

Hasso-Plattner-Institut fuer Software systemtechnik  
an der Universitaet Potsdam

---

# **Multi-Visualization and Hybrid Segmentation**

## **Approaches within Telemedicine Framework**

Dissertation  
zur Erlangung des akademischen Grades  
"doctor rerum naturalium"  
(Dr. rer. nat.)  
am Fachgebiet Internet Technologien und Systeme

eingereicht an der  
Mathematisch-Naturwissenschaftlichen Fakultät  
der Universitaet Potsdam

von  
**Chunyan Jiang**

Potsdam, 2007



**Gutachter:**

Prof. Dr. Christoph Meinel, Hasso-Plattner-Institut

Prof. Dr. Martin Bettag, Krankenhaus der Barmherzigen Brüder

Prof. Dr. Baocai Yin, Beijing University of Technology

**Prüfungskommission:**

Prof. Dr. Werner Zorn, Hasso-Plattner-Institut

Prof. Dr. Jürgen Döllner, Hasso-Plattner-Institut

Prof. Dr. Andreas Polze, Hasso-Plattner-Institut

Prof. Dr. Robert Hirschfeld, Hasso-Plattner-Institut

Prof. Dr. Micheal Gössel, Institut für Informatik

Dr. Georgi Graschew, Max Delbrück Centrum für Molekulare Medizin  
(MDC) Berlin-Buch / Robert-Rössle-Klinik der Charité,

Datum der Disputation: 05, 03, 2007

Die Dissertation wurde am 30, 06, 2006 an der  
Mathematisch-Naturwissenschaftlichen Fakultät der Universität Potsdam  
eingereicht und am 05, 03, 2007 mit „magna cum laude“ angenommen.

©2007 Chunyan Jiang

# **Abstract**

## **Multi-Visualization and Hybrid Segmentation Approaches within Telemedicine Framework**

By  
Chunyan Jiang

The innovation of information techniques has changed many aspects of our life. In health care field, we can obtain, manage and communicate high-quality large volumetric image data by computer integrated devices, to support medical care. In this dissertation I propose several promising methods that could assist physicians in processing, observing and communicating the image data. They are included in my three research aspects: telemedicine integration, medical image visualization and image segmentation. And these methods are also demonstrated by the demo software that I developed.

One of my research point focuses on medical information storage standard in telemedicine, for example DICOM, which is the predominant standard for the storage and communication of medical images. I propose a novel 3D image data storage method, which was lacking in current DICOM standard. I also created a mechanism to make use of the non-standard or private DICOM files.

In this thesis I present several rendering techniques on medical image visualization to offer different display manners, both 2D and 3D, for example, cut through data volume in arbitrary degree, rendering the surface shell of the data, and rendering the semi-transparent volume of the data.

A hybrid segmentation approach, designed for semi-automated segmentation of radiological image, such as CT, MRI, etc, is proposed in this thesis to get the organ

or interested area from the image. This approach takes advantage of the region-based method and boundary-based methods. Three steps compose the hybrid approach: the first step gets coarse segmentation by fuzzy affinity and generates homogeneity operator; the second step divides the image by Voronoi Diagram and reclassifies the regions by the operator to refine segmentation from the previous step; the third step handles vague boundary by level set model.

Topics for future research are mentioned in the end, including new supplement for DICOM standard for segmentation information storage, visualization of multimodal image information, and improvement of the segmentation approach to higher dimension.

## ACKNOWLEDGEMENTS

I am extremely fortunate to have had the opportunity to pursue the doctoral degree in Germany, and perform my research under the supervision of Prof. Dr. Christoph Meinel at University of Trier, and later at University of Potsdam (Hasso-Plattner-Institute). I am grateful to Prof. Meinel for his support, guidance and encouragement throughout my studies. I have learned considerably through his insight into problems. I also wish to express my sincere gratitude to Prof. Dr. Martin Bettag for his interests of my work and taking the time to review my dissertation. I greatly appreciate Prof. Baocai Yin and Prof. Dehui Kong for their time and effort that they have invested in judging the contents of my thesis. I thank PD Dr. Guenther Sigmund and PD Dr. Matthias Gutberlet for their helpful discussions concerning my research, and their supplying the medical images.

I would like to thank my colleagues in Institute for Telematics, and Hasso-Plattner-Institut. They showed me the team spirit in each project group. I was very glad to join them in the projects. And I also learned different cultures from them since we came from different countries. I am grateful to my friends around me for their warm friendship and helps in the hard times. Their encouragements helped me sail smoothly through the tasks of PhD study.

My deepest gratitude goes to my husband, Xinhua Zhang, for love, support and encouragement. I also cherish the color that my little son brought to me. His smile relaxed me when I was exhaust. I am profoundly grateful to my parents and my sister for their long-distance support, and always being there for me while I struggled.





# TABLE OF CONTENT

|  |            |
|--|------------|
| <b>I INTRODUCTION .....</b>  | <b>1</b>   |
| 1.1 MOTIVATIONS .....  | 1          |
| 1.2 MEDICAL IMAGE VISUALIZATION .....                                      | 2          |
| 1.3 MEDICAL IMAGE SEGMENTATION .....                                       | 4          |
| 1.4 TELEMEDICINE AND COMPUTATIONAL ANALYSIS .....                          | 5          |
| 1.5 THESIS CONTRIBUTIONS .....   | 6          |
| 1.6 ROADMAP .....  | 9          |
| <b>II TELEMED-VS: A VISUALIZATION AND SEGMENTATION TELEMEDICINE SYSTEM</b> | <b>10</b>  |
| 2.1 TELEMEDICINE COMPLIED FILE PROCESSING .....                            | 13         |
| 2.2 VOLUME, SURFACE AND RESLICING RENDERING .....                          | 14         |
| 2.3 MEDICAL IMAGE SEGMENTATION .....                                       | 17         |
| <b>III INTEGRATED TO TELEMEDICINE FRAMEWORK.....</b>                       | <b>22</b>  |
| 3.1 A SURVEY OF TELEMEDICINE.....  | 22         |
| 3.2 OVERVIEW OF DICOM STANDARD.....  | 23         |
| 3.3 TELEMED-VS MANIPULATES DICOM FILES .....                               | 30         |
| 3.4 A NEW SUPPLEMENT FOR DICOM ON 3D IMAGE DATA STORAGE.....               | 33         |
| 3.5 CHAPTER SUMMARY .....  | 37         |
| <b>IV THREE RENDERING WAYS FOR MEDICAL IMAGE.....</b>                      | <b>39</b>  |
| 4.1 VOLUME RENDERING.....  | 40         |
| 4.2 SURFACE RENDERING .....  | 43         |
| 4.3 CUT THROUGH RENDERING .....  | 45         |
| 4.4 CHAPTER SUMMARY .....  | 47         |
| <b>V A NEW HYBRID APPROACH OF SEGMENTATION.....</b>                        | <b>49</b>  |
| 5.1 INTRODUCTION OF RELATED WORK.....                                      | 50         |
| 5.1.1 Terminology Related to Segmentation .....                            | 51         |
| 5.1.2 Classification of Segmentation Methods .....                         | 54         |
| 5.2 OVERVIEW OF THE HYBRID APPROACH.....                                   | 61         |
| 5.3 GENERATING HOMOGENEITY OPERATOR .....                                  | 62         |
| 5.3.1 One Characteristic Intensity Pattern: Hanging Togetherness.....      | 62         |
| 5.3.2 Defining Affinity for Target Object .....                            | 64         |
| 5.3.3 Defining Homogeneity Criteria .....                                  | 68         |
| 5.3.4 Experiments.....   | 70         |
| 5.4 RECLASSIFYING EXTERIOR, INTERIOR AND BOUNDARY REGIONS .....            | 73         |
| 5.4.1 Concept of Voronoi Diagram .....                                     | 74         |
| 5.4.2 VD – Based Segmentation Algorithm.....                               | 76         |
| 5.4.3 Experiments.....   | 79         |
| 5.5 REFINING THE VAGUE BOUNDARY .....                                      | 81         |
| 5.5.1 Deformable Form.....   | 83         |
| 5.5.2 Front Propagation Problem .....                                      | 84         |
| 5.5.3 Shape Recovery with Front Propagation .....                          | 88         |
| 5.5.4 Extending the Speed Function .....                                   | 90         |
| 5.5.5 Experiments on Hybrid Framework .....                                | 95         |
| 5.6 EVALUATION OF THE HYBRID SEGMENTATION APPROACH.....                    | 98         |
| 5.7 CHAPTER SUMMARY .....  | 102        |
| <b>VI CONCLUSION AND FUTURE WORK.....</b>                                  | <b>104</b> |
| 6.1 CONTRIBUTIONS SUMMARY .....  | 104        |

|                           |  |            |
|---------------------------|--|------------|
| 6.1.1                     | Contributions in Integrated to Telemedicine Framework .....  | 104        |
| 6.1.2                     | Contributions in Medical Image Visualization.....            | 105        |
| 6.1.3                     | Contributions in Medical Image Segmentation.....             | 105        |
| 6.2                       | FUTURE WORK.....   | 106        |
| 6.2.1                     | Additional Supplement to DICOM Standard.....                 | 106        |
| 6.2.2                     | Multi-Modalities Visualization .....                         | 107        |
| 6.2.3                     | Segmentation Extensions and Additional Evaluation Study..... | 107        |
| 6.3                       | CHAPTER SUMMARY .....  | 108        |
| <b>BIBLIOGRAPHY .....</b> |  | <b>110</b> |
| <b>APPENDIXES .....</b>   |  | <b>120</b> |
|                           | <b>RAY CASTING ALGORITHM.....</b>                            | <b>121</b> |
|                           | <b>MARCHING CUBE ALGORITHM .....</b>                         | <b>132</b> |

# CHAPTER I

## INTRODUCTION

### 1.1 Motivations

Nowadays, many kinds of medical image devices are used in diagnosis to capture the inside images of patient body. Looking back to first internal image, it came from one German physicist Wilhelm Conrad Roentgen on November 8, 1895. He recorded a photograph of his wife's hand with one unknown mysterious ray, labelled as "X". It became the internal imaging that doctor's future dependence on. And three months later, X-rays were first used clinically in the United States.

In the subsequent century the technological innovations have increased the



Figure 1.1. First X-ray Picture. In 1895, X-ray version of Bertha Roentgen's hand and wedding ring fascinated the public and puzzled scientists

value of doctors' "X-ray vision". While the original radiographs revealed only 2D

projections, today's Computed Tomography (CT) scanners rotate the imaging apparatus to reconstruct 3D volumetric maps of X-ray attenuation coefficients. Magnetic Resonance Imaging (MRI) scanners can differentiate between various soft tissues. Functional information can be acquired by functional MRI (fMRI) or Positron Emission Tomography (PET).

While the advances in medical imaging have been impressive, the need for scientific progress does not end with the image acquisition process. Post-processing, or computational analysis of the image data, has attracted researchers in artificial intelligence, pattern recognition, neurobiology, and applied mathematics. Many clinical applications of medical image analysis rely on computers to embody the capability to understand the image data to some degree, for example, surgical planning help surgeons to ascertain the operability or identify the optimum approach trajectory before the operation. Surgeons can benefit not only from pre-operative planning, but also online guidance for precise, intra-operative localization [BMAS97], that says, surgical guidance aims to equip the surgeons with an enhanced vision of reality during operation; computer aided diagnosis [Gig00, GRV02], volumetric analysis, and so on.

To make full use of medical images, the images and related information should be able to be stored and transferred standardly between different departments in one hospital, between different hospitals, or even between different countries. So telemedicine is very important in such context. The standard defining such data storage and communication is required in telemedicine system. I will discuss it later in the thesis.

## **1.2 Medical Image Visualization**

One related work should be introduced here is medical image visualization, because it is an essential component function for computational medical image analysis system. It offers an intuitive and realistic rendering of inside of patients for physicians. A number of techniques have been developed to directly visualize 3D scalar fields on Cartesian grids such as data sets from medical imaging modalities. Two widely used volume visualization methods which are often applied in medical applications are surface fitting [Lev88] and direct volume rendering [Kau91]. For surface fitting method, an intermediate representation (usually polygons extracted via Marching Cube [LC87] or other surface extraction methods) is generated and then displayed. For complex surface extraction situation, for example, organs or tumour in medical image data, many segmentation techniques are developed to fulfil this task. It relates to another topic discussed in this thesis.

Direct volume rendering uses the original data. Volume rendering consists of integrating colour and opacity values across a 3D space. This integration is often performed by sampling the volume at regular intervals: hardware algorithms slice through the volume using a stack of closely spaced polygons, while software algorithms typically sample viewing rays at regular intervals. Many approaches for direct volume rendering have been presented in the past [KH84], [GK96], [WG91]. As in most volume rendering enhancement methods [CMHKG01, LMERH02], the visual cues or features are evaluated based on local volume characteristics (e.g., gradients). These advanced techniques make volume visualization faster and more realistic.

Although the techniques of surface rendering and direct volume rendering have been developed to perfection, the physician's requirement of visualization can not be fulfilled relying on one or two such techniques. Some time simple 2D slices

display can fit the requirement in clinic routine better than complex volume rendering. Therefore medical image application should offer appropriate visualization manners for actual requirement.

### **1.3 Medical Image Segmentation**

This research makes plenty of efforts on medical image segmentation. So here I want to introduce some background knowledge about segmentation. Image segmentation is a fundamental task in medical image analysis. In segmentation, objects of interest in the image are extracted so that we can analyze their properties. Such properties can include object's size, pixel (voxel) intensities, centroid location, shape and orientation. The property information from object segmentations is routinely used in many different applications, such as: diagnosis [Tay95], treatment planning [KDFPTL97], study of anatomical structure [DP95], organ motion tracking [HLRC98], and computer-aided surgery [ACCCLM96].

Methods for performing segmentations vary widely depending on the specific application, imaging modality, and other factors. Operator controlled method segments image manually incorporating with using a graphical interface to apply basic segmentation methods such as threshold (e.g. [LP89]), mathematical morphology operators [Ser82], seeded region growing [AB94], or combinations of these [LTA02] and to draw borders of regions of interest. The statistic of signal intensity can be used for segmentation, such as [GGK02], [WZW02]. [GMAKW04] classifies the remaining pixels using k-means style classification. Deformable models (e.g. [YZK03], [YD03]) directly define object boundaries, by identifying transitions in intensity properties between neighbouring but different anatomical regions. Anatomical knowledge can guide the segmentation process [CGHPVDT02]. Identifying structures based upon a priori knowledge and a pre-segmented anatomical

map [SRN00] relies on the assumption that the individual image, for example, brain image, and the atlas template are in the same frame of reference.

Haralick and Shapiro state that “Image segmentation techniques are basically ad hoc and differ precisely in the way they emphasize one or more of the desired properties” [HS85]. For example, the segmentation technique of brain tissue has different property requirements from the segmentation of the liver. Following Gonzalez and Wintz, “segmentation algorithms are generally based on two basic properties of grey-level values: discontinuity and similarity” [GW87]. However these properties can be affected by some general imaging artefacts, such as noise, partial volume effects, and motion. Therefore they can also have significant consequences on the performance of segmentation algorithms. Furthermore, each imaging modality has its own idiosyncrasies with which to contend. There is currently no single segmentation method that yields acceptable results for every medical image. Methods do exist that are more general and can be applied to a variety of data. However, methods that are specialized to particular applications can often achieve better performance by taking into account prior knowledge. Selection of an appropriate approach to a segmentation problem can therefore be a difficult dilemma.

#### **1.4 Telemedicine and Computational Analysis**

As I mentioned above, medical image processing can help doctor’s diagnosis. However, if doctors in different locations want to cooperate with each other, it is necessary for them to get the same picture. Therefore the storage and communication of the image and related information should be standardized. As the development of telecommunication technology, telemedicine is going to be reality.

Computational image analysis system should be part of telemedicine system obeying to its standard defined in many aspects, such as data storage method,

communication protocol, and so on. The physicians in different locations can discuss the result of the same case that analyzed by computational system, and make surgical plan. There exists one standard for medical image data storage, exchange and communication in telemedicine system. It is the Digital Imaging and Communications in Medicine (DICOM) standard. This standard has been accepted by most of main medical devices providers in the world. For the integration of computational image analysis system to telemedicine system, DICOM standard should be adapted. Hence, one DICOM complied computational image analysis system can communicate with other medical system in the telemedicine environment.

## **1.5 Thesis Contributions**

Concerning my research, the contributions can be concluded in three aspects: telemedicine framework integration; multi-ways medical data visualization; hybrid medical image segmentation approach development. These three parts build up the thesis.

- **Integration to Telemedicine Framework**

I developed a system to demonstrate the methods I proposed during my PhD research, and this system is developed within telemedicine framework. Although my system mainly focuses in medical data visualization and segmentation, it can be combined with other big integrated telemedicine system, such as picture archiving and communication system (PACS) or hospital information system (HIS). I use DICOM files as processing files by reading them according to DICOM standard. DICOM standard is very flexible. Most retailers use this advantage in order to define their own structures which will be adapted to the offered equipment by those retailers. However, these structures constructed by different retailers can only be read by their own system. It obstructs the communication between different systems. I propose a



mechanism in order to prevent private structures. The application removes every private or retired structure from a DICOM file in order to make it readable for most of the offered viewers, so that every hospital and institute is capable to construct their own imaging and communication system by using software and equipment of different retailers instead of only one.

The other contribution of my work to telemedicine framework is that I submit a supplement for 3D image data storage in DICOM standard. Currently, DICOM only defines medical image data storage format for 2D image data. There is no definition of 3D data storage in DICOM standard. A volume is usually presented by a series of 2D DICOM files, those are parallel slices of the volume. Therefore the volume data has some repeated descriptions of same properties of those slices, for example, patient's name. Furthermore there is another problem that it is difficult to keep the data's integrality, for example, once one piece of the series is missing, the volume is not integrated any more. In this supplement, I suggest one new storing format for DICOM about 3D data, and also add some tags describing the related properties of 3D data.

- **Multi - Ways Visualization**

In the thesis, I describe three methods for image data rendering. One method is volume rendering. Here, the volume is rendered as semitransparent colour object. Different intensity of the data is rendered as different colour. It is easy to distinguish different parts by the contrasting colours. The blank part of the data is set as transparent part, for example, the margin in image is shown as black for 2D presenting while rendered here transparently. Hence, this non-information part would not shelter the inner structure anymore. The other rendering method is surface fitting by defining iso-value surface. In some medical image modality, for example, CT data,

some organs have iso-value, such as bone or skin. These iso-values can be extracted and rendered as surface. The segmented 3D object can also be rendered as surface mode. If the image data is consisted by a stack of parallel slices, and the target object is segmented slice by slice, then the contours in each slice can construct the surface of the object same as iso-value. The surface rendering gives different impression as semi-transparent volume rendering. The third rendering method is the cut through of volume. If the whole row data is drawn on screen as their intensity value, then only the outside of the data can be rendered since inside is covered. I define one cut plane as any degree to reslice the data. The data on one side of the plane is moved while the other side preserved. So the cross section can be shown. The cut plane can be moved along its perpendicular to show the cross section in different position as fly-in, fly-out manner. These cross section images can also be saved as DICOM files.

- **Hybrid Segmentation Approach**

In my work, I proposed one new hybrid approach for medical image segmentation. This hybrid approach is designed for semi-automated segmentation of radiological image, such as CT, MRI, etc, to get the organ or interested area from the image. The approach integrates region-based method and boundary-based method. And it reduces the drawbacks of both methods and enlarges the advantages of them. Firstly, the target object is segmented roughly by fuzzy affinity. And the homogeneity operator is created in this step to distinguish the intensity pattern of object and background. Then Voronoi Diagram (VD)-based method refines the pre-step's result. VD-based method divides the image into Voronoi regions, and classified these regions to three types, inside, outside and boundary, by the homogeneity operator. Through a number of iterations of subdivision and reclassification of boundary regions, the segmentation of interest area is near to real boundary. Finally level set model handles some vague or

missed boundary, and gets smooth and accurate segmentation. This three-step segmentation method is fitting for some complex anatomical structures segmentation, such as brain white matter or tumour. This hybrid approach is semi-automated, since the whole segmentation procedure doesn't need much manual intervention, except the initial seed area selection for the first segmentation step.

## **1.6 Roadmap**

This thesis is organized as follows. In Chapter II, I firstly give an overview of a demo system, which demonstrates the proposed approaches that I will in detail introduce in later chapters. In Chapter III, I describe the telemedicine integrated property of my work. I give an introduction of telemedicine concept and DICOM standard, and how the system processes DICOM image files and repairs non-standard files. I also describe one submitted supplement for DICOM standard about 3D image data storage format. In Chapter IV, I present three different image data rendering methods. Each method shows different properties of the original data. One is for volume rendering. The other is for surface shell rendering. And the third one is for cut through the volume data in arbitrary degree. In Chapter V, I describe the framework for medical image segmentation. It is a hybrid approach that consists of three steps. They are coarse segmentation, refining segmentation and handle vague boundary. Each step is based on the last step's result and improves the segmentation. Finally, in Chapter VI the conclusion for the whole thesis is presented. And the further research works are described.

## **CHAPTER II**

### **TeleMed-VS: A VISUALIZATION AND SEGMENTATION TELEMEDICINE SYSTEM**

My research work aims to help doctors to store, observe and process medical images. Some proposals around this goal are presented in this thesis. In order to evaluate all of these new ideas, I develop one demo system that is named TeleMed-VS. It is a kind of medical image visualization and segmentation software integrated with telemedicine concept. The demo system not only demonstrates the proposals in DICOM file processing, in multi-way visualization and in hybrid segmentation framework, but also has friendly user interface and some practical functions. And it gains good feedback from users evaluation testing. In this chapter, I will simply introduce some operation procedures and graphic interfaces of this software, and in the following chapters, I will describe in details the novel methods and technologies used or implemented in this system.

The framework of the demo system is shown as below, figure 2.1. There are three main parts in this software. Telemedicine part is the base, that includes DICOM file read, write and repair. It offers the data material for the other two parts: visualization and segmentation; that they perform some processing of the data material. And this telemedicine-based system can be combined with other telemedicine system, for example, it can be one part of PACS (picture archiving and communication system).

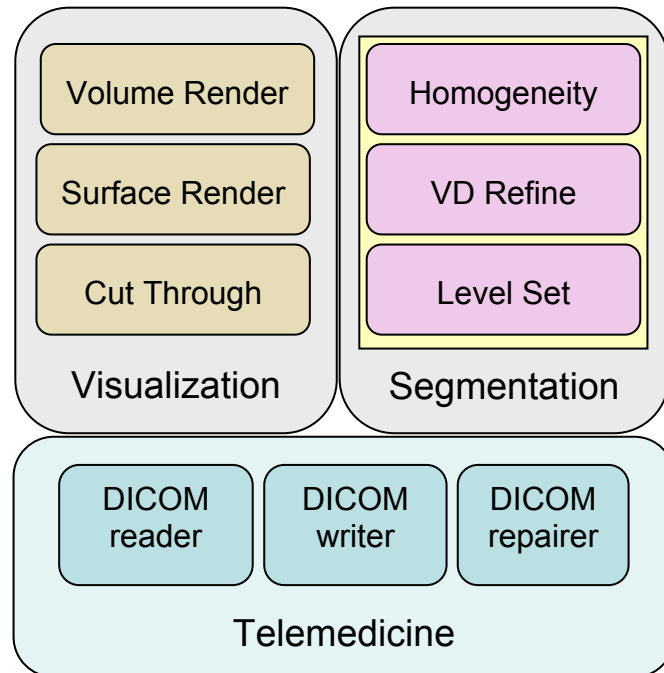


Figure 2.1 Framework of System

The system is developed with C++ language under Windows XP operation system. The system user interface is designed with FLTK (Fast Light Tool Kit), which is a C++ graphical user interface toolkit. Some other function libraries are called in the system for data processing, such as mathematical calculation. Following are two screenshots of the main window of this software:

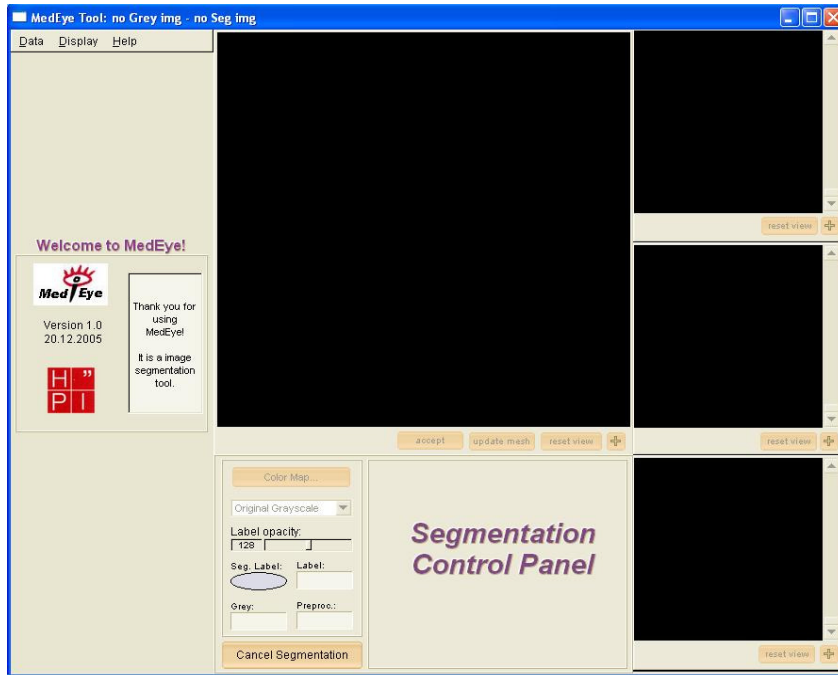


Figure 2.2 User interface of system

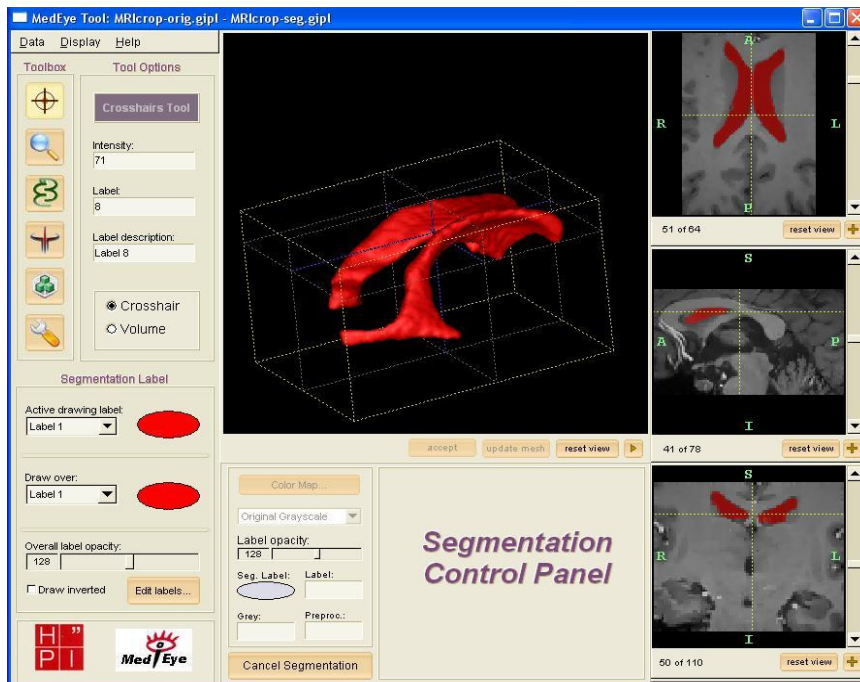


Figure 2.3 One example for image segmentation

## 2.1 Telemedicine Complied File Processing

This system is telemedicine integrated system so that it accomplishes come telemedicine concepts. Some DICOM related components are included in this system; they are DICOM reader, DICOM writer and DICOM repairer. Firstly, reader part imports DICOM data according to DICOM standard. During the read procedure, some information related image is extracted for future use, such as visualization or computing. The result of reading is shown as in figure 2.4. It shows the information of DICOM file header, as well as the image. The information relates to image modality, image size, data stored method, and some other things. I will explain in detail the content and structure of DICOM file in the next chapter. The DICOM writer component can write one image file as DICOM format, that image file can be generated in the system or modified from other DICOM file. DICOM repairer is responsible for checking the imported DICOM file to see if it contains some private or retired property tags. Those tags will be deleted, since they can not be read by most popular DICOM readers, so that they are useless for communication. DICOM repairer gets rid of un-useful tags and modifies the DICOM file according to DICOM standard. After repairing the file can be read by any DICOM reader, therefore it can be communicated in different systems.

```

First 128 bytes: unused by DICOM format
Followed by the characters 'D','I','C','M'
This preamble is followed by extra information e.g.:

0002,0000,File Meta Elements Group Len: 132
0002,0001,File Meta Info Version: 256
0002,0010,Transfer Syntax UID: 1.2.840.10008.1.2.1.
0008,0000,Identifying Group Length: 152
0008,0060,Modality: MR
0008,0070,Manufacturer: MRlcro
0018,0000,Acquisition Group Length: 28
0018,0050,Slice Thickness: 2.00
0018,1020,Software Version: 46\64\37
0028,0000,Image Presentation Group Length: 148
0028,0002,Samples Per Pixel: 1
0028,0004,Photometric Interpretation: MONOCHROME2
0028,0008,Number of Frames: 2
0028,0010,Rows: 109
0028,0011,Columns: 91
0028,0030,Pixel Spacing: 2.00\2.00
0028,0100,Bits Allocated: 8
0028,0101,Bits Stored: 8
0028,0102,High Bit: 7
0028,0103,Pixel Representation: 0
0028,1052,Rescale Intercept: 0.00
0028,1053,Rescale Slope: 0.00392157
7FE0,0000,Pixel Data Group Length: 19850
7FE0,0010,Pixel Data: 19838

```

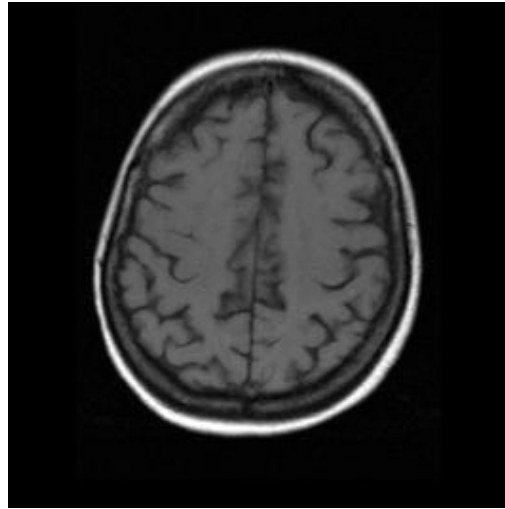


Figure 2.4 DICOM file reading example

## 2.2 Volume, Surface and Reslicing Rendering

Along with development of medical image acquiring devices, one acquired data set can include many kinds of information. The visualization methods are trying to render as most as possible useful information to aid physicians' work. The second part in the system is image data visualization. There are three methods realized in this software for image data render. Every method presents different characters' aspect of data. One method is volume rendering. It gives the whole presentation for the data by using semi-transparent render technique. Different grey scale will be rendered as different colours. And blank part in the image data will be rendered as transparent. Figure 2.5 below gives an example of this method.



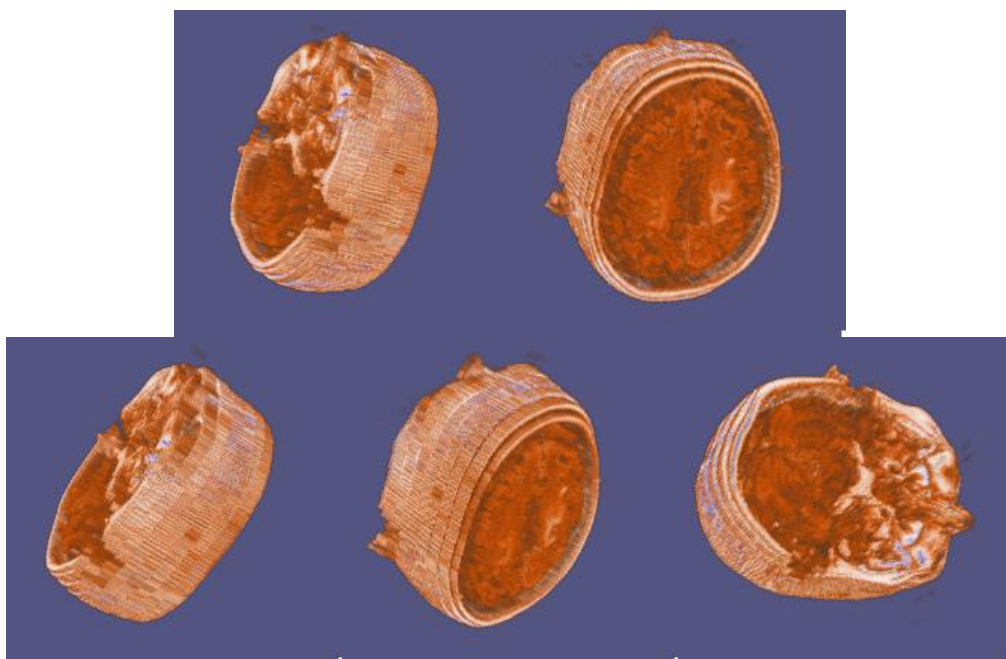


Figure 2.5 Volume rendering example

The other method is surface rendering. It renders the iso-value grey scale of the image data as geometry shell, for example bone or skin, as shown in figure 2.6. The figure 2.6 shows one data set. By using different iso-value, the bone information and the skin information can be rendered individually. So that the physician can choose what he/she wants from the data set in order to observe it better. This method concentrates on one iso-value, and gives explicit presentation of the shape. However, it ignores the other data. And if there is noise in the data, the surface will be disturbed. Hence, this method is one assistant manner for data rendering.

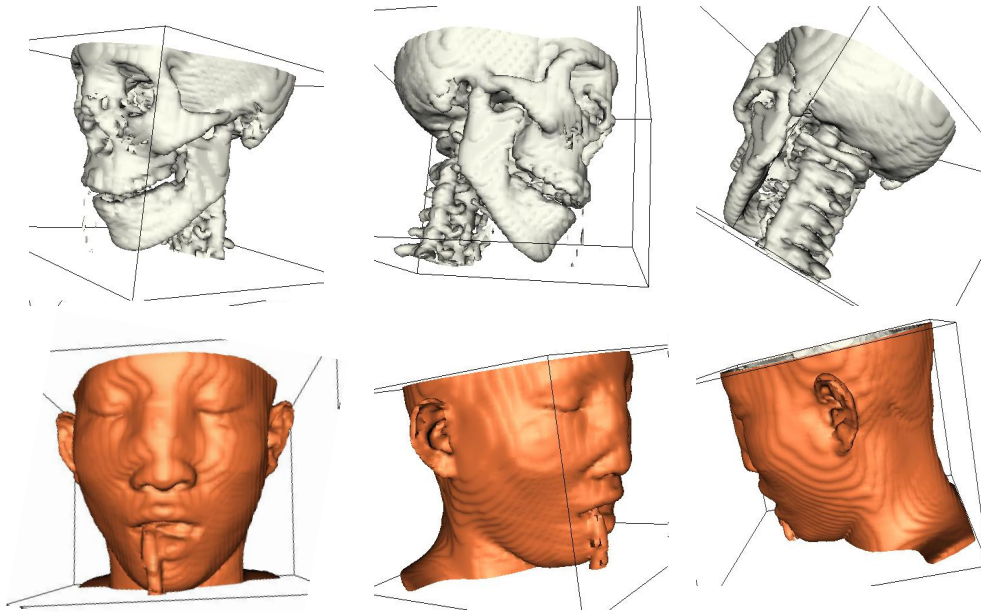


Figure 2.6 Surface rendering example

The third method is cut through method, also named reslicing method. It renders all data without any manipulations in advance. It just uses one plane to cut the data in any degree and position. The data in one side of the plane is removed, while the other side's data is held. Therefore the cross section is the inside situation of the data. By changing the position and degree of the plane, the cross section of data in any position can be observed. The figure 2.7 shows one cut through example. The left part is one data volume, which is cut by the plane. The right part shows one reference cube. The red plane is positioned as same as the cut plane in the data volume. Since the reference cube is semi-transparent, the red plane inside the cube can be seen clearly. Therefore it is served as the reference for cut plane's position. Furthermore, the image at cross section can be saved as DICOM format in my system by DICOM writer.

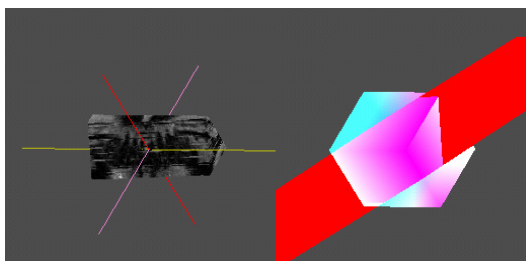


Figure 2.7 Cut through render example

### 2.3 Medical Image Segmentation

The third part of this system is image segmentation. I develop one hybrid approach for semi-automatic segmentation, which integrates region-based and edge-based techniques, and is composed of three main steps. The process starts from an initial seed inside the target object, and uses the pixels' fuzzy affinity to get an estimation of the object's boundary. This estimation also generates the homogeneity operator for later segmentation. Then with the Voronoi Diagram (VD), the image is redefined to outside, inside or boundary regions by the region classification with the homogeneity operator. After that, the boundary will be extracted to fill in the missing boundary and to override the spurious boundary data with a deformable surface model. This hybrid approach amplifies the strengths of both region-based and edge-based techniques but diminishes the weaknesses of them. The area of target object can be measured after segmentation. As we know, the measurement is very useful for disease diagnosis and monitoring.

The segmentation is performed slice by slice. It starts from the initial slice where the seed is put in. When the segmentation of this slice is finished, the result will be mapped to the neighbor slices. Because the images in neighbor slices are similar, it is not necessary to put other seeds in the neighbor slices. And the

segmentation in the other slices besides the initial slice can be performed automatically. After the segmentation in the whole data volume is finished, the segmented contour in each slice is extracted to generate one geometry shell. As a result, the segmented object is shown as surface rendering model in 3D.

The following figure 2.8 presents an example of segmentation. It is the lateral ventricle in the brain anatomy. This object is segmented from one MRI data volume.

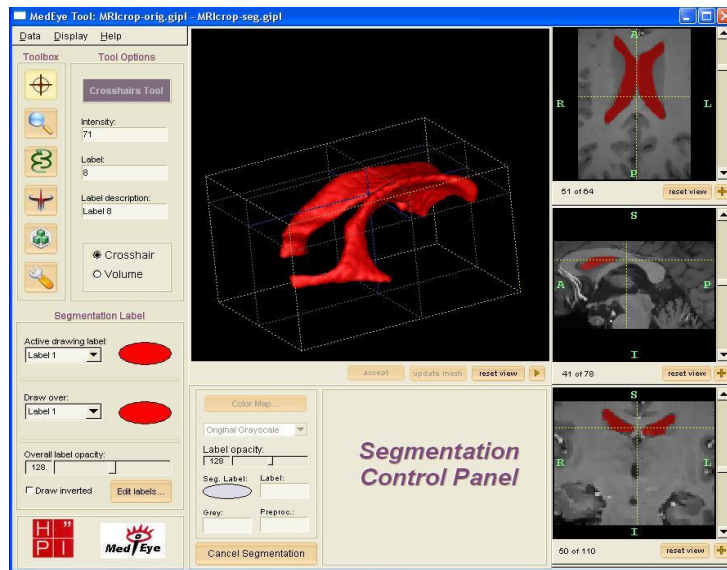


Figure 2.8 One segmentation

Here I take an example to illustrate the practical operational procedure of image segmentation by using this demo system. Firstly, the user defines an interest area of the image by dragging the square outline. The segmentation will be accomplished in this defined area, instead of the whole image. It increases the speed of the processing. The area-define-operation is shown as figure 2.9.

The top part in figure 2.9 is the segmentation control panel. In the middle of the figure is slicing view of image data. The blue-line box in the image presents the redefined region of interested area. In the bottom is the slicing view of the redefined

region. After defining the interested region, user can rearrange the range of image greyscale so that the target object is inside the range.

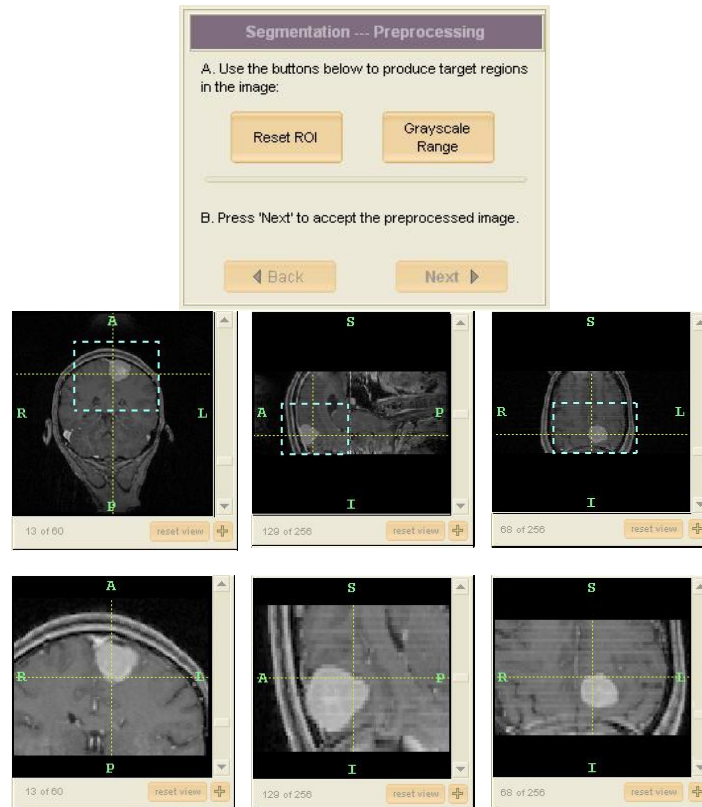


Figure 2.9 Reset interested region

Figure 2.10 shows the grayscale setting procedure. The following step is drawing initial area in the target object, as shown in figure 2.11.

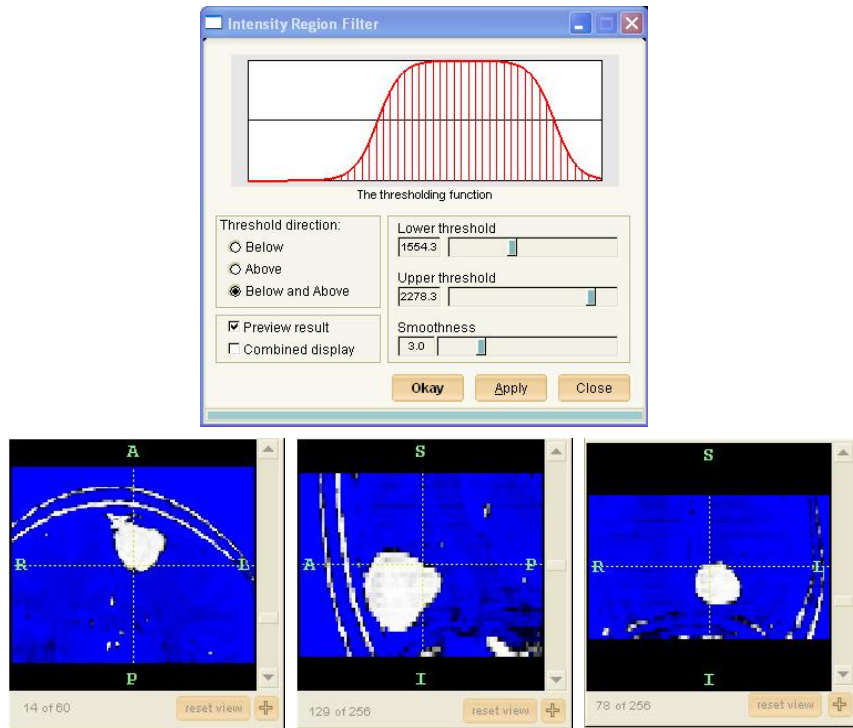


Figure 2.10 Set the range of image greyscale

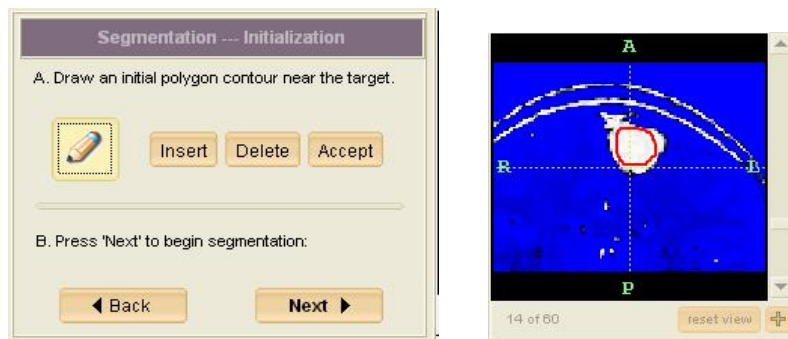


Figure 2.11 Draw initial seed area

Then system will start segmentation automatically, but user still can also control the segmentation process if it needed. When the segmentation is finished, the segmented volume can be presented in main window, like figure 2.12. The left picture of top part in this figure is the segmentation control panel. The right picture of top part is the segmented volume object. It is shown as geometry shell of object. The

bottom part is the slicing view of image data after segmentation, the segmented area is labeled by red color.

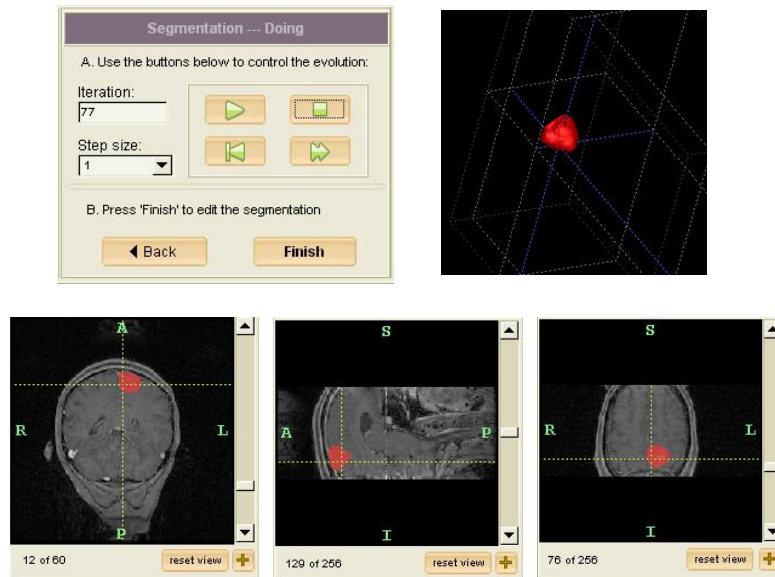


Figure 2.12 Segmentation and volume rendering

The system also offers some other useful functions, for example, saving the segmented object, calculating the segmented volume and doing some statistics. The user interface is quite friendly and easy to use.

## **CHAPTER III**

# **INTEGRATED TO TELEMEDICINE FRAMEWORK**

In this chapter, I will in particular describe the characters in telemedicine aspects that underlay my system. I will firstly introduce the concept of telemedicine and the concept of the Digital Imaging and Communications in Medicine (DICOM) standard. And then I detail how my system conforms to DICOM standard, how it manipulates the DICOM files, such as reading, writing and repairing of DICOM file. DICOM standard is one uncompleted standard, and it is always renewed and supplemented. In the last part of this chapter, I will describe in detail a supplement that I proposed concerning the 3D image data storage for DICOM standard. Although this system concentrates more on medical image visualization and segmentation, it is in deed a telemedicine based system. Therefore it can be easily integrated to other large commercial telemedicine systems.

### **3.1 A Survey of Telemedicine**

The Institute of Medicine defines telemedicine as “...the use of electronic information and communications technologies to provide and support health care when distance separates the participants...” Telemedicine is the delivery of health care and the exchange of health care information across distance. The prefix “tele” derives from the Greek “at a distance”, and hence, more simply telemedicine is medicine at a distance. As such it encompasses the whole range of medical activities including



diagnosis, treatment and prevention of disease, continuing education of health care providers and consumers, and research and evaluation, performed when distance is an issue.

The most common telemedicine applications today are in transmission of high-resolution X-rays, cardiology, orthopedics, dermatology and psychiatry. Often, interactive video and audio are used for patient consultations and guidance on procedures; sometimes video briefings and records of specific operations are kept on a network in digital form. Groups of physicians, teachers and researchers often “meet” across large distances. Telemedicine also embraces the management of electronic patient records, access to libraries and databases on the Web and on private networks, and extensive use of e-mail by many in the medical profession.

Providing healthcare services via telemedicine offers many advantages. It can make specialty care more accessible to underserved rural and urban populations. Video consultations or medical image from a rural clinic to a specialist can decrease the travel and associated costs for patients. Now with ease in the costs and availability of high end technology it is much easier and faster to implement telemedicine applications for remote areas which are medically underprivileged.

Telemedicine will soon be just another way to see a healthcare professional, just as seeing friends and family while talking to them on the phone is becoming commonplace.

### **3.2 Overview of DICOM Standard**

In order to realize telemedicine, the material that communicated remotely is concerned firstly. There should exist one standard about the material that each part in the telemedicine system would accept it and perform obey it. One big branch of material used in telemedicine system is digital medical images, since modern

medicine relies on it greatly. DICOM standard is such a standard in telemedicine concept that concerning digital medical imaging and communications.

The history of DICOM can be traced back to several decades ago. The digital medical image sources were introduced in the 1970's. And the computers were used in processing these images after their acquisition. These two factors led the American College of Radiology (ACR) and the National Electrical Manufacturers Association (NEMA) to form a joint committee in order to create a standard method for the transmission of medical images and their associated information. This committee, formed in 1983, published in 1985 the ACR-NEMA Standards Publication No. 300-1985. Prior to this, most devices stored images in a proprietary format and transferred files of these proprietary formats over a network or on removable media in order to perform image communication. While the initial versions of the ACR-NEMA effort (version 2.0 was published in 1988) created standardized terminology, an information structure, and unsanctioned file encoding, most of the promise of a standard method of communicating digital image information was not realized until the release of version 3.0 of the standard in 1993. The release of version 3.0 saw the name change to Digital Imaging and Communications in Medicine (DICOM), and numerous enhancements that delivered on the promise of standardized communications. With the enhancements made in DICOM (Version 3.0), the standard is now ready to deliver on its promise not only of permitting the transfer of medical images in a multi-vendor environment, but also facilitating the development and expansion of picture archiving and communication systems (PACS) and interfacing with medical information systems.

The goals of DICOM are to achieve compatibility and to improve workflow efficiency between imaging systems and other information systems in healthcare

environments worldwide. DICOM is a cooperative standard, Therefore the DICOM standard is extremely adaptable, a planned feature that has led to the adoption of DICOM by other specialties that generate images (e.g., pathology, endoscopy, dentistry). DICOM is used or will soon be used by virtually every medical profession that utilizes images within the healthcare industry. These include cardiology, dentistry, endoscopy, mammography, ophthalmology, orthopedics, pathology, pediatrics, radiation therapy, radiology, surgery, etc. Besides this, DICOM standard is modified and improved frequently. Thus, DICOM standard contains as more as possible features to satisfy the different requirements. It is clear that the use of DICOM objects and services in commonly used information technology applications will grow in the future.

My system uses DICOM files as processing material. It manipulates these files in special ways that differ from other system. Therefore I will introduce the structure of DICOM file in the following. The structure is so complex that the standard describes it by using more than one thousand pages. Only with understanding its structure, the manipulations of DICOM file, include reading, writing and repairing of DICOM file, are possible. A single DICOM file contains both a header (which stores information about the patient's name, the type of scan, image dimensions, etc), as well as all of the image data. This is different from the popular Analyze format, which stores the image data in one file (\*.img) and the header data in another file (\*.hdr). Another difference between DICOM and Analyze is that the DICOM image data can be compressed (encapsulated) to reduce the image size. Files can be compressed using lossy or lossless variants of the JPEG format, as well as a lossless Run-Length Encoding format (which is identical to the packed-bits compression found in some TIFF format images).

DICOM file consists of a set of data elements. Header part includes several data elements; those are information related to image. Image data is also contained in one data element, or more data elements if there are more than one part image in this DICOM file. Each data element is stored as depicted in figure 3.1.

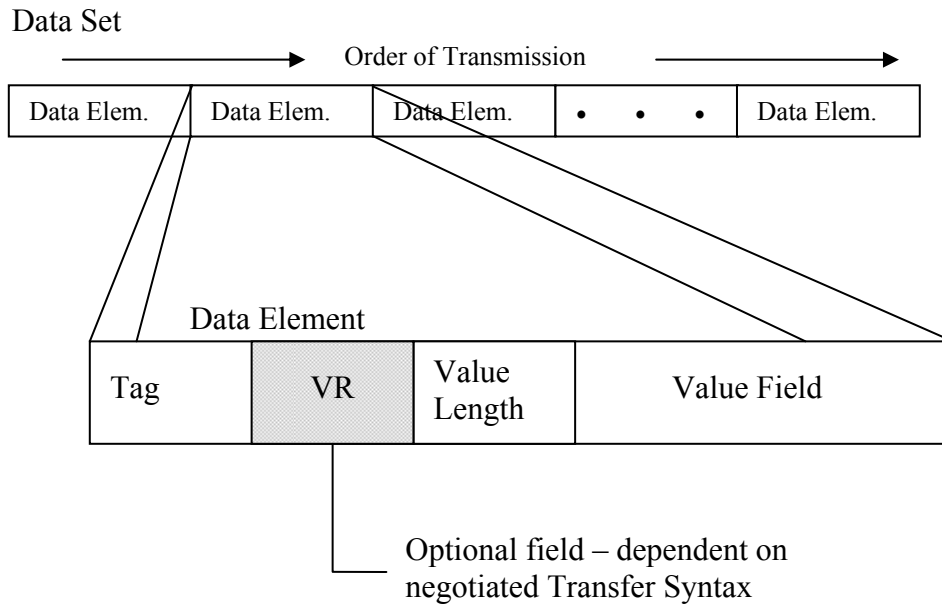


Figure 3.1 DICOM data set structure consists of several data elements. Each data element consists of a tag, a value representation, a value length, and a value field. [NEMA00]

A data element contains a field for the tag number of the DICOM attribute specified in part 6 of the standard. The Value Representation (“VR” in figure 3.1) may be defined depending on negotiated transfer syntax. Afterwards the value length and the value of the attribute follow. There are four kinds of transfer syntaxes, as table 3.1: little endian implicit, big endian implicit, little endian explicit and big endian explicit. This transfer syntax is of particular importance, and is depicted by data element 0002:0010. This value reports the structure of the image data, revealing whether the data has been compressed. DICOM images can be compressed both by the common lossy JPEG compression scheme (where some high frequency

information is lost) as well as a lossless JPEG scheme that is rarely seen outside of medical imaging (this is the original and rare Huffman lossless JPEG, not the more recent and efficient JPEG-LS algorithm). These codes are described in Part 5 of the DICOM standard.

| Transfer Syntax UID    | Definition  |
|------------------------|---|
| 1.2.840.10008.1.2      | Raw data, Implicit VR, Little Endian                                    |
| 1.2.840.10008.1.2.x    | Raw data, Explicit VR<br>x = 1: Little Endian<br>x = 2: Big Endian      |
| 1.2.840.10008.1.2.4.xx | JPEG compression<br>xx = 50-64: Lossy JPEG<br>xx = 65-70: Lossless JPEG |
| 1.2.840.10008.1.2.5    | Lossless Run Length Encoding  |

Table 3.1 Transfer syntax UID table

Note that as well as reporting the compression technique, the Transfer Syntax UID also reports the byte order for raw data. Different computers store integer values differently, so called “big endian” and “little endian” ordering. Consider a 16-bit integer with the value 257: the most significant byte stores the value 01 (=255), while the least significant byte stores the value 02. Some computers would save this value as 01:02, while others will store it as 02:01. Therefore, for data with more than 8-bits per sample, a DICOM viewer may need to swap the byte-order of the data to match the ordering used by specified computer.

The header of a DICOM file is a dataset that is represented by a set of attributes. The following table 3.2 specifies the most essential attributes of the header.

| <b>Attribute Name</b>           | <b>Tag</b>               | <b>Type</b> |
|---------------------------------|--------------------------|-------------|
| File Preamble                   | No Tags or Length Fields | 1           |
| DICOM Prefix                    | No Tags or Length Fields | 1           |
| Group Length                    | (0002,0000)              | 1           |
| File Meta Information Version   | (0002,0001)              | 1           |
| Media Storage SOP Class UID     | (0002,0002)              | 1           |
| Media Storage SOP Instance UID  | (0002,0003)              | 1           |
| Transfer Syntax UID             | (0002,0010)              | 1           |
| Implementation Class UID        | (0002,0012)              | 1           |
| Implementation Version Name     | (0002,0013)              | 3           |
| Source Application Entity Title | (0002,0016)              | 3           |
| Private Information Creator UID | (0002,0100)              | 3           |
| Private Information             | (0002,0102)              | 1C          |

Table 3.2 Some tag groups in DICOM header file [NEMA00]

The DICOM header starts with 128 empty bytes. In every DICOM file, there are 128 empty bytes at first. It is preserved for the future use. Behind the 128 Bytes 4 bytes follow. These four bytes contain the character string “DICM”. The prefix is intended to be used to recognize that this file is a DICOM file or not. The Group Length is used because of the optional attributes and therefore the header does not have a fixed size. The Metafile version has a fix value. Afterwards the information what DICOM class and instance is contained in the file follows. Each kind of image (CT is computer tomography, MR is magnetic resonance tomography) has an identifier as well as the instance of such a class.

After the header a dataset follows, which represents the content of the file. The dataset can be an image, a presentation state, a structured report or another DICOM object. All data elements of DICOM header and DICOM document content contain standard attributes which group number of the tag is even. The datasets may contain

private attributes which tags have odd group numbers. Some attributes are not used in newer versions of the standard and therefore they are defined as retired.

In addition to the Transfer Syntax UID, the image is also specified by the Samples per Pixel (0028:0002), Photometric Interpretation (0028:0004) and the Bits Allocated (0028:0100). For most MRI and CT images, the photometric interpretation is a continuous monochrome (e.g. typically depicted with pixels in grayscale). In DICOM, these monochrome images are given a photometric interpretation of “MONOCHROME1” (low values=bright, high values=dim) or “MONOCHROME2” (low values=dim, high values=bright). However, many ultrasound images and medical photographs include color, and these are described by different photometric interpretations (e.g. Palette, RGB, CMYK, YBR, etc). Some color images (e.g. RGB) store 3-samples per pixel (each for red, green and blue), while monochrome and palette images typically store only one sample per image. These images store 8-bits (256 levels) or 16-bits per sample (65,535 levels), though some scanners save data in 12-bit or 32-bit resolution. So a RGB image that stores 3 samples per pixel at 8-bits per byte can potentially describe 16 million colors (256 cubed).

People familiar with the medical imaging typically talk about the “window centre” and the “window width” of an image. This is simply a way of describing the “brightness” and “contrast” of the image. These values are particularly important for X-ray/CT/PET scanners that tend to generate consistently calibrated intensities so you can use a specific C: W pair for every image you see (e.g. 400:2000 might be good for visualizing bone, while 50:350 might be a better choice for soft tissue). Note that contrast in MRI scanners is relative, and so a C: W pair that works well for one protocol will probably be useless with a different protocol or on a different scanner. The figure 3.2 illustrates the concept of changes to “window centre” and “window

width”. Along the top row you can see three views of the same image with different C: W settings. The bottom row illustrates the colour mapping for each image (with the vertical axis of the graph showing rendered brightness and the horizontal axis showing the image intensity). Consider this image with intensities ranging from 0 to 170. A good starting estimate for this image might be a centre of 85 (mean intensity) and width of 171 (range of values), as shown in the middle panel. Reducing the width to 71 would increase the contrast (left panel). On the other hand, keeping a width of 171 but reducing the centre to 40 would make the whole image appear brighter.

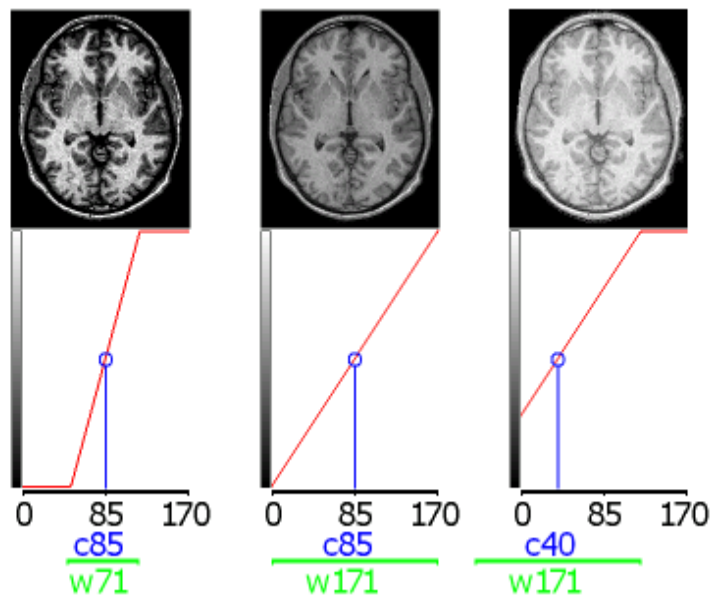


Figure 3.2 Example of changing window width and window centre

### 3.3 TeleMed-VS Manipulates DICOM Files

TeleMed-VS implies powerful and flexible methods to manipulate the DICOM files, such as reading the files to get image and related information, writing one image to DICOM file format with some related information, and repairing DICOM files to remove private and retired tag groups. This repairing operation is particularly designed in my work that differs from the other systems.



For the implementation of reading procedure, the system creates one data dictionary, which stores all kinds of tag groups in current standard. According to DICOM standard, DICOM file consists of a set of data elements. With this data dictionary, every data element can be read correctly. Some files may be very simple, which only has a few necessary tag groups and image data. Others are quite complex, that may be nested with other files or include overlay image layers, and so on. My system can handle those “complex” files, and give header information as while as render image. It can also read a set of files, which are a series of parallel slices of one object. To be easy to handle the same files in the future, the system creates one “group” file to store these series so that user can just select the “group” file next time to open the series. The figure 3.3 illustrates one example of reading DICOM file. The left part is header information of the file; the right up corner is image; and the right down corner is to render the image as different layer if there are more than 8 bits in image data, for some file stores different image in different layer.

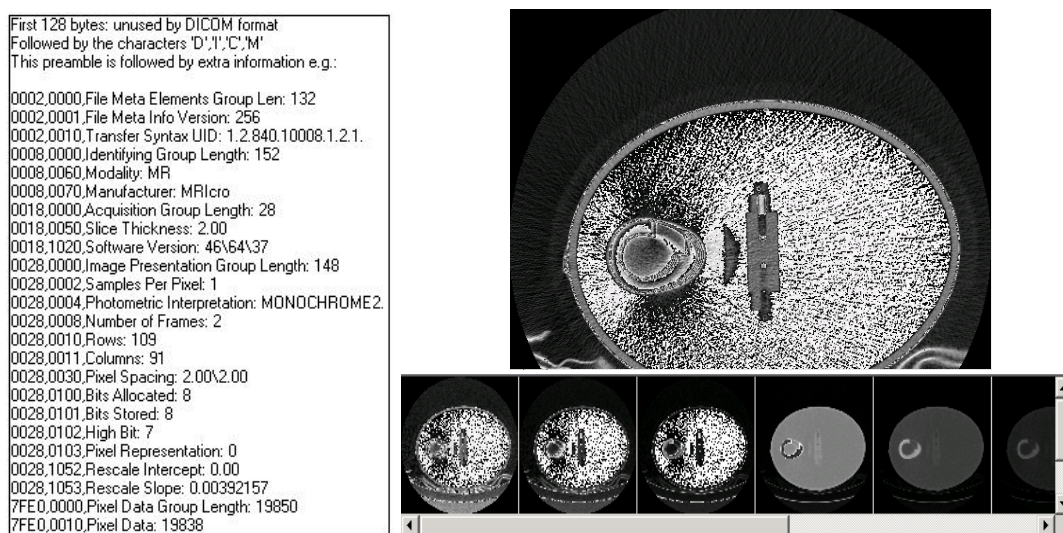


Figure 3.3 Example of read DICOM file

Writing DICOM file is just a converse procedure of reading DICOM file. For header part, some tag groups are necessary for DICOM file. If the new file is derived from other DICOM file, some information can be borrowed from the source file. The image information, such as modality, size, position, and bit allocated, can be gotten from the image itself. And image data can be written according to transfer syntax. The writing procedure can be used in DICOM file repairing process or reslicing volume data that will be depicted later. The DICOM files written by this system are complete, so that they can be read as well as by other DICOM read applications.

The aim of repairing DICOM file is to enable the communication with nonnormal DICOM files. The means of storing image data differs from retailer to retailer, therefore some DICOM files contain private tag groups, which can not be read by popular DICOM reader applications, except the file providers. This is not encouraged in communication between different systems. Besides, some DICOM files possibly contain retired tag groups since they are too old or created according to retired standard. Both kinds of DICOM files are nonstandard files. While looking for a DICOM viewer in the internet, there are many offered solutions. But there is no application that supports all DICOM images. Because DICOM applications normally have to handle each piece of data, such as attributes, tags .etc, not only standard part but nonstandard ones. Most of the viewers will fail while reading files with private or retired attributes. This is the reason why I developed a solution to repair such DICOM files. The repair procedure removes those nonstandard attributes like private and retired attributes. It keeps the necessary and standard attributes, and writes them to a new DICOM file. After performing this process each popular viewer is able to display those standardized images. The contribution about this point has been published in [VJM02] and [JVM02] in the international conferences.

### **3.4 A New Supplement for DICOM on 3D Image Data Storage**

While people mention “image”, their reflex normally is just a 2D picture. Thus, when we want to describe an image which is more than 2D, we like to call 3D or 4D, or simply say multi-dimension (MD). The domain of this supplement is multi-dimensional (MD) instances created during acquisition, post-processing, interpretation and treatment. A growing number of applications create and work with more and more complex data types for which there is no representation within DICOM. The supplement provides a way to encode objects with dimensions of space, time, acquisition context, or measured properties (channels). It is intended for composite data objects of any modality or clinical specialty. The framework for multi-dimensional encoding has sufficient power and extensibility to meet the needs of applications anticipated in the foreseeable future.

The motivation for this supplement stems from several DICOM limitations. There is no multi-dimension object definition in DICOM standard currently. Spatial image data is described as an ordered set of 2D arrays of pixels in DICOM, each of which may have multiple components of the same size and representation. For example, because the DICOM standard is lack of spatial coordinate system, there is no representation for direction dependent quantities, such as spatial vectors or higher order tensors that are required for describing motion of image or diffusion tensor imaging. These limitations restrict the usage of some applications for more complex performances.

The proposed supplement in this thesis introduces a new method for representing multi-dimensional data of any clinical specialty, modality, or post-processing application. It supports and extends the multi-frame encoding

approach for viewable image data. For encoding medical imaging data with spatial, this method is a modality independent way.

Normally 2D image has width and height dimension. And the image data is stored in an array with size “width×height” stored bit by bit from left to right in a line and from top to bottom line by line. For 3D image, another dimension should be added, which is depth dimension. The 3D image format should be width×height×depth. Dimension indices typically scale to Real World Domains, particularly space and time. One new sequence should be defined in standard, which contains items that describe the mapping of the dimension indices to their Real World Domains. Each mapping uses a mapping macro—time, space, generic scaled, or other—that identifies the Real World Domain, its units, and the array dimension, expressed as an ordinal number starting from the most to least frequently varying indices.

The representation of 3D image is set in Cartesian coordinate system. Similarly, the Cartesian spatial mapping macro will be used to map one, two or three array dimensions to the dimensions of the spatial reference coordinate system. Cartesian mapping uses a 4x4 homogeneous matrix to realize the spatial scaling. Non-Cartesian spatial mappings (e.g. spherical, projection or other) would require mapping macros in order to be created for the particular sampling geometry. And the patient’s orientation is also defined according to this reference coordinate system. For example, the value shall be L/P/H to indicate that the Reference Coordinate System is patient aligned according to DICOM convention that X is left, Y is posterior, and Z is toward the head. If the orientation relative to the patient is unknown, this attribute will not be present. Each value of the orientation attribute shall contain at least one of these characters. When present, one or two additional letters in each value specify

refinements in the orientation. Within each value, the order of the letters is with the principal orientation designated in the first character.

For 3D object rendering, one important feature is the rotation, which is to show different views of the 3D object. The application that renders the 3D image can rotate the object by multiplying rotating matrix. When the physician wants to annotate something at one specific angle, the annotation and the orientation should be recorded and coded in DICOM file. The rotating matrix specifies a Cartesian mapping of index or stored values to the reference coordinate system. Matrix elements will be listed in row-major order. The matrix is like:

$$\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} M_{11} & M_{12} & M_{13} & T_x \\ M_{21} & M_{22} & M_{23} & T_y \\ M_{31} & M_{32} & M_{33} & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} Q_1 \\ Q_2 \\ Q_3 \\ 1 \end{bmatrix} \quad (3.1)$$

The matrix coefficients will appear so that the column index varies most frequently:  $M_{x1}$ ,  $M_{x2}$ ,  $M_{x3}$ ,  $T_x$ ,  $M_{y1}$ ,  $M_{y2}$ ,  $M_{y3}$ ,  $T_y$ , etc.  $M_{ij}$  defines the rotational orientation and  $T_k$  defines the start position. The matrix shall obey orthogonality constraints for each of  $j=1,2,3$  and  $k=1,2,3$ . When applied to the indices of component array dimensions, the Q value is a particular array dimension identified by its order (1,2, ...) as specified in the mapped dimensions. When applied to component values, it is the component value descriptor sequence item ordinarily.

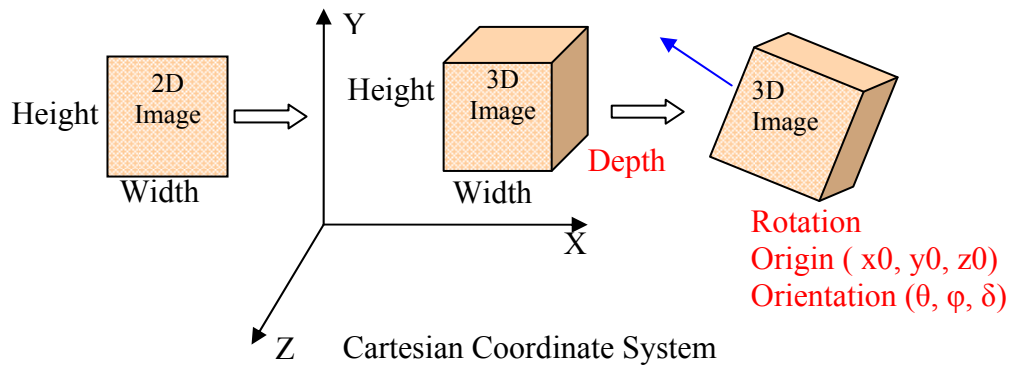


Figure 3.4 3D image data definition

As figure 3.4 shows, 2D image data is defined by width and height character in a plane coordinate system. For 3D image data, the third character is added, which is the depth. The reference coordinate system is right-hand Cartesian coordinate system. If the reference coordinate system is defined, the operations on the 3D image could have rules. Every 3D image object has the orientation in the reference coordinate system. This orientation can be defined by the angle between the orientation direction and the  $x, y, z$  axis. It is  $(\theta, \varphi, \delta)$ . The 3D image can be rotated in the reference coordinate system. The rotation angle is also defined as the rotated degree according to three axes.

Some new attributes are added to standard as this supplement [JVM03]. Those are described in table 3.3. These new features are about 3D image data concerning storage and movement in Cartesian coordinate system.

| Attribute Name | Tag          | Type | Attribute Description  |
|----------------|--------------|------|--|
| Dimension      | (xxxx, xxxx) | 1    | Indicate if the image is 3D  |
| Depths         | (0028,0012)  | 1c   | Number of depths in the image  |
| Origin         | (xxxx, xxxx) | 1    | The origin of 3D image in reference coordinate system  |
| Orientation    | (0020,0037)  | 1    | The direction cosines of the first row, the first column and first depth with respect to the reference coordinate system |
| Rotation       | (xxxx, xxxx) | 1    | The direction cosines of rotated angles with respect to the reference coordinate system                                  |

Table 3.3 New attributes added for the supplement

### 3.5 Chapter Summary

In this chapter I present the telemedicine integrated system. I introduce the concept of telemedicine firstly. It is the motivation that I develop this system. I also introduce DICOM standard, for DICOM file is a main kind of files that this system can process. This system is not only a simple DICOM viewer, but also does some modifications on files. It can write new DICOM file according to the standard. This feature provides a new capability for the system that it can standardize some old or non-standard DICOM files. After the standardization procedure, the old files or the files containing some private attributes can be treated as other normal files.

Besides, a new supplement is proposed for multi-dimension image data storage. As acquisition technique developed to multi-dimensional (3D or more), DICOM standard should be updated to follow this trend, and many new related attributes should be defined along with this development. In the proposed supplement, I defined the storage format and reference coordinate system so that some actions can

be performed on the 3D data object, such as translation or rotation. Surely it is just one aspect of the whole requirements, and other related definitions are still needed. In addition, I also suggest some new attributes to support those performances. These new attributes should be supplemented to the standard.



## **CHAPTER IV**

### **THREE RENDERING WAYS FOR MEDICAL IMAGE**

Image visualization of tomography volume data, as obtained in computer tomography (CT) or magnetic resonance imaging (MRI), is an important aid for diagnosis, treatment planning, surgery rehearsal, education, and research. For the medical image processing system, visualization of image is a necessary component. It gives the first impression for users whether such visualization mode reveals the information inside the image. Every physician wants to get as much as possible information from the image shown on screen to help the diagnosis of the disease, so that the visualization must be comprehensive and accurate. For these reasons, interactive manipulation and high quality rendering are essential features for the application in a clinical environment. It is also the main goal of computer aided diagnosis system. The visualization techniques are discussed in many medical image processing systems, also in some surgical planning system [PDMCEO96], [HDKJW97] and surgical guidance [GNK+01]. The visualization of image data includes the raw data rendering, and also processed data rendering.

In my system, there have been key advances in the three main approaches to the visualization of volumetric data: volume rendering, iso-surfacing and cut through, which all together make up the field of medical image visualization. These three visualization manners reveal both the local information within single slice images and 3D representation conveying spatial information of lesions and related structures. Every method shows different aspect character of image data, and supplies each other

to make the observer understand the data better. Besides, these three visualization methods performed in my system are in real-time rate. User can operate the visualized object intuitively, for example, 3D rotation around an arbitrary rotation axis, zooming, translating and etc. This interactive visualization is one of the most important means for the investigation of tomography data resulting from CT and MR scanners. Volume rendering method shows the whole data in three-dimension by setting semi-transparent character to data so that the whole data can be rendered without deletion or hiding. Iso-surface rendering is to show certain part of the whole data, which has some similarities. The surface could be one organ's surface by segmentation or just iso-value part of the raw data simply. The cut through method also renders part of raw data. It shows the cross-section of the data that is cut by a plane in arbitrary degree, and gives different view of the data comparing to the acquisitions. In this chapter, different visualization methods will be introduced. And the results of each method in the system will be shown

## **4.1 Volume Rendering**

The term “volume rendering” is used to describe techniques which allow the visualization of three-dimensional data directly, i.e. without first fitting geometric primitives to it. Volume rendering is a technique for visualizing sampled functions of three spatial dimensions by computing 2D projections of a colored semi-transparent volume. It solves one typical visualization problem in the medical context that tissues have no defined surface. Also, the visualization of semi-transparent objects, where objects can be visualized within their anatomical context, has shown to give decisive information for successful diagnosis or therapy. Currently, the major application area of volume rendering is medical imaging, where volume data is available from X-ray Computer Tomography (CT) scanners and Positron Emission Tomography (PET)

scanners. CT scanners produce three-dimensional stacks of parallel plane images, each of which consist of an array of X-ray absorption coefficients. In the two-dimensional domain, these slides can be viewed one at a time. The advantage of CT images over conventional X-ray images is that they only contain information from that one plane. A conventional X-ray image, on the other hand, contains information from all the planes, and the result is an accumulation of shadows that are a function of the density of the tissue, bone, organs, etc., anything that absorbs the X-rays.

Direct volume rendering denotes a set of techniques used to directly display volume data, where the images are generated through the transformation, shading, and projection of 3D voxels onto 2D pixels [Kau91]. A subset of direct volume rendering techniques is based on the ray casting algorithm introduced by [Lev90], in which a color and an opacity is assigned to each voxel, and a 2D projection of the resulting colored semi-transparent volume is computed. The principal advantages of these techniques over other visualization methods are their superior image quality and the ability to generate images without explicitly defining surface geometry. The principal drawback of these techniques is their cost. Since all voxels participate in the generation of each image, rendering time grows linearly with the size of the dataset. The basic ray casting principle and some optimized algorithms are described in details in the appendix A.

I use the optimized ray casting algorithm to realize volume rendering in the system. The reduction in image generation time obtained by applying these optimizations is highly dependent on the depth complexity of the scene. I focus on visualizations consisting of opaque or semi-transparent surfaces. A plot of opacity along a line perpendicular to one of these surfaces typically exhibits a bump shape with several voxels wide, and voxels not in the vicinity of surfaces have opacity of

zero. For these scenes, savings of up to an order of magnitude over brute - force rendering algorithms have been observed. For scenes consisting solely of opaque surfaces, the cost of generating images has been observed to grow nearly linearly with the size of the image rather than linearly with the size of the dataset.

Here I will give some results by using the optimized ray casting volume rendering algorithm, shown as figure 4.5 and 4.6. The figure 4.5 shows one MRI data set, volume space is (256, 256, 109). And another MRI data set shown in the figure 4.6 is in volume space (256, 256, 127). This algorithm employs both hierarchical spatial enumeration and adaptive termination of ray casting to reduce rendering costs, which are presented in details in appendix A. Any opacity assignment operator that partitions a volume dataset into coherent regions of opaque and transparent voxels is a candidate for this algorithm. Although the amount of time saved depends on the depth complexity of the partitioned scene, savings of more than an order of magnitude have been observed for many datasets.

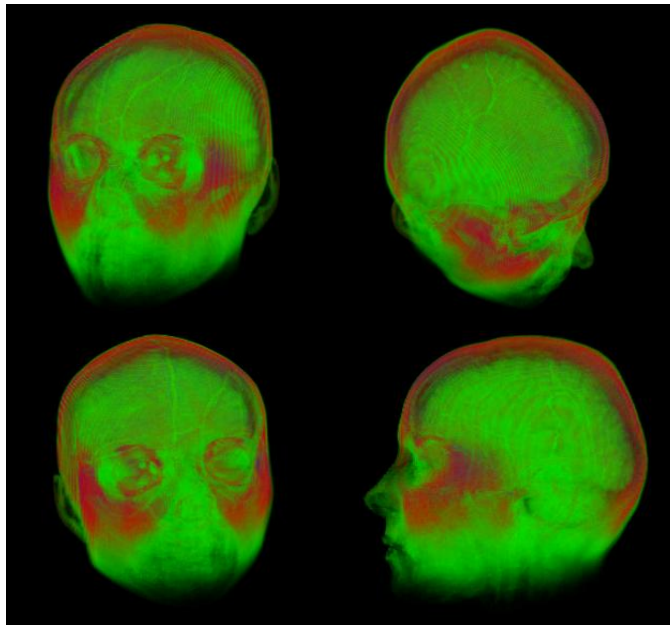


Figure 4.5 Ray casting volume rendering example

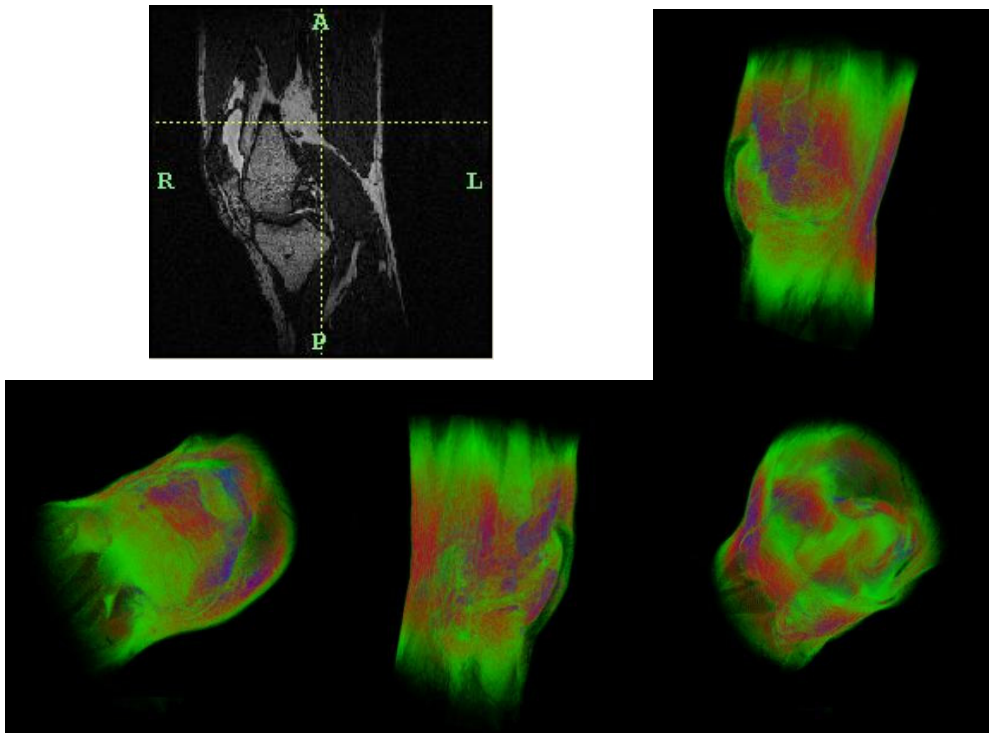


Figure 4.6 Another ray casting volume rendering example

## 4.2 Surface Rendering

Rendering 3D surface of anatomy is valuable in medicine context. The geometry surface extracted from 3D data visualizes the 3D anatomic structures, which are sometime difficult to be pictured mentally. Surface rendering is typically faster than volume rendering, since it only travels the whole volume data once to extract surface primitives, such as polygons or patches. After extracting the surfaces, rendering hardware and well-known rendering method can be used to quickly render the surface primitives each time when the user changes a viewing or lighting parameters. Furthermore, the extracted surface meshes could be used in finite element simulation.

One kind of surface rendering algorithm for getting surface meshes is typically fitting surface primitives to constant value contour surface in volumetric datasets. Firstly, a threshold is chosen, and then geometric primitives are automatically fit for

the high-contrast contours in the volume data that match the threshold. Appendix B gives a brief review of the related algorithms. This method is used in my system for CT data visualization, since it is easy to define one meaningful threshold for this modality, such as skin or bone. Those organs have relative fixed value in CT data. However, for some soft tissues in MRI data, it is difficult to get their surfaces by given a threshold. The complex segmentation algorithms are used for this aim. For rendering the segmented contours, the other method is employed that is contour-connecting. It is one of the first-invented methods for generating 3D surface visualization [Kep75]. And the descendents of contour-connecting algorithms are still in use in many disciplines [CK02]. The basic idea is to trace one closed contour in each slice, and then connect contours in adjacent slices. The optimal tessellation, usually of triangles, is used for connecting the curves in each two adjacent slices. The algorithms find an approximation to the surface passing through the high-gradient cells in the data.

In my system these surface rendering algorithms are employed. The figure 4.7 shows surface shell extracted by marching cube method. The top row is bone surface, and the bottom row is skin surface. The original image data is a CT data set with 94 slices. In these slices, the greyscale value of the whole skin or bone is a stable number. Using this number as threshold, the surface meshes can be extracted. Figure 4.8 shows another example. Here the object to be rendered as surface is brain tumour, shown in left picture in figure 4.8. It is not easy to be extracted merely by threshold. My system firstly segments the tumour with the approach presented later in this thesis. And the right picture shows the surface rendering result by contour-connecting method.

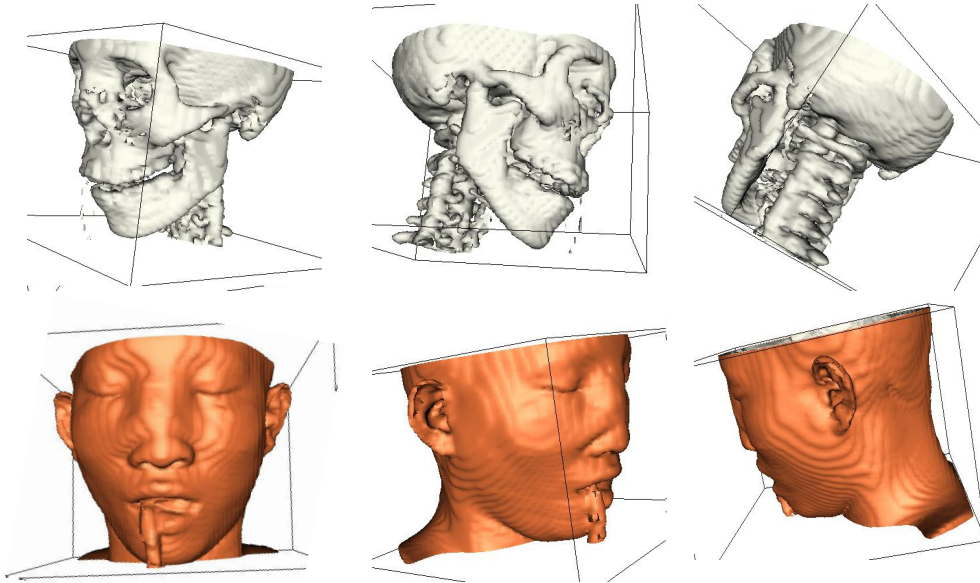


Figure 4.7 Iso-surface rendering example

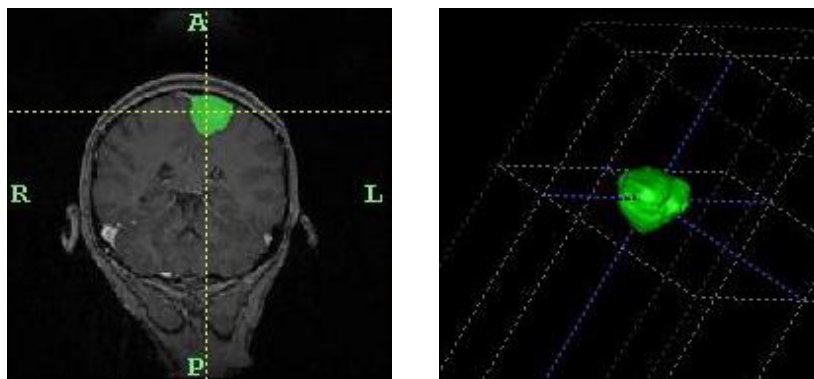


Figure 4.8 Contour-connecting rendering example

### 4.3 Cut Through Rendering

Cut through rendering is to cut the volume data with an arbitrary degree slicing plane. By removing the cut volume in one side of the plane, the cross section is shown on the screen. This rendering method offers seeing-through capability by moving the cut

plane along its perpendicular axis or adjusting the plane's degree. The cross section can be stored as a new slice. The generation of arbitrary slices of volume data is important for medical applications, because it doesn't need additional data acquirement. Hence, it saves the cost. Besides, some arbitrary slices are physically impossible to be acquired. In my system these slices are stored in term of the DICOM format standard, therefore they can be used for analysis and communication in the future.

The naive approach to cut through is to do an exhaustive examination of all the voxels of the volume data to test for intersection with the slicing plane. If the slicing plane is orthogonal to one of the principle axes, the intersection grid are rectilinear. It is easy to speedup the presorting of the grid for rendering. However, if the slicing plane is arbitrary, then the intersection grid is unstructured. In this case, the similar approach can be used for presorting if the orientation of the slicing plane is known. Normally the position of the slicing plane is set by user, so that the orientation is gained during the user's actions. The principle of this speedup approach is that the rotational component of the slicing plane from the  $xy$ -plane is calculated and its inverse is applied to each voxel in turn and its minimum and maximum  $z$ -component recorded along with the voxel ID. Once all voxels have been inspected a search tree can be constructed, e.g. a Kd-tree [Ben75], based on the minimum/maximum  $z$ -values of the voxels. The query value then is simply the distance from the origin to the slicing plane. The tree can be used to quickly find slices at any distance from the origin as long as they keep the original orientation. This allows planes to be rapidly swept through the data to give a good impression of the structures within. This method can be used for planes of arbitrary orientation on any unstructured grid.



Figure 4.9 shows one example. The left object with color crosshair is one 3d image data set. It is a deformed jaw MRI data set. The right color transparent cube with red plane is reference cube to show the position of cut plane. The bottom four images is the cross section images when cut plane lies in different positions.

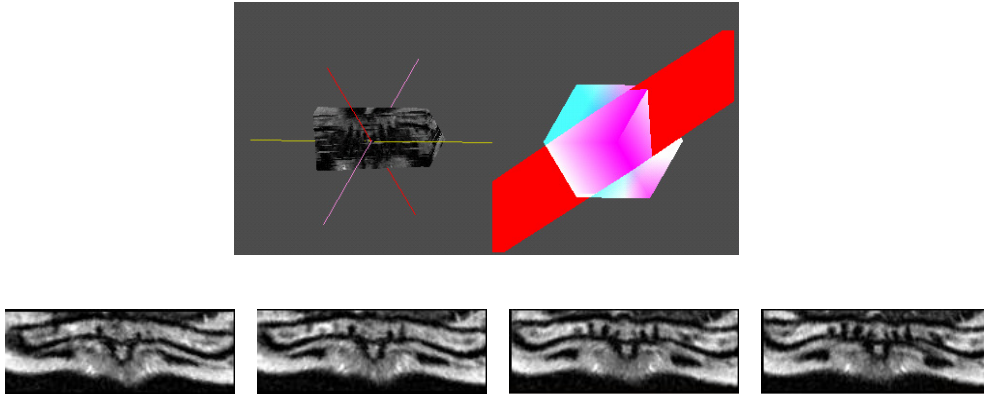


Figure 4.9 Cut through rendering example

#### 4.4 Chapter Summary

In this chapter, I introduce the different volume rendering methods those are applied in my system. The subject of volume visualization has come a long way over the past twenty years. The main approaches of iso-surface rendering through marching cubes, and volume rendering through ray casting and splatting were all crystallized during the late 1980s, but the past decade has seen these approaches develop a maturity - in terms of robustness, accuracy and performance.

There are strong advocates of the surface extraction approach, arguing that it gives excellent definition of features within a dataset, and exploits polygon rendering hardware to give fast performance. There are equally strong voices arguing for the volume rendering approach, and their case is strengthened by the new hardware developments. The truth is probably that both approaches are useful and the winner in

all the competition between researchers is the user - who now has a battery of very powerful techniques to apply to any volume visualization problem.

My system employs both surface extraction and volume rendering techniques to explore different character aspects of medical image data. It is essential part for medical image processing system. Although 3D rendering can give deep impression to users, 2D rendering is still non - replaceable since it shows planar information of image by original data and by cut through slicing rendering. The different rendering manners offered by the system gives completed visualization function.

## **CHAPTER V**

### **A NEW HYBRID APPROACH OF SEGMENTATION**

In this chapter, I will focus on the medical image segmentation, which is one important component in my research. Today, the role of medical imaging is not limited to simple visualization and inspection of anatomic structures, but goes beyond that to patient diagnosis, advanced surgical planning and simulation, radiotherapy planning etc. Although modern volume visualization techniques provide extremely accurate and high quality 3D view of anatomical structures, as introduced in last chapter, their utilization for accurate and efficient analysis is still limited.

Segmentation in medical imaging is generally considered a very difficult problem. This difficulty mainly arises due to the sheer size of the datasets coupled with the complexity and variability of the anatomic organs. The situation is worsened by the shortcomings of imaging modalities, such as sampling artefacts, noise, low contrast etc. which may cause the boundaries of anatomical structures to be indistinct and disconnected. Thus the main challenge of segmentation algorithms is to accurately extract the boundary of the organ or region-of-interest and separate it out from the rest of the dataset.

There are many approaches for segmentation proposed in literature. These approaches differ widely in several aspects, such as the specific application, imaging modality (CT, MRI, etc.), and other factors. For example, the segmentation of lungs has different issues than the segmentation of colon. The same algorithm which gives excellent results for one application, might not even work for another. Besides these,

general imaging artefacts like noise, motion and partial volume effect can significantly affect the outcome of a segmentation algorithm. For example, a segmentation algorithm could be robust against noise, but at the same time, it might fail miserably in the presence of partial volume effects. This variability is what makes segmentation a very challenging problem. There is currently no segmentation method that provides acceptable results for every type of medical dataset. There are methods in existence which are generalized and can be applied to a variety of data, but on the other hand, methods specialized for the particular problems always give better results.

In the following sections, I will give a review of related works firstly. Then I will introduce my hybrid segmentation approach, which consists of three steps: the first step gets a coarse segmentation sample and generates the homogeneity operator; the second step reclassifies the regions with the operator so that the boundary of target object is found; the third step refines the boundary got by the last step. These three steps improve the segmentation result progressively. This hybrid approach is appropriate for the complex structure segmentation, such as brain tissue, or brain tumour.

## **5.1 Introduction of Related Work**

The number of segmentation algorithms found in the literature is very high. Due to the nature of the problem of segmentation, most of these algorithms are specific to a particular problem, thus, having little significance for most other problems. I will try to cover all the algorithms that have a generalized scope and are the basis of most of the segmentation techniques today. Also, I refer only to the most commonly used radiological modalities for imaging anatomy: magnetic resonance imaging (MRI), X-ray computed tomography (CT), ultrasound, and X-ray projection radiography.

Most of the concepts described here, however, are applicable to other imaging modalities as well.

### 5.1.1 Terminology Related to Segmentation

#### Image Segmentation

Classically, image segmentation is defined as the partitioning of an image into non-overlapping, constituent regions which are homogeneous with respect to some characteristic such as intensity or texture [GW92][HS85]. If the domain of the image is given by  $I$ , then the segmentation problem is to determine the sets  $S_k \subset I$  whose union is the entire image  $I$ . Thus, the sets that make up segmentation must satisfy

$$I = \bigcup_{k=1}^K S_k \quad (5.1)$$

where  $S_k \cap S_j = \emptyset$  for  $k \neq j$ , and each  $S_k$  is connected. Ideally, a segmentation method finds those sets that correspond to distinct anatomical structures or regions of interest in the image.

When the constraint that regions be connected is removed, then determining the sets  $S_k$  is called pixel classification and the sets themselves are called classes. Pixel classification rather than classical segmentation is often a desirable goal in medical images, particularly when disconnected regions belonging to the same tissue class need to be identified. Determination of the total number of classes  $K$  in pixel classification can be a difficult problem. Often, the value of  $K$  is assumed to be known based on prior knowledge of the anatomy being considered.

#### Dimensionality

Dimensionality refers to whether a segmentation method operates in a 2D image domain or a 3D image domain. Methods that rely solely on image intensities are independent of the image domain. However, certain methods such as deformable

models, Markov random fields, and region growing, incorporate spatial information, and might therefore operate differently depending on the dimensionality of the image. Generally, 2D methods are applied to 2D images and 3D methods are applied to 3D images. In some cases, however, 2D methods can be applied sequentially to the slices of a 3D image [PPDX97], and my system also follows this rule in order to segment 3D object. For image data situation reason, if the image acquirement is not dense enough, that is, the resolution in one axis is much lower than in the other two axes, 3D methods can not work correctly. For the practical reasons, 2D methods have some superior characters, such as ease of implementation, lower computational complexity, and reduced memory requirements. In addition, certain structures are more easily defined along 2D slices.

### **Continuous or Discrete Segmentation**

Nearly all medical images used for image segmentation are represented as discrete samples on a uniform grid. Segmentation methods typically operate on the same discrete grid as the image. However, certain methods such as deformable models are capable of operating in the continuous spatial domain, thereby providing the potential for sub-pixel accuracy in delineating structures. Sub-pixel accuracy is desirable particularly when the resolution of the image is on the same order of magnitude as the structure of interest.

Although continuous segmentation methods have sub-pixel or sub-voxel resolution, their precision and accuracy are still dependent on the resolution of the original data. Furthermore, this level of precision can be difficult to validate on real data.

### **Interaction**

The trade off between manual interaction and performance is an important consideration in any segmentation application. Manual interaction can improve accuracy by incorporating prior knowledge of an operator. However, for large population studies, this can be laborious and time consuming.

The type of interaction required by segmentation methods can range from completely manual delineation of an anatomical structure, to the selection of a seed point for a region growing algorithm, which the hybrid segmentation approach described in this thesis belongs to. The differences in these types of interaction are the amount of time and effort required, as well as the amount of training required by an operator. Methods that rely on manual interaction can also be vulnerable to reliability issues. However, even “automated” segmentation methods typically require some interaction for specifying initial parameters that can significantly affect performance.

### **Validation**

In order to quantify the performance of a segmentation method, validation experiments are necessary. Validation is typically performed using as the following two different types of truth models. The validation for my hybrid approach uses both of two types that will be discussed later in this chapter. The most straightforward approach to validation is by comparing the automated segmentations with manually obtained segmentations (cf. [WZW02]). This approach, besides suffering from the drawbacks outlined in the previous section, does not guarantee a perfect truth model since an operator’s performance can also be flawed. The other common approach to validating segmentation methods is through the use of physical phantoms [LS92] or computational phantoms [CZKSK98]. Physical phantoms provide an accurate depiction of the image acquisition process but typically do not present a realistic

representation of anatomy. Computational phantoms can be more realistic in this latter regard, but simulate the image acquisition process using only simplified models.

Once a truth model is available, a figure of merit must be defined for quantifying accuracy or precision. The choice of the figure of merit is dependent on the application and can be based on region information such as the number of pixels misclassified, or boundary information such as distance to the true boundary.

### **5.1.2 Classification of Segmentation Methods**

In this section, I briefly describe several common approaches that have appeared in the recent literature on medical image segmentation. This classification defines each method, provides an overview of how the method is implemented, and discusses its advantages and disadvantages. Although each technique is described separately, multiple techniques are often used in conjunction with one another for solving different segmentation problems.

The image segmentation methods can be sorted as two main classes, region-based methods and edge-based methods. For the description convenient, I divide segmentation methods into eight categories: (1) thresholding approaches, (2) region growing approaches, (3) classifiers, (4) clustering approaches, (5) Markov random field models, (6) artificial neural networks, (7) deformable models, and (8) atlas guided approaches. The first four categories belong to region-based, and the other four categories belong to edge-based methods.

#### **Thresholding**

Thresholding approaches segment scalar images by creating a binary partitioning of the image intensities. Figure 5.1(a) shows the histogram of a scalar image that possesses three apparent classes corresponding to the three modes. A thresholding procedure attempts to determine an intensity value, called the threshold, which



separates the desired classes. The segmentation is then achieved by grouping all pixels with intensity greater than the threshold into one class, and all other pixels into another class. Two potential thresholds are shown in Figure 5.1(a) at the valleys of the histogram. Determination of more than one threshold value is a process called multi-thresholding [SSW88].

Thresholding is a simple yet often effective means for obtaining the segmentation in images where different structures have contrasting intensities or other quantifiable features. The partition is usually generated interactively, although automated methods do exist [SSW88]. Thresholding is often used as an initial step in a sequence of image processing operations. Its main limitations are that in its simplest form only two classes are generated and it can not be applied to multi-channel images. In addition, thresholding typically does not take into account the spatial characteristics of an image. This causes it to be sensitive to noise and intensity inhomogeneities, which can occur in magnetic resonance images.

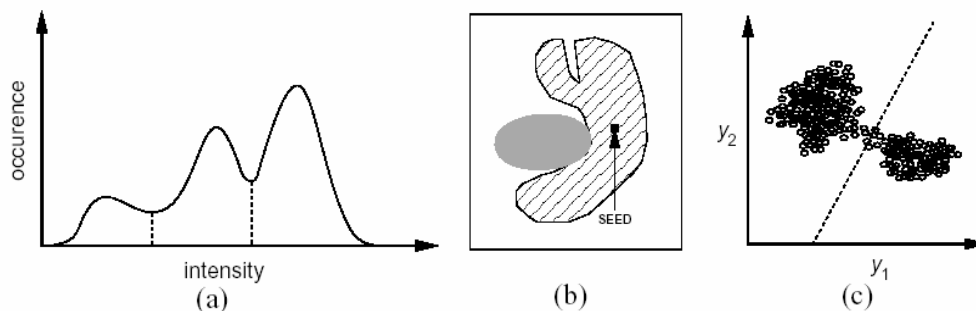


Figure 5.1 Feature space methods and region growing: (a) a histogram showing three apparent classes, (b) a 2-D feature space, (c) example of region growing.

### Region Growing

Region growing is a technique for extracting a region of the image that is connected based on some predefined criteria. These criteria can be based on intensity

information and/or edges in the image [HS85]. This is depicted in Figure 5.1(b), where region growing has been used to isolate one of the structures as seed put inside.

Like thresholding, region growing is not often used alone but within a set of image processing operations, particularly for the delineation of small, simple structures such as tumours and lesions [GBBH96]. Its primary disadvantage is that it requires manual interaction to obtain the seed point. Thus, for each region that needs to be extracted, a seed must be planted. Split and merge algorithms are related to region growing but do not require a seed point [MUCR98]. Region growing can also be sensitive to noise, causing extracted regions to have holes or even become disconnected.

### **Classifiers**

Classifier methods are pattern recognition techniques that seek to partition a feature space derived from the image using data with known labels [BHC93]. A histogram, as shown in Figure 5.1(a), is an example of a 1-D feature space. Figure 5.1(c) shows an example of a partitioned 2-D feature space with two apparent classes. All pixels with their associated features on the left side of the partition would be grouped into one class.

Classifiers are known as supervised methods since they require training data that are manually segmented and then used as references for automatically segmenting new data. A simple classifier is the nearest-neighbour classifier, where each pixel or voxel is classified in the same class as the training datum with the closest intensity. A commonly-used parametric classifier is the maximum likelihood (ML) or Bayes classifier [ZD94]. It assumes that the pixel intensities are independent samples from a mixture of probability distributions, usually Gaussian.

Standard classifiers require that the structures to be segmented possess distinct quantifiable features. Being non-iterative, they are relatively computationally efficient and unlike thresholding methods, they can be applied to multi-channel images. A disadvantage of classifiers is that they generally do not perform any spatial modelling. This weakness has been addressed in extending classifier methods to segmenting images that are corrupted by intensity inhomogeneities [KGKW98]. Another disadvantage is the requirement of manual interaction for obtaining training data, which is time consuming and laborious. On the other hand, use of the same training set for a large number of scans can lead to biased results which do not take into account anatomical and physiological variability between different subjects.

### **Clustering**

Clustering algorithms essentially perform the same function as classifier methods without the use of training data. Thus, they are termed unsupervised methods. In order to compensate for the lack of training data, clustering methods iterate between segmenting the image and characterizing the properties of the each class. In a sense, clustering methods train themselves using the available data.

Three commonly used clustering algorithms are the K-means or isodata algorithm [CA79], the fuzzy c-means algorithm [AYMF02], and the expectation-maximization (EM) algorithm [MBLG02]. The K-means clustering algorithm clusters data by iteratively computing a mean intensity for each class and segmenting the image by classifying each pixel in the class with the closest mean. The fuzzy c-means algorithm generalizes the K-means algorithm, allowing for soft segmentations based on fuzzy set theory [Zad65]. The EM algorithm applies the same clustering principles with the underlying assumption that the data follows a Gaussian mixture model. It iterates between computing the posterior probabilities and

computing maximum likelihood estimates of the means, covariances, and mixing coefficients of the mixture model.

Although clustering algorithms do not require training data, they do require an initial segmentation (or equivalently, initial parameters). The EM algorithm has demonstrated greater sensitivity to initialization than the K-means or fuzzy c-means algorithms. Like classifier methods, clustering algorithms do not directly incorporate spatial modelling and can therefore be sensitive to noise and intensity inhomogeneities. This lack of spatial modelling, however, can provide significant advantages for fast computation. Work on improving the robustness of clustering algorithms to intensity inhomogeneities in MR images has demonstrated excellent success [WZW02]. Robustness to noise can be incorporated using Markov random field modelling as described in the next section.

### **Markov Random Field Models**

Markov random field (MRF) modelling itself is not a segmentation method but a statistical model which can be used within segmentation methods. MRFs model spatial interactions between neighbouring or nearby pixels. These local correlations provide a mechanism for modelling a variety of image properties [Li95]. In medical imaging, they are typically used to take into account the fact that most pixels belong to the same class as their neighbouring pixels.

MRFs are often incorporated into clustering segmentation algorithms such as the K-means algorithm under a Bayesian prior model [SG04]. The segmentation is then obtained by maximizing the a posterior probability of the segmentation given the image data using iterative methods such as iterated conditional modes or simulated annealing.

A difficulty associated with MRF models is proper selection of the parameters controlling the strength of spatial interactions [Li95]. Too high a setting can result in an excessively smooth segmentation and a loss of important structural details. In addition, MRF methods usually require computationally intensive algorithms. Despite these disadvantages, MRFs are widely used not only to model segmentation classes, but also to model intensity inhomogeneities that can occur in MR images and texture properties.

### **Artificial Neural Networks**

Artificial neural networks (ANNs) are massively parallel networks of processing elements or nodes that simulate biological learning. Each node in an ANN is capable of performing elementary computations. Learning is achieved through the adaptation of weights assigned to the connections between nodes. A thorough treatment on neural networks can be found in [Hay94].

ANNs represent a paradigm for machine learning and can be used in a variety of ways for image segmentation. The most widely applied use in medical imaging is as a classifier [GFK96], where the weights are determined using training data, and the ANN is then used to segment new data. ANNs can also be used in an unsupervised fashion as a clustering method [BHC93], as well as for deformable models.

Because of the many interconnections used in a neural network, spatial information can easily be incorporated into its classification procedures. Although ANNs are inherently parallel, their processing is usually simulated on a standard serial computer, thus reducing this potential computational advantage.

### **Deformable Models**

Deformable models are physically motivated, model-based techniques for delineating region boundaries using closed parametric curves or surfaces that deform under the

influence of internal and external forces. To delineate an object boundary in an image, a closed curve or surface must first be placed near the desired boundary and then allowed to undergo an iterative relaxation process. Internal forces are computed from within the curve or surface to keep it smooth throughout the deformation. External forces are usually derived from the image to drive the curve or surface towards the desired feature of interest. Mathematically, a deformable model moves according to its dynamic equations and seeks the minimum of a given energy functional [KWT88].

The main advantages of deformable models are their ability to directly generate closed parametric curves or surfaces from images and their incorporation of a smoothness constraint that provides robustness to noise and spurious edges. A disadvantage is that they require manual interaction to place an initial model and choose appropriate parameters. Some extensions of deformable models have been done, such as, [HBG02] reduces sensitivity to initialization, [WZL01] uses pressure forces and other modified external force to reduce model's poor convergence to concave boundaries, [YD04] uses an implicit representation to enhance the adaptability of model topology.

### **Atlas – Guided Approach**

Atlas-guided approach is a powerful tool for medical image segmentation when a standard atlas or template is available. The atlas is generated by compiling information on the anatomy that requires segmenting. This atlas is then used as a reference frame for segmenting new images. Conceptually, atlas-guided approaches are similar to classifiers except they are implemented in the spatial domain of the image rather than in a feature space.

The standard atlas-guided approach treats segmentation as a registration problem. It first finds a one-to-one transformation that maps a pre-segmented atlas

image to the target image that requires segmenting. This process is often referred to as atlas warping. The warping can be performed using linear [LRSFFETM97] transformations but because of anatomical variability, a sequential application of linear and non-linear [CGHPVDT02] transformations is often used. Because the atlas is already segmented, all structural information is transferred to the target image.

An advantage of atlas-guided approaches is that labels are transferred as well as the segmentation. They also provide a standard system for studying morphometrics properties [KWNBJK01]. Even with non-linear registration methods however, accurate segmentations of complex structures is difficult due to anatomical variability. Thus, atlas-guided approaches are generally better-suited for segmentation of structures that are stable over the population of study.

## **5.2 Overview of the Hybrid Approach**

As the description introduced above, different method is appropriate for certain image modality and target object. My research focuses on MR and CT images segmentation. It aims to accurately perform some complex segmentation, such as segmentation on brain structure or brain tumour. In order to reach this aim one new hybrid segmentation approach is presented, which is the integration of both region-based method and edge-based method.

By this hybrid segmentation method, the process starts from an initial seed area in the target object, and uses the hanging togetherness concept to get an estimation of the object's region separating from the background. This sample region also generates the homogeneity operator for next step processing. Then with the Voronoi diagram (VD), the image is redefined to outside, inside or boundary regions of the target with the homogeneity operator. After that, the boundary will be extracted to fill in the missing boundary and to override the spurious boundary data with a

deformable model. This hybrid approach amplifies the strengths of both region-based and edge-based techniques but reduces the weaknesses of them. In the following three sections, I will in detail introduce the each step and corresponding algorithm that composes the hybrid approach.

### **5.3 Generating Homogeneity Operator**

The first step in the hybrid approach is to generate the homogeneity operator. This operator defines the intensity pattern of the target tissue. One affinity relationship between pixels in the image is calculated from the user defined seed area. Using the affinity factor the sample of target tissue is segmented. It derives the binary mask that distinguishes object and background. The homogeneity operator is achieved by measuring the mean and standard deviation of the target object. This operator is used for next processing step. This initial procedure of the hybrid approach is relative simple, and it doesn't need much manual work. The user only needs to draw a rough seed area in the target tissue.

#### **5.3.1 One Characteristic Intensity Pattern: Hanging Togetherness**

Besides the knowledge that the content of one image shows us, the greyscale of the image presents two characters: continuous or discontinuous. The main imaging operations of visualization, manipulation, and analysis are usually aimed toward certain "objects" which are represented in the image data as "continuous" area. When the object of interest is distinctly discontinuous to other objects, it is often possible to segment the object in a hard sense into a binary image. However, such continuity-based strategy does not account for most of the inaccuracies in acquired data. When the resolution of the image is not so good that one can not clearly tell the object's intensity pattern is "continuous" or "discontinuous", another concept about intensity pattern is needed to describe this situation. In this thesis I propose a new



concept: fuzzy hanging togetherness, which describes the character of the image element appropriately when the image is not so accurate. It is considered that medical images captured by devices have inherent inaccuracies [US96]. The degree of this inaccuracy depends on a number of factors including limitations in spatial, temporal, and parametric resolutions and other physical limitations of the device. Therefore hanging togetherness is useful for medical image segmentation.

Compared to those using hard (binary) segmentation, the strategy of hanging togetherness is aim to retain the relative accuracy of data. Its principle is to keep the data inaccuracies as realistically as possible in object representations and subsequently in object renditions and analysis. In order to apply this strategy in the operations of inaccurate image, the object should be defined in an appropriate mathematical format. Fuzzy setting is a good choice for data inaccuracies.

The basic mathematical framework toward this goal should be addressing issues of the following form: How are objects to be defined in a fuzzy setting? How are topological concepts such as connectivity and boundary to be handled in fuzzy situations? What are the algorithms to efficiently extract fuzzy hanging together components and fuzzy boundaries? Although the theory of fuzzy subsets is an appropriate mathematical vehicle for addressing these issues, the published literature on dealing with fuzzy topological notions is limited.

Normally, it is not easy to design an effective image segmentation algorithm for inaccurate image data. The main hard-ships encountered in the design of effective segmentation algorithms are often attributable to the inflexibility of the rigid, often contradicting, and the requirements that attempt to distinguish between object and non-object regions. The flexibility afforded by fuzzy hanging togetherness eases these

requirements. It makes the fuzzy connected component to be a computable alternative to the notion of an object.

### 5.3.2 Defining Affinity for Target Object

Although fuzzy hanging togetherness describes the intensity pattern of the inaccurate image, the segmentation algorithm based on this concept should be able to classify which pixel in the image belongs to the target object, which does not. The fuzzy affinity between pixels is designed toward this goal. The affinity between the two given pixels in an image is defined as a combined weighted function of the degree of coordinate space adjacency, the degree of intensity space adjacency, and the degree of intensity gradient space adjacency to the corresponding target object features.

Let's define the affinity in the way of formulation. Firstly, a scene is defined over a fuzzy digital space  $(Z^n, \alpha)$  as a pair  $\zeta = (C, f)$ , where  $C$  is a  $n$ -dimensional array of pixels and  $f$  is a function in the domain  $C$ . Its range is a subset of the closed interval  $[0, 1]$ ,  $f : C \rightarrow [0, 1]$ . Fuzzy affinity  $k$  is any reflexive and symmetric fuzzy relation in  $C$ , that is:

$$\begin{aligned} k &= \{((c, d), \mu_k(c, d)) | (c, d) \in C\} \\ \mu_k &: C \times C \rightarrow [0, 1] \\ \mu_k(c, c) &= 1, \forall c \in C \\ \mu_k(c, d) &= \mu_k(d, c), \forall (c, d) \in C \end{aligned} \quad (5.2)$$

$\mu_k$  can be written as follows generally:

$$\mu_k(c, d) = h(\mu_\alpha(c, d), \mu_\phi(c, d), \mu_\phi(c, d), c, d) \forall (c, d) \in C$$

Where:  $\mu_\alpha(c, d)$  represents the degree of coordinate space adjacency of  $c$  and  $d$ ;  $\mu_\phi$  represents the degree of intensity space adjacency of  $c$  and  $d$ ; and  $\mu_\phi$  represents the degree of intensity gradient space adjacency of  $c$  and  $d$  to the corresponding target object features. Fuzzy  $k$ -affinity is a fuzzy relationship in  $C$ ,

where  $\mu_\kappa(c, d)$  is the strength of a path, which is the strongest path between  $c$  and  $d$ , and the strength of a path is the smallest affinity along the path. The hard binary relation  $K_\theta$  based on the fuzzy relation  $K$  is used to define the notion of a fuzzy connected component.

$$\mu_\kappa(c, d) = \begin{cases} 1 & \text{iff } \mu_\kappa(c, d) \geq \theta \in [0,1] \\ 0 & \text{otherwise} \end{cases} \quad (5.3)$$

Let  $O_\theta$  be an equivalence class of the relation  $K_\theta$  in  $C$ . A fuzzy  $k$ -component  $\Gamma_\theta$  of  $C$  of strength  $\theta$  is a fuzzy subset of  $C$  defined by the membership function:

$$\mu_{\Gamma_\theta} = \begin{cases} f(c) & \text{iff } c \in O_\theta \\ 0 & \text{otherwise} \end{cases} \quad (5.4)$$

The equivalence class  $O_\theta \subset C$ , such that for any  $(c, d) \in C$ ,  $\mu_\kappa(c, d) \geq \theta, \theta \in [0,1]$ , and for any  $e \in \{C - O_\theta\}, \mu_\kappa(c, d) < \theta$ . The notation  $[O]_\theta$  denotes the equivalence class of  $K_\theta$  that contains  $O$  for any  $O \in C$ . The fuzzy  $k$ -component of  $C$  contains  $O$ , denoted  $\Gamma_\theta(O)$ . It is a fuzzy subset of  $C$ , whose membership function is given by:

$$\mu_{\Gamma_\theta(O)} = \begin{cases} f(c) & \text{iff } c \in [O]_\theta \\ 0 & \text{otherwise} \end{cases} \quad (5.8)$$

A fuzzy  $k\theta$ -object of  $\zeta$  is a fuzzy  $k$ -component of  $\zeta$  of strength  $\theta$ . For any pixel  $O \in C$ , a fuzzy  $k\theta$ -object of  $\zeta$  that contains  $O$  is a fuzzy  $k$ -component of  $\zeta$  of strength  $\theta$  that contains  $O$ . Given  $k, O, \theta$ , and  $\zeta$ , a fuzzy  $k\theta$ -object of  $\zeta$  of strength  $\theta \in [0,1]$  containing  $O$ , for any  $O \in C$ , can be computed via dynamic programming.

In the generic implementation for  $c, d \in C$ :  $\mu_\kappa(c, d) = h(\mu_\alpha(c, d), f(c), f(d), c, d)$  where  $c, d$  are the image locations of the two pixels,  $\mu_\alpha(c, d)$  is an adjacency function based on the distance of the two pixels, and  $f(c)$  and  $f(d)$  are the intensity of pixels  $c$  and  $d$ , respectively. In this general form,  $\mu_\kappa(c, d)$  is shift-variant. In other words, it is dependent on the location of pixels  $c$  and  $d$ . A more specific and shift-variant definition for a fuzzy affinity is:

$$\begin{aligned} \mu_\kappa(c, d) &= \mu_\alpha(c, d)[\omega_1 h_1(f(c), f(d)) + \omega_2 h_2(f(c), f(d))] \\ \mu_\kappa(c, c) &= 1 \end{aligned} \quad (5.5)$$

where,  $\mu_\alpha(c, d)$  is a linear combination of  $h_1(f(c), f(d))$  and  $h_2(f(c), f(d))$ , with  $\omega_1 + \omega_2 = 1$ . The three features taken into consideration are: the adjacency between the pixel  $\mu_\alpha(c, d)$ , the intensity of the pixels  $h_1(f(c), f(d))$ , and the gradient of the pixels  $h_2(f(c), f(d))$ .

The adjacency function  $\mu_\alpha(c, d)$  is assumed to be a hard adjacency relation, such that:

$$\mu_\alpha(c, d) \begin{cases} 1 & \text{if } \sqrt{\sum_i (c_i - d_i)^2} \leq 1 \\ 0 & \text{otherwise} \end{cases} \quad (5.6)$$

where  $c_i (0 \leq i \leq n)$  are the pixel's coordinates in  $n$  dimensions. The functions  $h_1$  and  $h_2$  are Gaussian functions of  $\frac{1}{2}(f(c) + f(d))$  and  $|f(c) - f(d)|$ , respectively,

such that:

$$\begin{aligned} h_1(f(c), f(d)) &= e^{-\frac{1}{2} \left[ \frac{1}{2}(f(c) + f(d)) \right]^2} \\ h_2(f(c), f(d)) &= e^{-\frac{1}{2} [f(c) - f(d)]^2} \end{aligned} \quad (5.7)$$

In the following I present the pseudo code for computing the fuzzy affinity function, which is based on the concept of hanging togetherness introduced above.

**Input:**  $\zeta, o, k$  as defined previously

**Output:**  $K_o$  - scene of  $\zeta$ , denoted  $\zeta_o$

**Auxiliary Data Structures:** An  $n$ D array representing the  $K_o$  - scene  $\zeta_o = (C_o, f_o)$  of  $\zeta$  and a queue  $Q$  of pixels

**Pseudo Code:**

Set all elements of  $\zeta_o$  to 0 except  $o$ , which is set to 1;

Push all pixels  $c \in \zeta_o$  such that  $\mu_k(o, c) > 0$  to  $Q$ ;

While  $Q$  is not empty do {

remove a pixel  $c$  from  $Q$ ;

find  $f_{\max} = \max_{d \in C_o} [\min(f_o(d), \mu_k(c, d))]$ ;

if  $f_{\max} > f_o(c)$  {

set  $f_o(c) = f_{\max}$ ;

push all pixels  $e$  such that  $\mu_k(c, e) > 0$  to  $Q$ ;

}

}

The fuzzy affinity  $\mu_k(c, d)$  is computed as:

$$\mu_k(c, d) = \mu_\alpha(c, d) \left[ \frac{h_1^2}{h_1 + h_2} + \frac{h_2^2}{h_1 + h_2} \right] \quad (5.8)$$

$$\mu_k(c, c) = 1$$

where  $h_1$  and  $h_2$  are described as equation (5.7)

By using fuzzy affinity the sample of target object is segmented. It is also the coarse segmentation of the image. The segmented result gives a binary mask of object so that the image is divided into object region and background region coarsely.

### 5.3.3 Defining Homogeneity Criteria

We know that homogeneity is largely related to the local information extracted from an image and reflects how uniform a region is [GW87]. It plays an important role in image segmentation since the segmented result should be several homogeneous regions. In this section I define one homogeneity criteria in order to present different regions homogeneous characteristic that distinguishes object region, background region and boundary region. And one homogeneity operator is generated to measure the uniform distance from one region to object region or background region, which is used by the following procedures in my hybrid segmentation approach.

The homogeneity criterion is defined as the composition of two components: standard deviation and discontinuity of the intensity. Standard deviation describes the contrast within a local region. Discontinuity is a measure of abrupt changes in grey levels.

Suppose  $f(w)$  is the intensity of a pixel  $P_{ij}$  at the location  $(i, j)$  in the image.  $W_d$  is a size  $d \times d$  window centered at  $(i, j)$  for the computation of variation. The standard deviation of  $P_{ij}$  is calculated:

$$S_c = \sqrt{\frac{1}{d^2} \sum_{f(w) \in W_d} [f(w) - \nu_c]^2} \quad (5.9)$$

$\nu_c$  is the mean of the intensity within window  $W_d$ , and calculated as:

$$\nu_c = \frac{1}{d^2} \sum_{f(w) \in W_d} f(w) \quad (5.10)$$

The discontinuity for pixel  $P_{ij}$  at the location  $(i, j)$  is measured as:

$$D_c = \sqrt{G_x^2 + G_y^2} \quad (5.11)$$

Where  $G_x$  and  $G_y$  are the components of the gradient in the  $x$  and  $y$  directions, respectively.

The standard deviation and discontinuity values are normalized in order to achieve computational consistence. Therefore the homogeneity is represented as:

$$H_c = 1 - \left( \frac{D_c}{D_{\max}} \right) \times \left( \frac{S_c}{S_{\max}} \right) \quad (5.12)$$

The value of the homogeneity at each location of an image has a range from 0 to 1. The more uniform the local region surrounding a pixel is, the larger the homogeneity value the pixel has.

The homogeneity value of one area is obtained by the mean of all pixels' homogeneity value in the area. In the following procedure of the hybrid segmentation approach, there are some regions to be classified as inside object area, outside object area or boundary area according to their the homogeneity characteristics. The homogeneity operator is generated as the above criteria only the mean intensity value in the equation (5.9) is replaced by the value of compared region. For instance, if one certain region  $X$  is compared to target object area in order to see whether it is highly homogeneous to target object, then the mean intensity value in the equation is replaced by the mean intensity value of the target object area.

The homogeneity operator is presented as:

$$H_{operator} = \begin{cases} inside & H_{X \rightarrow object} > \varepsilon \\ outside & H_{X \rightarrow background} > \varepsilon \\ boundary & else \end{cases} \quad (5.13)$$

where  $H_{X \rightarrow object}$  is the homogeneity value when a certain region  $X$  is compared to target object area.  $H_{X \rightarrow background}$  is the homogeneity value when the

region is compared to the background area.  $\varepsilon$  is the tolerant value that is close to the homogeneity value of the target object region and the background region.

### 5.3.4 Experiments

One advantages of fuzzy hanging togetherness segmentation is that in the application some appropriate parameters in fuzzy affinity function can be automatically created by the algorithm, based on some minimal information supplied by a user. With some pixels that user defined as target object in the image, the program can compute some statistics of the pixels that are identified, and then the parameters can be automatically gotten on the basis of such information.

The homogeneity operator is defined based on the segmentation, where a sample of tissue is segmented and compared against its background. And the homogeneity operator will be used in the following processing step, Voronoi Diagram based segmentation. Hanging togetherness concept captures the characteristics of the sample region (of target tissue) that is not inherently homogeneous. That is the finesse of this approach. Comparing to the general method for generating experimentally homogeneity statistics of inherently inhomogeneous regions that represented one organ, this approach is not a very sophisticated way to solve the problem. Understanding the power of this approach is helpful in understanding why inhomogeneous regions still can form a “whole” structure, that people can perceptually recognize those fuzzy components as one structure. Likewise, homogeneous operator really means how we can describe the homogeneous component (the “strength” of homogeneity) of something that is not homogeneous by nature.

Here I present some experimental results generated by applying fuzzy hanging togetherness algorithm both on phantom data and on actual MR data.



A synthetic example is shown as figure 5.2. It is one phantom data set, which possesses a region of interest similar in structure to that of a short-axis MR-scan of a left ventricle. Figure 5.2 shows the synthetic image in (a), the affinity map of the mean gradient of the image in (b), the affinity map of the directional gradient of the image in (c), the affinity map of the image in (d), the resultant histogram of (d) in (e), and the binary threshold of the segmentation result in (f).

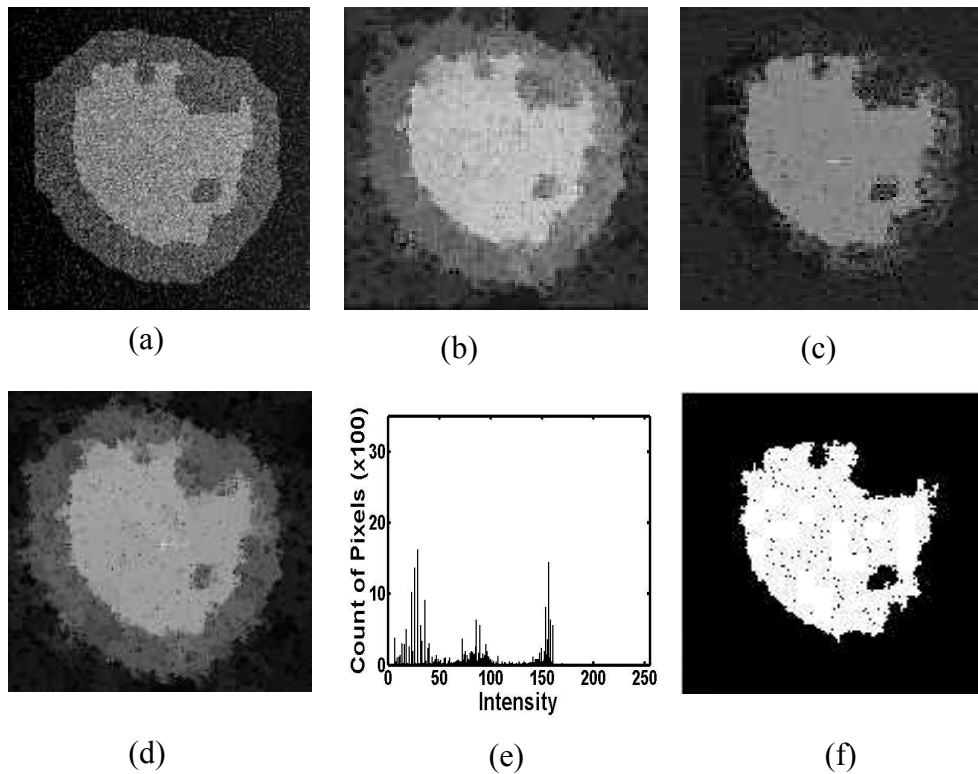


Figure 5.2 (a) a synthetic image, (b) the affinity map of the mean gradient, (c) the affinity map of the directional gradient, (d) the affinity map of the segmentation, (e) the resultant histogram, (f) the binary threshold of the segmentation result

As it can be seen from figure 5.2, comparing to mean gradient and directional gradient, the algorithm enhances the values of affinity attached to the pixels of the target object, and thus resulting in better defined fuzzy objects. It attaches much higher and uniform affinities to the target object relative to rest of the image, the edge magnitudes of segmented image are pronounced which is crucial for further

automation of the segmentation of the target object. As can be seen from figure 5.2(e), a valley between the target region and the rest of the image in affinity image histogram is broad. It can be detected easily by Otsu's auto thresholding algorithm [Ots79]. This automatic threshold detection of the target region relieves the user of deciding threshold of affinity map for object extraction. The only manual interaction left is the selection of the seed pixel.

Figure 5.3 shows one slice of the segmentation results for one  $T_1$  real-weighted MR images using the fuzzy hanging togetherness algorithm. The number of tissue classes in the segmentation was set to three, which corresponds to gray matter (GM), white matter (WM) and cerebrospinal fluid (CSF). Background pixels are ignored in the computation. One can see from figure 5.5 that, the GM and WM in the top region of the image can be relative good separated by this algorithm. However, in the middle region of the image, the boundary between GM and WM is quite blurred; hence the segmentation result is not so satisfied.

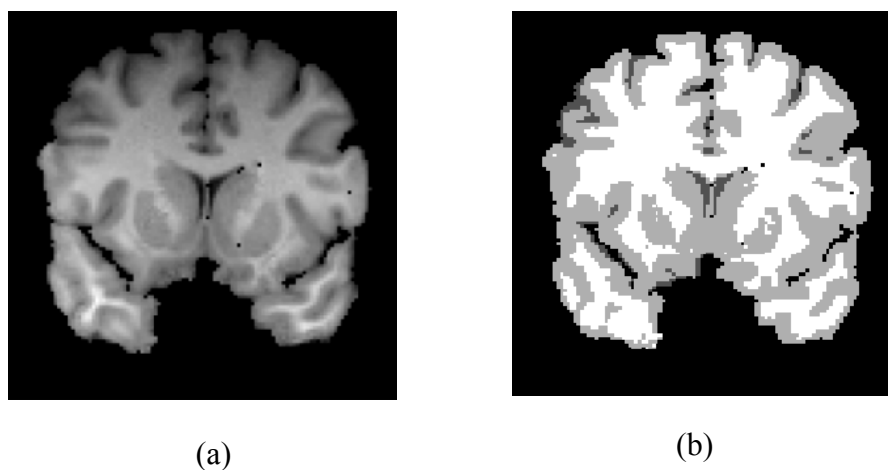


Figure 5.3 (a) original image, (b) segmentation result

The examples above show that this algorithm is an effective approach. Nevertheless, it is not universally applicable. For instance, if the region of interest is separated from another region with similar characteristics by a narrow wall, then

noise in the image may cause a break in this wall and make the two regions ‘leak’ into each other. Or the image has some blurs, which causes the algorithm sticks locally, instead of propagation correctly. However, the hybrid segmentation approach presented in this thesis will show how a correct segmentation can be achieved by applying the following segmentation procedures base on the result of this step. The fuzzy hanging togetherness algorithm is used as the first step to get the sample of target object. This task is carried out well by the algorithm. And the homogeneity operator is defined in this step. The segmentation result can be used for further refined segmentation in the next step of the hybrid approach.

#### **5.4 Reclassifying Exterior, Interior and Boundary Regions**

The second step in the hybrid segmentation approach is implemented by one region-based algorithm. It can quickly converge round an accurate boundary and requires minimal user interaction. The basic idea is to subdivide an image into smaller regions, and classify each region into either target object or background (outside the boundary of the target object) by the homogeneity operator. In principle, the homogeneity operator gained by last step should be able to separate the target area from the background. However, between target area and background area, there are some regions, whose homogeneous characteristic doesn’t belong to target area or background. These regions are boundary regions. It is difficulty to accurately classify them into target object or background in the level of homogeneity during the first step. These boundary regions should be subdivided into smaller regions; afterwards the smaller regions will be classified again into target object or background by the homogeneity operator. The smaller the boundary regions are divided, the more precise target object is segmented. This process can be repeated as many times as the user wishes, within the bounds of hardware limitations, in order to refine the calculated

boundary. In fact, the sample of target object region that segmented by last step is normally close to the aim so that the reclassification procedure does not need to repeat too many times.

For the region subdivision, the concept of Voronoi Diagram is adopted, which is a very useful tool for the image segmentation. Bertin and Chassery have presented a grey scale region-based segmentation method for microscopic data which makes use of Voronoi Diagrams to divide the image into smaller regions [BPC93]. The definition of the Voronoi Diagram is detailed described in [PS88]. I give a brief review of it here.

#### 5.4.1 Concept of Voronoi Diagram

Let  $S$  be a set of  $N$  points in the plane, indexed by  $i \in \{1, \dots, N\}$ . The Voronoi region associated to one point  $p_i \in S$  denoted by  $Vor_S(p_i)$  is the set of the points closer to  $p_i$  than to any other points of  $S$ . According to this definition it is easy to show that each Voronoi region is polygonal and convex as an intersection of the half-plane. Let us denote  $H(p_i, p_j)$  the half-plane containing  $p_i$  that is defined by the perpendicular bisector of  $\overline{p_i p_j}$ . It is written as below:

$$Vor_S(p_i) = \bigcap_{i \neq j} H(p_i, p_j) \quad (5.14)$$

The Voronoi Diagram is defined by the set of all Voronoi polygons. Figure 5.4 shows the procedure how the Voronoi diagram is built by adding sites one by one.

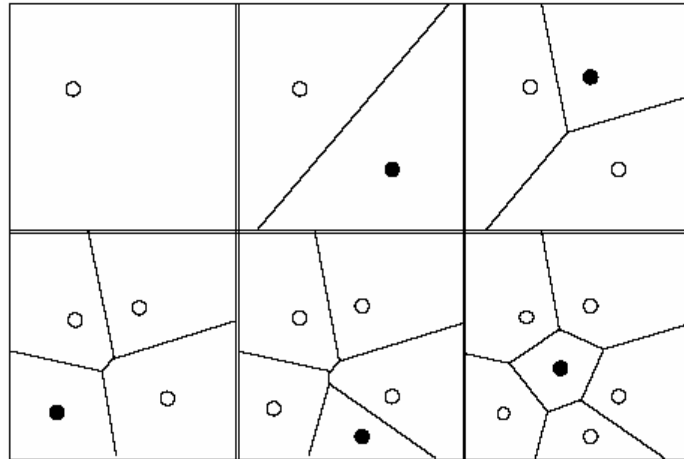


Figure 5.4 Construction of the Voronoi Diagram by adding successively sites and local modification of the diagram. The black dot is the last site to add.

An interesting property is that the dual graph of the Voronoi Diagram is the Delaunay graph with the following properties: the Delaunay graph is a triangulation such that each circle  $C$  circumscribed by every triangle  $\overline{p_i p_j, p_k}$  does not contain any point of  $S$  in its interior. The proof is that assume there exists a point  $p_l$  of  $S$  in the interior of  $C$ . Then the distance between the centre  $c$  of  $C$  and  $p_l$  is smaller than the distance between  $c$  and any  $p_n \in S, n \neq l$ . According to the definition of a Voronoi polygon,  $c$  belongs to the interior of  $Vor_S(p_l)$ , which is contradictory. As shown in figure 5.5, the black dots are points in one plane. The solid lines are Delaunay triangles on these points. The dark lines are Voronoi Diagram about these points, each region divided by those lines are Voronoi region. The circle in the figure is the circumscribed circle of one Delaunay triangle. To proof the Delaunay graph property, it shows that in this circle there is no other dot of the points set.

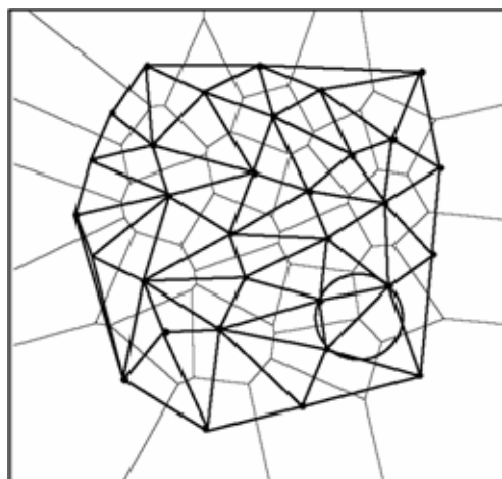


Figure 5.5 Voronoi polygons and Delaunay triangulation. The circle does not contain any point of  $S$  (Black dots) in its interior.

In this algorithm, the Delaunay triangulation is used to connect the final boundary regions to form an outline.

#### 5.4.2 VD – Based Segmentation Algorithm

Comparing to other dividing methods, for instance, the most common method: quadtree [Sam84], Voronoi Diagram has some obvious advantages, and its concept exactly answers for the idea of the hybrid segmentation approach. For example, the Voronoi Diagram has the only subdivision graph for certain data set, and its dual graph, Delaunay triangulation, can present any directions of the edge while quadtree can only present horizontal and vertical directions. As the second step in the hybrid approach, the Voronoi Diagram-based segmentation method processes the image based on the result of the previous step. This hybrid approach has in the first step generated the initial segmented area of the target object. It also provides the second step with the statistic homogeneity operator for the exterior part, interior part and boundary. In the second step, the VD-based segmentation algorithm will divide the image into regions by distributing automatically a number of seed points throughout the plane of the image and then generating the Voronoi Diagram of these points. Each

Voronoi region is a convex polygon which can be efficiently analyzed for various statistics. For each region, the homogeneity operator will reclassify it to exterior, interior or boundary.

Once the regions have been classified, the algorithm can label all boundary regions, and then each region exterior and interior the boundary regions will respectively share an edge. Afterwards the algorithm constructs the Delaunay triangulation and selects those edges which connect the seed points in the neighbouring boundary regions. This generates an approximation of the outline of the target object. In order to improve the accuracy of the results, the algorithm adds a seed point on the midpoint of each edge of every boundary region, recalculate the Voronoi Diagram with these new seeds, and repeat the process. In each iteration procedure, the Delaunay triangulation of the boundary regions will be constructed. It shows the outline of the target object boundary. After a few iterations, the outline is close to the real boundary. The user can choose to quit manually or let the algorithm quit the processing automatically. The procedures for this DV-based method are shown as the following :

- 1) Input some points in the image.
- 2) Compute Voronoi Diagram of those points.
- 3) Classify each region as interior, exterior or boundary.
- 4) Compute Delaunay triangulation and show the outline of the computed boundary regions.
- 5) Add seeds to the edges and inside of boundary regions.
- 6) Goto step 2 until a specified number of iterations procedures are processed or quitted by the users.

Figure 5.6 shows an example of VD-based segmentation of visceral adipose tissue in a single slice. We can see that the divided mesh near the boundary is denser than in the middle region. Therefore the outline of the boundary by Delaunay triangulation is quite precise even though the subdivision procedure is performed only a few times.

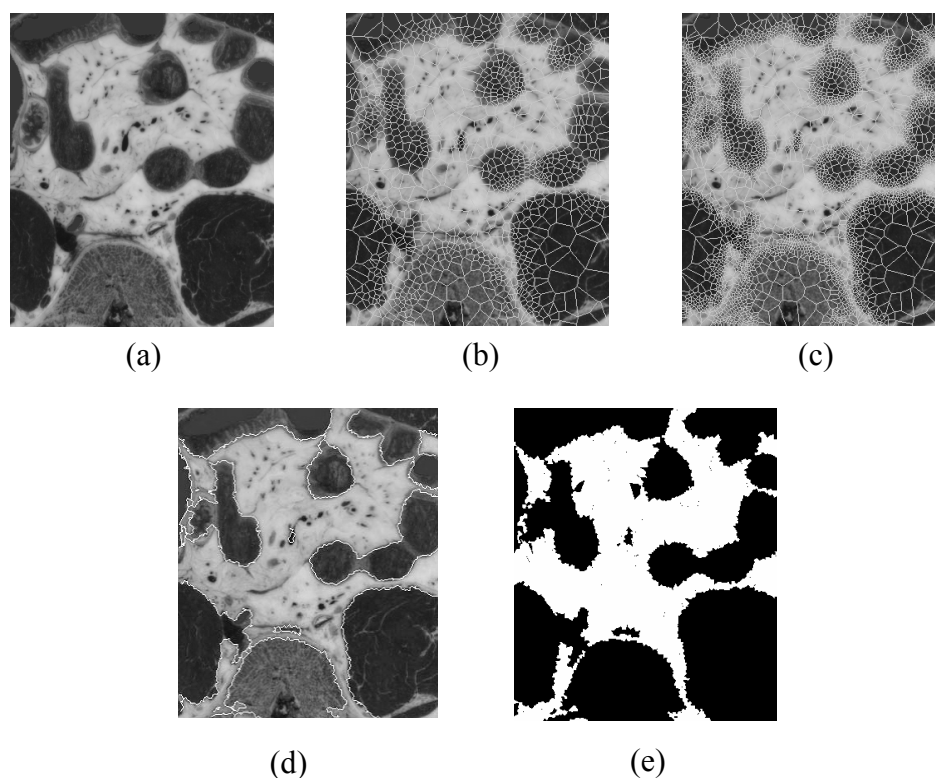


Figure 5.6 Voronoi Diagram classification. (a) Input image. (b)(c): Voronoi regions after 2 and 8 iterations, respectively. (d) The final boundary. (e) The final segmented region: a binary object.

This method is quite robust. Normally in a few iterations, the accuracy of the boundary outline computed by this method is acceptable. Since it is region-based algorithm, the search procedure can only be concentrated on the specified area by the result of last step. It improves the algorithm both in speed and in accuracy. The Voronoi Diagram based segmentation is only performed on the boundary region. This strategy has another advantage that the definition of interior region, say, target tissue, is a “compatibility” measure, but rather “homogeneity” measure. The interior region



might be intensity various, for example, brain tumour, which have complex structure interior, and it is difficult to be segmented. Applying this hybrid segmentation approach, the complex interior structure will not be over-partitioned by “compatibility” measure, while the boundary of target can be refined by Voronoi subdivision.

### 5.4.3 Experiments

In this section I present some experiment results generated after finishing two steps of the hybrid segmentation method described above. Some different tissue types are tested: muscle tissue (figure. 5.7), brain tissue (figure. 5.8), and MRI patient data (figure. 5.9). Figure 5.7 shows the segmentation of temporalis muscle, a structure in the head region. In figure 5.8, the brain gray matter is segmented. The hybrid method is tested with the Visible Human data as well as with a sample of MRI patient data.

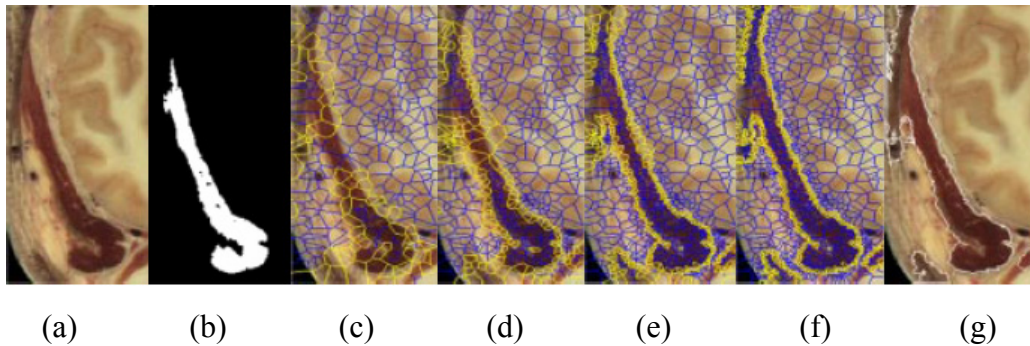


Figure 5.7 Hybrid Method of two steps (segmentation of temporalis muscles): (a) Color male cryosection slice, (b) a fuzzy segmented component, (c)-(f) iterations of the VD-based algorithm, (g) an outline of the boundary.

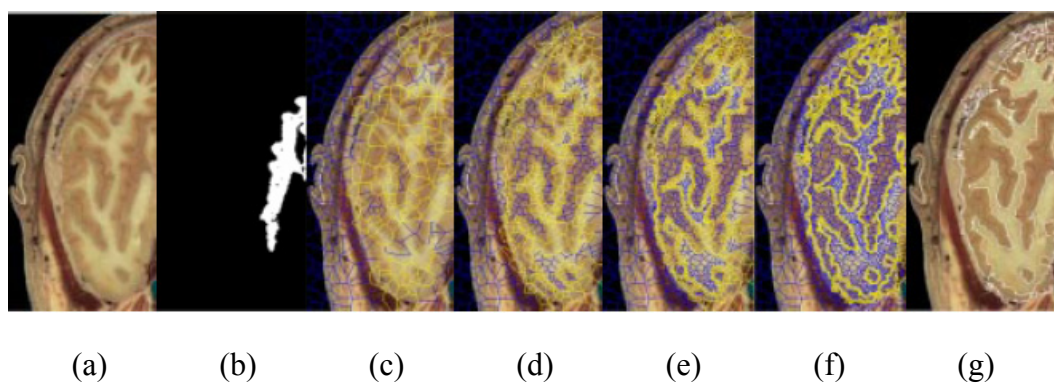


Figure 5.8 Hybrid Method of two steps (segmentation of brain gray matter): (a) Color male cryosection slice, (b) a fuzzy segmented component, (c)-(f) VD-based algorithm, (g) an outline of the boundary.

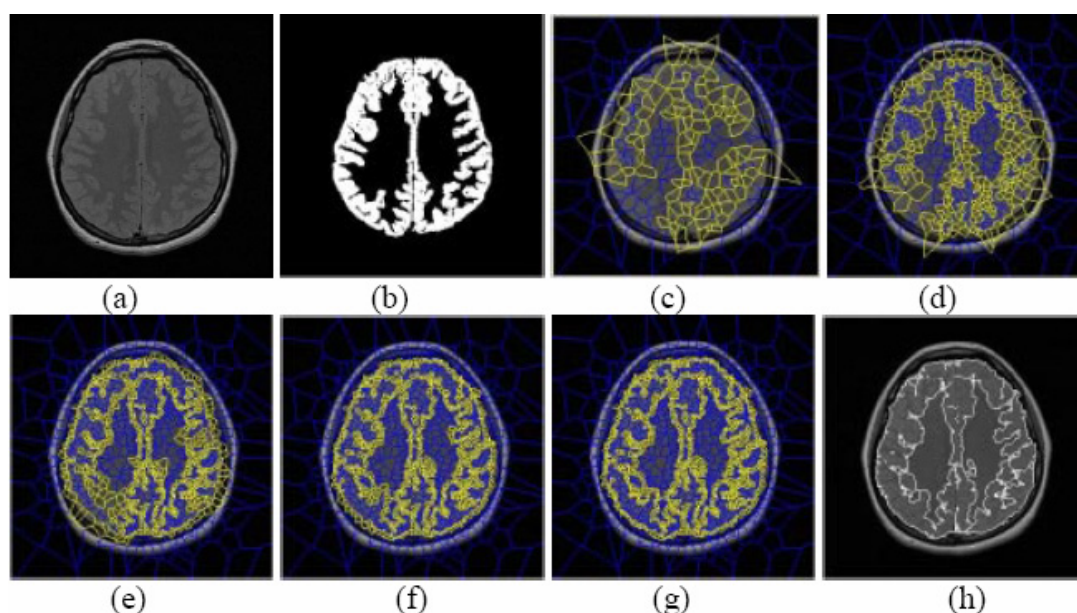


Figure 5.9 Hybrid Method of two steps (segmentation of MRI gray matter): (a) MRI patient slice, (b) fuzzy segmented component, (c)-(g) iterations of the VD-based algorithm, (h) an outline the boundary.

The above illustrations show some experiment results of the hybrid approach. It aims to improve the robustness and performance of the segmentation, and to reduce the need for user interactions. Starting from the seed set by user inside the object, the fuzzy affinity between pixels is calculated for justifying the hanging together target object. As a result, the sample of target tissue is segmented. From the segmented

sample tissue a homogeneity operator, used to classify regions in the Voronoi Diagram segmentation, is consequently derived. Thereafter this method generates Voronoi Diagram from randomly distributed seed points over the image. The Voronoi regions are classified, in accordance with the homogeneity operator, into interior, exterior and boundary regions. The boundary regions are subsequently subdivided by adding seed points in their edges and re-compute the Voronoi Diagram, and then the Voronoi regions are classified again. This approach iterates until the boundary regions converge to the final segmentation and each boundary region reaches a relative small area.

Although a relative accurate segmentation can be achieved after the first two steps of the hybrid segmentation approach, the boundary yielded by the Voronoi Diagram classification still has a “noisy” appearance because it doesn’t take the smoothness factor of the boundary into account. Therefore, we need another process step to get smooth final result. Following these two steps, the third step in the hybrid segmentation approach is to refine the vague boundary. I will describe it in the next section.

## **5.5 Refining the Vague Boundary**

This hybrid method of medical image segmentation is a combination of the region-based method and the boundary-based method. It not only encourages the advantages of these two kinds of image segmentation methods, but also discourages their disadvantages. The first step and second step introduced above belong to region-based method. The result produced by last two steps is the segmented target object labeled a boundary, the outline of Delaunay triangulation. Although the segmentation is quite precise, it does not allow for the smoothness factor of the boundary outline. Therefore the accuracy of the segmentation might be affected by

the noise of the data. Because the deformable model is very proper for handling the outline of segmentation, the level-set model, a boundary-based method, is used in the third step of the hybrid approach.

Deformable models provide an explicit representation of the boundary and the shape of the object. They combine several desirable features such as inherent connectivity and smoothness, which counteract noise and boundary irregularities, as well as the ability to incorporate knowledge about the object of interest [XPP00]. However, parametric deformable models have two main limitations. First, in situations where the initial model and desired object boundary differ greatly in size and shape, the model must be re-parameterized dynamically to faithfully recover the object boundary. The second limitation is that it has difficulty dealing with topological adaptation such as splitting or merging model parts, a useful property for recovering either multiple objects or objects with unknown topology. This difficulty is caused by the fact that a new parameterization must be constructed whenever topology change occurs, which requires sophisticated schemes.

Level set deformable models [OS88] [MSV95] also referred to as geometric deformable models, provide an elegant solution to address the primary limitations of parametric deformable models. These methods have drawn a great deal of attention since their introduction in 1988. Advantages of the contour implicit formulation of the deformable model over parametric formulation include: (1) no parameterization of the contour, (2) topological flexibility, (3) good numerical stability, (4) straightforward extension of the 2D formulation to n-D.

The hybrid method of medical image segmentation uses level set model as its third step. It improves and smoothes the segmentation result of previous two steps by its special characteristics. Since level set model method is a boundary-based

segmentation method that needs to be initialized near the solution. The first and second step in the hybrid method are good prior to initialize the model for the following process in the third step.

In this section, I will dwell on the principles of level set model which is integrated to the hybrid segmentation approach as deformable model and the accompanying numerical algorithms. I firstly discuss the application of this technique to image segmentation problems, and consider the speed function and stopping criteria. And then I give a conclusion and analysis about the level set model used in the hybrid segmentation approach. At last I will show some experimental results of applying level set model to handle the segmented image from last two steps in the hybrid method. You will see that the segmentation becomes much better both in preciseness and in smoothness after level set model processing.

### **5.5.1 Deformable Form**

Model-based techniques have an important usability that is shape recovery from various types of visual data. Broadly speaking, these techniques involve the use of a model whose boundary representation is matched to the image to recover the object of interest. These models can either be rigid, such as correlation-based template matching techniques, or non-rigid, as those used in dynamic model fitting techniques.

The modeling technique may be viewed as a form of active modeling such as “snakes” and deformable surfaces since the model, which consists of a moving front, may be molded into any desired shape by externally applied halting criteria synthesized from the image data. The “snakes” or deformable surfaces may be viewed as Lagrangian geometric formulations wherein the boundary of the model is represented in a parametric form. These parameterized boundary representations will encounter difficulties when the dynamic model embedded in a noisy data set is

expanding/shrinking along its normal field and sharp corners or cusps develop or pieces of the boundary intersect.

By exploiting recent advances in interface techniques, level set modeling technique avoids this Lagrangian geometric view and instead capitalizes on a related initial value partial differential equation. In this setting, several advantages are apparent, including the ability to evolve the model in the presence of sharp corners, cusps and changes in topology, model shapes with significant protrusions and holes in a seamless fashion, and extension to three dimensions in an extremely straightforward way.

### **5.5.2 Front Propagation Problem**

many deformable models will meet the front propagation problem during the interface propagating. Level set solves this problem by its special features. Therefore these features are adopted by the hybrid segmentation approach in the deformable model for image segmentation. As a starting point and motivation for the level set approach, let's consider a closed curve moving in the plane, that is, let  $\gamma(0)$  be a smooth, closed initial curve in Euclidean plane  $R^2$ , and let  $\gamma(t)$  be the one-parameter family of curves generated by moving  $\gamma(0)$  along its normal vector field with speed  $F(K)$ , which is a given scalar function of the curvature  $K$ . Let  $X(s,t)$ , be the position vector which parameterizes  $\gamma(t)$  by  $s, 0 \leq s \leq S$ . One numerical approach to formulate the moving curve is the Lagrangian description. It produces equations of motion for the position vector  $X(s,t)$ , and then discretizes the parameterization with a set of discrete marker particles lying on the moving front. These discrete markers are updated in time by approximating the spatial derivatives in the equations of motion, and advancing their positions.

However, there are still several problems with this approach, as discussed in [OS88]. First, small errors in the computed particle positions will be tremendously amplified by the curvature term, and calculations are prone to instability unless an extremely small time step is employed. Second, in the absence of a smoothing curvature (viscous) term, singularities develop in the propagating front, and an entropy condition must be observed to extract the correct weak solution. Third, topological changes are difficult to manage as the evolving interface breaks and merges. And fourth, significant bookkeeping problems occur in the extension of this technique to three dimensions.

As an alternative, the central idea in the level set approach is to represent the front  $\gamma(t)$  as the level set  $\{\psi = 0\}$  of a function  $\psi$ . Thus, given a moving closed hyper surface, that is,  $\gamma(t=0): [0, \infty) \rightarrow R^N$ , one wish is to produce an Eulerian formulation for the motion of the hyper surface propagating along its normal direction with speed  $F$ , where  $F$  can be a function of various arguments, including the curvature, normal direction, etc. The main idea is to embed this propagating interface as the zero level set of a higher dimensional function  $\psi$ . Let  $\psi(X, t=0)$ , where  $X \in R^N$  is defined by

$$\psi(x, t=0) = \pm d \tag{5.15}$$

where  $d$  is the distance from  $X$  to  $\gamma(t=0)$ , and the plus (minus) sign is chosen if the point  $X$  is outside (inside) the initial hyper surface  $\gamma(t=0)$ . Thus, an initial function  $\psi(X, t=0): R^N \rightarrow R$  with the property is presented as

$$\gamma(t=0) = (X | \psi(X, t=0) = 0) \tag{5.16}$$

As illustration of figure 5.10, the front propagating is considered as the example of an expanding circle. It supposes that the initial front  $\gamma$  at  $(t=0)$  is a

circle in the  $xy$ -plane (figure 5.10(a)). Imagine that the circle is the level set  $\{\psi = 0\}$  of an initial surface  $Z = \psi(x, y, t = 0)$  in  $R^3$  (see figure 5.10(b)). Then match the one-parameter family of moving curves  $\gamma(t)$  with a one-parameter family of moving surfaces in such a way that the level set  $\{\psi = 0\}$  always yields the moving front (see figure 5.10(c) and figure 5.10(d)).

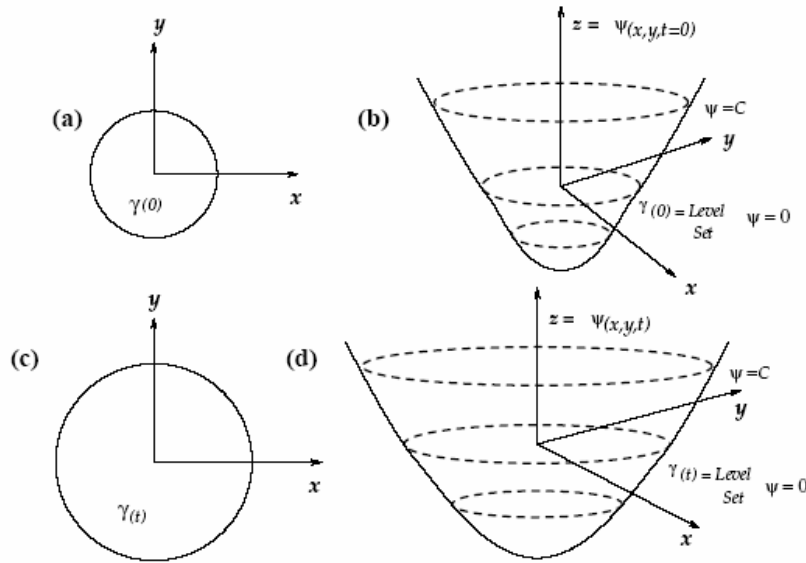


Figure 5.10 Level set formulation of equations of motion - (a) and (b) show the curve  $\gamma$  and the surface  $\psi(x, y)$  at  $t = 0$ , and (c) and (d) show the curve  $\gamma$  and the corresponding surface  $\psi(x, y)$  at time  $t$ .

In the general case, let  $\gamma(0)$  be a closed, nonintersecting,  $(N-1)$  dimensional hyper-surface. Let  $\psi(X, t), X \in R^N$ , be the scalar function such that  $\psi(X, 0) = \pm d(X)$ , where  $d(X)$  is the signed distance from  $X$  to the hyper-surface  $\gamma(0)$ . The plus sign means  $X$  is outside  $\gamma(0)$  and minus sign means  $X$  is inside. Each level set of  $\psi$  flows along its gradient field with speed  $F(K)$ . The gradient  $\nabla \psi(X, t)$  is normal to the  $(N-1)$  dimensional level set passing through  $X$ . Now, the equation of motion for function  $\psi$  is derived.



Consider the motion of some level set  $\{\psi = C\}$ . Let  $X(t)$  be the trajectory of a particle located on this level set, so

$$\psi(X(t), t) = C \quad (5.17)$$

The particle speed  $\partial X / \partial t$  in the direction  $n$  normal to  $\gamma(t)$  is given by the speed function  $F$ . Thus,

$$\frac{\partial X}{\partial t} \cdot n = F \quad (5.18)$$

where the normal vector  $n$  is given by  $n = \nabla \psi / |\nabla \psi|$ . By the chain rule,

$$\psi_t + \frac{\partial X}{\partial t} \cdot \nabla \psi = 0 \quad (5.19)$$

and substitution yields

$$\psi_t + F|\nabla \psi| = 0 \quad (5.20)$$

with an initial condition  $\psi(X, 0) = \pm d(X)$ . The equation (5.20) is referred as level set ‘‘Hamilton-Jacobi’’ formulation. Note that at any time, the moving front  $\gamma(t)$  is simply the level set  $\{\psi(X, t) = 0\}$ . There are several advantages to this approach. First, since the underlying coordinate system is fixed, discrete mesh points used in the numerical update equations do not move that resulting in a stable computation. Topological changes in the front can be handled naturally by exploiting the property that the level surface  $\{\psi = 0\}$  need not be simply connected.  $\psi(X, t)$  always remains a function, even if the level surface  $\{\psi = 0\}$  corresponding to the front  $\gamma(t)$  changes topology, or forms sharp corners. The geometric and differential properties of  $\gamma(t)$  are captured in the function  $\psi$  and can be readily extracted.

As an example, if  $X \in R^2$ , the curvature is given by

$$K = \frac{(\psi_{yy}\psi_x^2 - 2\psi_x\psi_y\psi_{xy} + \psi_{xx}\psi_y^2)}{(\psi_x^2 + \psi_y^2)^{3/2}} \quad (5.21)$$

This approach can also be easily extended to higher dimensions and appropriate expressions can be obtained for the mean curvature and the Gaussian curvature.

By substituting  $F(K)=1-\varepsilon K$  as a typical speed function in equation (5.20), the equation of motion becomes

$$\psi_t + |\nabla \psi| = \varepsilon K |\nabla \psi| \quad (5.22)$$

Equation (5.22) resembles a Hamilton-Jacobi equation with viscosity, where “viscosity” refers to the second-order parabolic right-hand side. This equation can be solved using the stable, entropy-satisfying finite difference schemes, borrowed from the literature on hyperbolic conservation laws.

### 5.5.3 Shape Recovery with Front Propagation

In this section, I will describe how the level set formulation for the front propagation problem, discussed in the previous section, can be used for shape recovery. The front represents the boundary of an evolving shape. Since the idea is to extract the object’s shape from a given image, the front should be forced to stop in the vicinity of the desired object’s boundary. This is analogous to the force criterion used to push the active contour model towards desired shapes. The final shape is defined to be the configuration when all the points on the front come to a stop, thereby bringing the computation to an end.

The goal now is to define a speed function from the image data that can be applied on the propagating front as a halting criterion. The speed function  $F$  can be split into two components:  $F = F_A + F_G$ . The term  $F_A$ , referred to as the advection term, is independent of the moving front's geometry. The front uniformly expands or

contracts with speed  $F_A$  depending on its sign that is analogous to the inflation force. The second term  $F_G$  is the part which depends on the geometry of the front, such as its local curvature. This diffusion term smoothes out the high curvature regions of the front and has the same regularizing effect on the front as the internal deformation energy term in thin-plate-membrane splines. This smooth term can make the segmented boundary have a good appearance and decrease the effect by the “noise” in the image. My hybrid segmentation approach makes use of this feature to smooth the boundary generated from last two steps. Equation 5.20 is rewritten by splitting the influence of  $F$  as

$$\psi_t + F_A |\nabla \psi| + F_G |\nabla \psi| = 0 \quad (5.23)$$

First, if consider the case when the front moves with a constant speed, that is,  $F_G = 0 \Rightarrow F = F_A$ , a negative speed  $F_l$  is defined as

$$F_l(x, y) = \frac{-F_A}{(M_1 - M_2)} \{ |\nabla G_\sigma * I(x, y)| - M_2 \} \quad (5.24)$$

where  $M_1$  and  $M_2$  are the maximum and minimum values of the magnitude of image gradient  $|\nabla G_\sigma * I(x, y)|, (x, y) \in \Omega$ . The expression  $G_\sigma * I$  denotes the image convolved with a Gaussian smoothing filter whose characteristic width is  $\sigma$ . Alternately, a smoothed zero-crossing image is used to synthesize the negative speed function. The zero-crossing image is produced by detecting zero-crossings in the function  $\nabla^2 G_\sigma * I$ , which is the original image convolved with a Laplacian-of-Gaussian filter whose characteristic width is  $\sigma$ . The value of  $F_l$ , lies in the range  $[-F_A, 0]$  as the value of image gradient varies between  $M_1$  and  $M_2$ . From this argument it is clear that, if  $|\nabla G_\sigma * I(x, y)|$  approaches the maximum  $M_1$

at the object boundaries, then the front gradually attains zero speed as it gets closer to the object boundaries and eventually comes to a stop.

If  $F_G \neq 0$ , then it is not possible to find an additive speed term from the image, that will cause the net speed of the front to approach zero in the neighborhood of a desired shape. Instead, multiply the speed function  $F = F_A + F_G$  with a quantity  $k_l$ . The term  $k_l$  is defined as

$$K_l(x, y) = \frac{l}{l + |\nabla G_\sigma * I(x, y)|} \quad (5.25)$$

have values that are closer to zero in regions of high image gradient and values that are closer to unity in regions with relatively constant intensity. By employing

$$k_l(x, y) = e^{-|\nabla G_\sigma * I(x, y)|} \quad (5.26)$$

the speed function that falls to zero can be faster than the reciprocal function.

#### 5.5.4 Extending the Speed Function

The image-based speed terms have meaning only on the boundary  $\gamma(t)$ , that is, on the level set  $\{\psi = 0\}$ . This follows from the fact that they were designed to force the propagating level set  $\{\psi = 0\}$  to a complete stop in the neighborhood of an object boundary. However, the level set equation of motion is written for the function  $\psi$  defined over the entire domain. Consequently, the evolution equation has a consistent physical meaning for all the level sets, that is, at every point  $(x, y) \in \Omega$ . The speed function  $F_l$  derives its meaning not from the geometry of  $\psi$  but from the configuration of the level set  $\{\psi = 0\}$  in the image plane. Thus, the goal should be to construct an image-based speed function  $\hat{F}_l$  that is globally defined. I call it an extension of  $F_l$ , off the level set  $\{\psi = 0\}$  because it extends the meaning of  $F_l$ .

Note that the level set  $\{\psi = 0\}$  lies in the image plane and therefore  $\hat{F}_l$  must equal  $F_l$ , on  $\{\psi = 0\}$ . The same argument applies to the coefficient  $k_l$ . With the extensions so defined, the equation of motion for the case  $F = F_A$  is given by

$$\psi_t + (F_A + \hat{F}_l)|\nabla \psi| = 0 \tag{5.27}$$

and

$$\psi_t + \hat{k}_l(F_A + F_G)|\nabla \psi| = 0 \tag{5.28}$$

when  $F = F_A + F_G$ .

If the level curves are moving with a constant speed, that is,  $F_G = 0$ , then at any time  $t$ , a typical level set  $\{\psi = C\}$ ,  $C \in R$ , is a distance  $C$  away from the level set  $\{\psi = 0\}$  (see figure 5.11). This statement is a rephrased version of Huygen's principle which, from a geometrical standpoint, stipulates that the position of a front propagating with unit speed at a given time  $t$  should consist of only the set of points located a distance  $t$  away from the initial front. On the other hand, for  $F_G \neq 0$ , the level sets will not remain a constant distance apart.

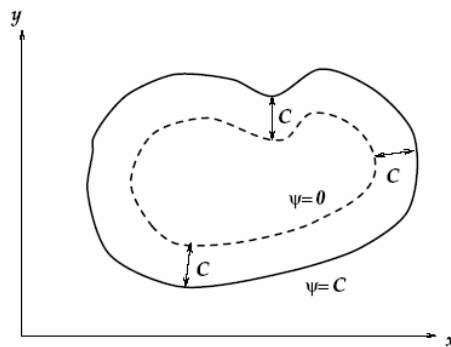


Figure 5.11 Huygen's principle construction

The global extension is constructed to the image-based speed function by (see figure 5.12) assigning the value of  $\hat{F}_l(\hat{k}_l)$  at a point  $P$  lying on a level set  $\{\psi = C\}$

be the value of  $\hat{F}_l(\hat{k}_l)$  at a point  $Q$ , such that point  $Q$  is the closest point to  $P$  and lies on the level set  $\{\psi = 0\}$ . Thus,  $\hat{F}_l(\hat{k}_l)$  reduces to  $F_l(k_l)$  on  $\{\psi = 0\}$ .

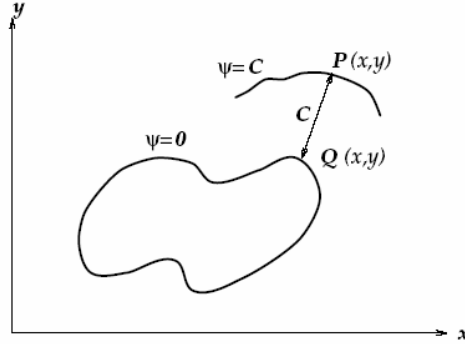


Figure 5.12 Extension of image-based speed terms to other level set

By updating the level set function on a grid, the level sets are moving without constructing explicitly. This construction can create a discontinuous velocity extension away from the zero level set, since the distance function is not differentiable. A straightforward way to do this is to re-compute the distance from each point of the grid to the zero level set. However, this is an  $O(N^3)$  operation, if we assume that there are  $N$  points in each coordinate direction, plus approximately  $O(N)$  points on the interfaces.

An alternative to this reconstruction is to iterate on the level set function at a given time according to the following equation:

$$\psi_{k+1} = \psi_k + S(\psi)(1 - |\nabla \psi|) \quad (5.29)$$

In the limit as  $k \rightarrow \infty$ , this converges to the distance function, with some error in relocating the original zero level set.

The most expensive step in either of these algorithms is the computation of the extension for image-based speed term. This is because at each grid point, it must search for the closest point lying on the level set  $\{\psi = 0\}$ . Moreover, if  $F_G = 0$ , then

the stability requirement for the explicit method for solving the level set equation is  $\Delta t = O(\Delta x)$ . For the full Equation (5.26), the stability requirement is  $\Delta t = O(\Delta x^2)$ . This could potentially force a very small time step for fine grids. These two effects, individually and compounded, make the computation exceedingly slow.

An efficient alternative is to move the front by updating the level set function at a small set of points in the neighborhood of the zero set instead of updating it at all the points on the grid. This concept fits the requirement of my hybrid segmentation approach very well, because the boundary segmented by last two steps is close to the real target object's boundary. Therefore updating the level set model in the neighborhood range is a reasonable and efficient way. In figure 5.13 the bold curve depicts the level set  $\{\psi = 0\}$  and the shaded region around it is the narrow band. The narrow band is bounded on either side by two curves which are a distance  $\delta$  apart. That is, the two curves are the level sets  $\{|\psi| = \pm\delta/2\}$ . The value of  $\delta$  determines the number of grid points that fall within the narrow band. Since, during a given time step the value of  $\psi_{ij}$  is not updated at points lying outside the narrow band, the level sets  $\{|\psi| > \delta/2\}$  remain stationary. The zero set which lies inside moves until it collides with the boundary of the narrow band. Which boundary the front collides with depends on whether it is moving inward or outward; either way, it cannot move past the narrow band.

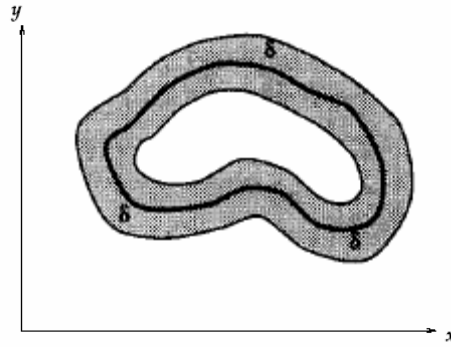


Figure 5.13 A narrow band of width  $\delta$  around the level set  $\{\psi = 0\}$

As a consequence of the update strategy, the front can be moved through a maximum distance of  $\delta/2$ , either inward or outward. Reinitialize the  $\psi$  function by treating the current zero set configuration, that is,  $\{\psi = 0\}$ , as the initial curve  $\gamma(0)$ . It is observed that the re-initialization step can be made cheaper by treating the interior and exterior mesh points as sign holders. Note that the re-initialization procedure must account for the case when  $\{\psi = 0\}$  changes topology. This procedure will restore the meaning of  $\psi$  function by correcting the inaccuracies introduced as a result of the update algorithm. Once a new  $\psi$  function is defined on the grid, the algorithm can create a new narrow band around the zero set, and go through another set of, say  $l$ , iterations in time to move the front ahead by a distance equal to  $\delta/2$ . The value of  $l$  is set to the number of time steps required to move the front by a distance roughly equal to  $\delta/2$ . One fast algorithm for shape recovery consists of the following steps:

- 1) Set the iteration number  $m = 0$  and go to step 2.
- 2) At each grid point  $(i, j)$  lying inside the narrow band, compute the extension  $\hat{k}_l$  of image-based speed term.



- 3) With the above value of extended speed term  $(\hat{k}_l^m)_{i,j}$ , and  $\psi_{i,j}^m$ , calculate  $\psi_{i,j}^{m+1}$  using the upwind, finite difference scheme using in [SS92].
- 4) Construct a polygonal approximation for the level set  $\{\psi = 0\}$  from  $\psi_{i,j}^{m+1}$ . A contour tracing procedure is used to obtain a polygonal approximation. Given a cell  $(i, j)$  which contains  $\gamma(t)$ , this procedure traces the contour by scanning the neighboring cells in order to find the next cell which contains  $\gamma(t)$ . Once such a cell is found, the process is repeated until the contour closes on itself. The set of nodes visited during this tracing process constitutes the polygonal approximation to  $\gamma(t)$ . In general, to collect all the closed contours, the above tracing procedure is started at a new, as yet unvisited cell, which contains the level set  $\{\psi = 0\}$ . A polygonal approximation is required in step 2 for the evaluation of image-based speed term and more importantly, in step 6 for reinitializing the  $\psi$  function.
- 5) Increment  $m$  by one. If the value of  $m$  equals  $l$ , goes to step 6, else, go to step 2.
- 6) Compute the value of signed distance function  $\psi$  by treating the polygonal approximation of  $\{\psi = 0\}$  as the initial contour  $\gamma(0)$ . As mentioned earlier, a more general method of reinitialization is required when  $\{\psi = 0\}$  changes topology. Go to step 1.

This approach only updates  $\psi$  at points lying in the narrow band, the issue of specifying boundary conditions for points lying on the edge of the band becomes pertinent. With this relatively simple speed motion, the free-end boundary condition is adequate.

### 5.5.5 Experiments on Hybrid Framework

The above sections present the construction of level set model and the expanding algorithm of the speed function. Level set model, as one kind of deformable model, is inherent suitable for image segmentation, because it constructs closed dynamic boundary. The model is driven by two forces during front propagating. The outside force makes the front move towards destination boundary, while the inside force keeps the smoothness of the model. Therefore this model plays important role in my hybrid segmentation approach in order to get precise and smooth segmentation. On the other hand, the segmented result comes from the previous two processing steps is very near to the real boundary, so that it can be good initialization for this third processing step. Consequently the narrow band method presented above is one efficient solution for the model calculation.

The level set model in this hybrid approach can significantly refine the segmented boundary from the previous processing procedures. The following experiment results will give us some intuitionistic understanding of what the level set can do.

Two examples of the hybrid segmentation approach are illustrated in figure 5.14 and 5.15. For the example in figure 5.14, the MRI proton density brain image is processed, in which the light part of the original image is the destination. The example in figure 5.15, a single MRI T1 weighted image is processed, where the adipose tissue of image should be segmented. In both cases, the result of each step is shown in sequence. As we can see from figure 5.14 and 5.15, in the first step, the sample of target object is segmented. But it is not the whole object. The segmentation procedure stops locally due to the grey level variance. This segmentation result is much improved in the second step, and the segmented area covers the whole target tissue. However, its boundary is yet quite rough because the image might contain

some noise. The third step refines the rough boundary of the last step. The final segmentation result gets both precise and smooth property of the boundary of the target tissue.

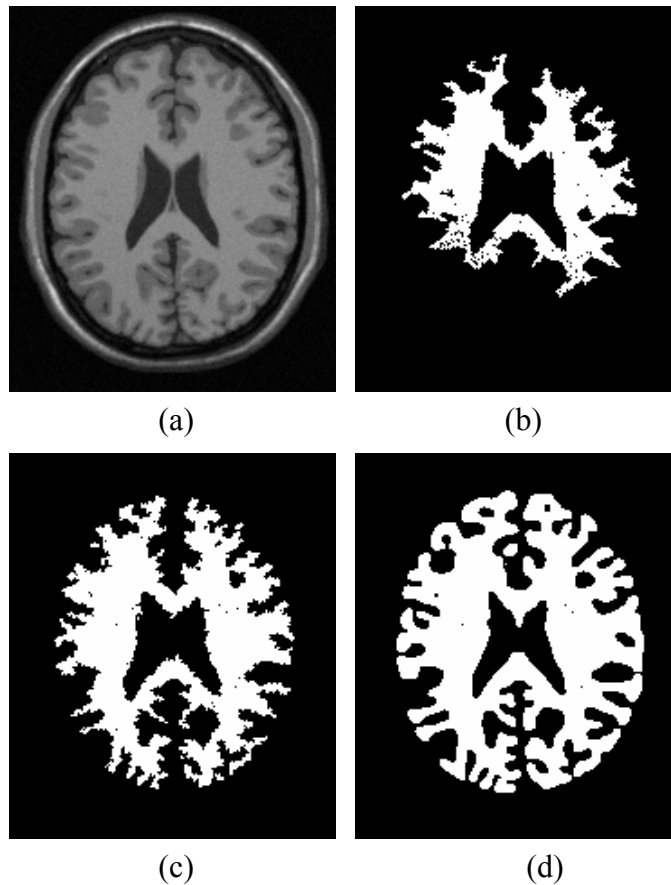


Figure 5.14. This figure shows one example of my hybrid approach. The image (a) is one MRI proton density brain image. (b) Fuzzy affinity segmentation. (c) Voronoi diagram segmentation using (b) as a prior. (d) Level-set model smoothing on (c).

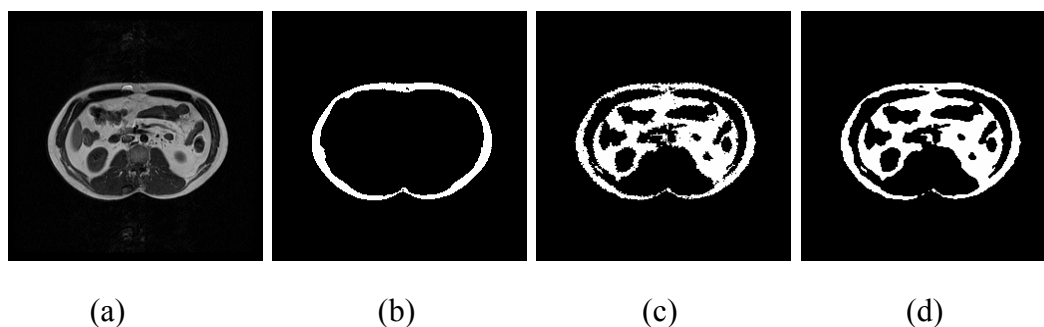


Figure 5.15. This figure shows another example of our hybrid approach. (a) Input image (MRI T1). The following images (b), (c), (d) are the result of three segmentation steps, respectively

## 5.6 Evaluation of the Hybrid Segmentation Approach

In this chapter, I have introduced a hybrid segmentation approach. I have described the theory and the implementation of every component in the hybrid approach. In this section, an objective evaluation is taken in order to underline the performance of this method.

The question of how valid image segmentations are is difficult to answer, because there is no ground truth or a “gold standard” to compare with. Realistic simulation of MRI data that covers all aspects of clinical data is not yet possible, and dissection of actual tissue does not preserve shape or location of structures of interest. A performance criterion suitable for one application may not be suitable for another application, and that the results could depend on the criteria chosen, on object size, on contrast in the images, on variance in object and background, and on noise, and so forth.

The quality measure of segmentation should be general, objective, and quantitative. For specific segmentation tasks, the quality measure could be defined in terms of specific features of the desired segmentations (shape, topology, etc.) or even ultimately by the ability to facilitate a correct diagnosis. However, for a general

evaluation method, the measure must be simple and geometric. This allows area or boundary oriented measures such as the area overlap or the mean minimal boundary distance between the ground truth segmentation and the segmentation being evaluated.

In order to evaluate the performance of my hybrid segmentation approach, both simulated and real MRI scans are tested. The simulated MRI brain scans is from the BrainWeb [CZKSKHE98], which allows one to generate high quality simulated MRI volumes from known (ground truth) anatomical models, for different levels of noise and spatial inhomogeneities. The real data is from the Internet Brain Segmentation Repository (IBSR), where the expert manual segmentation is offered. Even though manual segmentations cannot be considered to be 100% “ground truth”, they are a good way to begin comparing automated segmentation methods. As figure 5.16 shows, (a) is a simulated T1 MR brain scan with ground truth white and gray matter. The volume is 181x217x181 with coronal slice thickness 1 mm, intensity non-uniformity level 20%, and noise level 9%. The data comes from <http://www.bic.mni.mcgill.ca/brainweb>. (b) is a real MR brain scan image, which is available from IBSR. Manually-guided segmentation results (shown as green outlines) are provided for some data sets along with the corresponding MR images. The MR image data is a T1-weighted 3D coronal brain scan after it has been normalized. The segmentation or “outline” files are the result of semi-automated segmentation techniques with user manually controlled, which require many hours of effort by a trained expert.

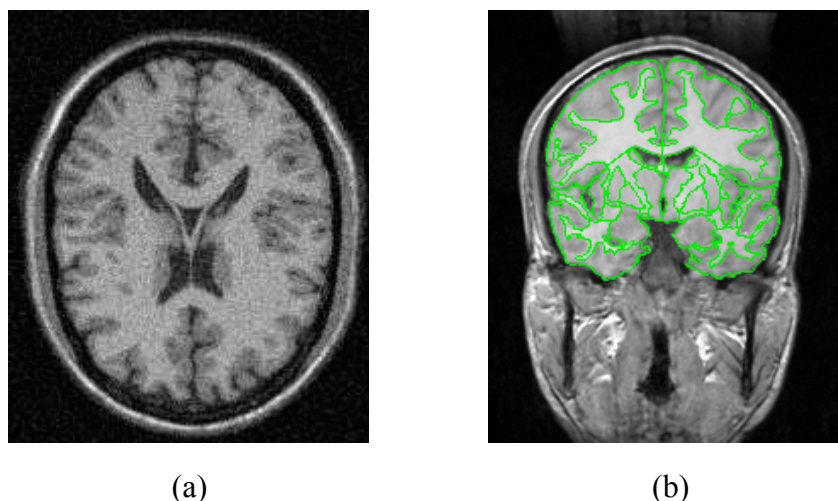


Figure 5.16 Simulated image and real image. (a) Simulated T1 MR brain scan with ground truth white and gray matter. (b) Real MR brain scan image

I compare the volume of overlap between the hybrid segmentation approach and ground truth or expert manual segmentation to analysis the accuracy of the segmentation result. Accuracy is defined as the overall number of correctly segmented pixels with respect to the standard segmentation

$$A = \frac{V_{TP} + V_{TN}}{V_T} \% \quad (5.30)$$

Relative to the total number of pixels  $V_T = 256 \times 256$  of the 2D MRI.  $V_{TP}$  (true positive volume or sensitivity) denotes the number of pixels correctly segmented as the structure of interest and  $V_{TN}$  (true negative volume specificity) denotes the number of pixels correctly segmented as background.

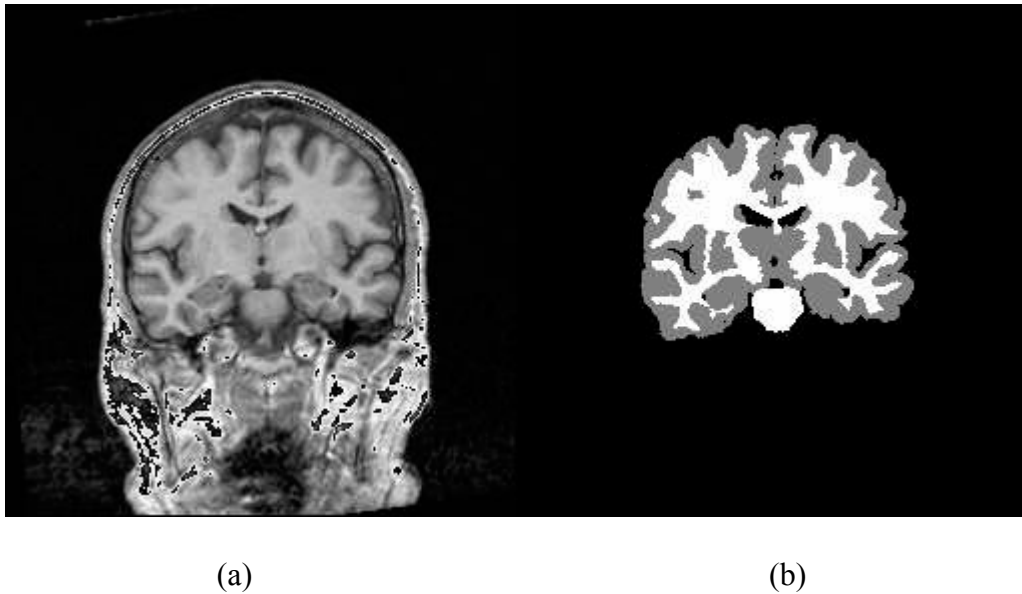


Figure 5.17 Example of segmentation. (a) is T1 MR brain scan. (b) is the white and gray segmentation

In order to evaluate my approach, I select fifty images both simulated and real images for this study. One result is shown in figure 5.17. Segmentation accuracy is illustrated in figure 5.18 and figure 5.19. The average accuracies for simulated image and real image are  $97.38 \pm 0.29\%$ ,  $96.82 \pm 0.33\%$ , separately (mean  $\pm$  standard deviation).

The running time of the algorithm is not evaluated since as current computing capability of the hardware, the running time is not main factor of the algorithm. There is not big distinguishing in running time from one method to the others.

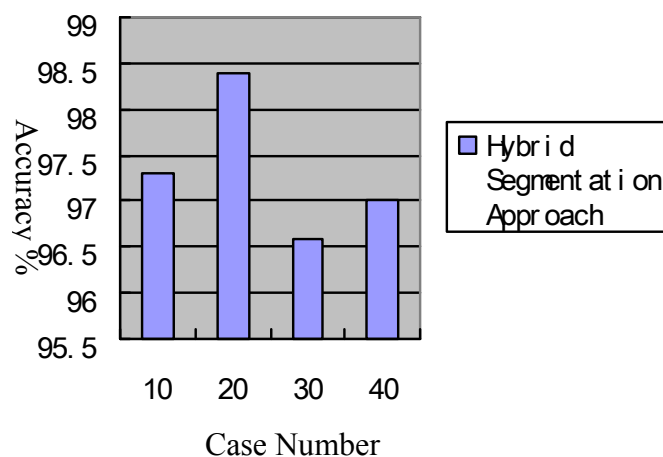


Figure 5.18 Segmentation accuracy illustrations for simulated images of 50 cases

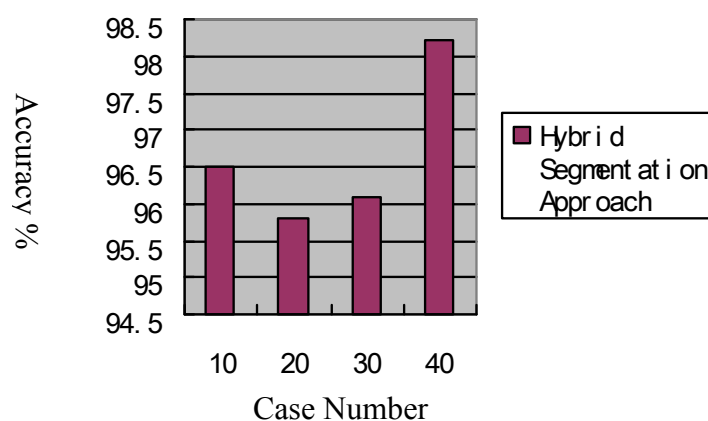


Figure 5.19 Segmentation accuracy illustrations for real images of 50 cases

## 5.7 Chapter Summary

In this chapter, I present a new hybrid image segmentation approach, and each segmentation step in the hybrid approach has been discussed in details. The experimental results have also been shown and the accuracy of the segmentation has been evaluated. This hybrid approach is designed for semi-automated segmentation of radiological image, such as CT, MRI, etc, to get the organ or interested area from the



image. This approach integrates region-based method and boundary-based method. Such integration reduces the drawbacks of both methods and enlarges the advantages of them.

The process starts from an initial seed area in the target object, and uses the fuzzy affinity to get an estimation of the object's boundary. At the same time, the algorithm generates the homogeneity operator for the target object. This can be the classifier in next step. Then with the Voronoi Diagram (VD), the image is redefined to outside, inside or boundary regions by the region classification method with the homogeneity operator. After it, the boundary will be refined to fill in the missing boundary and to override the spurious boundary data with a deformable surface model, level set model. This hybrid approach amplifies the strengths of both region-based and edge-based techniques but reduces the weaknesses of them.

This hybrid approach is semi-automated, since the whole segmentation procedure doesn't need much manual intervention, except for the initial seed position selection for the first step segmentation. The semi-automated also yields another advantage of the hybrid approach, that is, it doesn't need training set for the segmentation, and it's easy to put this approach into the practice.

Medical image segmentation is the main part in my research. But as I introduced in the previous chapters, it is not an isolated part in the telemedicine system. For instance, the segmentation result can be displayed by several rendering methods introduced in the last chapter. Besides with certain kind of mean, the processed result can be stored into the header part of a DICOM file, so that different viewer tool can display the same segmentation result. Just as I have realized in the demo system.

## **CHAPTER VI**

### **CONCLUSION AND FUTURE WORK**

On-going technology innovations increase the ease with which we can obtain, manage and communicate high quality large volumetric image data sets to support medical care. We need accurate and efficient image processing tools to realize the potential benefits embodied within image information, and all the information should be communicated remotely. The research that I have described in this dissertation demonstrated promising methods and a system that could assist radiology experts in processing, observing and communicating the image data.

#### **6.1 Contributions Summary**

The contributions of this work relate to image data processing within telemedicine framework, as necessary to support diagnosis, quantification, and therapy planning. In the following sections, I will discuss specific contributions. Section 6.1.1 discusses contributions to my system integrated to telemedicine framework. Section 6.1.2 describes contributions to the multiple data visualization methods. Section 6.1.3 outlines contributions to image segmentation. These contributions are also summarized in my related papers, see bibliography.

##### **6.1.1 Contributions in Integrated to Telemedicine Framework**

My system is developed within telemedicine framework. One telemedicine system can involve many kinds of techniques. My research work focuses on medical data storage in telemedicine. In telemedicine field, DICOM standard is the predominant standard for the communication of medical images. I illustrate in my system the main

technology of operation process of DICOM files, in addition, I also propose a method to repair the non-standard DICOM files so that they can be recognized by general DICOM viewers. Those non-standard files could be very old files, or offered by specific DICOM devices manufacture vendors. The aim of repairing is to make use of old files, and also overcome the barriers in different vendors to facilitate communications. Concerning image data storage, I proposed a novel 3D image data storage method, which is lacking in DICOM standard.

### **6.1.2 Contributions in Medical Image Visualization**

For medical image visualization, some techniques for rendering 3D medical data are described. My system offers different manners for image data visualization both 2D and 3D. They consist of (1) 2D image data display techniques, which render image as different window width and window center, and cut through 3D data volume by a plane in arbitrary degree to fly-in-out; (2) surface-based techniques, which apply a surface detector to the sample array, then fit geometric primitives to the detected surfaces, and finally render the resulting geometric representation; (3) volume-rendering techniques, a variant of the binary voxel techniques in which a color and a partial opacity are assigned to each voxel; images are formed from the resulting colored, semitransparent volume by blending together voxels projecting to the same pixel on the picture plane. These visualization approaches reveal the different aspects of image data from 2D to 3D; hence my system benefits physicians to observe images.

### **6.1.3 Contributions in Medical Image Segmentation**

Medical image segmentation is one difficult task in image processing work. In this thesis I presented a novel, accurate and efficient and method: the hybrid segmentation approach, and it is an optimized combination of region-base and edge-based

segmentation methods. Firstly, I use fuzzy affinity to get an initial segmentation result, and the homogeneity operator is also generated in this step. Then I use Voronoi Diagram-based method to refine the last step's result. Finally the level set model is used to handle some vague or missed boundary, and get smooth and accurate segmentation.

The segmentation procedure is relative automatic. Although my system is not full automatic, the semi-automatic processing can still reduce a lot of labor work. It only needs a few manual works to give the initial seed area in the target region. In the case the segmentation work is totally done by expert manually, it will cost several hours in one set of images. The image segmentation taken by computer system reduces the operator time from several hours to a few minutes, so that this makes it practical to consider the integration of computerized segmentation into daily clinical routine.

## **6.2 Future Work**

I have described my multi-visualization and hybrid segmentation approaches within telemedicine framework, and presented a medical image process system which aims to demonstrate the methods I introduced in this dissertation. However, several avenues for further study remain open at the end of this work. This section contains a very brief overview of potential directions for future work

### **6.2.1 Additional Supplement to DICOM Standard**

DICOM file is an information composite file, which contains both image data and related information. After the segmentation processing, some new information is created, such as segmented boundary, segmented area size or some descript text. All of such information should be stored in one segmented file so that it can be retrieved and communicated with other physicians, which is even the advantage of DICOM

format file. In current standard, there are two possible ways to store them. One way were, to use “Color Look Up Table” (add one additional for the Alpha Component) into a Composite Object, which also contains a reference to the image, on which the “Transfer Function” shall be applied. The other option were, to define a “Structured Report” template for storing “Transfer Functions” as (part of a) DICOM SR Object. However, both of them are not sufficient for special storage requirement. Some additional id and information object definition are required. Therefore it should be a significative research point in the future to make a new supplement to DICOM standard for storage of the segmented information, furthermore, for display of segmented result and possible comments.

### **6.2.2 Multi-Modalities Visualization**

In order to help decision-making, it requires the integration of several types of information in a meaningful, reliable, and efficient manner. However, this integration involves voluminous amounts of data, thereby presenting a formidable problem in the representation, display, and interpretation of multidimensional information. In the current research, my approach can only render one kind of modality data in one process. In the future research, the objective should be to develop an unified methodology to objectively and accurately integrate the diverse types of modality data, even with sparse and noisy data, which will allow to correctly display and interactively manipulate the synthesized information data, and which will support interpretive decision-making at higher levels of abstraction.

### **6.2.3 Segmentation Extensions and Additional Evaluation Study**

Medical image volumes are amenable to segmentation with 3D methods, which process voxel arrangements in the same manner independent of orientation, or with 2D methods, which process 2D sections independently and then combine the

results to form the segmented volume. In my current research on segmentation, I focus on the approaches with 2D methods. But in fact, this hybrid segmentation method is readily extensible to 3D in cases where the assumptions that we make for 2D segmentation apply also to 3D datasets. Two considerations are relevant to the decision of whether to perform volume segmentation in 2D or in 3D: the volume resolution, and the complexity of the algorithm. 2D algorithms are likely to produce more accurate results when the volume resolution is anisotropic and the resolution for one axis is significantly lower than the resolution for the other two axes. 2D algorithms are likely to be more practical if the complexity of the algorithm is greater than linear in the number of voxels. It is clearly that the segmentation algorithms' extension to 3D certainly gives birth to more computational complexity. Further research should be focused on how to decrease the complexity of 3D segmentation algorithms and make my hybrid approach more efficient.

On the other hand, the segmentation hybrid approach presented in this thesis currently could be used for the preparation of data for image guided surgery in a clinical setting. In an ongoing study, the method will be further evaluated based on a larger number of patient cases. I should design experiments to quantify the characteristics so that this system assumes for the input data (e.g., contrast-to-noise ratio) to allow more realistic expectations about performance.

### **6.3 Chapter Summary**

In this chapter, I have firstly summarized the contributions of my research in three major aspects: the telemedicine, such as the supplement of DICOM standard and process techniques of DICOM data; the image visualization, such as three rendering methods of medical image; and the novel hybrid medical image segmentation approach. Then, I highlighted directions for future research that should be done on the

basis of the current work, and hopefully lead my research more progress, for example, the supplement of segmented object storage in DICOM standard, multi-modalities data visualization, and segmentation approach for 3D images, and so on.

**BIBLIOGRAPHY**

- [AB94] R. Adams and L. Bischof. Seeded region growing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(6):642–645, 1994.
- [ACCCLM96] N. Ayache, P. Cinquin, I. Cohen, L. Cohen, F. Leitner, and O. Monga, “Segmentation of complex three-dimensional medical objects: a challenge and a requirement for computer-assisted surgery planning and performance,” *Computer-Integrated Surgery: Technology and Clinical Applications*, pp. 59–74, MIT Press, 1996.
- [AS94] D. Adalsteinsson and J. A. Sethian, “A fast level set method for propagating interfaces,” *Journal. of Computational Physics*, pp. 269-277, 1994.
- [AYMF02] M. N. Ahmed, S. M. Yamany, N. Mohamed, A. A. Farag. “A Modified Fuzzy C-Means Algorithm for Bias Field Estimation and Segmentation of MRI Data.” *IEEE TRANSACTIONS ON MEDICAL IMAGING*, VOL. 21, NO. 3, MARCH 2002
- [Ben75] J. L. Bentley, “Multidimensional Binary Search Trees Used for Associative Searching.” *Communications of ACM* 18(9): pp.509-517, 1975
- [BGCK93] K.T. Bae, M.L. Giger, C. Chen, C.E. Kahn. “Automatic segmentation of liver structure in CT images.” *Medical Physics*, 20:71–78, 1993.
- [BHC93] J.C. Bezdek, L.O. Hall, and L.P. Clarke. “Review of MR image segmentation techniques using pattern recognition.” *Medical Physics*, 20:1033–1048, 1993.
- [BMAS97] P.M. Black, T. Moriarty, E. Alexander III, P. Stieg, E.J. Woodard, P.L. Gleason, C.H. Martin, R. Kikinis, R. Schwartz, F.A.Jolesz. “The Development and Implementation of Intraoperative MRI and its Neurosurgical Applications.” *Neurosurgery* October 1997; 41:831-842.
- [BMRS01] R. Bellazi, S. Montani, A. Riva, and M. Stefanelli, “Web-based telemedicine systems for home-care: Technical issues and experiences,” *Computer Methods and Programs in Biomedicine*, vol. 64, pp. 175–187, 2001.
- [BPC93] Bertin E., Parazza F., and Chassery J.M., “Segmentation and Measurement Based on 3D Voronoi Diagram: Application to



Confocal Microscopy”, *Special Issues in Computerized Medical Imaging and Graphics*, vol.17, pp.175-182, 1993.

- [BusinessWeek02] <http://www.businessweek.com>. “Focusing on Picture-Perfect Diagnoses”. *Business Week* October, 15<sup>th</sup>, 2002.
- [CA79] G.B. Coleman, H.C. Andrews. “Image segmentation by clustering.” *Proceedings of IEEE*,67(5):773–785, 1979.
- [CGHPVDT02] M. Bach Cuadra, J. Gomez1, P. Hagmann1, C. Pollo, J.-G. Villemure, B.M. Dawant, J. Ph. Thiran. “Atlas-Based Segmentation of Pathological Brains Using a Model of Tumor Growth.” *Proceedings of the 5th International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2002, Pages: 380 - 387
- [CK02] S. H. Choi, K. T. Kwok, Hierarchical slice contours for layered-manufacturing, *Computers in Industry*, v.48 n.3, p.219-239, August 2002
- [CLKJ90] H.E. Cline, E. Lorensen, R. Kikinis, F.A. Jolesz. “Three-Dimensional Segmentation of MR Images of the Head using Probability and Connectivity.” *Journal of Computer Assisted Tomography*, 14(6):1037–1045, 1990.
- [CMHKG01] Csebfalvi B., Mroz L., Hauser H., König A., Gröller E. “Fast visualization of object contours by non-photorealistic volume rendering”. *Computer Graphics Forum* 20, 3 (2001), 452–460.
- [Com01] Committee on Quality of Health Care in America, Institute of Medicine, “Crossing the Quality Chasm: A New Health System for the 21st Century”. *Washington, D.C.: National Academy of Sciences*, 2001.
- [CVPSAS93] L.P. Clarke, R.P. Velthuizen, S. Phuphanich, J.D. Schellenberg, J.A. Arrington, M. Silbinger. “MRI: Stability of Three Supervised Segmentation Techniques.” *Magnetic Resonance Imaging*, 11(1):95–106, 1993.
- [CZKSK98] D.L. Collins, A.P. Zijdenbos, V. Kollokian, J.G. Sled, N.J. Kabani, et al. “Design and construction of a realistic digital brain phantom.” *IEEE Transactions on Medical Imaging*, 17:463–468, 1998.
- [CZKSKHE98] D.L. Collins, A.P. Zijdenbos, V. Kollokian, J.G. Sled, N.J. Kabani, C.J. Holmes, A.C. Evans. Design and Construction of a Realistic Digital Brain Phantom. *IEEE Transactions on Medical Imaging*, 17, June 1998. <http://www.bic.mni.mcgill.ca/brainweb/>.

- [DP95] C. A. Davatzikos and J. L. Prince, "An active contour model for mapping the cortex," *IEEE Trans. on Medical Imaging*, vol. 14, pp. 65–80, 1995.
- [EFAM97] A.C. Evans, J.A. Frank, J. Antel, D.H. Miller. "The Role of MRI in Clinical Trials of Multiple Sclerosis: Comparison of Image Processing Techniques." *Ann Neurol*, 41:125–132, 1997.
- [GBBH96] P. Gibbs, D.L. Buckley, S.J. Blackband, A. Horsman. "Tumour volume detection from MR images by morphological segmentation." *Physics in Medicine and Biology*, 41:2437–2446, 1996.
- [GFK96] E. Gelenbe, Y. Feng, K.R.R. Krishnan. "Neural network methods for volumetric magnetic resonance imaging of the human brain." *Proceedings of IEEE*, 84:1488–1496, 1996.
- [Gig00] M.L. Giger. "Computer-aided Diagnosis of Breast Lesions in Medical Images". *Computing in Science & Engineering* Sept-Oct 2000; 2:39-45.
- [GGK02] D. T. Gering, W. L. Grimson<sup>1</sup>, R. Kikinis. "Recognizing Deviations from Normalcy for Brain Tumor Segmentation" *Proceedings of the 5th International Conference on Medical Image Computing and Computer-Assisted Intervention Part I*: 388 - 395 (2002)
- [GK96] A. Van Gelder, K. Kim. "Direct volume rendering with shading via three-dimensional textures". *Volume Visualization Symposium Proceedings*, pages 23–30, San Francisco, CA, October 1996.
- [GKRR02] R. Goldenberg, R. Kimmel, E. Rivlin, M. Rudzsky. "Cortex Segmentation: A Fast Variational Geometric Approach." *IEEE Transactions on Medical Imaging*, 21(2):1544-51, December 2002
- [GMAKW04] V. Grau, A. U. J. Mewes, M. Alcañiz, R. Kikinis, S. K. Warfield. "Improved Watershed Transform for Medical Image Segmentation Using Prior Information." *IEEE TRANSACTIONS ON MEDICAL IMAGING*, VOL. 23, NO. 4, APRIL 2004, P. 447-458
- [GNK+01] D.T. Gering, A. Nabavi, R. Kikinis, N. Hata, L.J. Odonnell, W. Eric L. Grimson, F. A. Jolesz, P. Black, W. Wells III. "An Integrated Visualization System for Surgical Planning and Guidance Using Image Fusion and an Open MR". *Journal of Magnetic Resonance Imaging* June 2001; 13:967-975.
- [GRV02] B.V. Ginneken, B.M.H. Romeny, M.A. Viergever. "Computer-Aided Diagnosis in Chest Radiography: A Survey". *IEEE Transaction on Medical Imaging* December 2001; 20:1228-1241.

- [GW87] R.C. Gonzalez, P. Wintz. *Digital Image Processing*. Addison-Wesley, Reading, MA, USA, 1987.
- [GW92] R.C. Gonzalez, R.E. Woods. *Digital Image Processing*. Addison-Wesley, 1992.
- [Hay94] S. Haykin. *Neural networks: a comprehensive foundation*. Macmillan College, New York, 1994.
- [HBG02] S. Ho, E. Bullitt, G. Gerig. "Level Set Evolution with Region Competition: Automatic 3-D Segmentation of Brain Tumors." *Proceedings of 16th International Conference on Pattern Recognition*, Aug 2002, pp. 532-535
- [HDKJW97] Hata N, Dohi T, Kikinis R, Jolesz FA, Wells III WM. "Computer assisted intra-operative MR-guided therapy: pre and intra-operative image registration, enhanced three-dimensional display, deformable registration." *Proc. 7th Annual Meeting of Japan Society of Computer Aided Surgery*, Sapporo, Japan, 1997. p 119–120.
- [HLRC98] P. Haigron, G. Lefaix, X. Riot, and R. Collorec, "Application of spherical harmonics to the modelling of anatomical shapes," *Journal of Computing and Information Technology*, vol. 6, no. 4, pp. 449–461, December 1998.
- [HS85] R.M. Haralick and L.G. Shapiro. "Image segmentation techniques." *Computer Vision, Graphics, and Image Processing*, 29:100–132, 1985.
- [Jae91] B. Jaehne. *Digital Image Processing: Concepts, Algorithms and Scientific Application*. Springer-Verlag, Berlin-Heidelberg-Tokyo, 1991.
- [JM05] Ch., X. Zhang, Ch. Meinel, "Hybrid Framework for Medical Image Segmentation." *Proc. CAIP 05*, Paris (France), 2005, pp.264-272
- [JVM02] Ch. Jiang,L. Vorwerk, Ch. Meinel, "Standardizing DICOM File and Rendering 3-D Medical Image." *Proc. EuroPACS 2002*, Oulu (Finland), 2002, pp. 225
- [JVM02a] Ch. Jiang, L. Vorwerk, Ch. Meinel, "Deformation and Construction of 3-D Medical Image." *Proc. ICDIA 2002*, Shanghai (China), 2002, pp. 48-53
- [JVM03] Ch. Jiang, L. Vorwerk, Ch. Meinel, "Concept for DICOM about 3D data interchange." *Proc. TTC 2003 Tromso (Norway)*, 2003, pp. 40

- [JZGM04] Ch. Jiang, X. Zhang, M. Gevantmakher, Ch. Meinel, "A new Practical Tool for 3D Medical Image Segmentation, Visualization and Measurement." *Proc. CCCT 2004* Austin (USA), 2004, pp.7-10
- [JZHM04] Ch. Jiang, X. Zhang, W. Huang, Ch. Meinel, "Segmentation and Quantification of a Brain Tumor." *Proc. IEEE VECIMS 2004* Boston/MA (USA), 2004, pp. 61-66
- [JZM05] Ch. Jiang, X. Zhang, Ch. Meinel, "Medical Image Segmentation Using a Combined Approach." *Proc. VISION 05*, Las Vegas (USA), 2005, pp.84-90
- [Kau91] A. Kaufman. "Introduction to Volume Visualization". *IEEE Computer Society Press*, 1991
- [Kep75] Keppel, E., "Approximating Complex Surfaces by Triangulation of Contour Lines," *IBM Journal of Research and Development*, Volume 19, Number 1, January 1975, pp. 2-11.
- [KDFPTL97] V. S. Khoo, D. P. Dearnaley, D. J. Finnigan, A. Padhani, S. F. Tanner, M. O. Leach, "Magnetic resonance imaging (MRI): considerations and applications in radiotherapy treatment planning," *Radiotherapy and Oncology*, vol. 42, no. 1, pp. 1–15, 1997.
- [KGKW98] T. Kapur, W.E.L. Grimson, R. Kikinis, W.M. Wells. "Enhanced spatial priors for segmentation of magnetic resonance imagery." *Proceedings of the 1st International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 457–468, 1998.
- [KH84] J. T. Kajiya, T. Von Herzen. "Ray Tracing Volume Densities". *Computer Graphics*, 18(3):165–173, July 1984.
- [KSCF95] M. Kamber, R. Shinghal, D.L. Collins, G.S. Francis, A.C. Evans. "Model-Based 3-D Segmentation of Multiple Sclerosis Lesions in Magnetic Resonance Brain Images." *IEEE Transactions on Medical Imaging*, 14(3):442–453, 1995.
- [KWT88] M. Kass, A. Witkin, D. Terzopoulos. "Snakes: Active contour models." *International Journal of Computer Vision*, 1:321–331, 1988.
- [KWNBJK01] M. R. Kaus, S. K. Warfield, A. Nabavi, P. M. Black, F. A. Jolesz, R. Kikinis. "Automated Segmentation of MRI of Brain Tumors." *Radiology*. 2001;218:586-591.
- [Lev88] M. Levoy. "Display of Surfaces from Volume Data". *IEEE Computer Graphics and Applications*, 8(3):29–37, March 1988.

- [Lev90] M. Levoy. "Efficient Ray-Tracing of Volume Data." *ACM Transactions on Graphics*, v. 9, n. 3, 1990, p. 245–261.
- [Li95] S.Z. Li. *Markov random field modeling in computer vision*. Springer, 1995.
- [LL94] P. Lacroute, M. Levoy. "Fast Volume Rendering Using a Shear-Warp factorization of the Viewing Transform". *Computer Graphics, Proceedings of SIGGRAPH 94*, pages 451–457, July 1994.
- [LP89] K.O. Lim, A. Pfefferbaum. "Segmentation of MR Brain Images into Cerebrospinal Fluid Spaces, White and Grey Matter." *Journal of Computer Assisted Tomography*, 13(4):588–593, 1989.
- [LRSFFETM97] J.L. Lancaster, L.H. Rainey, J.L. Summerlin, C.S. Freitas, P.T. Fox, A.C. Evans, A.W. Toga, and J.C. Mazziotta. "Automated labeling of the human brain: A preliminary report on the development and evaluation of a forward-transform method." *Human Brain Mapping*, 5:238–242, 1997.
- [LS92] T. Lei, W. Sewchand. "Statistical approach to X-Ray CT imaging and its applications in image analysis – part II: A new stochastic model-based image segmentation technique for X-Ray CT image." *IEEE Transactions on Medical Imaging*, 11(1):62–69, 1992.
- [LTA02] Rudy J Lapeera, AC Tan, Richard Aldridge. "A Combined Approach to 3D Medical Image Segmentation Using Marker-based Watersheds and Active Contours: the Active Watershed Method." *Proceedings of Medical Image Understanding - MIUA*, 2002
- [LC87] W. E. Lorensen, H. E. Cline. "Marching-cubes: A high resolution 3d surface construction algorithm". *Computer Graphics, Proceedings of SIGGRAPH 87*, pages 163–169, 1987.
- [LMERH02] Lu A., Morris C. J., Ebert D. S., Rheingans P., Hansen C. "Non-photorealistic volume rendering using stippling techniques". *Proceedings of IEEE Visualization '02* (Oct. 2002), 211–218.
- [MBLG02] N. Moon, E. Bullitt, K. V. Leemput, G. Gerig. "Automatic Brain and Tumor Segmentation." *Proceedings of the 5th International Conference on Medical Image Computing and Computer-Assisted Intervention –Part I 2002*: 372-379
- [MLC99] F. Magrabi, N. H. Lovell, and B. G. Celler, "A web-based approach for electrocardiogram monitoring in the home," *International Journal of Medical Informatics*, vol. 54, pp. 145–153, 1999.

- [MSV95] R. Malladi, J. A. Sethian, and B. C. Vemuri, "Shape modeling with front propagation: A level set approach," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, pp. 158-175, 1995.
- [MUCR98] I.N. Manousakas, P.E. Undrill, G.G. Cameron, T.W. Redpath. "Split-and-merge segmentation of magnetic resonance medical images: performance evaluation and extension to three dimensions." *Computers and Biomedical Research*, 31:393-412, 1998.
- [NEMA00] NEMA, *Digital Imaging and Communications in Medicine*, Part 1-15, (NEMA Standards Publication PS3.x, 2000)
- [OS88] S. Osher and J. A. Sethian, "Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations." *Journal of Computational Physics*, vol. 79, pp. 12-49, 1988.
- [Ots79] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms." *IEEE Transactions on Systems, Man, and Cybernetics*, volume SMC-9, pages 62-66, January 1979.
- [PA95] DA. Perednia, A. Allen, "Telemedicine technology and clinical applications," *The Journal of the American Medical Association*, 1995; 273: 483-488
- [PDMCEO96] Peters T, Davey B, Munger P, Comeau R, Evans A, Olivier A. "Threedimensional multimodal image-guidance for neurosurgery." *IEEE Transactions on Medical Imaging*, 1996;15:121-128.
- [PD84] Porter, T. and Duff, T. "Compositing digital images." *Computer Graphics* 18, 3 (July, 1984), 253-259.
- [PEG03] Marcel Prastawa, Elizabeth Bullitt, Guido Gerig. "Robust Estimation for Brain Tumor Segmentation." *MICCAI (Medical Image Computing and Computer-Assisted Intervention)*, Vol. 2: 530-537, 2003
- [PPDX97] D.L. Pham, J.L. Prince, A.P. Dagher, C. Xu. "An automated technique for statistical characterization of brain tissues in magnetic resonance imaging." *International Journal of Pattern Recognition and Artificial Intelligence*, 11(8): 1189-1211, 1997.
- [PS88] F. P. Preparata and M. I. S. Shamos, *Computational Geometry, an Introduction*. Springer Verlag, New York, 1988
- [RGKFC92] J. Rademacher, A.M Galaburda, D.N. Kennedy, P.A. Filipek, V.S. Caviness. "Human cerebral cortex: localization, parcellation and morphometry with magnetic resonance imaging." *Journal of Cognitive Neuroscience*, 4:352-374, 1992.

- [SABT92] A. Simmons, S.R. Arridge, G.J. Barker, P.S. Tofts. "Segmentation of Neuroanatomy in Magnetic Resonance Images." *SPIE Medical Imaging VI: Image Processing*, 1652:2–13, 1992.
- [Sam84] Samet, H., "The Quadtree and Related Hierarchical Data Structures." *Computation Surveys*, vol. 16, No. 2, pp. 187-230, 1984.
- [Ser82] J. Serra. *Image Analysis and Mathematical Morphology*. Academic Press, 1982.
- [SG04] J. Sun, D. Gu. "Bayesian Image Segmentation Based on an Inhomogeneous Hidden Markov Random Field." *Pattern Recognition, 17th International Conference on (ICPR'04) Volume 1*, August 23 - 26, 2004, pp. 596-599
- [SJR+95] J.F. Schenk, F.A. Jolesz, P.B. Roemer, others. "Superconducting open-configuration MR imaging system for image-guided therapy". *Radiology* 1995; 195:805-814.
- [SK01] J. H. Schneider, D.Kofos, "Consumer Internet medical records: Benefits for pediatrics, problems and proposed standards," *American Telemedicine Association Annual Meeting*, Ft. Lauderdale, 2001.
- [SML96] Will Schroeder, Ken Martin, Bill Lorensen. *The Visualization Toolkit: An Object-Oriented Approach to 3D Graphics*. Prentice-Hall,1996. See also the vtk website: [www.kitware.com](http://www.kitware.com).
- [SRN00] S. Shiffman, G. D. Rubin, S. Napel. "Medical Image Segmentation Using Analysis of Isolable-Contour Maps" *IEEE TRANSACTIONS ON MEDICAL IMAGING*, VOL. 19, NO. 11, NOVEMBER 2000, PP. 1064-1074
- [SS92] J. A. Sethian and J. Strain. "Crystal growth and dendritic solidification." *Journal of Computational Physics*, vol. 98, pp. 231-253, 1992.
- [SSW88] P. K. Sahoo, S. Soltani, A.K.C.Wong. "A survey of thresholding techniques." *Computer Vision, Graphics, and Image Processing*, 41:233–260, 1988.
- [Tay95] P. Taylor, "Invited review: computer aids for decision-making in diagnostic radiology - a literature review," *British Journal of Radiology*, vol. 68, pp. 945–957, 1995.
- [US96] J. K. Udupa, S. Samarasekera. "Fuzzy connectedness and object definition: Theory, algorithms and applications in image

- segmentation.” *Graphic Models and Image Processing*, 58(3):246–261, 1996.
- [VJM02] L. Vorwerk, Ch. Jiang, Ch. Meinel, “Application for Repairing and Presenting DICOM Objects.” *Proc. IMSA 2002*, Kauai (Hawaii, USA), 2002, pp. 318-323
- [VJBM03] Lutz Vorwerk, Chunyan Jiang, Ulf Birkel, Christoph Meinel, “Deformierung von kieferorthopädischen 3D-Aufnahmen aus 2D-DICOM Bildern. ” *Telemedizinführer Deutschland*, pp. 184-186, Achim Jäckel, Deutsches Medizin Forum, 2003
- [VPSNRS91] M.W. Vannier, T.K. Pilgram, C.M. Speidel, L.R. Neumann, D.L. Rickman, L.D. Schertz. “Validation of Magnetic Resonance Imaging (MRI) multi-spectral Tissue Classification.” *Computerized Medical Imaging and Graphics*, 15:217–223, 1991.
- [VS91] L. Vincent, P. Soille. “Watersheds in digital spaces: an efficient algorithm based on immersion simulation.” *IEEE Transactions of Pattern Analysis Machine Intelligence*, 13:583–598, 1991.
- [WG91] J. Wilhelms, A. Van Gelder. “A Coherent Projection Approach for Direct Volume Rendering”. *Computer Graphics, Proceedings of SIGGRAPH 91*, pages 275–284, 1991.
- [WMW86] G. Wyvill, C. McPheeters, B. Wyvill. “Data structure for soft objects.” *The Visual Computer*, 2:227–234, 1986.
- [WZL01] S. Wang, W. Zhu, Z. Liang. “Shape Deformation: SVM Regression and Application to Medical Image Segmentation.” *IEEE International Conference on Computer Vision (ICCV)*, II:209-216, Vancouver, Canada, 2001
- [WZW02] Simon K. Warfield, Kelly H. Zou, and William M. Wells. “Validation of Image Segmentation and Expert Quality with an Expectation-Maximization Algorithm.” *Proceedings of the 5th International Conference on Medical Image Computing and Computer-Assisted Intervention-Part I* Pages: 298 - 306 (2002)
- [XPP00] C. Xu, D. Pham, and J. Prince, "Image segmentation using deformable models.," in *Handbook of Medical Imaging*, vol. 2: SPIE, 2000, pp. 129-174.
- [YD03] J. Yang, J. S. Duncan. “3D Image Segmentation of Deformable Objects with Shape-Appearance Joint Prior Models.” *MICCAI (Medical Image Computing and Computer-Assisted Intervention)*, vol.1, 573-580, 2003.



- [YD04] J. Yang, J. S. Duncan. "3D image segmentation of deformable objects with joint shape-intensity prior models using level sets." *Medical Image Analysis* 8 (2004) 285-294
- [YZK03] A. Yezzi , L. Zollei , T. Kapur. „A variational framework for integrating segmentation and registration through active contours” *Medical Image Analysis* 7 (2003) 171–185
- [Zad65] L.A. Zadeh. "Fuzzy sets." *Information and Control*, 8:338–353, 1965.
- [ZD94] A.P. Zijdenbos, B.M. Dawant. "Brain segmentation and white matter lesion detection in MR images." *Critical Reviews in Biomedical Engineering*, 22:401–465, 1994.

## **APPENDIXES**

## APPENDIX A

### RAY CASTING ALGORITHM

Ray casting is a method used to render high quality images of solid objects. The basic goal of ray casting is to allow the best use of the three-dimensional data and not attempt to impose any geometric structure on it.

#### Brute - Force Algorithm

First consider the brute - force volume rendering algorithm as outlined in figure A.1. Begin with a 3D array of data samples. For simplicity, assume a scalar - valued array forming a cube  $N$  voxels on a side. Here, voxels are as point samples of a continuous function rather than as volumes of homogeneous value. Voxels are indexed by a vector  $\mathbf{i} = (i, j, k)$ , where  $i, j, k = 1, \dots, N$ , and the value of voxel  $\mathbf{i}$  is denoted  $f(\mathbf{i})$ . Using local operators, a scalar or vector color  $C(\mathbf{i})$  and an opacity  $\alpha(\mathbf{i})$  is derived for each voxel.

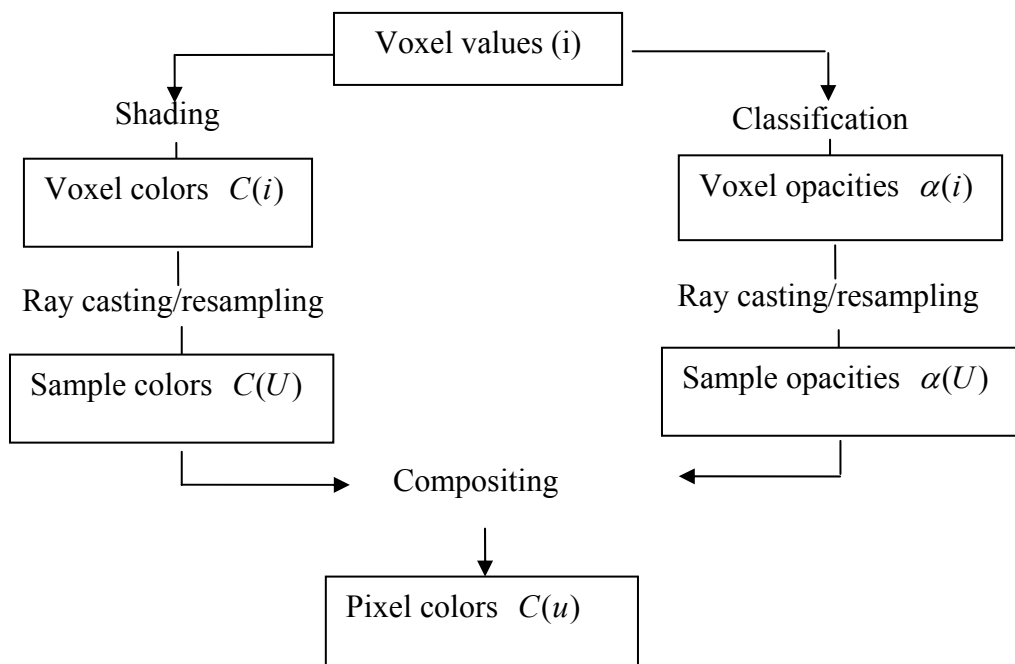


Figure A.1 Overview of volume rendering algorithm

Parallel rays then trace into the data from an observer position as shown in figure A.2. Assume that the image is a square measuring  $P$  pixels on a side, and that one ray is cast per pixel. Pixels and hence rays are indexed by a vector  $u = (u, v)$ , where  $u, v = 1, \dots, P$ . For each ray, a vector of colors and opacities is computed by resampling the data at  $W$  evenly spaced locations along the ray and trilinearly interpolating from the colors and opacities in the eight voxels surrounding each sample location. Samples are indexed by a vector  $U = (u, v, w)$  where  $(u, v)$  identifies the ray, and  $w = 1, \dots, W$  corresponds to distance along the ray with  $w = 1$  being closest to the eye. The color and opacity of sample  $U$  are denoted  $C(U)$  and  $\alpha(U)$  respectively. Finally, a fully opaque background is draped behind the dataset, and the resampled colors and opacities are composite with each other and with the background to yield a color for the ray. This color is denoted  $C(u)$ .

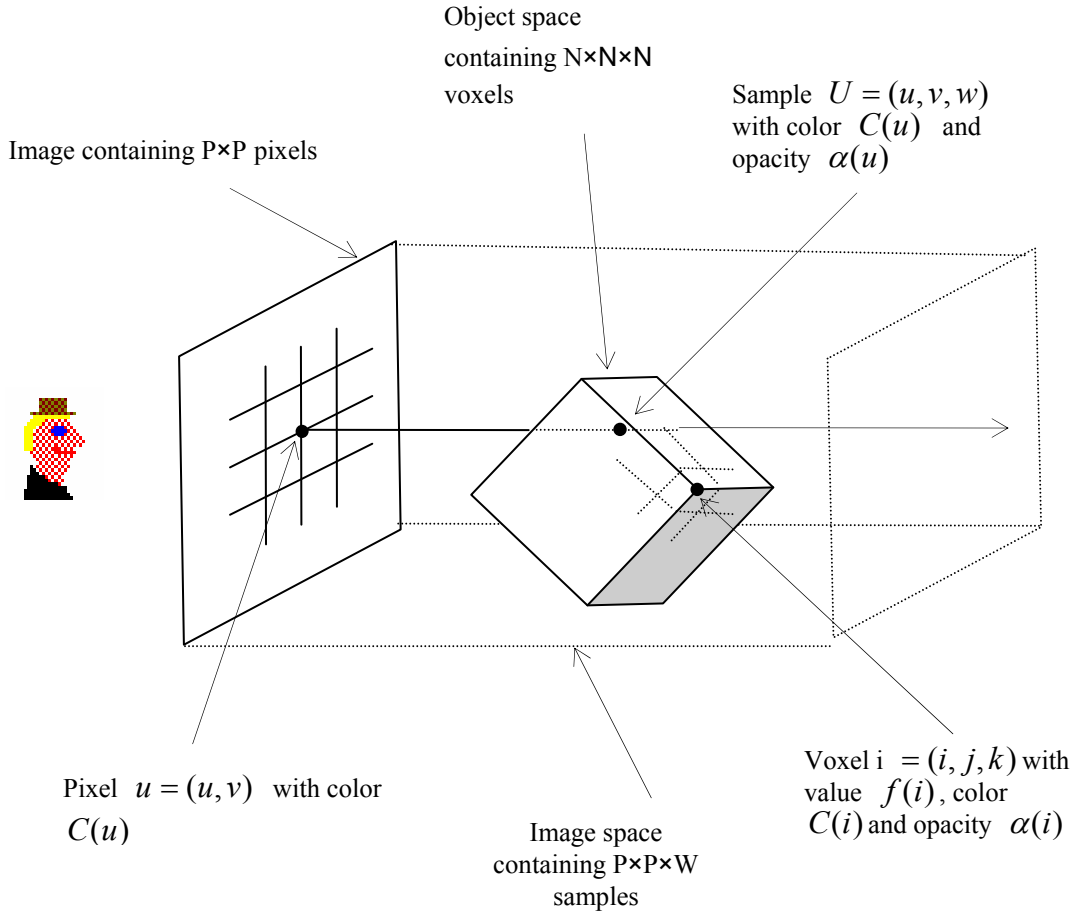


Figure A.2 Coordinate system used during volume rendering

This algorithm works from front to back, compositing the color and opacity at each sample location under the ray in the sense of [PD84]. Specifically, the color  $C_{out}(u;U)$  and opacity  $\alpha_{out}(u;U)$  of ray  $u$  after processing sample  $U$  is related to the color  $C_{in}(u;U)$  and opacity  $\alpha_{in}(u;U)$  of the ray before processing the sample and the color  $C(U)$  and opacity  $\alpha(U)$  of the sample by the transparency formula

$$\hat{C}_{out}(u;U) = \hat{C}_{in}(u;U) + \hat{C}(U)(1 - \alpha_{in}(u;U)) \quad (1a)$$

and

$$\alpha_{out}(u;U) = \alpha_{in}(u;U) + \alpha(U)(1 - \alpha_{in}(u;U)) \quad (1b)$$

where  $\hat{C}_{in}(u;U) = C_{in}(u;U)\alpha_{in}(u;U)$  ,  $\hat{C}_{out}(u;U) = C_{out}(u;U)\alpha_{out}(u;U)$  , and  $\hat{C}(U) = C(U)\alpha(U)$ .

After all samples along a ray have been processed, the color  $C(u)$  of the ray is obtained from the expression  $C(u) = C_{out}(u;W) / \alpha_{out}(u;W)$  where  $W = (u, v, W)$ . If a fully opaque background is draped behind the dataset at  $w' = W + 1$  and composed under the ray after it has passed through the data, then  $\alpha_{out}(u, W') = 1$  where  $W' = (u, v, w')$ , and this normalization step can be omitted.

The ray tracing, resampling, and compositing steps of the rendering algorithm are thus summarized as follows:

**Procedure** *TraceRay*<sub>1</sub>(*u*) **begin**

$\hat{C}(u) = 0$ ;

$\alpha(u) = 0$ ;

$x_1 = First(u)$ ;

$x_2 = Last(u)$ ;

$U_1 = [image(x_1)]$ ;

$U_2 = [image(x_2)]$ ;

{Loop through all samples falling within data}

**for**  $U = U_1$  **to**  $U_2$  **do begin**

$x = object(U)$ ;

{if sample opacity>0}

{then reample color and composite into ray}

$\alpha(U) = sample(\alpha, x)$ ;

**if**  $\alpha(U) > 0$  **then begin**

$$\hat{C}(U) = \text{sample}(\hat{C}, x);$$

$$\hat{C}(u) = \hat{C}(u) + \hat{C}(U)(1 - \alpha(u));$$

$$\alpha(u) = \alpha(u) + \alpha(U)(1 - \alpha(u));$$

**end**

**end**

**end** *TraceRay*<sub>1</sub>.

The **First** and **Last** procedures accept a ray index and return the object - space coordinates of the points where the ray enters and leaves the data respectively. These coordinates are denoted by real vectors of the form  $\mathbf{x} = (x, y, z)$  where  $1 \leq x, y, z \leq N$ . The **Object** and **Image** procedures convert between object-space coordinates and image - space coordinates. Although these calculations normally require matrix multiplications, they can be simplified for the restricted case of an orthographic viewing projection by retaining the coordinates computed in the previous invocation and using differencing. The **Sample** procedure accepts a 3D array of colours or opacities and the object - space coordinates of a point, and returns an approximation to the colour or opacity at that point by trilinearly interpolating from the eight surrounding voxels.

### **Optimized Algorithm**

The first optimization technique considered is hierarchical spatial enumeration. For a dataset measuring  $N$  voxels on a side where  $N = 2^M + 1$  for some integer  $M$ , this enumeration is represented by a pyramid of  $M + 1$  binary volumes as shown in figure A.3 for the case of  $N = 5$ . Volumes in this pyramid are indexed by a level number  $m$  where  $m = 0, \dots, M$  and the volume at level  $m$  is denoted  $V_m$ . Volume  $V_0$

measures  $N - 1$  cells on a side, volume  $V_1$  measures  $(N - 1)/2$  cells on a side, and so on up to volume  $V_m$ , which is a single cell. Cells are indexed by a level number  $m$  and a vector  $\mathbf{i} = (i, j, k)$  where  $i, j, k = 1, \dots, N - 1$ , and the value contained in cell  $\mathbf{i}$  on level  $m$  is denoted  $V_m(i)$ . It defines the size of cells on level  $m$  to be  $2^m$  times the spacing between voxels. Since voxels are treated as points, whereas cells fill the space between voxels, each volume is one cell smaller in each direction than the underlying dataset as shown in the figure. It also places voxel  $(1, 1, 1)$  at the front - lower - right corner of cell  $(1, 1, 1)$ . Thus, for example, cell  $(1, 1, 1)$  on level zero encloses the space between voxels  $(1, 1, 1)$  and  $(2, 2, 2)$ .

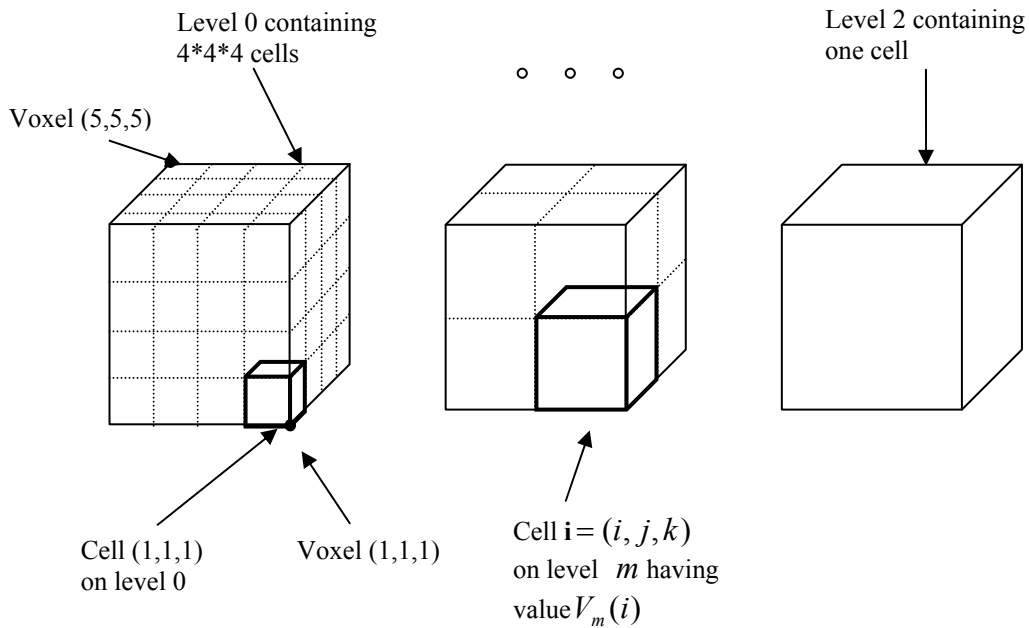


Figure A.3 Hierarchical enumeration of space object for  $N = 5$

The pyramid is constructed as follows. Cell  $\mathbf{i}$  in the base volume  $V_0$  contains a zero if all eight voxels lying at its vertices have opacity equal to zero. Cell  $\mathbf{i}$  in any volume  $V_m, m > 0$ , contains a zero if all eight cells on level  $m - 1$  that form its octants contain zeros. In other words, let  $\{1, 2, \dots, k\}^n$  be the set of all  $n$  - vectors with



entries  $\{1,2,\dots,k\}$ . In particular,  $\{1,2,\dots,k\}^3$  is the set of all vectors in 3- space with integer entries between 1 and  $k$ . Here then defines

$$V_0(i) = \begin{cases} 1 & \text{if } \alpha(i + \Delta i) = 1 \quad \text{for } i \in \{1,2,\dots,N-1\}^3 \text{ and } \text{any } \Delta i \in \{0,1\}^3 \\ 0 & \text{otherwise} \end{cases} \quad (2a)$$

and

$$V_m(i) = \begin{cases} 1 & \text{if } V_{m-1}(2i - \Delta i) = 1 \quad \text{for } i \in \{1,2,\dots,(N-1)/(m+1)\}^3 \text{ and } \text{any } \Delta i \in \{0,1\}^3 \\ 0 & \text{otherwise} \end{cases} \quad (2b)$$

for  $m = 1, \dots, M$ .

It now reformulates the ray tracing, resampling, and compositing steps of the rendering algorithm to use this pyramidal data structure. For each ray, it first computes the point where the ray enters the single cell at the top level. It then traverses the pyramid in the following manner. When a cell is entered, its value is tested. If it contains a zero, it advances along the ray to the next cell on the same level. If the parent of the new cell differs from the parent of the old cell, it moves up to the parent of the new cell. The reason to do this is because if the parent of the new cell is unoccupied, it can advance the ray further on the next iteration than if it had remained on a lower level. This ability to advance quickly across empty regions of space is where the algorithm saves its time. If, however, the cell being tested contains a one, it moves down one level, entering whichever cell encloses the current location. If it is already at the lowest level, it knows that one or more of the eight voxels lying at the vertices of the cell have opacity greater than zero. It then draws samples at evenly spaced locations along that portion of the ray falling within the cell, resamples the data at these sample locations, and composites the resulting color and opacity into the color and opacity of the ray.

The second optimization technique will be considered is adaptive termination of ray casting. The goal is to quickly identify the last sample location along a ray that significantly changes the color of the ray. Returning to equation (1a), it defines a significant color change as one in which  $C_{out}(u;U) - C_{in}(u;U) > \varepsilon$  for some small  $\varepsilon > 0$ . Since  $\alpha_{in}(u;U)$  in equation (1b) increases monotonically along the ray, no significant color changes occur beyond the point where  $\alpha_{out}(u;U)$  first exceeds  $1 - \varepsilon$ . This becomes the termination criterion. Higher values of  $\varepsilon$  reduce rendering time, while lower values reduce image artifacts.

Combining both of these optimizations gives the following algorithm:

**procedure**  $TraceRay_2(u)$  **begin**

$\hat{C}(u) = 0$  ;

$\alpha(u) = 0$  ;

$x = First(u)$  ;

$m = m_{max}$  ;

{Loop until beyond data or opacity > threshold}

**while**  $InBounds(x)$  **and**  $\alpha(u) \leq 1 - \varepsilon$  **do begin**

$i = Index(m, x)$  ;

{if high level cell contains a one, drop a level}

**if**  $V_m(i)$  **and**  $m > m_{min}$  **then**  $m = m - 1$  ;

**else begin**

{if level zero cell contains a one, render it}

**if**  $V_m(i)$  **then**  $RenderCell(\mathbf{u}, \mathbf{x}, Next(m, \mathbf{x}, \mathbf{u}))$  ;

{Advance to next cell and maybe jump to higher level}

```

while    $Parent(m, Index(m, Next(m, \mathbf{x}, \mathbf{u}))) \neq Parent(m, \mathbf{i})$ 

and  $m < M$ 

    do begin
         $\mathbf{i} = Parent(m, \mathbf{i});$ 
         $m = m + 1;$ 
    end
     $\mathbf{x} = Next(m, \mathbf{x}, \mathbf{u});$ 
end
end
end  $TraceRay_2$ 

procedure  $RenderCell(\mathbf{u}, \mathbf{x}_1, \mathbf{x}_2)$  begin
     $U_1 = \lceil Image(x_1) \rceil;$ 
     $U_2 = \lfloor Image(x_2) \rfloor;$ 
    {Loop through all sample falling within cell}
    for  $U = U_1$  to  $U_2$  do begin
         $x = object(U);$ 
        {If any of eight surrounding voxels have opacity>0,}
        {then resample color and opacity and composite into ray}
        if  $V_0(Index(0, x))$  then begin
             $\hat{C}(U) = sample(\hat{C}, x);$ 
             $\alpha(U) = sample(\alpha, x);$ 
             $\hat{C}(u) = \hat{C}(u) + \hat{C}(U)(1 - \alpha(u));$ 
             $\alpha(u) = \alpha(u) + \alpha(U)(1 - \alpha(u));$ 
        end
    end

```

**end**

**end** *RenderCell*.

The **Index** procedure accepts a level number and the object - space coordinates of a point, and returns the index of the cell that contains it. The **Parent** procedure accepts a level number and cell index, and returns the index of the parent cell. The **Next** procedure accepts a level number and a point on a ray, and computes using a method the coordinates of the point where the ray enters the next cell on the same level. The **RenderCell** procedure composites the contribution made to a ray by the specified interval of volume data. The algorithm terminates when the ray leaves the data as detected by the **InBounds** procedure.

Figure A.4 shows in two dimensions how a typical ray might traverse a hierarchical enumeration. The level zero cell corresponding to each non-empty voxel is denoted by a shaded box. The largest empty cell enclosing each empty voxel is denoted by an unshaded box. The sequence of points computed by the **Next** procedure is denoted by circular dots. In regions containing many non-empty level zero cells, the spacing between these dots is close to the spacing between voxels. There is therefore led to ask the question: why not simply resample the data at these points? It is observed, however, that these points are not evenly spaced along the ray. If the data is resampled at such non - uniformly spaced points, a noise component may be added to the resulting image. To avoid these artifacts, it is superimposed a set of evenly spaced sample locations as shown by the rectangular tick marks in the figure, then limit to resampling the data at these locations.

Assuming that it is rendering a non-empty dataset, most cells on the top levels of the pyramid will contain ones. It is therefore inefficient to begin the traversal there.

For the datasets used, traversal costs were minimized by setting  $m_{\max} = M - 2$  for all values of  $M$ . Assuming an orthographic viewing projection, the cost of advancing a ray from one cell to the next by computing ray - cell intersections is higher than the cost of advancing the ray from one sample location to the next using differencing. It is therefore inefficient to descend to level zero. Instead, it is descended to some higher level, loop through the sample locations falling within that cell, and render those for which  $V_0(\text{Index}(0, x)) = 1$ .

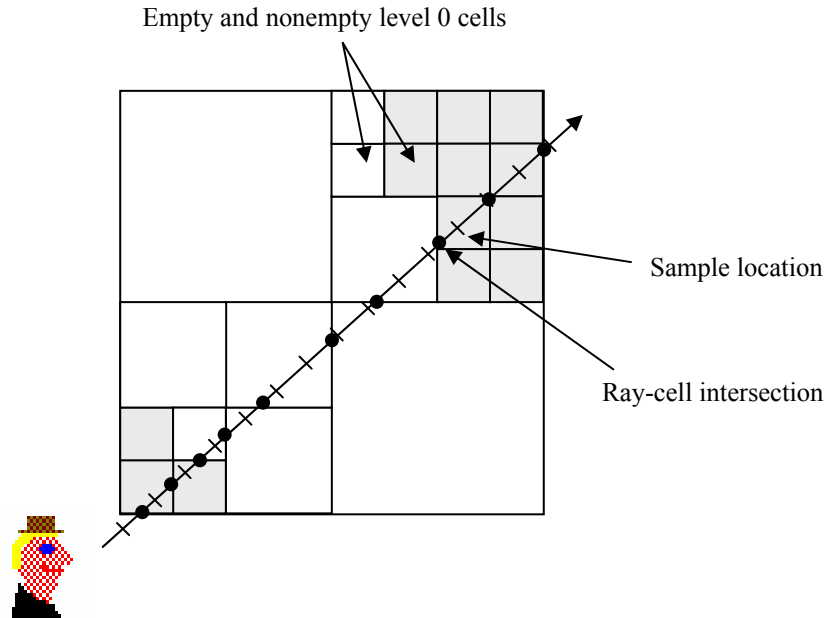


Figure A.4 Ray casting of hierarchical enumeration

The memory required for the optimized algorithm is  $2N^3$  bytes to hold a monochrome colour and opacity for each voxel,  $(8^{M+1} - 1)/7$  bits to hold the pyramid of binary volumes, and  $P^2$  bytes to hold a monochrome output image. Condensed representations of the pyramid such as linear octrees are possible, although the amount of memory saved would be small compared to the size of the colour and opacity arrays, and the cost of accessing a cell would generally be higher.

## APPENDIX B

### MARCHING CUBE ALGORITHM

The classical approach to surface extraction is the Marching Cubes algorithm, proposed by [LC87], with a similar suggestion from [WMW86]. This assumes data is on a structured grid and conceptually it processes each cell independently, one after the other. Each cell in a structured grid is topologically equivalent to a cube, and so the method focuses on the extraction of an isosurface within a unit cube - hence the term “marching cubes”. The method is described as follows. Each vertex of a cube can be either greater than or less than the threshold value  $k$ , giving 256 different scenarios. An estimate  $F(x, y, z)$  can be constructed as a trilinear interpolation of the values at the cube vertices. The intersections of the isosurface  $F(x, y, z) = k$  with the edges of the cube are easily and accurately calculated by inverse linear interpolation. As mentioned earlier, the behaviour of  $F(x, y, z) = k$  inside the cube is non-trivial and is a cubic surface. However a simplistic estimate of  $F$  within the cube can be made by joining intersection points into a set of triangles. Lorensen and Cline argued that for reasons of symmetry and complementarity there are only 15 canonical configurations, and proposed corresponding triangulations of the isosurface (see Figure B.1). For a given configuration (from the set of 256), they provide a look-up table to give the corresponding canonical configuration and hence its triangulation.

This algorithm has been much used over the years since 1987, and has proved very effective in combination with fast triangle rendering hardware as provided on Silicon Graphics workstations, and more recently on PC graphics boards supporting OpenGL.

There are two major aspects of the algorithm which have received attention in recent years:

**Surface representation** The classical marching cubes algorithm has a naive approach to forming the interior representation. It was discovered that holes can appear when two adjacent cells have certain configurations. Much work has gone into making the algorithm more robust. In addition, there has been recent attention to the issue of accuracy, and gaining a more faithful representation of the true isosurface within each cell. Recent efforts are put to make the marching cubes isosurface algorithm give an improved representation of the surface - in terms of robustness, topological correctness and accuracy. The original marching cubes algorithm reduced the 256 possible cases to one of 15 canonical configurations. This enabled a small look up table and efficient coding, but caused inconsistent matching of surfaces between adjacent cells, so that ‘holes’ could appear. For robustness, a remedy is to return to a full 256 case table of triangulations, and this is used for example in the *vtk* implementation of the algorithm [SML96]. The case which has ambiguities will be subdivided into subcases to make sure the topological correctness. Some recent work has attempted to develop beyond the robustness and topological correctness, in order to increase the accuracy of the internal representation of the isosurface within the cell. The trilinear interpolation or progressive mesh refinement can improve the accuracy of the rendering.

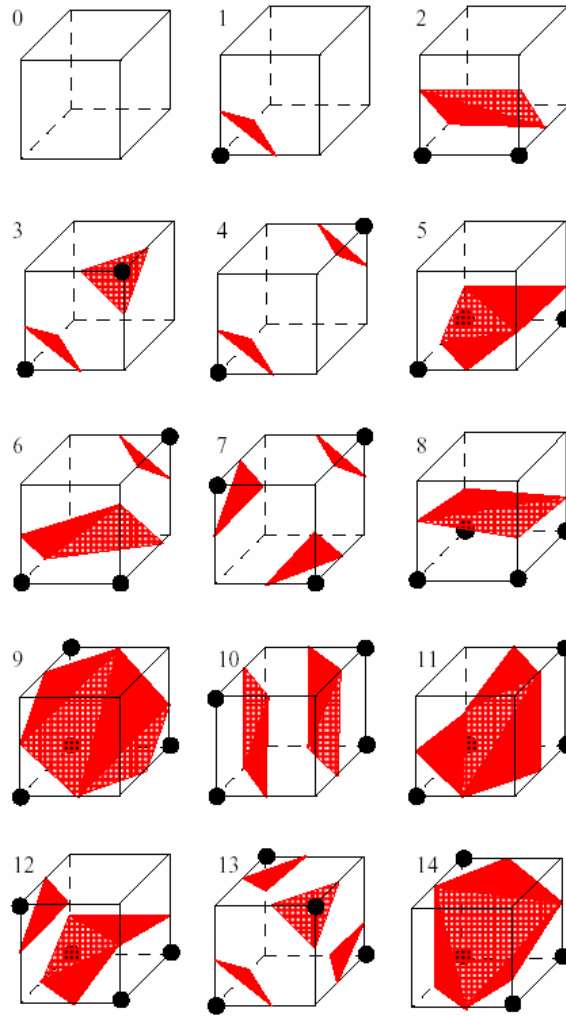


Figure B.1 The 15 marching cubes configurations

**Performance** As computing power and measurement technology has been increased, so has the size of datasets that users wish to analysis. The marching cubes algorithm can be rather slow - both in terms of locating cells which contain segments of the isosurface, and also in terms of rendering the large number of triangles which may result.

Algorithm performance has become increasingly important as our ability to capture or create data has grown at a rate that has outstripped the rate of improvement of computing technology. A number of strategies exist for improving the performance of isosurfacing extraction from large structured and unstructured data sets. One



approach is to presort the data so that the location of the isosurface can be found more quickly. This is especially efficient when a sequence of isosurfaces is to be generated (as is often the case in practice). The initial overhead of presorting is rapidly outweighed by the faster subsequent searching. Another approach is to extract at different levels of resolution, which is called multiresolution approaches. It seeks to improve isosurfacing time by actually reducing the number of cells used to represent the dataset by combining small cells together using a series of rules or conditions. The parallel algorithm is used to speed up isosurface extraction from large data sets as long as the data can be suitably partitioned. Another way of isosurfacing large datasets while only having access to a desktop workstation is to make use of remote resources through distributed processing. Out-of-core algorithms of all types are designed for situations where the amount of data to be processed is simply too large to fit in main memory. These algorithms avoid the time wasted by disk thrashing by employing unique data structures on disk that offer optimal I/O performance and also seek to exploit locality.