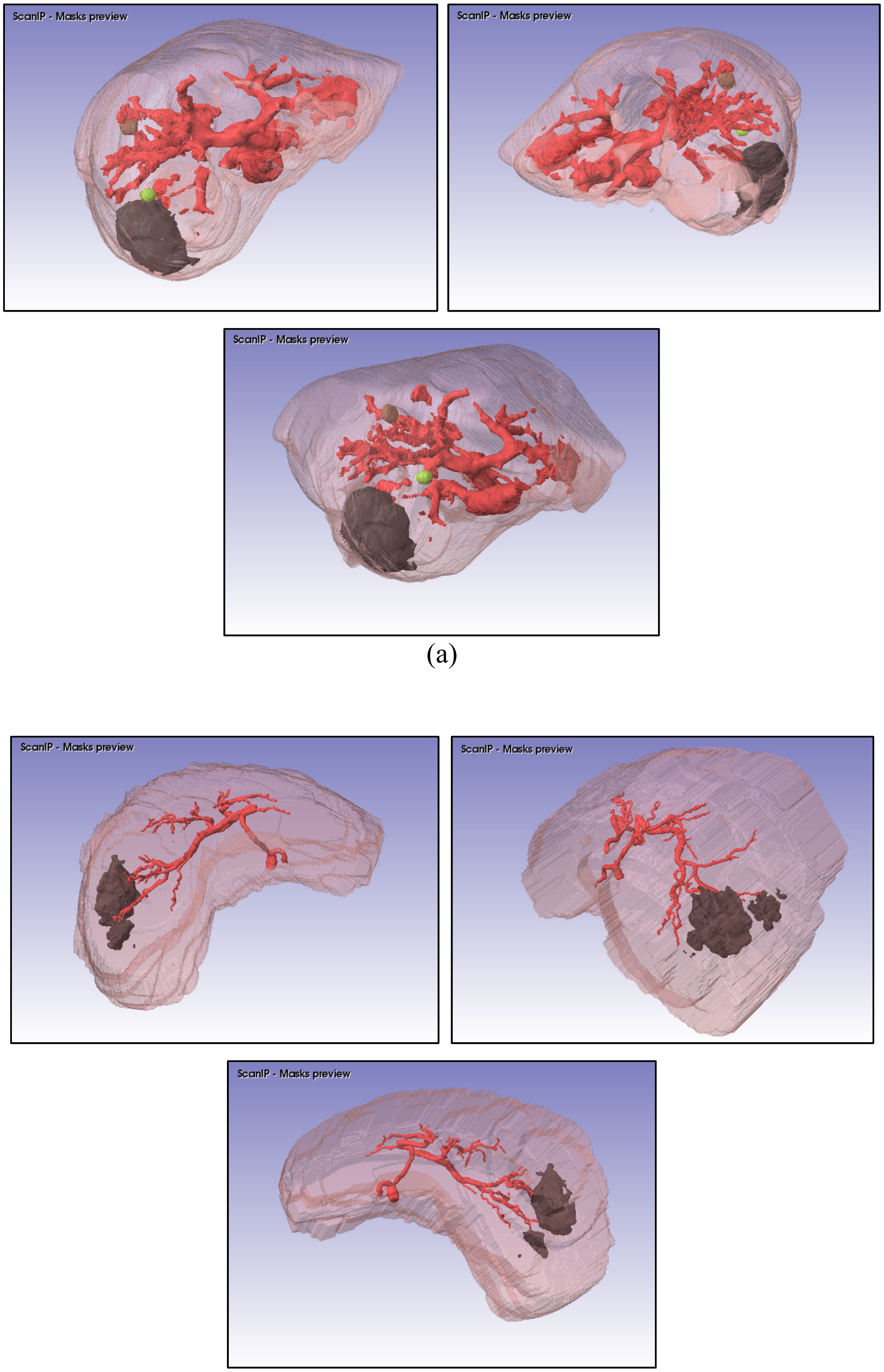


Figure 6.9: Generating the 3-D fused solution containing the liver, tumor(s) and blood vessels for enhanced visualization



(a)

(b)

Figure 6.10: Examples of 3-D fused solution. (a) Subject 4 (b) Subject 17

6.4. Extending the algorithm’s extent towards other applications

The algorithm for the liver segmentation process is modular in nature where some of the sections of the algorithm are capable of being executed as a standalone process. This nature of the algorithm enables it to be adapted for different applications of segmentation other than liver segmentation for which it was designed. This section provides two instances for which a part or a modified version of the algorithm has been applied for segmentation.

The algorithm has been modified for use in two applications we have encountered in the laboratory at the Center for Advanced Technology and Education (CATE) at the Florida International University, Miami, Florida. These two applications are:

1. Thermal imaging as a biometrics approach to facial signature authentication as performed by Ana Guzman, a colleague in the CATE laboratory (Guzman, 2011; Guzman et al., 2012).
2. Segmentation of brain from MR images for calculation of patient specific white and gray matter volumes (Cabrerizo et al., 2011).

What we have demonstrated through these two different additional applications, beyond the merit of the generalized construct of the segmentation algorithm, is its robustness for segmenting other organs and vasculatures.

In the first application using thermal imaging, we have proven that in the case of the facial segmentation process where the intensity distribution of the face is fairly homogenous, the Yezzi energy function (Li and Yezzi, 2007) is better suited than the

Chan-Vese method used for the localized contouring algorithm in the liver which had a heterogeneous intensity distribution. Furthermore, the results show a high accuracy in matching facial signatures as biometric markers.

For the second application, we have shown that an under-estimation of the white matter volumes may result in incorrect 3-D source localization, and where volume differences near 102 ml between the results of the proposed method and the BET (Brain Extraction Tool), which is considered the gold standard in neuroscience (Smith, 2002), can be a significant factor towards the determination of the seizure focus.

CHAPTER VII

DISCUSSION

This discussion centers on the four main topics that make up this dissertation. These are liver segmentation, parallel computing, tumor segmentation and the contextual blood vessel extraction with respect to the tumor location. Understanding the merits of each of these topics allows us to project a potential extension of this fully integrated algorithm towards other biomedical applications.

7.1. Liver Segmentation Algorithm

Results have shown that the liver region is accurately delineated from the heart, spleen and other organs even as they share similar intensities. Very little leakage to the adjoining organs is thus seen due to the use of local properties in the contouring algorithm. This segmentation is independent of the structure, size, position and intensity distribution of the liver region, even in the presence of the challenging fact that the intensity distribution across the liver is not uniform.

Furthermore, the size of the liver is vastly different from patient to patient, especially in the case where one patient had actually undergone an earlier resection. This makes the algorithm reliable when used in post-treatment follow-ups. The liver segmentation algorithm provides highly accurate results for livers of all sizes ranging from 1102 ml to 5336 ml. This is an important feature for any liver segmentation approach to be applied for cancer study and especially SIRT since the size of the liver is greatly affected by the disease. Also, resection of the liver due to surgical procedures leads to smaller size and atypical liver shapes. The algorithm is not affected by these situations. This makes the algorithm robust and applicable for use in post-treatment and follow-ups as well.

It can also be visualized that the span of the liver is different for the datasets under consideration where some liver are located and confined to the right side of the body (see Figure 6.1(a) whereas in some cases the liver spans the entire width of the body (see Figure 6.3(c)). It is seen that the algorithm is capable of extracting the liver correctly with an accuracy of less than 10%, which is considered the clinical error threshold for critical applications like SIRT. This is an advantage of the segmentation algorithm over model-based and learning-based approaches, which fail in situations where such large deviations from 'normal' structures are seen. Also, since the algorithm does not need to be trained like in model-based approaches, independent analysis is reinforced and influence of one dataset over others is removed. Another feature evident from Figure 6.3(d) is that the dataset is rotated such that the patient data is obtained as if the patient was scanned feet first instead of head first. It is seen that since the algorithm is not dependent on prior knowledge of the dataset under consideration, it did not fail under such a circumstance.

Results displayed in Table 6.1 for the *Ver. 1.0* algorithm demonstrate that the percent error in determining liver volume was anywhere in the range of 0.38% to 10.12% for the 20 datasets under consideration with a mean error of only 1.6%. Similar results for the *Ver. 2.0* algorithm are given in Table 6.2 to demonstrate that the improvised initialization step and reduction in the interaction from *Ver. 1.0* to *Ver. 2.0* had little or no effect on the accuracy of the volume determination. The results of the *Ver. 3.0* algorithm shown in Table 6.3 do shows a slight increase in the mean error obtained by the algorithm to around 3.24% but still this is well below the accepted clinical norms. In *Ver. 3.0* it is emphasized that the initialization process relied on a single slice, minimizing to a great extent human intervention in this critical initialization process.

This fact is also evidenced from the statistical analysis performed in section 6.2.1 which demonstrated that the percentage error seen in determining liver volume was strongly independent of the version of the algorithm utilized (p = 0.96). Accurate volume calculation specific to each patient's anatomy would help in the calculation of a more precise dose to deliver to that patient, reducing as a consequence the risks for either excess dosing which may damage healthy surrounding liver tissue or under dosing with a potential for tumor relapsing.

In comparison to other techniques found in the literature, an important feature to note is the low standard deviation attained in the volume calculation, which suggests a higher consistency in the results. The algorithm fairs well with respect to the computational burden imposed by the subtle intricacies of the algorithm itself as well as the amount of imaging data required to process each of the 3-D CT datasets. The algorithm as designed is parallel aware and can be deployed on larger computer clusters if the need arises to reduce significantly the processing time to seconds.

Although the comparison process provided encouraging results, it should be noted that the different studies used for the comparison process used different datasets and hardware systems for the deployment of the algorithm. Also, in addition to the different datasets and hardware configurations the various algorithms listed in the comparison use different programming languages such as C, C++, Java, Visual Basic and MatLab. The developed algorithm was performed on the available 3-D liver datasets and using a system as outlined in section 5.4.3. For a more meaningful comparison it would be essential to test the algorithm on the same datasets using the same hardware configuration preferably programmed with the same language. However such a comparison is difficult to achieve given the critical nature of the data that was available for this study.

The statistical analysis demonstrated that the results of the segmentation were not dependent on user initialization ($p>0.05$). Such a feature is essential as different medical staff may be called to perform the initialization.

Furthermore, the selection of the seed points for the K-means algorithm is an important feature introduced in the algorithm. The modified K-means approach as deployed in this study segments effectively the different regions of the CT slice around the user selected points which act as seeds for each of the aforementioned five masks. The selection of the seeds, rather than a random selection or uniform selection of points in the entire intensity range of the image, yielded much better segmentation results as shown in Figure 3.4 for any given slice. These seed points clearly differentiate the various organs of interest as compared to the other two methods of seed selection where the liver region is not at all visible. Based on empirical results, it was determined that the optimal mask would require a combination of the first two identified regions since in some cases the entire liver is not seen in mask M1 due to the inhomogeneous intensity distribution across the entire liver region. Combining masks M1 and M2 ensures that no part of the liver is missed out.

Although the results presented in the study demonstrated very high accuracies in liver volume calculation, some precautions are required in the presence of truncated datasets. Such truncations are evident in the 3D renderings as illustrated earlier in Figure 6.4 where the dataset has been truncated without the entire liver being imaged. In order to avoid such errors, which result in underestimation of the liver volume, it is advised to validate the integrity of the dataset before applying the liver segmentation algorithm. In situations where the entire liver is not imaged, either the imaging study must be repeated or the physician must be informed of such cases.

Finally, it should be noted that although MatLab provides an excellent environment for research and development of algorithms, their eventual deployment for commercialization would likely rely on object oriented languages like C, C++ or Java.

7.2. Parallel Computing

The results achieved by system 1 on the first experiments led us to believe that Amdahl's law (Hill & Marty, 2008), which is a model for the relationship between the expected speedup of parallelized implementations of an algorithm relative to the serial algorithm, was the explanation of the results obtained in Figure 5.4. However, after running the same algorithm on system 2, we conclude that the effect observed was a mere resource saturation of the system being deployed and not an example of Amdahl’s law.

The result of the statistical analysis of the experiment performed on system 1 is shown in Table 7.1.

Table 7.1: ANOVA Test results for experiments on System 1

Source	Sum of Squares	*df*	Mean Square	*F Value*	*p-value Prob > F
Model	3.51E+05	7	50122.9	221.32	< 0.0001
OS	0.079	1	0.079	3.4E-04	0.9852
Number of *Workers*	12610.02	3	4203.34	18.56	< 0.0001
Size of Image	3.38E+05	3	1.1E+05	497.85	< 0.0001
Residual	19929.56	88	226.47	.	.
Lack of Fit	19882.18	24	828.42	1118.9	< 0.0001
Pure Error	47.39	64	0.74	.	.
Corrected Total	3.71E+05	95		.	.

* Values of "Prob> F" less than 0.0500 indicate model terms are significant.
* *df indicates degrees of freedom.*

The results of the ANOVA test suggest no significance on the factors represented by the operating system, while there is significance on the factors represented by the number of *workers* utilized and the number of images processed. The Model F-value of 221.32 implies that the difference in the speed up is statistically significant and not due to chance. There is only a 0.01% chance that a "Model F-Value" this large could occur due to noise.

Based on these ANOVA test results, it can be concluded that there is no significant difference between the results obtained under Windows Vista and Linux operating systems. However, the difference in the processing loads, expressed as a function of the size of image or the number of images being processed, is viewed as statistically significant.

The results from the FFT-based benchmarking process show a speed up of around 5 when 8 *workers* were employed are in agreement with the experiments carried out in system 2 for the FFT of the images, but are different from the results achieved for the equivalent experiment on system 1, where a modest 70% gain was reached. A similar speed up of around 5 was accomplished when segmenting the liver images with 8 *workers* for each of the three algorithms as shown in section 5.4.3. This suggests that the parallelization process is dependent on the resources available for the hardware platform (available memory, number of cores in the processor) and independent of the processing employed (segmentation or FFT) or the type of image employed (the abdominal CT images used in the experiments). However, we have to be cautious in this reasoning since parallelization of tasks is not possible on all algorithms, so there is an individual level of parallelism that applies to each algorithm. In our experiments, it happens to have the same effect in performance execution. Also, the graph shows a fairly linear rise in the speed up with the number of *workers*, and saturation for higher number of *workers* did not occur for up to 8 *workers*. This is an important feature since it gives us the opportunity to further increase the number of *workers* in the future.

Also, from the comparative results between MAC and Windows operating systems one can observe the following relationship:

$$T_{cMac}(n) = T_{cWindows}(n/2) \qquad (17)$$

where n is the number of *workers* employed for the computational task, as defined earlier, and T_c is as defined earlier in Eq. 16. This outcome is true for every experimental situation that was considered in this study. It is also observed that the computational gains from a single worker to multiple workers (shown by speed up factor in Figure 5.24) for the Mac OS-X system was greater than the Windows based system.

The results of the statistical analysis of the experiment performed on system 2 are shown in Table 7.2. These results suggest that there is significance on the type of operating system used, the number of *workers* utilized, and the number of images processed. The Model F-value of 29.09 implies that the differences in the speed up are statistically significant. It is also noted that there is significant differences not only in terms of load (number of images) and computing resources available (*workers*) but also in terms of the performance of the operating system, with Windows Vista performing better than MAC-OS 10.5.

The time required to complete the image processing tasks after converting them to a C code was monitored with the expectation that it may yield a reduced processing time. The MatLab compiler was used for the conversion of the serial codes to their C counterparts. The details of these results are not included here since no significant improvement in computational time was obtained by this conversion process. The reason for this outcome is that although the generated C code is a standalone version of the MatLab code; it still calls MatLab routines for processing the task. Also, the high level of optimization seen in MatLab's pre-defined functions such as the FFT reduces the chance of any improvement that could be made by developing an in-house C code. Although such improvement is possible, a very high level of optimization would be required to outmatch the processing task obtained by using MatLab.

Table 7.2: ANOVA Test result for experiments in system 2

Source	Sum of Squares	*df*	Mean Square	F Value	*p-value Prob > F
Model	4.3E+05	7	62196.66	29.09	< 0.0001
OS	18019.3	1	18019.3	8.43	0.0047
Numberof *Workers*	79287.5	3	26429.17	12.36	< 0.0001
Size of Image	3.3E+05	3	1.13E+05	52.7	< 0.0001
Residual	1.8E+05	88	2138.28		
Lack of Fit	1.8E+05	24	7838.88	14008.7	< 0.0001
Pure Error	35.81	64	0.56		
Corrected Total	6.2E+05	95			

* Values of "Prob> F" less than 0.0500 indicate model terms are significant.
* *df indicate degrees of freedom.*

As a note of caution, when installing the PCT/DCT in MatLab version R2008, a maximum of 4 local *workers* can be defined. However, the MatLab version R2009 allows defining a maximum of 8 local *workers*. This feature is independent of the hardware used, although the number of usable *workers* depends on the available system resources. In fact, the first set of experiments demonstrated that the hardware limitation establishes constraints in the possible speed up of the application expected. For a system with limited hardware resources the possible gain achieved quickly converged to a plateau.

In retrospect, the contribution here is in developing compute-aware algorithms before investing on large scale distributed system or parallel machines. An important feature about the proposed segmentation algorithm to be noted is that the entire parallelization is carried out on a single desktop computer. This feature allows the algorithm to be portable and adaptable in hospital settings with minimal cost and space requirements. More importantly, the hybrid algorithm as designed displayed extremely good accuracies in calculating the volumes of the different livers.

7.3. Tumor and Blood Vessel Segmentation

The tumor segmentation results shown in section 6.3.1 demonstrate the capability of the algorithm in determining the tumor volumes. It is seen from the results shown in Figure 6.5 that the algorithm is capable of extracting the tumors with good detail, and works well for both calcified and non-calcified tumors. From the results presented in Table 6.13 the algorithm estimated the volume of the tumor(s) from CT images with an accuracy of about 86% for the 13 datasets out of the 20 that had livers with tumors. Although the accuracy is not as high as the liver segmentation approach, this accuracy is comparable to the results obtained from different studies (Abdel-massieh et al., 2010; Choudhary et al., 2008; Moltz et al., 2008; Qi et al., 2008; Xu et al., 2010; Yoav et al., 2008).

Tumor segmentation from CT on the other hand remains an extremely challenging task due to the extensive amount of discrepancies seen among the datasets. Also, in as many as 25% of the cases, tumors are not visible in the CT datasets altogether due to the similar intensity of the tumor and the surrounding liver (Xu et al., 2011). Even when manually performed, experts find it difficult differentiating tumor from background. Studies have shown that about a 17% variability is found in delineating the tumor manually by different experts (Moltz et al., 2011). In Figure 6.5, the segmentation algorithm yielded good results when the intensity distribution of the tumors was fairly uniform. However, in situations where a calcified ring is seen around the tumor, the algorithm fails to extract the ring since the intensity of the calcified ring is much higher than the central region. This is attributed to the intensity based region-growing step, which encompasses pixels only within a 10% deviation of the selected seed point. An increment in the threshold to include the calcified ring is not possible due to the intensity of the surrounding liver tissue. It is seen that the tumor segmentation approach underestimates the volume of the tumor in most of the studied cases. The exclusion of such calcified regions may contribute towards the underestimation of the tumors.

In case of SIRT, the ability to display blood vessels in context with the liver and the tumor(s) in a 3-D space provides a map for the insertion point of the radio-isotope (Gulec and Fong, 2007), but also shows which part of the vasculature is actually feeding the tumor(s) . Such an assessment of liver vasculature is essential in liver transplants cases in order to gauge the branching patterns of all hepatic vessel system and for understanding the underlying vasculature before surgical intervention. From the results displayed in Figure 6.8, the algorithm is able to extract meticulously the vasculature to include not only the larger structures of the hepatic blood vessel system such as the aorta and hepatic artery but also the smaller and finer blood vessels.

Blood vessel segmentation is usually validated using corrosion cast where an agent is inserted in the vasculature of the liver of a corpse followed by corrosion of the liver to obtain the liver vasculature structure. Corrosion cast techniques are very difficult to master as well as expensive due to the need of the specialized equipment, which are available at very few centers in the USA.

Although, the algorithms developed in the study for tumor and blood vessel segmentation are accurate, better segmentation results can be obtained by using the right set of images. Although CT has been established as the backbone imaging technique that is most prominent in oncology programs, it is observed that up to 25% of the liver metastases are often missed in CT scans (Scott et al., 2001; Valls et al., 2001). In contrast to this fact, PET/CT is better than other modalities for the identification of CRLM (Bipat et al., 2005; Mainenti et al., 2010). The use of PET/CT for tumor segmentation and volume calculations would greatly improve the results.

Also, in contrast enhanced CT images it is essential to segment the tumor from images acquired in the correct phase. It is believed that the best contrast for segmenting the tumors is available in the portal or venous phase where the tumor to background contrast ratio is the optimal. In addition to this it should be said that the poor selectivity of CT towards tumor differentiation makes CT a non-ideal imaging choice for tumor segmentation. However, CT has been widely used for tumor/disease staging due to the ease of use and lower costs as compared to PET/CT.

Finally, we demonstrated that the usefulness of the liver segmentation algorithm extends to other applications in biometrics using thermal imaging, and in brain research using Magnetic Resonance Imaging clearly demonstrates the modular nature of the algorithm and its adaptation to other organs and parts of the body (Guzman, 2011).

7.4. Contributions of the study

The following section provides a summary of the contributions provided by the development of the semi-automatic approaches for liver, tumor and blood vessel towards the field of liver cancer imaging.

The dissertation demonstrated the development of a robust and accurate method for the segmentation of the liver from CT images for the purpose of volume calculations. The algorithm was demonstrated to be independent of the dataset-properties namely structure, size, position and intensity distribution of the liver region mainly due to the novel hybrid segmentation method that coupled the k-means algorithm with a newly established contouring method that relied on radio density of the CT images. The experiment results reported in this study display very high accuracies of 98.27% in agreement with manual expert analysis.

Also, the development of the novel initialization method, which aims at reducing human/user interaction, achieved highly accurate results with only a single slice of the dataset initialized by the user, minimizing greatly human intervention as a consequence.

The proposed algorithm is also structured to lend itself to parallel computing, which is viewed as an important characteristic in applications that require heavy computational loads such as in medical imaging. Parallel computing provided henceforth a scalable approach applicable for single workstations poised to serve as a portable method that is cost-effective and highly desirable in clinical settings.

The results obtained on the execution of the parallelization of the liver image segmentation indicates a significant speed up in the performance, achieving more than 500% increase by using 8 MatLab workers on a dual quad-core machine.

SIRT relies on the calculation of the tumor to liver volume ratio for the calculation of the radioactive dose to the patient. The novel technique for tumor segmentation outlined in the dissertation achieved highly accurate results in segmenting the tumor(s) from the CT scans. Also, the algorithm's capability to extract the tumor(s) using very little human interaction serves the purpose as a fast yet reliable technique for tumor(s) segmentation. Finally, the blood vessel segmentation approach delineated in the dissertation provided a means to segment the entire blood vessel architecture of the liver with the initialization of a single point of the vasculature.

The successful implementation of the various algorithms in the study enables the physician to obtain a complete picture of the disease under investigation and to plan the best treatment for the disease. Also, accurate volume calculation specific to each patient's anatomy would help in the calculation of the absolute precise dose to deliver to that patient. This would reduce the risk for excess dosing which may damage healthy surrounding liver tissue or reduce the risk of under dosing and the potential relapsing of the tumor.

CHAPTER VIII

CONCLUSIONS AND FUTURE WORK

The study demonstrated the development of a robust and accurate liver segmentation technique requiring minimal human interaction. This was achieved by going through progressive research steps that led to the ultimate algorithm that relied solely on a single initialized slice and two seed points manually selected from the first and last slices. The algorithm demonstrated a very high accuracy in determining liver volumes, which demonstrate the effectiveness of the segmentation process. Statistical analyses have also proven consistency in the results and independence on user initialization during the segmentation process.

Moreover, the tumor segmentation algorithm led to volume estimations of the tumor(s) with good accuracy and comparable to those found in the literature while minimizing human intervention in delineating manually the tumor borders in every slice, requiring instead a single point in a single slice for the entire tumor. Together with the developed blood vessel segmentation approach the liver and tumor segmentation algorithm provide a complete solution towards pre-surgical evaluation.

The proposed algorithm is also structured to lend itself to parallel computing, which is viewed as an important characteristic in applications that require heavy computational loads such as in medical imaging. The multicore parallel processing deployed on a single workstation with a dual quad core processor demonstrated a speed up of around 5 for the segmentation.

Since calculation of the radioactive dose to be delivered to the patient relies on the tumor to volume of liver ratio, accurate calculation of the liver volume helps significantly in the accurate determination of the radioactive dosage to the patient. Calculation of accurate radioactive dose has the merits of avoiding under-dosing resulting in recurrence of tumor or over-dosing resulting in damage to healthy tissue.

The future of algorithm development for cancer therapy planning is rooted in the use of multiple modalities for the determination of the anatomical and functional extent of the disease. Development of techniques and algorithms to fuse information from PET, MRI and CT is of prime importance.

The next steps towards the enhancement of the developed algorithm will be to devise techniques to reduce the interaction of the liver segmentation algorithm to make it fully automatic. This can be achieved by using image-processing techniques to estimate the largest slice for *Ver. 3.0* algorithm or develop a completely different technique. Irrespective of the techniques used reduction in accuracy is not an option.

Further validation of the tumor segmentation algorithm with more datasets and with greater number of experts is required for a complete quantification of the approach. Also, as it has been repeatedly said the validation of blood vessels remains a challenge for the future.

Finally, in order to demonstrate the advantages of the complete 3-D solution and the novel dosimetry, clinical trials will be of utmost importance in order to truly gauge the merits of the proposed method to conventional methods as currently performed in clinical settings.

References

Abdel-massieh NH, Hadhoud MM, Amin KM. (2010): Fully automatic liver tumor segmentation from abdominal CT scans. 2010 International Conference on Computer Engineering and Systems (ICCES):197-202.

Adam R, Akpinar E, Johann M, Kunstlinger F, Majno P, Bismuth H. (1997): Place of cryosurgery in the treatment of malignant liver tumors. Ann Surg 225(1):39-8; discussion 48-50.

Amos SM, Duong CP, Westwood JA, Ritchie DS, Junghans RP, Darcy PK, Kershaw MH. (2011): Autoimmunity associated with immunotherapy of cancer. Blood 118(3):499-509.

Arce G, McLoughlin M. (1987): Theoretical analysis of the max/Median filter. IEEE Transactions on Acoustics, Speech and Signal Processing 35(1):60-69.

Ariel IM, Padula G. (1978): Treatment of symptomatic metastatic cancer to the liver from primary colon and rectal cancer by the intra-arterial administration of chemotherapy and radioactive isotopes. Prog Clin Cancer 7:247-54.

Ariel IM, Padula G. (1982): Treatment of asymptomatic metastatic cancer to the liver from primary colon and rectal-cancer by the intra-arterial administration of chemotherapy and radioactive isotopes. Journal of Surgical Oncology 20(3):151-156.

Armstrong JG, Anderson LL, Harrison LB. (1994): Treatment of liver metastases from colorectal cancer with radioactive implants. Cancer 73(7):1800-4.

Aylward S, Bullitt E, Pizer S, Eberly D. 1996. Intensity ridge and widths for tubular object segmentation and description. Mathematical Methods in Biomedical Image Analysis, 1996., Proceedings of the Workshop on. p 131-138.

Ballantyne GH, Quin J. (1993): Surgical treatment of liver metastases in patients with colorectal cancer. Cancer 71(12 Suppl):4252-66.

Barot T, Suthar R, Bhatt R, Goryawala M, McGoron A, Gulec S. (2010): Segmental liver dosimetry integrated comprehensive treatment planning for Y-90 microsphere selective internal radiation therapy in liver cancer. J NUCL MED 51(Supplement 2)(2_MeetingAbstracts):12.

Beck A, Aurich V. (2007): HepaTux-a semiautomatic liver segmentation system. Proc. MICCAI Workshop on 3-D Segmentat. Clinic: A Grand Challenge:225-233.

Beichel R, Bauer C, Bornik A, Sorantin E, Bischof H. (2007): Liver segmentation in CT data: A segmentation refinement approach. Proc. MICCAI Workshop on 3-D Segmentat. Clinic: A Grand Challenge:235-245.

Ben S, Azaiz O, Chaieb F, M'Hiri S, Ghorbel F. (2010): Segmentation of liver tumor using HMRF-EM algorithm with Bootstrap resampling. 5th International Symposium on I/V Communications and Mobile Network (ISVC):1-4.

Ben-Dan I, Shenhav E. (2008): Liver Tumor segmentation in CT images using probabilistic. The MIDAS Journal - Grand Challenge Liver Tumor Segmentation (2008 MICCAI Workshop).1-6.

Bennett GL, Petersein A, Mayo-Smith WW, Hahn PF, Schima W, Saini S. (2000): Addition of gadolinium chelates to heavily T2-weighted MR imaging: limited role in differentiating hepatic hemangiomas from metastases. AJR Am J Roentgenol 174(2):477-85.

Bernatik T, Strobel D, Hahn EG, Becker D. (2001): Detection of liver metastases: comparison of contrast-enhanced wide-band harmonic imaging with conventional ultrasonography. J Ultrasound Med 20(5):509-15.

Bipat S, van Leeuwen MS, Comans EF, Pijl ME, Bossuyt PM, Zwinderman AH, Stoker J. (2005): Colorectal liver metastases: CT, MR imaging, and PET for diagnosis--meta-analysis. Radiology 237(1):123-31.

Bismuth H, Adam R, Levi F, Farabos C, Waechter F, Castaing D, Majno P, Engerran L. (1996): Resection of nonresectable liver metastases from colorectal cancer after neoadjuvant chemotherapy. Ann Surg 224(4):509-20; discussion 520-2.

Bister M, Yap C, Ng K, CH. T. (2007): Increasing the speed of medical image processing in MatLab®. Biomed Imaging Interv J 3(1):e9.

Bliss NT, Kepner J. (2007): pMATLAB parallel MATLAB library. International Journal of High Performance Computing Applications 21(3):336-359.

Blyth S, Blakeborough A, Peterson M, Cameron IC, Majeed AW. (2008): Sensitivity of magnetic resonance imaging in the detection of colorectal liver metastases. Ann R Coll Surg Engl 90(1):25-8.

Cai GX, Cai SJ. (2012): Multi-modality treatment of colorectal liver metastases. World J Gastroenterol 18(1):16-24.

Campadelli P, Casiraghi E, Esposito A. (2009): Liver segmentation from computed tomography scans: A survey and a new algorithm. Artif. Intell. Med. 45(2-3):185-196.

Chakravarti A, Grad-Freilich S, Laure E, Jouvin M, Philippon G, Loomis C, Floros E. 2008. Enhancing e-Infrastructures with Advanced Technical Computing: Parallel MATLAB® on the Grid. Parallel MATLAB® on the Grid. EGEE- Transactions. p TR-2008-001.

Chan TE, Sandberg BY, Vese LA. (2000): Active contours without edges for vector-valued images. Journal of Visual Communication and Image Representation 11(2):130-141.

Chi Y, Cashman PMM, Bello F, Kitney RI. 2007. A discussion on the evaluation of a new automatic liver volume segmentation method for specified CT image datasets. Proc. MICCAI Workshop on 3-D Segmentat. Clinic: A Grand Challenge.

Choudhary A, Moretto N, Pizzorni Ferrarese F, Zamboni G. (2008): An entropy based multi-thresholding method for semi-automatic segmentation of liver tumors. The MIDAS Journal - Grand Challenge Liver Tumor Segmentation (2008 MICCAI Workshop).1-8.

Choy R, Edelman A. (2005): Parallel MATLAB: Doing it right. Proceedings of the IEEE 93(2):331-341.

Choy R, Edelman A, Gilbert JR, Shah V, Cheng D. 2004. Star-P: High productivity parallel computing. Eighth Annual Workshop on High-Performance Embedded Computing (HPEC 04).

Dawant BM, Li R, Lennon B, Li S. (2007): Semi-automatic segmentation of the liver and its evaluation on the MICCAI 2007 grand challenge data set. Proc. MICCAI Workshop on 3-D Segmentat. Clinic: A Grand Challenge:215-221.

Deng XL, Hong TS, Jiang XH, Li Z. (2010): A NEW FRAMEWORK OF UNSUPERVISED IMAGE SEGMENTATION WITH WEIGHT OPTIMIZATION. International Journal of Innovative Computing Information and Control 6(7):3289-3297.

Edelman A. 2007. The Star-P High Performance Computing Platform. IEEE International Conference on Acoustics, Speech and Signal Processing, 2007. Honolulu, HI. p IV 1197- IV 1200.

Elias D, Farace F, Triebel F, Hattchouel JM, Pignon JP, Lecesne A, Rougier P, Lasser P, Duvillard P, Escudier B. (1995): Phase I-II randomized study on prehepatectomy recombinant interleukin-2 immunotherapy in patients with metastatic carcinoma of the colon and rectum. J Am Coll Surg 181(4):303-10.

Elizondo G, Fretz CJ, Stark DD, Rocklage SM, Quay SC, Worah D, Tsang YM, Chen MC, Ferrucci JT. (1991): Preclinical evaluation of MnDPDP: new paramagnetic hepatobiliary contrast agent for MR imaging. Radiology 178(1):73-8.

Frei E, 3rd, Canellos GP. (1980): Dose: a critical factor in cancer chemotherapy. Am J Med 69(4):585-94.

Furukawa D, Shimizu A, H. K. 2007. Automatic liver segmentation based on maximum a posterior probability estimation and level set method. Proc. MICCAI Workshop on 3-D Segmentat. Clinic: A Grand Challenge. p 117.

Geoghegan JG, Scheele J. (1999): Treatment of colorectal liver metastases. Br J Surg 86(2):158-69.

Gray B, Van Hazel G, Hope M, Burton M, Moroz P, Anderson J, Gebski V. (2001): Randomised trial of SIR-Spheres® plus chemotherapy vs. chemotherapy alone for treating patients with liver metastases from primary large bowel cancer. Annals of Oncology 12(12):1711-1720.

Gray BN, Burton MA, Kelleher D, Klemp P, Matz L. (1990): Tolerance of the liver to the effects of Yttrium-90 radiation. Int J Radiat Oncol Biol Phys 18(3):619-23.

Gulec SA. (2007): A surgical perspective on positron emission tomography. J Surg Oncol 95(6):443-6.

Gulec SA, Fong Y. (2007): Yttrium 90 microsphere selective internal radiation treatment of hepatic colorectal metastases. Arch Surg 142(7):675-82.

Gulec SA, Mesoloras G, Stabin M. (2006): Dosimetric techniques in 90Y-microsphere therapy of liver cancer: The MIRD equations for dose calculations. J NUCL MED 47(7):1209-11.

Gulec SA, Sztejnberg ML, Siegel JA, Jevremovic T, Stabin M. (2010): Hepatic structural dosimetry in (90)Y microsphere treatment: a Monte Carlo modeling approach based on lobular microanatomy. J NUCL MED 51(2):301-10.

Guzman A. 2011. Thermal imaging as a biometrics approach to facial signature authentication. Miami: Florida International University.

Guzman AM, Goryawala M, Adjouadi M. 2012. Generating thermal facial signatures using thermal infrared images. Emerging Signal Processing Applications (ESPA), 2012 IEEE International Conference on. p 21-24.

Hardwigsen J, Baque P, Crespy B, Moutardier V, Delpero JR, Le Treut YP. (2001): Resection of the inferior vena cava for neoplasms with or without prosthetic replacement: a 14-patient series. Ann Surg 233(2):242-9.

Heimann T, Meinzer H-P. (2009): Statistical shape models for 3D medical image segmentation: A review. Medical Image Analysis 13(4):543-563.

Heimann T, van Ginneken B, Styner MA, Arzhaeva Y, Aurich V, Bauer C, Beck A, Becker C, Beichel R, Bekes G and others. (2009): Comparison and Evaluation of Methods for Liver Segmentation From CT Datasets. IEEE Transactions on Medical Imaging 28(8):1251-1265.

Higgins WE, Spyra WJT, Ritman EL. 1989. Automatic extraction of the arterial tree from 3-D angiograms. Engineering in Medicine and Biology Society, 1989. Images of the Twenty-First Century., Proceedings of the Annual International Conference of the IEEE Engineering in. p 563-564 vol.2.

Hiraoka A, Kume M, Miyagawa M, Tazuya N, Ichiryu M, Ochi H, Tanabe A, Nakahara H, Shinbata Y, Kan M and others. (2010): Diagnostic value of sonazoid for hepatic metastasis: comparison with FDG PET/CT. Hepatogastroenterology 57(102-103):1237-40.

Ho CM, Lee PH, Shau WY, Ho MC, Wu YM, Hu RH. (2012): Survival in patients with recurrent hepatocellular carcinoma after primary hepatectomy: Comparative effectiveness of treatment modalities. Surgery.

Hudak DE, Ludban N, Krishnamurthy A, Gadepally V, Samsi S, Nehrbass J. (2009): A computational science IDE for HPC systems: design and applications. Int. J. Parallel Program. 37(1):91-105.

Hughes KS, Simon R, Songhorabodi S, Adson MA, Ilstrup DM, Fortner JG, Maclean BJ, Foster JH, Daly JM, Fitzherbert D and others. (1986): Resection of the liver for colorectal carcinoma metastases: a multi-institutional study of patterns of recurrence. Surgery 100(2):278-84.

Hui L, Qiang L, Acharya RS, Gaborski R. 2000. Robust snake model. Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on. p 452-457 vol.1.

Jaffar MA, Iqbal A, Hussain A, Baig R, Mirza AM. (2011): GENETIC FUZZY BASED AUTOMATIC LUNGS SEGMENTATION FROM CT SCANS IMAGES. International Journal of Innovative Computing Information and Control 7(4):1875-1890.

Jimenez-Carretero D, Fernandez-de-Manuel L, Pascau J, Tellado JM, Ramon E, Desco M, Santos A, Ledesma-Carbayo MJ. 2011. Optimal multiresolution 3D level-set method for liver segmentation incorporating local curvature constraints. Engineering in Medicine and Biology Society,EMBC, 2011 Annual International Conference of the IEEE. p 3419-3422.

Jingran W, Xiaoyan Z, Ye X, Zuofeng L, Lei L. 2009. Comparison of AdaBoost and logistic regression for detecting colorectal cancer patients with synchronous liver metastasis. Biomedical and Pharmaceutical Engineering, 2009. ICBPE '09. International Conference on. p 1-6.

Kainmüller D, Lange T, Lamecker H. 2007. Shape constrained automatic segmentation of the liver based on a heuristic intensity model. Proc. MICCAI Workshop on 3-D Segmentat. Clinic: A Grand Challenge. p 109.

Kamel IR, Georgiades C, Fishman EK. (2003): Incremental value of advanced image processing of multislice computed tomography data in the evaluation of hypervascular liver lesions. J Comput Assist Tomogr 27(4):652-6.

Kanungo T, Mount DM, Netanyahu NS, Piatko CD, Silverman R, Wu AY. (2002): An efficient k-means clustering algorithm: Analysis and implementation. IEEE Transactions on Pattern Analysis and Machine Intelligence 24(7):881-892.

Kawata Y, Niki N, Kumazaki T. 1996. Feature extraction of convex surfaces on blood vessels using cone-beam CT images. Image Processing, 1996. Proceedings., International Conference on. p 315-318 vol.3.

Kinkel K, Lu Y, Both M, Warren RS, Thoeni RF. (2002): Detection of hepatic metastases from cancers of the gastrointestinal tract by using noninvasive imaging methods (US, CT, MR imaging, PET): a meta-analysis. Radiology 224(3):748-56.

Kirbas C, Quek FKH. (2003): Vessel extraction techniques and algorithms: a survey. Third IEEE Symposium on Bioinformatics and Bioengineering, 2003. Proceedings.:238-245.

Kobashi S, Hata Y. (2010): LUNG LOBAR SEGMENTATION USING TUBULAR TISSUE DENSITY FROM MULTIDETECTOR-ROW CT IMAGES. International Journal of Innovative Computing Information and Control 6(3A):829-842.

Kopp AF, Heuschmid M, Claussen CD. (2002): Multidetector helical CT of the liver for tumor detection and characterization. Eur Radiol 12(4):745-52.

Krishnamurthy A, Nehrbass J, Chaves JC, Samsi S. 2007. Survey of Parallel MATLAB Techniques and Applications to Signal and Image Processing. IEEE International Conference on Acoustics, Speech and Signal Processing.

Kulinna C, Helmberger T, Kessler M, Reiser M. (2001): [Improvement in diagnosis of liver metastases with the multi-detector CT]. Radiologe 41(1):16-23.

Lankton S, Tannenbaum A. (2008): Localizing Region-Based Active Contours. IEEE Transactions on Image Processing 17(11):2029-2039.

Lau WY, Ho S, Leung TW, Chan M, Ho R, Johnson PJ, Li AK. (1998): Selective internal radiation therapy for nonresectable hepatocellular carcinoma with intraarterial infusion of 90yttrium microspheres. Int J Radiat Oncol Biol Phys 40(3):583-92.

Lau WY, Lai EC, Lau SH. (2010): Methods of vascular control technique during liver resection: a comprehensive review. Hepatobiliary Pancreat Dis Int 9(5):473-81.

Lecornu L, Toumoulin C, Roux C, Senhadji L. 1992. Using SVD and analyzing wavelets for the analysis and the segmentation of digital subtraction angiography images. Computers in Cardiology 1992. Proceedings. p 655-658.

Lee J, Kim N, Lee H, Seo JB, Won HJ, Shin YM, Shin YG. (2007): Efficient liver segmentation exploiting level-set speed images with 2.5D shape propagation. Proc. MICCAI Workshop on 3-D Segmentat. Clinic: A Grand Challenge:189-196.

Li H, Yezzi A. (2007): Local or global minima: Flexible dual-front active contours. IEEE Transactions on Pattern Analysis and Machine Intelligence 29(1):1-14.

Lim S-J, Jeong Y-Y, Ho Y-S. 2005. Segmentation of the Liver Using the Deformable Contour Method on CT Images. In: Ho Y-S, Kim H, editors. Advances in Multimedia Information Processing - PCM 2005: Springer Berlin / Heidelberg. p 570-581.

Liu F, Zhao BS, Kijewski PK, Wang L, Schwartz LH. (2005): Liver segmentation for CT images using GVF snake. Medical Physics 32(12):3699-3706.

Maeda S, Kim H, Itai Y, Tan JK, Ishikawa S, Yamamoto A. (2011): REDUCTION OF PROCESSING TIMES FOR TEMPORAL SUBTRACTION ON LUNG CT IMAGE EMPLOYING OCTREE ALGORITHMS. International Journal of Innovative Computing Information and Control 7(5B):2603-2610.

Mainenti PP, Mancini M, Mainolfi C, Camera L, Maurea S, Manchia A, Tanga M, Persico F, Addeo P, D'Antonio D and others. (2010): Detection of colo-rectal liver metastases: prospective comparison of contrast enhanced US, multidetector

CT, PET/CT, and 1.5 Tesla MR with extracellular and reticulo-endothelial cell specific contrast agents. Abdom Imaging 35(5):511-21.

Malladi R, Kimmel R, Adalsteinsson D, Sapiro G, Caselles V, Sethian JA. 1996. A geometric approach to segmentation and analysis of 3D medical images. Mathematical Methods in Biomedical Image Analysis, 1996., Proceedings of the Workshop on. p 244-252.

Malladi R, Sethian JA. 1996. Level set and fast marching methods in image processing and computer vision. Image Processing, 1996. Proceedings., International Conference on. p 489-492 vol.1.

Massoptier L, Casciaro S. (2008): A new fully automatic and robust algorithm for fast segmentation of liver tissue and tumors from CT scans. European Radiology 18(8):1658-1665.

Mathworks I. 2009a. Distributed Computing Toolbox, Users Guide. The MathWorks, Inc.,: Natick, MA,.

Mathworks I. 2009b. MATLAB Distributed Computing Engine. The MathWorks, Inc.,: Natick, MA,.

Mathworks I. 2009c. Parallel Computing Toolbox, Users Guide. The MathWorks, Inc.,: Natick, MA,.

Miles KA, Hayball MP, Dixon AK. (1993): Functional images of hepatic perfusion obtained with dynamic CT. Radiology 188(2):405-11.

Mirman I. (2006): Going Parallel the New Way Desktop Engineering 11(10):24-25.

Moltz J, Bornemann L, Dicken V, Peitgen H. (2008): Segmentation of Liver Metastases in CT Scans by Adaptive Thresholding and Morphological Processing. The MIDAS Journal - Grand Challenge Liver Tumor Segmentation (2008 MICCAI Workshop).1-10.

Moltz JH, Braunewell S, Ruhaak J, Heckel F, Barbieri S, Tautz L, Hahn HK, Peitgen HO. 2011. Analysis of variability in manual liver tumor delineation in CT scans. Biomedical Imaging: From Nano to Macro, 2011 IEEE International Symposium on. p 1974-1977.

Murthy R, Nunez R, Szklaruk J, Erwin W, Madoff DC, Gupta S, Ahrar K, Wallace MJ, Cohen A, Coldwell DM and others. (2005): Yttrium-90 microsphere therapy for hepatic malignancy: Devices, indications, technical considerations, and potential complications. Radiographics 25 Suppl 1:S41-55.

Nekovei R, Ying S. 1991. Classification Of Digital Angiograms Using Artificial Neural Networks. Engineering in Medicine and Biology Society, 1991. Vol.13: 1991., Proceedings of the Annual International Conference of the IEEE. p 1440-1441.

Nekovei R, Ying S. (1995): Back-propagation network and its configuration for blood vessel detection in angiograms. Neural Networks, IEEE Transactions on 6(1):64-72.

Pappas TN. (1992): An Adaptive Clustering-Algorithm for Image Segmentation. IEEE Transactions on Signal Processing 40(4):901-914.

Pathak S, Jones R, Tang JM, Parmar C, Fenwick S, Malik H, Poston G. (2011): Ablative therapies for colorectal liver metastases: a systematic review. Colorectal Dis 13(9):e252-65.

Perona P, Malik J. (1990): Scale-Space and Edge-Detection Using Anisotropic Diffusion. IEEE Transactions on Pattern Analysis and Machine Intelligence 12(7):629-639.

Qi Y, Xiong W, Leow W, Tian Q, Zhou J, Liu J, Han T, Venkatesh S, Wang S. (2008): Semi-automatic Segmentation of Liver Tumors from CT Scans Using Bayesian Rule-based 3D Region Growing. The MIDAS Journal - Grand Challenge Liver Tumor Segmentation (2008 MICCAI Workshop).1-8.

Ricke J, Wust P. (2011): Computed tomography-guided brachytherapy for liver cancer. Semin Radiat Oncol 21(4):287-93.

Rikxoort E, Arzhaeva Y, Ginneken B. 2007. Automatic segmentation of the liver in computed tomography scans with voxel classification and atlas matching. Proc. MICCAI Workshop on 3-D Segmentat. Clinic: A Grand Challenge. p 101-108.

Ruskó L, G. B, G. N, M. F. 2007. Fully automatic liver segmentation for contrast-enhanced CT images. Proc. MICCAI Workshop on 3-D Segmentat. Clinic: A Grand Challenge. p 143-150.

Saddi M, Rousson KA, Chefd'hotel C, Cheriet F. 2007. Global-to-local shape matching for liver segmentation in CT imaging. Proc. MICCAI Workshop on 3-D Segmentat. Clinic: A Grand Challenge. p 207-214.

Sahani D, Mehta A, Blake M, Prasad S, Harris G, Saini S. (2004): Preoperative hepatic vascular evaluation with CT and MR angiography: implications for surgery. Radiographics 24(5):1367-80.

Sarwal A, Dhawan A. 1997. Three dimensional reconstruction of coronary arteries from two views. Engineering in Medicine and Biology Society, 1997. Proceedings of the 19th Annual International Conference of the IEEE. p 565-568 vol.2.

Scheele J, Rudroff C, Altendorf-Hofmann A. (1997): Resection of colorectal liver metastases revisited. J Gastrointest Surg 1(5):408-22.

Schima W, Saini S, Echeverri JA, Hahn PF, Harisinghani M, Mueller PR. (1997): Focal liver lesions: characterization with conventional spin-echo versus fast spin-echo T2-weighted MR imaging. Radiology 202(2):389-93.

Schmidt G, Athelogou MA, Schönmeyer R, Korn R, Binnig G. 2007. Cognition network technology for a fully automated 3-D segmentation of liver. Proc. MICCAI Workshop on 3-D Segmentat. Clinic: A Grand Challenge. p 125-133.

Scott DJ, Guthrie JA, Arnold P, Ward J, Atchley J, Wilson D, Robinson PJ. (2001): Dual phase helical CT versus portal venous phase CT for the detection of colorectal liver metastases: correlation with intra-operative sonography, surgical and pathological findings. Clin Radiol 56(3):235-42.

Seghers D, Slagmolen P, Lambelin Y, Hermans J, Loeckx D, Maes F, Suetens P. 2007. Landmark based liver segmentation using local shape and local intensity models. Proc. MICCAI Workshop on 3-D Segmentat. Clinic: A Grand Challenge. p 135-142.

Selle D, Preim B, Schenk A, Peitgen HO. (2002): Analysis of vasculature for liver surgical planning. Medical Imaging, IEEE Transactions on 21(11):1344-1357.

Seront E, Van den Eynde M. (2012): Liver-Directed Therapies: Does It Make Sense in the Current Therapeutic Strategy for Patients With Confined Liver Colorectal Metastases? Clin Colorectal Cancer.

Sharma RA, Van Hazel GA, Morgan B, Berry DP, Blanshard K, Price D, Bower G, Shannon JA, Gibbs P, Steward WP. (2007): Radioembolization of Liver Metastases From Colorectal Cancer Using Yttrium-90 Microspheres With Concomitant Systemic Oxaliplatin, Fluorouracil, and Leucovorin Chemotherapy. Journal of Clinical Oncology 25(9):1099-1106.

Shu SG, Lin HH, Kuo SW, Yu SS. (2009): EXCLUDING BACKGROUND INITIAL SEGMENTATION FOR RADIOGRAPHIC IMAGE SEGMENTATION. International Journal of Innovative Computing Information and Control 5(11A):3849-3860.

Sica GT, Ji H, Ros PR. (2000): CT and MR imaging of hepatic metastases. AJR Am J Roentgenol 174(3):691-8.

Siegel R, Ward E, Brawley O, Jemal A. (2011): Cancer statistics, 2011. CA: A Cancer Journal for Clinicians 61(4):212-236.

Smeets D, Stijnen B, Loeckx D, De Dobbelaer B, Suetens P. (2008): Segmentation of Liver Metastases Using a Level Set Method with Spiral-Scanning Technique and Supervised Fuzzy Pixel Classification. The MIDAS Journal - Grand Challenge Liver Tumor Segmentation (2008 MICCAI Workshop).1-7.

Smith D. (2009): The MathWorks Introduces New Versions of MATLAB Parallel Computing Products. The Mathworks Inc., DOI: http://www.mathworks.com/company/pressroom/articles/article34060.html.

Smith SM. (2002): Fast robust automated brain extraction. Hum Brain Mapp 17(3):143-55.

Sonka M, Stolpen A, Liang W, Stefancik RM. 2001. Vascular Imaging and Analysis. Handbook of Medical Imaging. Sonka M, Fitzpatrick JM, editors. Bellingham, WA: SPIE Press.

Susomboon R, Raicu DS, Furst J. (2007): A hybrid approach for liver segmentation. Proc. MICCAI Workshop on 3-D Segmentat. Clinic: A Grand Challenge:151-160.

Swanson EL, Indelicato DJ, Louis D, Flampouri S, Li Z, Morris CG, Paryani N, Slopsema R. (2012): Comparison of Three-Dimensional (3D) Conformal Proton Radiotherapy (RT), 3D Conformal Photon RT, and Intensity-Modulated RT for Retroperitoneal and Intra-Abdominal Sarcomas. Int J Radiat Oncol Biol Phys.

Toledo R, Orriols X, Radeva P, Binefa X, Vitria J, Canero C, Villanuev JJ. 2000. Eigensnakes for vessel segmentation in angiography. Pattern Recognition, 2000. Proceedings. 15th International Conference on. p 340-343 vol.4.

Tolias YA, Panas SM. 1997. An unsupervised fuzzy vessel tracking algorithm for retinal images. Fuzzy Systems, 1997., Proceedings of the Sixth IEEE International Conference on. p 325-329 vol.1.

Tozaki T, Kawata Y, Niki N, Ohmatsu H, Moriyama N. 1994. 3-D visualization of blood vessels and tumor using thin slice CT images. Nuclear Science Symposium and Medical Imaging Conference, 1994., 1994 IEEE Conference Record. p 1470-1474 vol.3.

Trefethen A, Menon V, Chang C, Czajkowski G, Myers C, Trefethen L. (1996): MultiMATLAB: MATLAB on Multiple Processors. Cornell Theory Center Technical Reports:96-239.

Tsai MH, Wang MH, Chang TY, Pai PY, Chan YK, Chen JM. (2010): AN ADAPTABLE THRESHOLD DECISION METHOD. International Journal of Innovative Computing Information and Control 6(5):2285-2299.

Valls C, Andia E, Sanchez A, Guma A, Figueras J, Torras J, Serrano T. (2001): Hepatic metastases from colorectal cancer: preoperative detection and assessment of resectability with helical CT. Radiology 218(1):55-60.

Van Hazel G, Blackwell A, Anderson J, Price D, Moroz P, Bower G, Cardaci G, Gray B. (2004): Randomised phase 2 trial of SIR-Spheres (R) plus fluorouracil/leucovorin chemotherapy versus fluorouracil/leucovorin chemotherapy alone in advanced colorectal cancer. Journal of Surgical Oncology 88(2):78-85.

Wang HQ, Yang JY, Yan LN. (2011): Hemihepatic versus total hepatic inflow occlusion during hepatectomy: a systematic review and meta-analysis. World J Gastroenterol 17(26):3158-64.

Ward J, Guthrie JA, Wilson D, Arnold P, Lodge JP, Toogood GJ, Wyatt JI, Robinson PJ. (2003): Colorectal hepatic metastases: detection with SPIO-enhanced breath-hold MR imaging--comparison of optimized sequences. Radiology 228(3):709-18.

Weg N, Scheer MR, Gabor MP. (1998): Liver lesions: improved detection with dual-detector-array CT and routine 2.5-mm thin collimation. Radiology 209(2):417-26.

Wong D, Liu J, Yin F, Tian Q, Xiong W, Zhou J, Yingyi Q, Han T, Venkatesh S, Wang S. (2008): A semi-automated method for liver tumor segmentation based on 2D region growing with knowledge-based constraints. The MIDAS Journal - Grand Challenge Liver Tumor Segmentation (2008 MICCAI Workshop).1-10.

Xing Z, Jie T, Dehui X, Xiuli L, Kexin D. 2011. Interactive liver tumor segmentation from ct scans using support vector classification with watershed. Engineering in Medicine and Biology Society,EMBC, 2011 Annual International Conference of the IEEE. p 6005-6008.

Xu LH, Cai SJ, Cai GX, Peng WJ. (2011): Imaging diagnosis of colorectal liver metastases. World J Gastroenterol 17(42):4654-9.

Xu Y, Liu Z, Ji L. 2010. Liver CT image segmentation by local entropy method. Computer Application and System Modeling (ICCASM), 2010 International Conference on. p V11-591-V11-594.

Yoav T, Ofer E, Moti F, Leo J, Jacob S. (2008): An iterative Bayesian approach for liver analysis: tumors validation study. Grand Challenge Liver Tumor Segmentation (2008 MICCAI Workshop).

Yuanzhong L, Hara S, Shimura K. 2006. A Machine Learning Approach for Locating Boundaries of Liver Tumors in CT Images. Pattern Recognition, 2006. ICPR 2006. 18th International Conference on. p 400-403.

VITA

MOHAMMED GORYAWALA

January 20, 1984	Born, Baroda, Gujarat, India
2002-2006	B.S., Electronics and Communication Engineering Veer Narmad South Gujarat University Surat, Gujarat, India
2006-2007	M.S., Biomedical Engineering Florida International University Miami, Florida
2008 - 2012	Doctoral Candidate Florida International University Miami, Florida

PUBLICATIONS AND PRESENTATIONS

R. Bhatt, M. Adjouadi, **M. Goryawala**, S. Gulec, A. McGoron, “An algorithm for PET tumor volume and activity quantification: Without specifying the camera’s point spread function (PSF),” accepted with revision Medical Physics. (Original article)

M. Cabrerizo, M. Ayala, **M. Goryawala**, P. Jayakar, M. Adjouadi, “A new parametric feature descriptor for the classification of epileptic and control EEG records in pediatric population”, International Journal of Neural Systems, 22:2, (16 pages), April 2012

M. Cabrerizo, **M. Goryawala**, P. Jayakar, A. Barreto, S. Khizroev, M. Adjouadi, Accurate 3D source localization of focal epileptic foci using interictal EEG spikes, in: Signal Processing in Medicine and Biology Symposium (SPMB), IEEE, pp. 1-6. 2011

M. Goryawala, M. R. Guillen, S. Gulec, T. Barot, R. Suthar, R. Bhatt, A. McGoron, M. Adjouadi “A New 3D Liver Segmentation Method with Parallel Computing for Selective Internal Radiation Therapy”, IEEE Transactions on Information Technology in BioMedicine, Vol. 16(1), pp. 62-69, January 2012.

M. Goryawala, M. R. Guillen, A. Barreto, R. Bhatt, S. Gulec, T. Barot, R. Suthar, A. McGoron, M. Adjouadi, Design and Evaluation of Parallel Processing Techniques for 3D Liver Segmentation and Volume Rendering, i-manager’s Journal on Software Engineering, Vol. 5(4), pp. 12-26 April – June 2011.

M. Goryawala, M. R. Guillen, S. Gulec, T. Barot, R. Suthar, R. Bhatt, A. McGoron and M. Adjouadi, “An Accurate 3D Liver Segmentation Method for Selective Internal Radiation Therapy Using a Modified K-Means Algorithm and Parallel Computing”, accepted in Int. J. of Innovative Computing Information and Control, in press Vol 8(7), (24 pages) July 2012.

M. Goryawala, I. Yaylali, M. Cabrerizo, K. Vedala, and M. Adjouadi, "An effective intra-operative neurophysiological monitoring scheme for Aneurysm Clipping and Spinal Fusion surgeries", Journal of Neural Engineering, in press Vol. 9, (12 pages), 2012

M. Goryawala, M. del Valle, Jiali Wang, James Byrne, Juan Franquiz and A. McGoron, "Low-cost respiratory motion tracking system", Proc. SPIE 6918, 691822, 2008, doi:10.1117/12.772922

M. Goryawala, M. R. Guillen, X. You and M. Adjouadi, "A New Algorithm as an Extension to the Gradient Descent Method for Functional Brain Activation Classification", IFMBE Proceedings, Vol. 24(9), pp. 125-128, 2009

M. Goryawala, M. Guillen, R. Bhatt, A. McGoron, M. Adjouadi, "A comparative study on the performance of the parallel and distributing computing operation in MatLab", Proc. International Conference on Advanced Information Networking and Applications, IEEE-AINA, Perth, WA, pp. 150-157, April 2010.

M. Goryawala, R. Bhatt, S. Gulec, A. McGoron and M. Adjouadi, A low-interaction automatic 3D liver segmentation method for selective internal radiation therapy, submitted to 2012 IEEE International Conference on Image Processing, 2012.

M. Goryawala, I. Yaylali, M. Cabrerizo, K. Vedala, M. Adjouadi, An effective novel patient specific Gaussian template based scheme for somatosensory evoked potential detection, in: Signal Processing in Medicine and Biology Symposium (SPMB), IEEE, pp. 1-6, 2011.

M. Goryawala, T. Barot, R. Suthar, M. Adjouadi, A. McGoron, S. Gulec, Anatomic liver volume determination for liver cancer SIRT planning, J NUCL MED, 51(Supplement 2) 1311, 2010

A. Guzman, **M. Goryawala**, and M. Adjouadi, "Generating Thermal Facial Signatures Using Thermal Infrared Images", IEEE-ESPA-2012, The 1st International Conference on Emerging Signal Processing Applications, Las Vegas, Nevada, pp. 21-24, January 12-14, 2012.

K. Vedala, I. Yaylali, M. Cabrerizo, **M. Goryawala**, M. Adjouadi, "Peak Detection of Somatosensory Evoked Potentials using an Integrated PCA-Walsh Method", Journal of Clinical Neurophysiology, in press.

J.L. Wang, M. del Valle, **M. Goryawala**, J.M. Franquiz, A.J. McGoron, Computer-assisted quantification of lung tumors in respiratory gated PET/CT images: phantom study, Med Biol. Eng. Computing, Vol. 48, pp. 49-58, 2010.