3D Human Heart Modeling
(Self-organizing Map Neural Network)

By

Mohd.Tharmezy B. MD.Tajuddin

A thesis presented to the Faculty of Computer Science and Information Technology University of Malaya in fulfillment the requirement for the Degree

Bachelor of Computer Science
University of Malaya
2003

To

My Family

Sadness Memory on 17 February 2003

Grandmother Yam Bt. Lebai.Hamid

And

To

My loving one's

Ena

ACKNOWLEDGEMENT

Begin with remember the god name's that most gracious and the most merciful

On the beginning I would like to express my grateful for the encouragement and teaching to my supervisor Mr. Woo Chaw Seng with his I know in detail the requirement for this project, you're like a lamp in the dark. And also in my mind I would like to express my special thanks to my moderator Miss. Mangalam Sankupelley with her encouragement. Special thank also to the most helpful person Assoc.Prof.Dr.Ir.Selvanathan Narainasamy for your material in this project. And also to my friends Azrizal b. Ismail, Mohd. Safwan B. Mohd.Makki, special thanks for all of you for helpful and having a patience in painful.

The morale support by the Department of Artificial Intelligent, Faculty of Science Computer and Information Technology, University Malaya must be recognized here, as the entire project would not have been accomplished and succeeded without requirements.

Lastly, I would also like to acknowledge the assistance of fellow course-mates and other people who have either directly or indirectly participated in this project. This project can only become a reality with the effort, assistance and cooperation of many individuals whom I sincerely appreciate.

"A lots of Thank for all of you"

Kohonen has developed an algorithm with self-organizing properties for a network of adaptive elements. These elements receives signal from an event space and the signal representations are automatically mapped onto a set of output responses in such way that these responses acquire the same topological order as that of the primary events.

In human heart modeling process, segmentations is the most important thing before create the 3 dimensional modeling of human heart. Heart images from the source can be processed to become the signal for the Self-Organizing Maps (SOM) network and the output neuron that have adapted to the images, present interesting features such as contour-extractions and edge detections.

In this work the Neural Network applications specific to the Self-Organizing network used to segmenting the human heart images to produced the 3 dimensional modeling of human heart.

TABLE OF CONTENTS

CHAPTER 4: SYSTEM ANALYSIS AND DESIGN

CHAPTER 5: SYSTEM IMPLEMENTATION

CHAPTER 6: SYSTEM TESTING

LIST OF FIGURE

LIST OF TABLE

1. AN INTRODUCTION

"HUMAN HEART MODELING"

Modeling an organ in the human body is one of the problem before 21^{st} century but now days it not going to be a problem again. In order to model a biological organ with accompany by an artificial intelligence technique, neural network is one of the most available technique choose to involve in this project. This chapter introduced the definition of this title.

1.0 An Introduction to the Human Heart

Human Heart Modeling is a concept to search information about human heart which is one of the advantages is to gives the heart pumping by including time domain. The heart is essentially a muscle (a little larger than the fist). Like any other muscle in the human body, it contracts and expands. The walls of the heart are made up of three layers, while the cavity is divided into four parts. There are two upper chambers, called the right and left *atria*, and two lower chambers, called the right and left *ventricles*. The Right Atrium, receives blood from the upper and lower body through the *superior vena cava* and the *inferior vena cava*, respectively, and from the heart muscle itself through the *coronary sinus*. The right atrium is the larger of the two atria, having very thin walls. The right atrium opens into the right ventricle through the *right atrioventicular valve* (tricuspid).

The left atrium smaller than the right atrium, but has thicker walls. The valve between the left atrium and the left ventricle, the *left atrioventicular valve* (bicuspid), is smaller than the tricuspid. It opens into the left ventricle and again is a one-way valve. The *Aorta*, the largest artery in the body, which originates from the left ventricle.

From the explanations know that the human heart have several partitions. For the purpose of modeling each partition must be extracted and initialized first before performing other processes. Something that must be thinks is about what is the technique for the extraction this partition. There are several techniques to perform this work in image processing field or in the artificial intelligence field. Figure 1.1 and figure 1.2 is the image view about human heart and what have already been discussed in previous paragraph to give an extra understanding to the heart partitions.

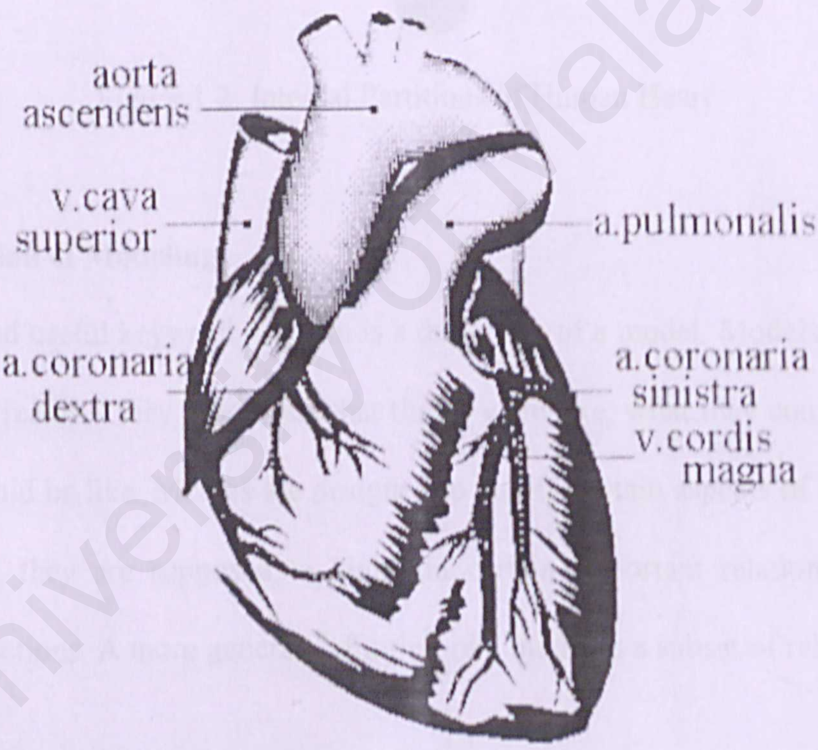


Figure 1.1: Human Heart Partitions

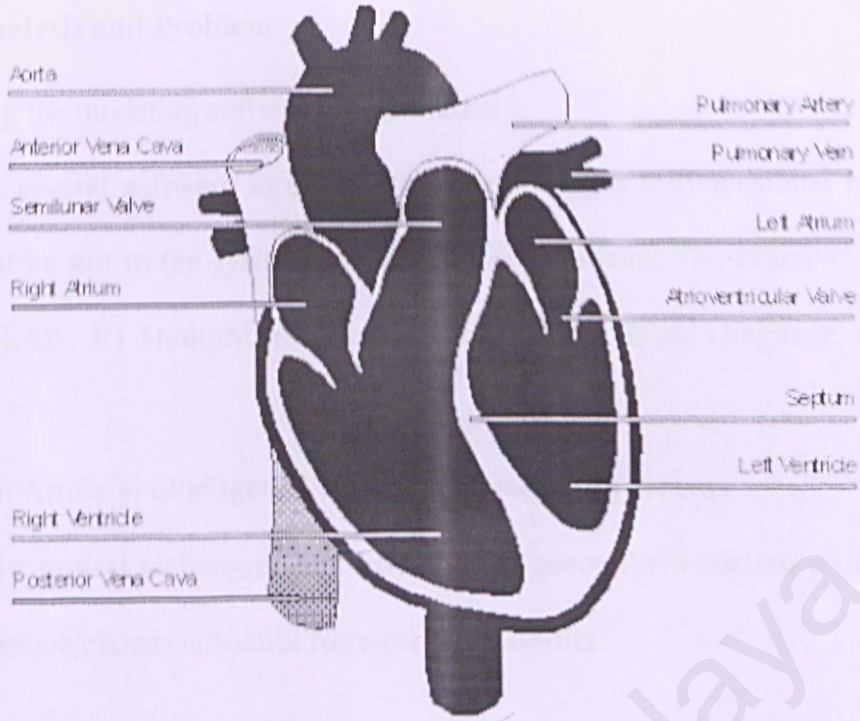


Figure 1.2: Internal Partitions of Human Heart

1.1 A Definition of Modeling

The second useful keyword to begin is a definition of a model. Models represent an abstraction of reality - they represent what things were like, what they could be like, or what they should be like. Models are designed to clarify certain aspects of a problem or problem area; they are supposed to highlight certain important relationships and certain key interactions. A more general definition of a model is a subset of relationships in the real world.

In this title contact, a model referred to the heart model that abstraction from reality of human heart. This model represents what the heart was like, what the heart could be like, or what the heart should be like.

1.2 Title Analysis and Problem

1) Choosing the modeling software in the market

There is several software in market that performing a 4 dimensional modeling and must be suit to the system development requirement. The example software is MATLAB, 3D StudioMax, Virtual Reality Manipulating Language, OpenGL and etc.

2) Using an Artificial Intelligence Technique in modeling process.

There are several techniques in artificial intelligence for modeling process. But the technique choose is Neural Network for classifier.

1.3 The Objectives

When considering to model the human heart the most important thing that should be thinks is the objectives of modeling, this modeling job is quick same as the others current modeling, that the main objective to achieve the main goal, presenting the acquired modeling with the good and exact shape. Nevertheless trying to model the human heart is come from many purposes other than the main goal.

The objectives are:

- **Using the Neural Network in the modeling process.**
- **Viewing heart in saggital, axial and coronal dimension.**
- **Design 3 dimensional human heart and included time element**
- **Present the human heart pumping in computer graphic**

- **Extract certain component in the human heart.**
- **Segment the partition of human heart.**

In order to follow these objectives, the useful and available application must be defined and determined.

1.4 Project Scope

The system can use by any person to learn about human heart. But in this subsection introduced the little bit about the use of this system.

- **Surgery doctor**
    - This system can model a patient's heart before physician doing the operation. From this performance the physician can extract the certain partitions in human heart (e.g: tumor, region of interest) before performing the operation to the patients. This system also can help to give the information about recognition the right partitions in the heart.
- **Medical student (university)**
    - For medical student purpose, this system can give the student view in coronal, saggital, axial and transverse and also can slice the heart to extract the certain partitions. It can give the information to the medical student in knowing the structure of the human heart in the study beside use the real organ from the dead person. Student also can use this system to integrate to other software in process of virtual surgery.

- **Student (school)**
  - In the purpose of school student, it about use of this system for introducing the human heart partitions. The school student also can view the heart in the coronal, saggital, axial and transverse. From this performance the school student can know the shape and contents of the human heart. Also the performances in showing the human heart in slices and segmentations can give the school student know more about the partitions in the heart and layer of the heart. The benefit all of this is to make school student understanding and know the correlation of human structure to their biological function that they have study in the school.

1.5 Project Schedule

To make sure the project that have been develop reach their objectives in the fixed time, a schedule have been constructed. The schedule is to ensure that the development application running in the smooth way. It has six main processes in the project schedule:

i. Project Title and Plan Suggestion

This level is about analysis the title and find out the attribute that will construct the plan of project. It is included project objectives, scope and the list for the project performance planning.

ii. Literature Review

In this step the technology and software consider involve in this project will be elaborate. Reference is use through the internet, library, discuss with supervisor and friends, and through the previous student research that correlation to this project.

iii. Requirement Analysis

The step introduced the information about system requirement. It included the function requirement that will be used to follow the objective of this project and other non-function requirement that needed by the system. Through this information, methodology that will use to construct the system will determine.

iv. System Design

This stage is all about physical design to the system. From this stage the user interface design will be create and also, the input and output design for the system.

v. Encoding

The code that will use in the system will recognize in this level.

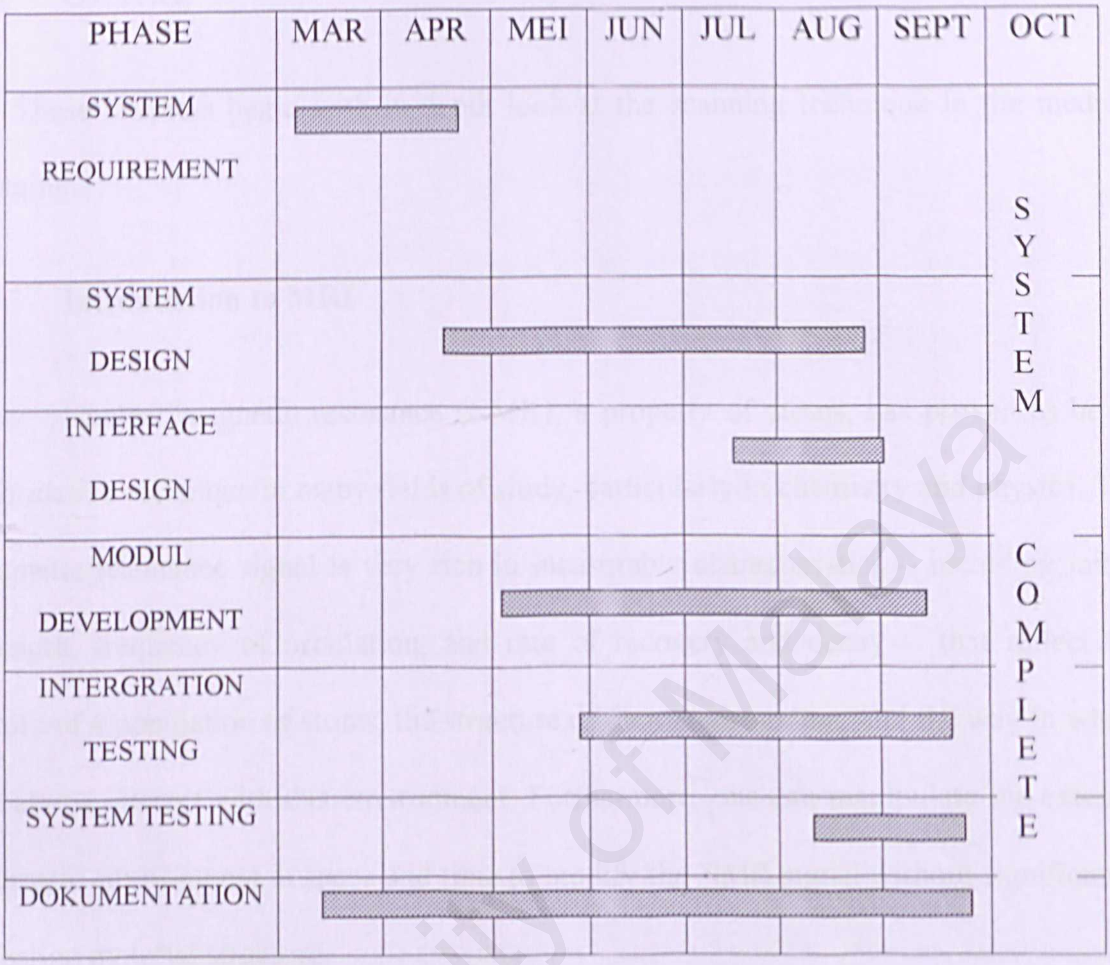
vi. Testing

For the formal in the testing level must follow the step in the each component in the system like module level, and then integration one module with another module and testing to all part of the system For this project, testing procedure is only for the system operation and accuracy because this is a small system and don't has too much module to be tested for the purpose of integrated.

vii. Documentation

The user manual will be constructed in this step, it is important if something fault operation occur or for the future modifying.

Table 1.1: Project Schedule



2. LITERATURE REVIEW

2.0 Overview

These chapters begin with in-depth look at the scanning technique in the medical imaging.

2.1 Introduction to MRI

Nuclear magnetic resonance (NMR), a property of atoms, has proven to be an informative technique in many fields of study, particularly in chemistry and physics. The magnetic resonance signal is very rich in measurable characteristics -- including initial strength, frequency of oscillation, and rate of recovery and decay -- that reflect the nature of a population of atoms, the structure of their environment, and the way in which the atoms interact with this environment. Furthermore, one can manipulate the external magnetic environment in space and time to modify the NMR signal without significantly affecting material structure.

Relatively recently, magnetic resonance was extended to the *in vivo* study of human anatomy. This was made possible by new, practical methods for exciting signal from limited volumes [3], and for generating spatial maps of this signal [4]. Relying primarily on the differential decay and recovery characteristics of the proton NMR signal (generally termed relaxation behavior), this technology can generate images with high contrast among various soft tissues and organs. As a result, magnetic resonance imaging (MRI) has become the modality of choice in many diagnostic studies of the head, spine, and joints. With ongoing developments to improve the image quality, acquisition speed and quantitative accuracy of related measures of local signal characteristics, the range of clinical applications for MRI continues to expand rapidly.

2.2 The Discovery of MRI

Bloch and Purcell, both of whom were awarded the Nobel Prize in 1952, discovered the magnetic resonance phenomenon independently in 1946. In the period between 1950 and 1970, NMR was developed and used for chemical and physical molecular analysis. In 1972, the X-ray based computerized tomography was introduced. This date is important to the MRI timeline because it showed hospitals were willing to spend a large amount of money for medical imaging hardware. In 1973, Lauterbur demonstrated imaging using NMR and the back projection technique used in CT. In 1975, Ernst proposed magnetic resonance imaging using phase and frequency encoding, the current MRI technique. Edelstein and co-workers demonstrated imaging of the body using this technique in 1980. A single image could be acquired in approximately five minutes. By 1986, the imaging time was reduced to about five seconds, without sacrificing to much on image quality.

In 1988, Dumoulin perfected MRI angiography, which allowed imaging of flowing blood without the use of contrast agents. In 1989, echo-planner imaging was introduced which permitted image acquisition a high-speed rate. Many clinicians thought this technique would be applied in dynamic MRI of joints; instead it has been used for imaging the regions of the brain responsible for thought and motor control. In 1991, Richard Ernst was rewarded for his achievements in pulsed NMR and MRI with the Nobel Prize in Chemistry. In 1994, researchers at the State University of New York at Stony Brook and Princeton University demonstrated the imaging of hyperpolarized 129Xe gas for respiration studies.

For more than a decade now, physicians have used MRI to produce accurate representations of internal and external anatomical structures. With the help of these images, abnormalities such as tumors can be easily detected. Today, MRI data are being used not only by physicians, but also medical device manufacturers. Through the use of 3D models derived from MRI data, medical devices such as implants or prostheses can be designed to accurately conform to the shape of the human body.

2.2.1 Image Analysis

The nature of the data produced with MRI is, in many ways, well suited to subsequent image analysis. Specifically, (i) many different organs can be distinguished with the flexible soft tissue contrast; (ii) resolution can be tailored to specific applications and noise is generally well behaved; (iii) signal can be resolved in the 3 spatial dimensions to sub-millimeter levels; (iv) one can resolve temporal signal changes at the sub-second level; finally, (v) one can obtain multiple measurements of the same volume element with different contrast characteristics. The vast volume of data alone often demands computer image analysis for interpretation and display.

Areas of research in image analysis include methods for image segmentation [12,5,6] and registration [6,7]. Image segmentation facilitates (i) volume estimation for characterizing disease (ejection fraction in the heart, tumor volumes), (ii) isolation of a tissue of interest from a three-dimensional data set, notably for the display of vascular anatomy, and (iii) multiple image registration through the identification of common landmarks in the images. Registration facilitates (i) temporal signal analysis of a specified volume within the same study, (ii) comparative studies of the same patient over multiple studies perhaps with data of different contrast weightings or even data from different imaging modalities, and (iii) comparative studies of anatomy across multiple individuals.

The task of image analysis is simplified greatly when attention is paid to the nature of the basis images. Image sequence parameters can be selected to maximize contrast between tissues, facilitating segmentation. Acquisition can be synchronized to motion or priority can be given to imaging speed to reduce the demands on registration algorithms. Similarly, registration of multiple images with different contrasts is aided by acquiring lines of **k** space from the different images in a time-interleaved manner. For segmentation, image resolution should be selected to minimize the difficulties associated with partial voluming, where multiple tissues are present in the same voxel, based on the sizes of the structures of interest. Finally, the additive, Gaussian nature of the noise can be exploited to optimize classification strategies in segmentation; as a caveat, the noise takes on a Rician distribution in magnitude images most often used for segmentation as a result of the magnitude operation.

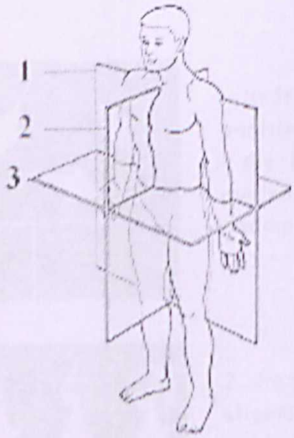
MR uses magnetic energy and radio waves to create cross-sectional images or "slices" of the human body. The main component of most MR systems is a large tube shaped or cylindrical magnet. Also now available are MR systems with a C-shaped magnet or other type of open design. The strength of the MR systems magnetic field is measured in metric units called "Tesla". Most of the cylindrical magnets have a strength between 0.5 and 1.5 Tesla and most of the Open or C-shaped magnets have a magnetic strength between 0.01 and 0.35 Tesla. A 1.5 Tesla MR system has a magnetic field 30,000 times stronger than the pull of gravity on the earth's surface. The patient aperture or bore of the cylindrical magnet is usually between 55 cm and 65 cm wide (21.6" to 25.6") and they have a total end-to-end length of 160 cm to 260 cm (5' 3" to 8' 6"). Approximately 3% of MR patients suffer from claustrophobia and may not tolerate an MR exam in a traditional cylindrical MR system. These patients may now have the option of having an MR study with an Open MR system where typically many sides of the system are open and claustrophobic anxiety is lessened.

To begin the MR examination, the patient is positioned on a special table and positioned inside the MR system opening where the magnetic field is created by the magnet. Each total MR examination typically is comprised of a series of 2 to 6 sequences, with each sequence lasting between 2 and 15 minutes. An "MR sequence" is an acquisition of data that yields a specific image orientation and a specific type of image appearance or "contrast." Thus a typical exam can last for a total of ten minutes to an hour, depending on the type of exam being run and the MR system being used.

During the examination, a radio signal is turned on and off, and subsequently the energy, which is absorbed by different atoms in the body, is echoed or reflected back out of the body. These echoes are continuously measured by the MR scanner and a digital computer reconstructs these echoes into images of the body. The tapping heard during the MR exam is created when "gradient coils" are switched on and off to measure the MR signal reflecting back out of the patient's body. A benefit of M RI is that it can easily acquire direct views of the body in almost any orientation, while CT scanners typically acquire images perpendicular to the long body axis.

The following diagram shows the three main planes of acquisition used in MR imaging.

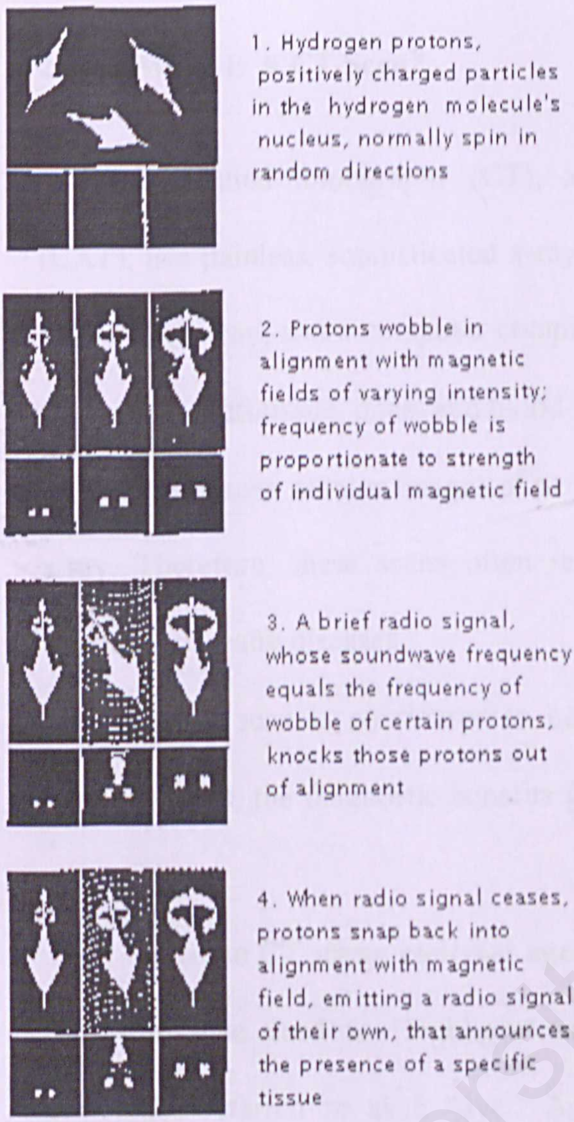
Medical images taken of the human body are acquired or displayed in three main orientations:



1. Coronal orientation: in a cross section (plane), for example, across the shoulders, dividing the body into front and back halves
2. Sagittal orientation: in a cross section (plane), for example, down the middle, dividing the body into left and right halves
3. Axial orientation: in a cross section (plane), perpendicular to the long axis of the body, dividing the body into upper and lower halves

Figure 2.0: 3 main planes of acquisition used in MR imaging

Views which are combinations of the above three orientations (called oblique views) can also be directly acquired with the MR system. These oblique views are especially important for orthopedic and sports medicine applications when tendons and ligaments that run at oblique angles need to be clearly imaged.



Like X ray, MRI is based on a discovery in the physic lab: when the nuclei of hydrogen atoms--single protons, all spinning randomly--are caught suddenly in a strong magnetic field, they tend to line up like so many compass needles. If the protons are then hit with a short, precisely tuned burst of radio waves, they will momentarily flip around. Then, in the process of returning to their original orientation, they resound with a brief radio signal of their own. The intensity of this emission reflects the number of protons in a particular "slice" of matter.

Figure 2.1: Explain how does MRI work

2.3 CT scanner: An Introduction [16]

2.3.1 What Is A CT Scan?

Computed tomography (CT), also known as Computed Axial Tomography (CAT), is a painless, sophisticated x-ray procedure. Multiple images are taken during a CT or CAT scan, and a computer compiles them into complete, cross-sectional pictures ("slices") of soft tissue, bone, and blood vessels.

A CT scan obtains images of parts of the body that cannot be seen on a standard x-ray. Therefore, these scans often result in earlier diagnosis and more successful treatment of many diseases.

A CT scan is considered to be a safe examination. While CT imaging does involve x-rays, the diagnostic benefits generally outweigh the risks of x-ray (radiation) exposure.

In some CT scans, **contrast agents** or sedatives may be used. A **contrast agent** is a substance used to "highlight" an organ or tissue during examination and is sometimes referred to as a "dye." Again, the benefits of early, accurate diagnosis generally outweigh any risks associated with the potential side effects of these agents.

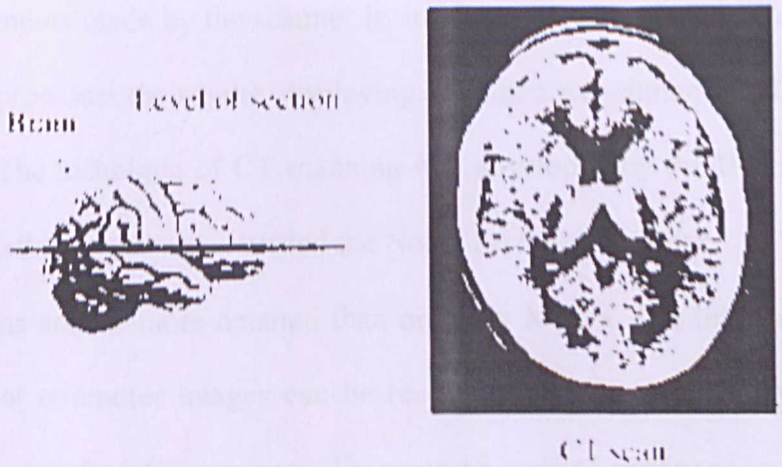


Figure 2.2: Example CT scan image

CT scanning was developed during the mid-1970s. The original systems were dedicated to head imaging and were very slow-it taken hours to acquire the images for each individual slice. The newest scanners collect as many as four slices of data in less than 350 microseconds.

This great improvement in the speed of CT scanning has been accompanied by increased patient comfort and higher resolution images. And, as scan times have become faster, the time of x-ray exposure has decreased, providing better image quality at lower x-ray doses.

A CT (computerized tomography) scanner is a special kind of X-ray machine. Instead of sending out a single X-ray through your body as with ordinary X-rays, several beams are sent simultaneously from different angle. The X-rays from the beams are detected after they have passed through the body and their strength is measured. Beams that have passed through less dense tissue such as the lungs will be stronger, whereas beams that have passed through denser tissue such as bone will be weaker. A computer can use this information to work out the relative density of the tissues examined. Each set of measurements made by the scanner is, in effect, a cross-section through the body. The computer processes the results, displaying them as a two-dimensional picture shown on a monitor. The technique of CT scanning was developed by the British inventor Sir Geoffrey Hounsfield, who was awarded the Nobel Prize for his work.

CT scans are far more detailed than ordinary X-rays. The information from the two-dimensional computer images can be reconstructed to produce three-dimensional images by some modern CT scanners. They can be used to produce virtual images that show what a surgeon would see during an operation. CT scans have already allowed doctors to inspect the inside of the body without having to operate or perform unpleasant examinations. CT scanning has also proven invaluable in pinpointing tumors and planning treatment with radiotherapy. The CT scanner was originally designed to take pictures of the brain. Now it is much more advanced and is used for taking pictures of virtually any part of the body. The scanner is particularly good at testing for bleeding in the brain, for aneurysms (when the wall of an artery swells up), brain tumors and brain damage. It can also find tumors and abscesses throughout the body and is used to assess types of lung disease. In addition, the CT scanner is used to look at internal injuries such as a torn kidney, spleen or liver; or bony injury, particularly in the spine. CT scanning can also be used to guide biopsies and therapeutic pain procedures.

If the patient is receiving an abdomen scan, for example, they will be asked not to eat for six hours before the test. They will be given a drink containing gastrografin, an aniseed flavored X-ray dye, 45 minutes before the procedure. This makes the intestines easier to see on the pictures. Sometimes a liquid X-ray dye is injected into the veins during the test. This also makes it easier to see the organs, blood vessels or, for example, a tumor. The injection might be a little uncomfortable, and some people also experience a feeling of warmth in their arm. The scanner looks like a large doughnut. During the scan the patient lies on a bed, with the body part under examination placed in the round tunnel or opening of the scanner. The bed then moves slowly backwards and forwards to allow the scanner to take pictures of the body, although it does not touch the patient. The length of the test depends on the number of pictures and the different angles taken. A CT scan can give the doctor a much clearer picture of the inside of the body than an ordinary X-ray. For example, different types of tissue such as bone, muscle and fatty tissue are easy to see on a CT scan. When looking at the abdomen, the scan shows various organs such as the pancreas, spleen and liver. When it is necessary to look at the brain, the areas containing liquid - the ventricles - are also clearly defined. Very small shadows on the lungs can also be detected using CT and there are now studies looking into using it as a screening test for lung cancer.

2.4 Ultrasonography (US)

Ultrasonography is based on sound reflections that producing on continuous or real-time images, which are viewed on a monitor. Sound waves a very high frequency are emitted from transducer that is in contact with the surface of the body [13]. The sound waves are reflected from the body's interior based into the same transducer, which then reconstructs the return waves information into images.

2.5 Positron Emission Tomography (PET)

Positron emission tomography based on the spatial distribution of radioactivity from radionuclides injected into the body [14]. The images produced by sampling the gamma emissions from the radionuclides within the body. Positron emission tomography detects metabolic activities of a tissue of interest.

2.6 Single Photon Emission ComputeTomography (SPECT)

Single photon Emission Compute Tomography also depends on radioactive decay of injected markers. It also used to detect metabolic activities.

2.7 Other

1) Electrical Source Imaging (ESI)

2) Electrical Impedance Tomography (EIT)

3) Magnetic Source Imaging (MSI)

4) Medical Optical Imaging (MOI)

2.8 Still in Development [15]

1) Microwave Imaging

2) Infrared Imaging

3) Electron Spin

4) Resonance

5) Interferomentary

2.9 Project Related Areas

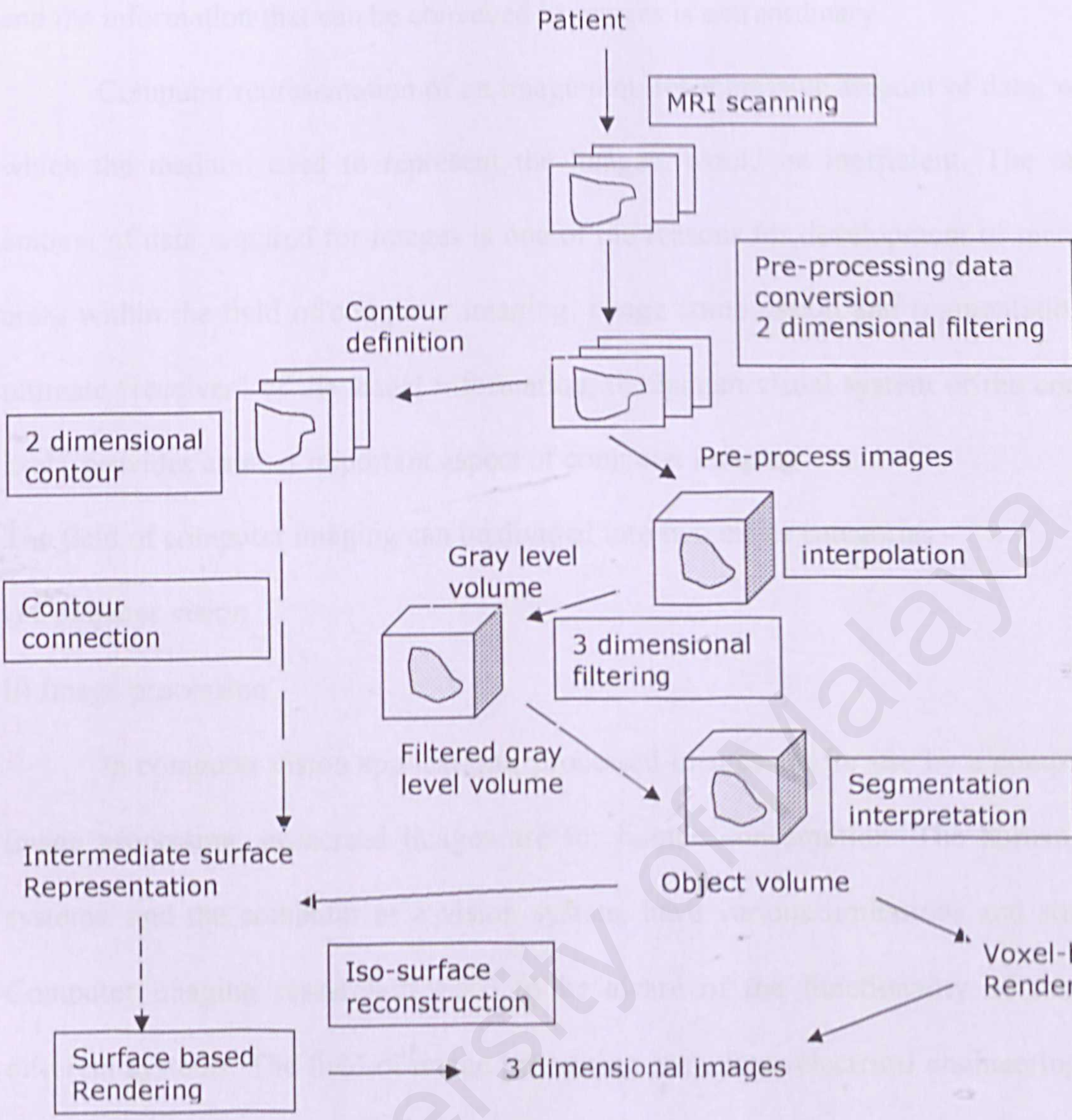


Figure 2.3: Image processing pipeline

2.9.0 Computer Imaging

Visual information, transmitted in the form of digital images, is becoming a major method of communication in the modern age. Computer imaging can be defined as the acquisition and processing of visual information by the computer. The importance of computer imaging is derived from the fact that our primary sense is our visual sense, and the information that can be conveyed in images is extraordinary.

Computer representation of an image requires a massive amount of data, without which the medium used to represent the images, would be inefficient. The massive amount of data required for images is one of the reasons for development of many sub-areas within the field of computer imaging, image compression and segmentation. The ultimate "receiver" of the visual information, the human visual system or the computer itself, provides another important aspect of computer imaging.

The field of computer imaging can be divided into two major categories:-

i) Computer vision

ii) Image processing

In computer vision applications, processed images are for use by a computer. In image processing, processed images are for human consumption. The human visual systems, and the computer as a vision system, have various limitations and strengths. Computer imaging researchers need to be aware of the functionality of these two different systems. The field of image processing grew from electrical engineering, as an extension of signal processing. Computer vision developed from the area of computer science. Computer imaging blends techniques of computer vision and image processing.

2.9.1 Computer Graphics

Computer graphics is a specialized field within the realm of computer science, that refers to the reproduction of visual data through the use of the computer. This includes the creation of computer images for display or print, and the process of generating or manipulating any images for output. Computer graphics provides methods to synthesize images from numerical descriptions. These techniques were originally developed for realistic displays of human-defined objects, such as models from computer aided design (CAD). Objects in 3D space are usually represented by infinitely thin surface patches such as triangles or higher order curves. Contributions or computer graphics to 3D imaging include data structures, projection techniques and shadowing models.

A major use of computer graphics is in design processes. Computer aided design (CAD) methods are used in designing. When object designs are complete, or nearly complete, realistic lighting models and surface rendering are applied to produce displays that will show the appearance of the final product.

Image processing and computer graphics are typically combined in medical applications. This technique is used to model and studying physical functions, design artificial limbs and plan, and practise surgery. 2D cross section of the body is obtained using imaging techniques. These slices are then viewed and manipulated using graphics methods to simulate actual surgical procedures and to try out different surgical cuts [8].

2.9.2 Volume visualization techniques

1) Direct surface rendering

Surface rendering is an image visualisation process that displays the exterior boundary of an image rather than slices through the volume. Volume rendering on the other hand preserves all the information about the slices so that we can see not only what the surface look like, but also what's inside. Surface rendering accounts only for the surfaces themselves and the inside is not visible. Volume rendering incorporates more information, but is more difficult to achieve (due to hardware and software requirements). In some cases, it is desirable to display both the internal image information, as well as the surface [15].

2) Direct surface rendering and shadows

This is an image visualisation process that displays the exterior boundary of an image and incorporates the shadows of the object.

3) Voxelisation

Voxelisation is the process of producing 3D data from an object description, such as polygonal meshes, so that each voxel has a value. Voxel (volume element)is the name given to a value in 3D space. The data can then be rendered using a volume visualisation technique to produce an image of the original object. Voxelisation is the process of converting an object of one description into volume data. Appropriate volume visualisation techniques can be applied to such data [10][11].

4) Volume rendering

Volume rendering is the process of obtaining images from three dimensional volume data by treating the data as cloudy material. Each value in the data has a colour and opacity assigned to it during a classification stage. Rays are cast for each pixel into the volume, and the colour and opacity are sampled at evenly distributed points along the ray. These samples are composited using standard techniques to produce the accumulative colour and opacity reaching the pixel. Image produced by this method usually have several differently coloured semi-transparent skin overlaid on opaque bone for an image of a CT scan. Volume rendering methods include standard volume rendering method, adaptive termination, no shading, template method, bilinear method, x-ray method, maximum value method, maximum value method with depth cue and Sabella's method.

One of the most popular algorithms for volume rendering involves generating colour and opacity values for each element in the data set and combining the values along the path of an imaginary ray fired through the data set. These values of colour and opacity are then used to produce the pixel value for the final image. There are three main steps involve in volume rendering [12][13]:-

i) Classification and shading

ii) Ray casting

iii) Splitting

Classification and shading

The first step of the algorithm involves obtaining a value for the opacity, classification, and the colour. There are two possible methods for this stage. In one isosurfaces are calculated by mapping an opacity to data values with represent boundaries. To avoid introducing artefacts, the data with values near that of a boundary value is assigned an opacity near that of the boundary opacity. In the other method, each material is assigned an opacity, and the linear mapping is used to convert the voxel values into opacities. This ensures that any thin, wispy regions will still appear in the final image. The overall effect of this classification method is to produce a superimposition of multiple semi-transparent surfaces [12][13].

Ray Casting

Once an opacity and colour has been obtained for all the volume elements in the data set, these must be combined to produce a final image. This is usually achieved by calculating the contribution of each volume element in the path of a ray 'fired' into the volume in the direction of the final image. At sample points along the ray, the colour and opacity are interpolated from the surrounding values for that points on the ray. The values along the ray are then combined and the final image can be displayed. These volume rendering techniques are all structure independent. This makes them more effective than surface rendering for complex scenes as they do not make any assumptions about the presence of features such as surfaces in the data.

Splitting

Another variation on volume rendering is known as splitting. In this case, the voxels in the datasets are traversed from front to back with one slice of voxels, and their individual contribution to the final image calculated using a filter, known as a reconstruction kernel. The term splitting comes from the analogy with throwing snowballs at a glass plate. The contribution of an individual snowball is higher nearer its centre and gradually tails of towards the edges. This is reflected in splitting in the filter used to calculate the contribution of the voxels to the final images.

Splitting has similar advantages and disadvantages to the standard volume rendering techniques. However it does have an additional advantage, in that it is possible for the user to see the image growing one slice at a time, rather than one pixel at a time as with the ray casting technique.

Develop a parallel graphical algorithm.

Visualisation is the process of creating images from data in order to assist comprehension. The generation of images used for visualisation relies an standard graphical techniques such as ray-tracing, 2-buffer and alpha-buffer rendering as well as algorithms such as volume rendering and isosurface generation which are specifically intended for dealing with 3D data .

Animations

All the methods of displaying volume data which are described above, have limitations in that they require the volume data to be reduced to 2 dimensional data for display. Another way in which the 3 dimensional natures of the data can be visualised is by making use an animation.

Animation involves moving the image plane in a defined path round the data set. This allows the data to be viewed from slightly different direction each time. If the frames, or individual images of the picture, can be viewed quickly enough then the effect of motion parallax will give the image the illusion of 2 dimensions. However, in order to make use of motion parallax with real time visualisation, images must be produced at the rate of at least 10 per seconds, the real time visualisation of volume data sets is still an active area of research .

In the meantime, an animation is still a useful tool for understanding volume data by storing the images and looking at the animation once all the images have been produced. An additional advantage to using animation to view volume data is that the individual images do not have to be as detailed as the effect of motion parallax can compensate for some loss of detail in the image .

5) Pre-processing

Data Conversions

The first step in the image processing pipeline after image acquisition is usually data conversion. Besides a change of data format, this also involves measures for data reduction to save storage space and processing time. 3D imaging usually deals with huge amounts of data. A typical CT study of 80 cross-sections with 512 x 512 pixels each takes 40 megabytes of memory. It an interpolation step is performed, the size of the data may be multiplied common techniques of data reduction are [1]:-

- Cutting

  -A region of interest is chosen , other parts of the image are cut away.

- Reduced spatial resolution

  -The matrix size is reduced, for example, by averaging from 512 x 512 pixels to 256 x 256 pixels.

- Reduced intensity resolution

  -For example, a 16 bit to 8 bit reduction

  -An intensity window is chosen which represents most of the contrast in the images. This is usually done with a histogram, which shows the distribution of the grey values.

The reduced spatial and intensity resolution methods will generally cause some loss of information, therefore these methods must be used with care [1].

Filtering

Another important aspect in the pre-processing is image filtering. Filtering is a general term for all kinds of image processing routines which are used to smoothen or enhance the information contents of an image. Atypical example is to improve the signal-to-noise ratio, especially in MRI images. Commonly used noise filters are average, median and Gaussian. Unfortunately these filters smooth out small details as well. Better results can be obtained with anisotropic diffusion methods. Other filter types are applied to emphasize special aspects of an image, for example, to enhance edges.

Filters can be designed to work on 1D lines, 2D images, 3D volumes, or higher dimensional data. A 1D filter can also be applied to the individual rows or columns of a 2D image. However, results are better if an image is filtered with a 2D filter and a volume with a 3D filter.

Data structures

Some of the important data structures used for volume data include:-

- Binary voxel-model

  -Voxels values are stored as either 1 (object) or 0 (no object). This simple model is not commonly used anymore. In order to reduce storage requirements, a data structure called octree, a hierarchical tree structure, in which binary volumes are recursively subdivided into homogenous sub-volumes, can be used.

- Grey level voxel-model

  -Each voxel holds intensity information

- Generalized voxel-model

  -Each voxel holds a grey value, and further information such as object membership label, material percentages or data from other sources. This data structure is used for many advanced applications.

6) Object definitions

A grey level volume usually represents a large number of different regions. To display a particular region of interest, which part of the volume or which voxels, constitute which region of the image, must be identified. This information is also needed for morphometric measurements of distances, angles, and volumes. Object definition involves the establishing of between voxels and meaningful anatomical terms. This task is performed through segmentation and interpretation.

7) Segmentations

This process involves partitioning the grey level volume into different regions which are homogenous with respect to some formal criteria while corresponding to real anatomical objects. An example of segmentation is specifying an intensity range with lower and upper threshold values. A voxel will belong to the selected class only if its intensity value falls within the specified range [1].

Thresholding is a method commonly used to select bone or soft tissue in CT. It is often performed during the rendering process so that no explicit segmentation step is required. A drawback of this sort of method, is that a small objects with cover only a small fraction of a voxel and cases of uncertainty, cannot be handled efficiently. To model the cases, fuzzy segmentation techniques have been developed, whereby a set of probabilities is assigned to every voxel, indicating the evidence for different materials [1].

8) Interpretation

Interpretation is a step in which various regions of an image are identified and labelled with meaningful terms such as "white matter" or "ventricle". This process can be performed interactively or with an automatic system [1].

9) Obtaining volumetric data

3D image are made up of voxels or "volume elements". The voxels is a three-dimensional analog of the pixel. A voxel has a position that uniquely identifies it's location in space. This position is specified as an (x,y,z) coordinate triplet and represents a value [10][11].

Three dimensional volumes are broken into smaller square elements, it can be divided into discrete, cubic, unit elements for storage in memory. Each voxel is a cubic base unit of space containing a value, usually representing colour. Once broken down, or voxelized and stored in a suitable data structure, a 3D object can be displayed with great detail in linear time, independent of complexity of the object, and dependent instead on the number of voxels used to represent it [10][11].

A volume is simply a 3D array of data. The type of data can be almost anything. Required information can also be precomputed and stored in a voxel. Typically a voxel is made up of three fields as shown in Figure 2.7 below.

Scalar	Normal	Gradient

Figure 2.4: Fields in a voxel

The data stored in the voxel includes an 8-bit scalar value and two precomputed fields. The scalar value yields volume information, or information needed to create the volume file, and is used for classification. The first precomputed field is a surface normal vector encoded in a 16-bit field. It gives surface information used for doing shading routines. The second precomputed is the gradient-magnitude of the scalar value. This field can be used for detecting surface boundaries during classification.

2.10 MATLAB

2.10.0 Overview

MATLAB is the one of the manipulating software that has the toolbox to design the 3 dimensional object included time domain. It can convert the MRI data format to the MATLAB data format. From this converted permission, the MRI data format can be manipulating on using MATLAB. Let review the explanations about the amazing of the MATLAB, but on this review the head will be an example.

2.10.1 Image Processing Toolbox

Displaying Multiframe Images

A multiframe image is an image file that contains more than one image. The MATLAB-supported formats that enable the reading and writing of multiframe images are HDF and TIFF. Once read into MATLAB, the image frames of a multiframe image are always handled in the fourth dimension. Multiframe images can be loaded from disk using a special syntax of imread, or created using MATLAB. Multiframe images can be displayed in several different ways; to display a multiframe image, can Display the frames individually, using the imshow function. See Displaying the Frames of a Multiframe Image Individually below. Display all of the frames at once, using the montage function. See Displaying All Frames of a Multiframe Image at Once. Convert the frames to a movie, using the immovie function. See Converting a Multiframe Image to a Movie below.

Displaying the Frames of a Multiframe Image Individually

In MATLAB, the frames of a multiframe image are handled in the fourth dimension. To view an individual frame, call imshow and specify the frame using standard MATLAB indexing notation. For example, to view the seventh frame in the intensity array I,

```
imshow(I(:,:,:,7))
```

The following example loads mri.tif in the MATLAB command windows and displays the third frame.

```
% Initialize an array to hold the 27 frames of mri.tif

mri = uint8(zeros(128,128,1,27));

for frame=1:27

% Read each frame into the appropriate frame in memory

[mri(:,:,:,frame),map] = imread('mri.tif',frame);

end

imshow(mri(:,:,:,3),map);
```

Figure 2.5:Third frame of mri.tif

Intensity, indexed, and binary multiframe images have a dimension of m-by-n-by-1-by-k, where k represents the total number of frames, and 1 signifies that the image data has just one color plane. Therefore, the following call, imshow(mri(:,:,:,3),map);

is equivalent to, imshow(mri(:,:,1,3),map).

RGB multiframe images have a dimension of m-by-n-by-3-by-k, where k represents the total number of frames, and 3 signifies the existence of the three color planes used in RGB images. This example, imshow(RGB(:,:,:,7)); shows all three color planes of the seventh frame, and is not equivalent to imshow(RGB(:,:,3,7)); which shows only the third color plane (blue) of the seventh frame. These two calls will only yield the same results if the image is RGB grayscale (R=G=B).

<u>Displaying All Frames of a Multiframe Image at Once</u>

To view all of the frames in a multiframe array at one time, use the montage function. montage divides a figure into multiple display regions and displays each image in a separate region. The syntax for montage is similar to the imshow syntax. To display a multiframe intensity image, the syntax is montage(I)

To display a multiframe indexed image, the syntax is montage(X,map)

Note: All of the frames in a multiframe indexed array must use the same colormap. This example loads and displays all frames of a multiframe indexed image.

```
% Initialize an array to hold the 27 frames of mri.tif.

mri = uint8(zeros(128,128,1,27));

for frame=1:27

% Read each frame into the appropriate frame in memory.

[mri(:,:,:,frame),map] = imread('mri.tif',frame);
```

```
end

montage(mri,map);
```

a square.

Converting a Multiframe Image to a Movie

To create a MATLAB movie from a multiframe image array, use the immovie function. This call creates a movie from a multiframe indexed image X mov = immovie(X,map); where X is a four-dimensional array of images that want to use for the movie. It can play the movie in MATLAB using the movie function. movie(mov);This example loads the multiframe image mri.tif in MATLAB command windows and makes a movie out of it. .

```
% Initialize and array to hold the 27 frames of mri.tif.

mri = uint8(zeros(128,128,1,27));

for frame=1:27

% Read each frame into the appropriate frame in memory.

[mri(:,:,:,frame),map] = imread('mri.tif',frame);

end

mov = immovie(mri,map);
```

movie(mov);

Note: that immovie displays the movie as it is being created, so will actually see the movie twice. The movie runs much faster the second time (using movie). Note MATLAB movies require MATLAB in order to be run. To make a movie that can be run outside of MATLAB, can use the MATLAB avifile and addframe functions to create an AVI file. AVI.

2.11 Artificial Neural Network

Neural network are a new information processing technique. They are computer-based simulation of living nervous system, which work quite differently than conventional computing [17]. From this definition know that the neural network is the new form of computing inspired by biological models. In this project Neural Network technique apply in segmentation for process of getting 3 dimensional imaging. In order to perform this modeling some operations over the image have to be completed, one of these, and perhaps the most important and complex is image segmentations. Segmentations as defined by Kapur [18] is a " a labeling problem in which the goal is to assign to each voxel in an input gray-level image, a unique label that represent an anatomical structure". Therefore the ultimate objective would be to properly identify some structures such as vena, artery the human heart tissue and etc. The segmentation of an image can be carried out by the different techniques that are based mostly on the discontinuity and similarity of the gray level of an image. In this project Neural network approach is used to segment the 2 dimensional heart images before goes to modeling.

In this project the Self-Organizing Maps used to perform segmentations. It will consist on a series of nodes or " neutronens " that will act upon a series of inputs. Each cell is densely interconnected, receives a primary input and a number of lateral interconnections from the output of other units. The lateral coupling of the neurons is though of as a function of the distance in two ways: Excitatory and inhibitory range. The excitatory is in a short range up to a certain radius and the inhibitory surrounds the excitatory area up to a bigger radius. Out side the inhibitory range a weaker and much bigger excitatory zone exists. A cluster or bubble around one particular node of the network is formed because of the lateral coupling around a given cell [19]. The primary input determines a winner node, which will have a certain cluster and then following the input the winner node with it surrounding cluster or neighborhood will adapt to the input. The process continues after the number of iteration until a certain degree of adaptation is reached. When the input is an image, certain feature can be extracted from the final adaptation of the neurons.

3. METHODOLOGY

3.0 Introduction

Software engineering process is consisting of set of the step that included the method, equipment and procedure. Set of step is always referring to paradigm of software engineering or software development life cycle (SDLC). This paradigm usually chosen by the kind of project that wants to development and application and also equipment and procedure to be conducted.

Paradigm or model for system development also call methodology is consist of several shape that have pro and contra. Nevertheless, all models have same generic phase, definition phase, development phase and maintenance phase.

For the whole perspective, methodology or steps for doing something consist of objectives for particular purpose in system development:

1) Investigating accurate system requirement.

2) Get ready the systematic step for system development. With that, the proceeds of system can be monitor in time of development.

3) Produce the system that easy to documentation and easy for maintenance.

4) Finding the changing of the system in the early stage of development.

5) Produce the system that can maximizing in using

6) Create the linear understanding between activities, source and limitations.

7) Help to finding un-consistent and redundancy in the process.

Models that want to consider below this title is the representation for the steps that consist phases in system development. This consideration is important to recognize the limitations and strengths that occur in the scale of phase. The available chosen methodology can help in system development for this project.

3.1 Considerations and Methodology Analysis

3.1.0 Spiral model

This model converge in risk reduce in system development. Below is the risk may be faced by the system in the scale of system development process refer to this model:

1. System may be not finish it goal or find the user requirement
2. System may be not achieving the quality needed.
3. System development cost is over from the budget.
4. Time taken in system development is too long from given time.
5. The experts involve leaving the project before ending the project-abandoned project
6. The same product develop but more excellent than the development project. It will cause the decrepit.

This model looking for produce the quality of product behind reduces the risk in the system development. Reducing the risk factor can reduce the cost and time especially in the testing phase. Accurate in risk analysis can make the maintenance phase become easily, behind perform the alternative way for problem solving in product or system development.

Nevertheless, this model needed connection or communication between user and system developer to initial the risk and method of solving problem. So this model only suitable for only internal system development which system developer and user situated in the same organization. Accompany for stakeholder in organization difference can makes many problem in contract agreement and non-aspect implications.

Behind that, the accurate risk analysis is not the easy way to be done. So it's not suitable for the small system development.

3.1.1 Rapid Prototyping Model

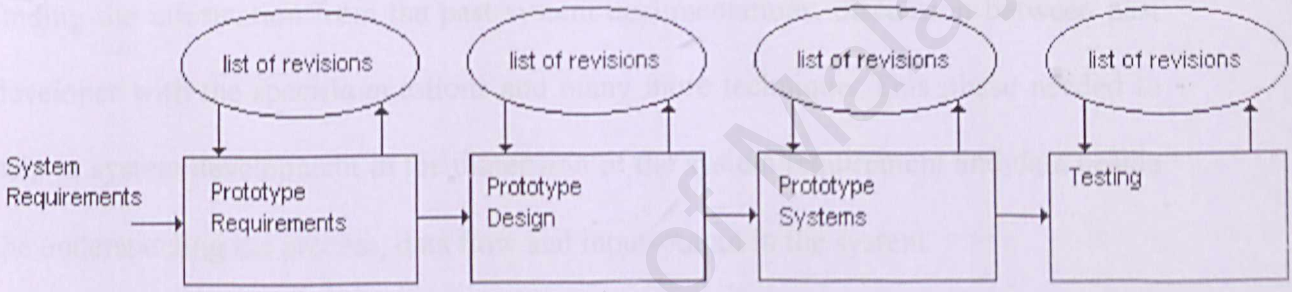


Figure 3.0: Rapid Prototyping model

- Start with user interface (probably most important to customer).
- When customer is happy with that, then work on the behind the scenes stuff.
- Customer is involved throughout the entire process.
- Linear model
- Does not turn into "product"
- Rapid prototyping may replace specification phase-never the design phase

<u>Rapid Prototyping Model: Specific focus</u>

Phase 0: System Requirement

It is not really the Rapid prototyping phase because it is find out before the first phase begins. But it is important to discuss it little bit for purpose of understanding. In this phase, the research and analysis in the past system perform to make understanding to develop the system and also identify the strength and limitation in the past system. With that it will help to find and identify the additional system requirement for the new system. Have several technique to reach this goal included finding the information from the past system documentations, discussion between past developer with the specific questions and many more technique. This phase needed to help in system development in the dimension of the system requirement and data beside the understanding the process, data flow and input-output in the system.

The techniques like data flow diagram (DFD), Entity relationship Diagram (ERD), structure chart and State Transition Diagram (STD) may be able to be use to understanding the behavior of the system in graphic manner.

This phase also must consider the user requirement assumed important for the user interface development (e.g. menu, algorithms box and windows), input screen, and manipulating screen. In this phase the developer must define the functions requirement and the non-function requirement. It is important for the success in system development. Un-accurate in purpose of introducing the functions requirement and non functions requirement will be deduct by the user may effect the development cost.

Phase 1: Prototyping the Requirement

This is the first phase in Rapid prototyping technique. In this phase the result for system requirement in the phase 0 prototyped. It also has the list of revisions to teach the right prototyping process. In the Rapid prototyping technique, start from this phase, interaction between user and developer begin until the system finish.

Phase 2: Prototyping Design

In this phase system design prototyped. It also has the list of revisions to taking the advising from user in the system design. The processes getting the information from user to develop the satisfied design continue repetitions until reach the goal.

Phase 3: Prototyping System

Like above 2 phase in this technique, where the user accompany to give the information for the purpose of system design. This system design included the create user interface, windows, manipulate icon and coding windows.

Phase 4: Testing

This is the last step in Rapid prototyping. The testing process is purpose to minimum the error before deliver the system to user. This process important to improve a quality of the system.

This testing consists of the several stages:

1) Function testing to ensure that system complete to operate.
2) Function testing to ensure that the user satisfied to the system.
3) Delivery function done by user before accept the system in formal.

The Assessment

System assessment doing after 3 to 6 month operated. It is looking for expert workers beside the stable system. This process is important to ensure that the system reach the objective.

Analysis and synthesis

From above considerations, rapid prototyping model is the most suitable model using in this project because the development process is quit fast with the need of interaction between users in each step of the prototyping process. It also keeps the revisions in each step of development. For the purpose of the human heart modeling the modeling process have a high interaction between developer and the user (physicians). The modeling process needs the fast result to perform the problem solving in the medical purpose. This model also deducts some step that consists in the othar model and suitable for purpose of modeling and their objectives. From this deducted also can fast in reach the objectives. Below is the summarization about why rapid prototyping model the most suitable model:

-suitable for small model or system

-the model doesn't need have too many steps to be followed

-the modeling need fast development

-the result from modeling process is used for limited time.

3.2 System Methodology

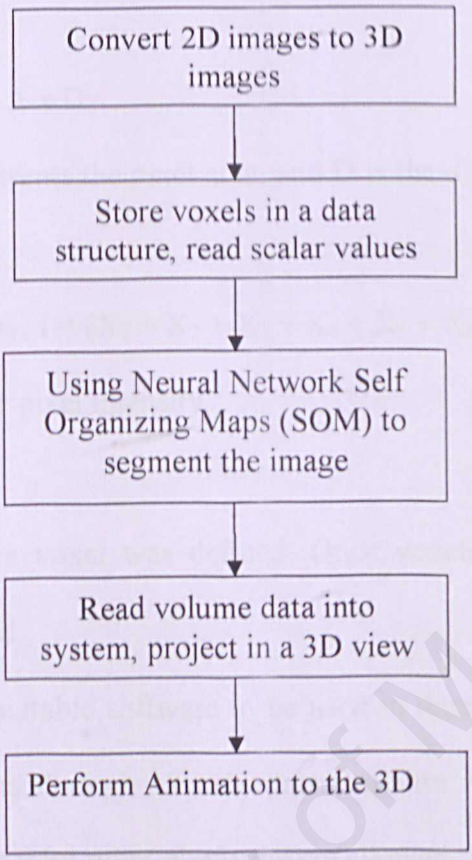


Figure 3.1 Overview of project processes

3.2.0 Reading of the 2D images files

2D slices had to be processed to obtain 3D volume images. As mentioned 2D MRI slices are available in the ACR format. These files have to be converted to the MAT format to be read in and manipulated in MatLab. Once the files were converted, voxels were calculated to obtain the 3D volume.

Voxel size and intensity computation was done using the following formulas. Voxel size was calculated using formula (1). The intensity of a voxel was calculated using formula (2).

(1) Voxel Size = A x D

where A represents the pixel area, and D is the distance between adjacent slices.

(2) Voxel Intensity, $I = (X_1 + X_2 + X_3 + X_4 + X_5 + X_6 + X_7 + X_8) / 8$

where X is the pixel intensity.

Through these calculations a voxel was defined. Once voxels had been created, 3D volume images were obtained.

MatLab was found to be a suitable software to be used in thus project. MatLab allows flexible manipulations of data through its array data structure. Various built in function can be used, and conversions from other platforms are allowed.

3.2.1 Defining a suitable data structure to store voxels

A suitable data structure must be used to store voxel values. For storage of voxels, a multi-dimensional data structure was used, since many values such as x, y, and z coordinates had to be stored. The MatLab Software allows various manipulation methods of the array data structure. Arrays provide multi-dimensional storage, computational and data organization features [7].

Multi-dimensional arrays in MatLab are an extension of the normal 2D matrix. A 2D matrix element can be accessed with two subscripts, one representing the row index, and the other representing the column index. Multi-dimensional arrays have three subscripts for indexing. The first subscript references array dimension 1, the row. The second references dimension 2, the column. The third references dimension 3, known as page.

The same techniques used to create 2D matrices, can be used to create a multi-dimensional array. MatLab provides a special concatenation function for building multi-dimensional arrays. These arrays can be generated using indexing, or using MatLab functions . Concatenation of the MAT format files were done, to obtain a 3D array. When the 3D array had been obtained, a MatLab program was written to calculate voxels, as explained above, thus producing a 3D volume.

3.2.2 Using Neural Network Self Organizing Maps to segment the image

When the image have store in the computer, image then performed the segmentations using self- organizing maps neural network proposed by Kohonen. It follow 2 basic thing: matching and finding the winner node determined by the minimum distance to the input and the update of the position of the neurons inside the cluster [19]. The updating process is a variation of the location of the node, proportional to the distance from the node to the input multiplied by the gain sequence if the node lies inside the neighborhood. If it is not inside the neighborhood, it position remain unchanged.

Beside that the neighborhood define with the present of two different cases, one if the network of nodes accepts that the neighborhood is limited by the edges of the network itself, and the other case that the neighborhood is not limit by the edges of the network[19]. From that we know that for example if the network consist of the n*1 neurons, the first case result in a linear network and the second an annular network. For the network is a two- dimensional array of neurons, the network itself generally limits the neighborhood.

For the image, it used as input and transform into an i*j*l matrix, where i*j is the dimension in the pixel of the image and l, the number of layer of the image. For color image l=3 and for the gray scale images l=1. For each (i,j,l) position there are exist a $K_{i,j,l}$ value in the range 0-255 depending on the intensity of the color or gray level. For example let $R_{i,j,l}$ be the region of existence of K. Also let looking for the example that the image transform to the 3D matrix like figure 3.2:

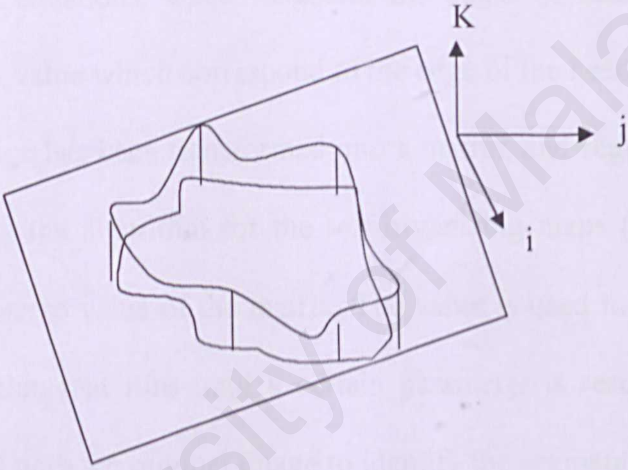


Figure 3.2: Matrix 3D of image

The i and j axis follow the original image and the value of the k-axis proportional to the intensity, darkness/brightness, of the pixel in the x, y positions of the original image. Let says the edges of the image appear with a higher k value and the while the other element appear in the lower values. From this example know that when applying to the human heart it can appear in the different value between the different kinds of pixel in heart. This format of the image allows several transformations. Since the k value represent the gray level, selecting a certain range of k can segment the image. For example with two different thresholds one lower, LT, and one higher, HT, the value of k can be transformed according to:

$$\forall\ i,j,l \qquad 0 \le LT < HT \le 255$$

$$0 < K_{i,j,l} \le LT \longrightarrow K_{i,j,l} = 0$$

$$LT < K_{i,j,l} \le HT \longrightarrow K_{i,j,l} = K_{i,j,l}$$

$$HT < K_{i,j,l} \le 255 \longrightarrow K_{i,j,l} = 0$$

According to above equations, when segment the edges of heart(for example) just selected the higher K value which correspond to the edge of the heart.

Once the image has been transformed into a matrix, and segmented if needed by gray level selections, the algorithm for the self-organizing maps (SOM) will have an input region the non-zero value of the matrix. The value is used for the self-organizing maps (SOM) algorithm that runs until a certain parameter is reached. Then the final network is compared with the original image to identify the segmented region.

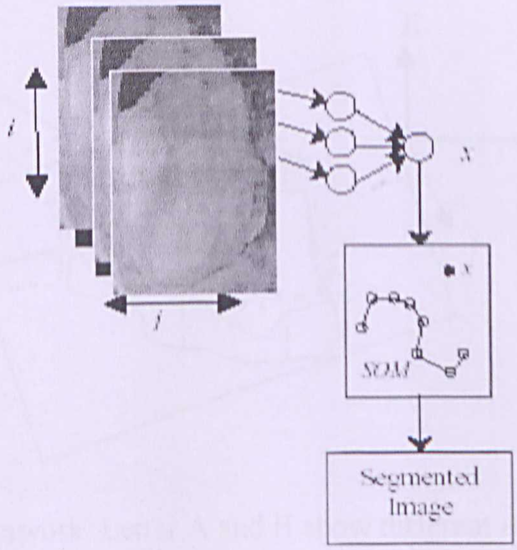


Figure 3.3: Segmentations Algorithm

The Self-organizing maps run with different images as input signal. For the medical images the network with the ring topology is the most suitable instead of a grid or linear [19]. It can provide the best results in order to extract the contour and "thin" the edges as well to give continuity to the shape of partitions in the heart. This result obtains after many iterations and run with the number of neurons.

One application of the algorithm is the volumetric object reconstructions: a collection of images produced by continuous slice of magnetic resonance of human heart can be segmented and for each slice, a number of neurons will represent point in the positions of the human heart. This volumetric algorithm will be use to produced the 3 dimensional model based on the point in a 3 dimensional space.

For example figure below show the output of network. The positions neurons will depend on density of pixels.

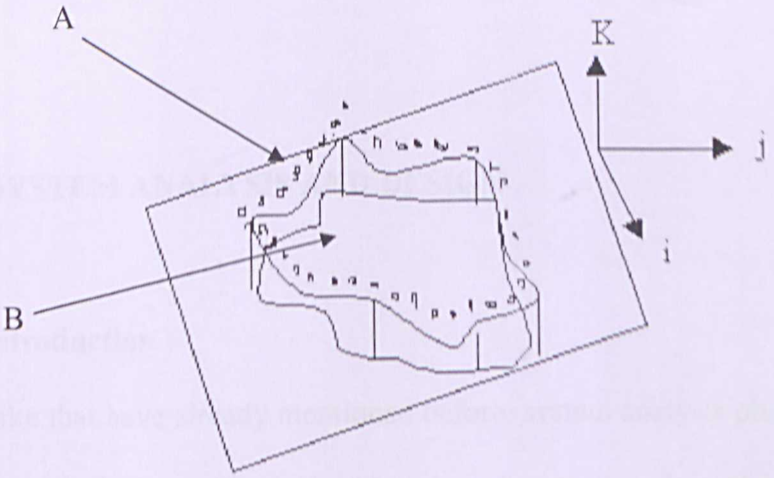


Figure 3.4: Output of network. Letter A and B show different distributions of the neurons

Letter A point a higher density of neurons along the edges and letter B show an unfilled region between the edges. Through this distance between neurons certain element in human heart can be identified like the artery or edges of heart like letter A.

3.2.3 Volume and sub-volume visualization

The volume was projected in a 3D view through manipulation of various MatLab functions. The Sub-volume was obtained by using the 'sub-volume' function in MatLab. Data ranges along the x-, y- and z-axis, corresponding to where the user would like to view the volume are read into the function. This portion of the volume is then extracted and displayed.

4. SYSTEM ANALYSIS AND DESIGN

4.0 Introduction

Like that have already mentioned before, system analysis phase is the early phase in a system development life cycle. This phase is important in getting an explanation and knowledge about important aspects that need to give an reluctant in the system development.

System analysis activity needs a specific approach including feedback from user sight, work analysis and another requirement specification that has been given by the organization.

This analysis is important to make sure that the system can execute and supported requirement and policy in the organization. For this reason, system analysis is divided into 2 main components, it is user analyze and system requirement specification that included function requirement and non-function requirement.

4.1 System requirement specification

The requirement is a characteristics or description that play in system development to complete a system in execution as suggested. It's not only explaining a flow of input and output information about the system but also explaining the restriction too in system execution.

To get an exact requirement, each process need a repetition and should be involve a feedback from end-user and system developer. An exact requirement is important to ensure that the system executed normally

For the system requirement specification that discuss in this chapter it is focus to function requirement and non-function requirement.

4.1.1 Function requirement

1) Image processing

2) Animation

3) Neural Network

4) 3 dimensional processing

4.1.2 Non-Functions Requirement

More reluctant must be given in user requirement satisfaction. It is not only about the function that is executed by the system, but it's need non-function requirements involved.

Other non-function requirements that must be followed by this system development is:-

Available to maintenance

The system is developed using an approach of a module that the system is consists of small modules. It will increase an understanding of the system execution and make the maintenance of the system become easy when the system is ready.

Reliability

The system may be able to give an output as the user need when using the system.

Efficient and correctness

The system can perform what the user want when it is need although the system has been used all the time and it executes every time without any problem. The speed in system interaction becomes one of an important issue in aspect of using system.

Available to used

The user needs to understand the system, easy to use and accepted by the user. The system must be able to learn by the user in a short time although they use the system for the first time. The system must have a guideline how to use the system.

Easy to understand

How the system work must be able to understand by the user. We can do any partial changing in the system without interrupt other part in the system.

4.2 Software and Hardware Requirement

Software

Operation system	: Windows XP, Me and above
Programming Language	: MatLab 6.5

Hardware

Central Processing Unit	: Pentium 3 500Mhz(minimum requirement)
Memory space HDD	: 20.0 Ghz
Random Access Memory	: 256 DDRAM
Other requirement	: CD-ROM, VGA and sound system

4.3 Graphical User Interfaces (GUI)

The GUI was developed by using the interactive Guide Editor in MatLab. The GUI development tools were found to be one of unstable layout, causing constant craches, and non-user friendly.

However, through the implementation of a GUI, the visualization system has been made user-friendly. The user can view the volume through just the clicks of buttons. Once the volume has been rendered and viewed, the user has to enter data ranges for the x-, y- and z-axis corresponding to where he would like to view the volume. A second click of the mouse displays a rendered sub-volume. Any portion of the volume can be extracted viewed as a sub-volume, depending on the values entered by the user.

5. SYSTEM IMPLEMENTATION

5.0 Overview of System Implementation.

In this chapter discussed about the process of development a system based on the given requirement. Implementation is the process to translate the whole process or detail project design to the code that perform individual function. There are several process included in this system development and all of that process will be defined in the rest of this chapter

5.1 Development Environment.

The most important thing to look is the platform of the system and the appropriate software and hardware chose for system development. In choosing the hardware and software also a platform or tool in development it would be speed up the development process.

5.1.0 Hardware Requirements.

For the hardware requirement is as same as mention in the early chapter that this system is running under the high performance personal computer with the high capacity of memory for the purpose of image processing pixel by pixel.

5.1.1 Software Requirements

In the software requirement also as mention in early chapter that the development of this system is in the Matlab software with in the Neural Network toolbox for the Self-Organizing Map Neural Network developing and the graphic toolbox for the purpose of development 3 dimension Human heart.

5.2 Development of Proposed System

There have 6 sub-sections to be proceeding in development of proposed system. But before doing the actual step in the system, data must be prepared to become a suitable for an input in the system.

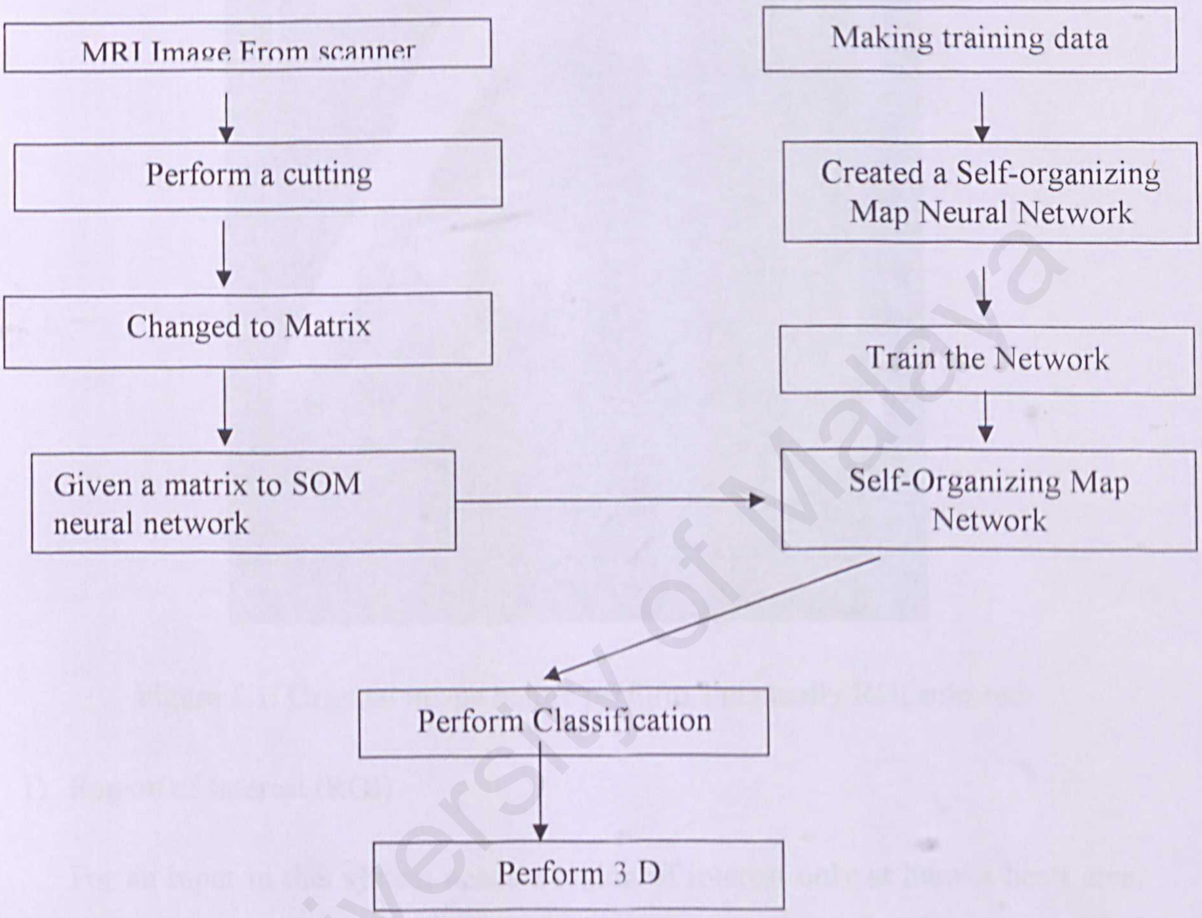


Figure 5.0: System development pipeline

1) Preparing for the data requirement.

1) Getting Material-an image acquiring from MRI

Image acquiring from the MRI scan is the image from the right-sided and consists of all partition of human abdomen. It is in 4-dimensional matrix and saved in current computer image format "*.tiff", which changed to 3-dimensional matrix image in 255-by-255-by-3.

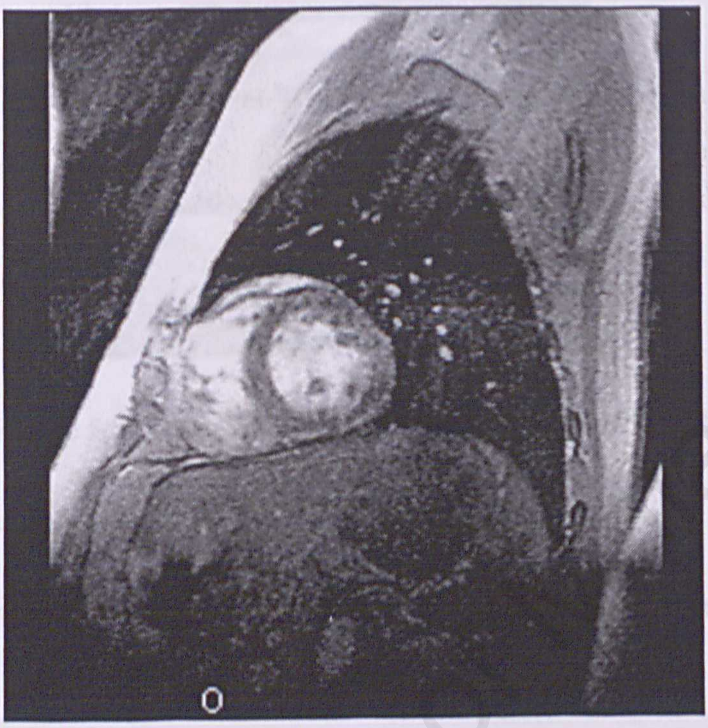


Figure 5.1: Original image before perform a manually ROI selected

1) Region of Interest (ROI)

For an input in this system needs a region of interest only at human heart area. The ROI process is doing in manually by using Adobe Photoshop software. The image then saved in ".tif' format that carry out an 2-Dimensional matrix. If MRI image that already show a heart area this step can be idled because image can read throughout Matlab and reduced it dimension by using "squeeze" command. This Region of interest operation does not remove the entire unwanted tissue region. From the cutting image in the figure 5.2 the dark color region is an abdomen tissue that classify in unwanted tissue. The white color is the elastic heart wall that has a ring occurs in less white color. This

-1 values. There are system referred used the random value to build this training data. But in this system not using this random value for training and the data is set as below to assure that distribution of neuron unit is not due to other value excluded.

T=[1 0.8 0.6 0.4 0.2 0 -0.2 -0.4 -0.6 -0.8 -1; 1 0.8 0.6 0.4 0.2 0 -0.2 -0.4 -0.6 -0.8 -1]

<u>Training data</u>

For example, when using a random value the data will be distribute in all over the space and in the training phase the neuron unit will adapt their synapse weight to be a winning neuron at their local maximum. It really useful when using many neuron but in this system consist only 6 neurons that want to capture large amount of training data. It is not so practical in case of accuracy in classification.

Creating the training data also consider in case of input data position out of space. For this case, the data will be respond back by the nearest neuron that will calculate by Euclidean distances.

1) Develop Self-Organizing Map (SOM) Neural Network.

The network that just created set to consist of 6 neurons unit that arranging in 6-by-1-matrix dimension. This type of network is unsupervised learning, which is not such instruction occur to be follow in perform a clustering. The basic idea of this neuron clustering is depend on class of pattern that will have something in common: they will judge as being similar. 6 neuron define a 6 different class that the SOM neural network will be perform for actual data given

During the training session, each of the neuron unit can be considered as competing to be awarded the training vector. When any training vector is presented, the distance of all cluster units is calculated and the neuron's unit that is closest to the training vector is denoted as a winning unit. The winning unit will then adapts its synapse weight in a way that moves that cluster unit even closer to the training vector.

The self-organizing map neural network has two phase of learning. In the first phase the neuron units are ordered to mirror the input space, and in the second phase fine-tuning takes place. In this system the distribution of training data can be plotted as below:

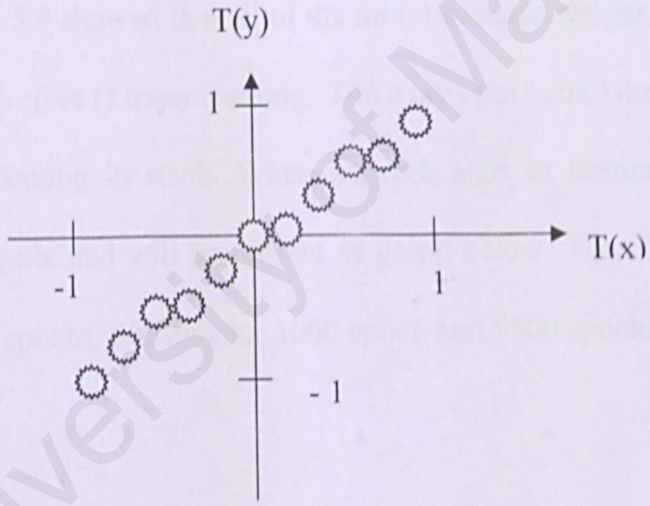


Figure 5.3: Training data plotted in graph

Before training self-organizing map neural network have initialization for newsom is midpoint. Thus, initial network neurons are all concentrated at the gray spot at (0.5, 0.5) as graph below:

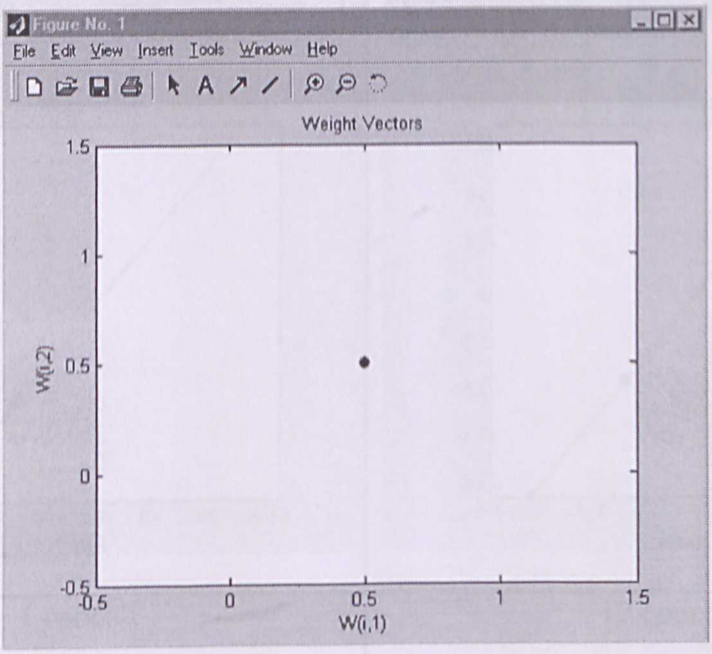


Figure 5.4: Network Initialization

The graph in figure 5.4 showed that all of the initial synapse weight for the 6 neuron located in the middle before it been training. The gray spot look like one spot but it consists of 6 spot arranging in stack. When network start to training the gray spot distributes along the space and will be present as graph below. These graphs take in 1 epoch, 10 epochs, 100 epochs, 500 epochs, 1000 epoch and 1500 epochs.

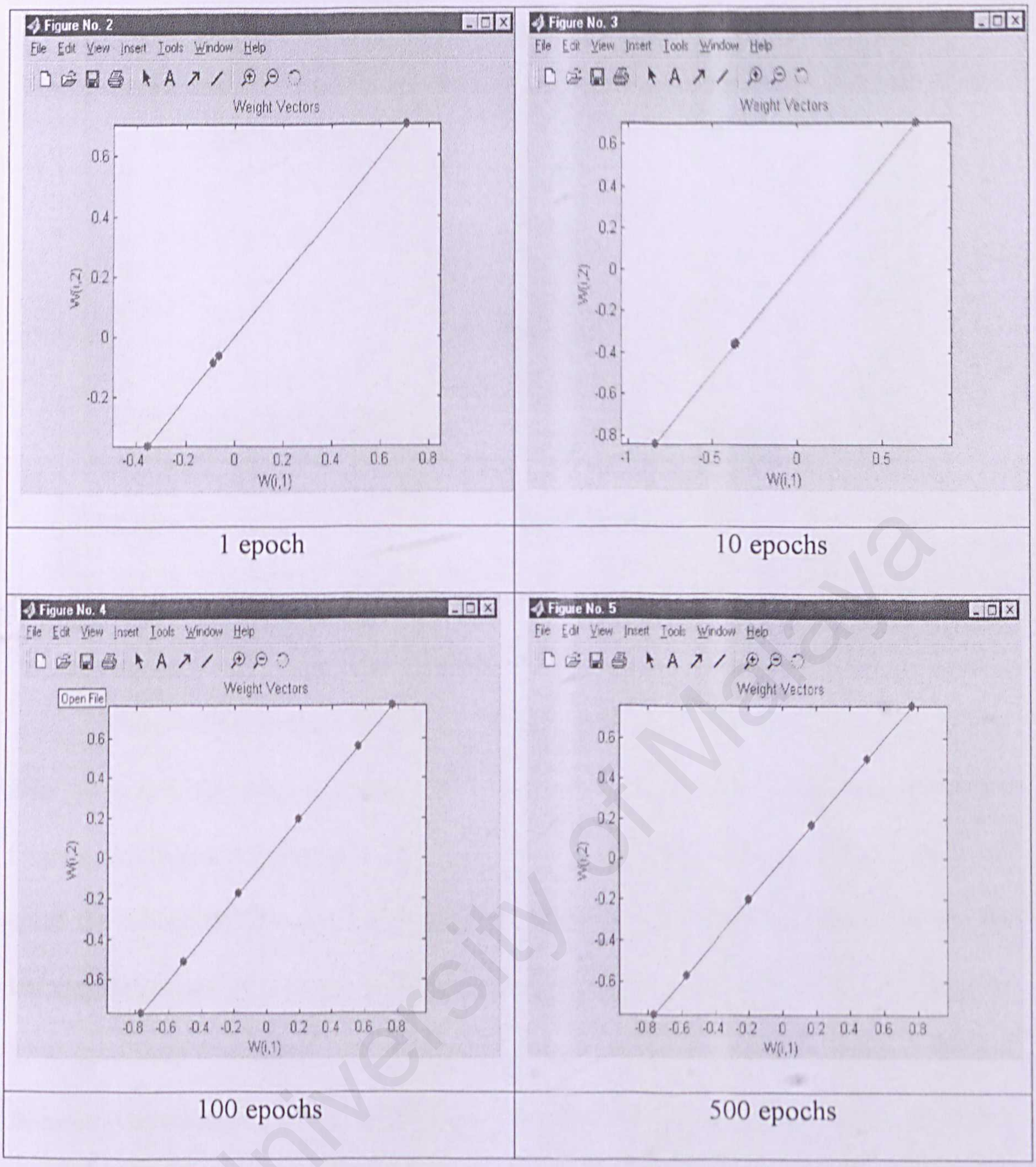
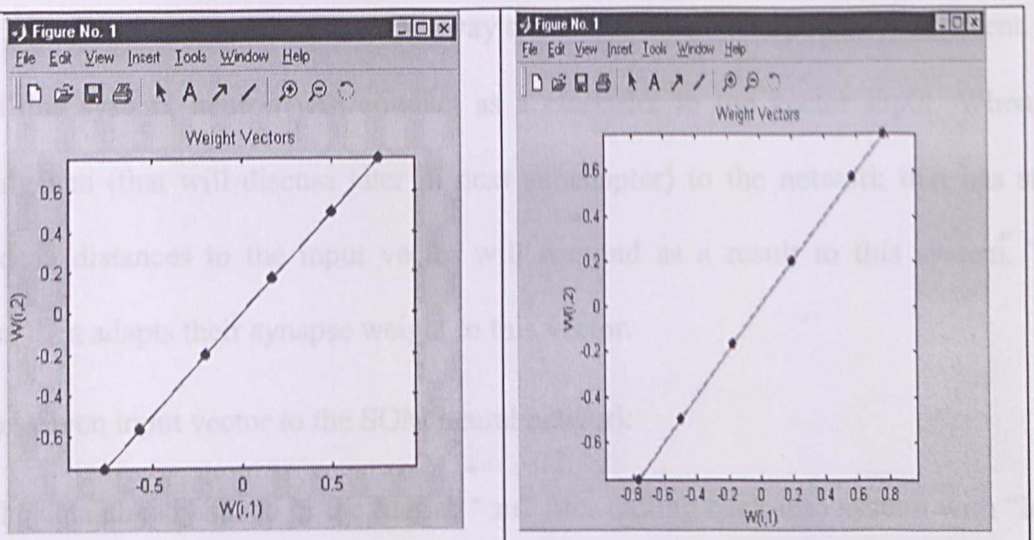


Figure 5.5: Initial synapse weight



1000 epochs	1500 epochs

Figure 5.5 cont…

After training session the synapse weight of neuron distribute along the training data space and become a winning neuron at their local area. The gray lines define the relative's connection between each neuron. More relative's each unit of neuron more small the distances it located. On this graph also show the distribution neuron along the training data space and synapse weight in each neuron become to adapt to the data after 1500 epochs. For example, lets says that this type of neural network want to grouping furniture according to use and appearance. All chair-like objects placed in one group and all table-like objects are another group. These groups are then inspected, and the table-like group is split to separate out desks. The desks group is similar to the table-like group and so these two groups are placed close to one another away from the chair-like group. So this neuron cluster algorithms do a similar job to pattern a data. The neuron that have located each other assume that have near as same pattern but different in some small part. For the neurons located far away show that the pattern perfectly different.

In this system, neuron will conduct as a classifier to the vector input. When the vector given (that will discuss later in next subchapter) to the network that has small Euclidean distances to the input vector will respond as a result to this system. That neuron then adapts their synapse weight to this vector.

1) Given input vector to the SOM neural network

The data already saved in the Matlab ".m" files calling back into system with "load" command. The data that already mention in the above subchapter have 2-dimensional that consists of 74-by-94-matrix. This data showed the intensity value of an image. The value between 0 to 255. For purpose of input to the neuron, this intensity value must be changed to the value between 1 to –1.

To perform this changing, normalization formula below used:

NIV= -1 + (OIV/128)

NIV= New value of intensity

OIV= old value of intensity.

The data of input vector changed to the value between 1 and –1 and presenting by one defined variable.

2) Network simulation

To simulate the network function "sim" used. This simulation method receives 2-by-1 matrix of input vector. When it going to simulate the input vector given to the self-organizing map neural network and the nearest neuron will be respond to that input vector. The output for this simulation command is in matrix x-by-1 and changed to index in Matlab. This simulation process can be described as figure 5.6.

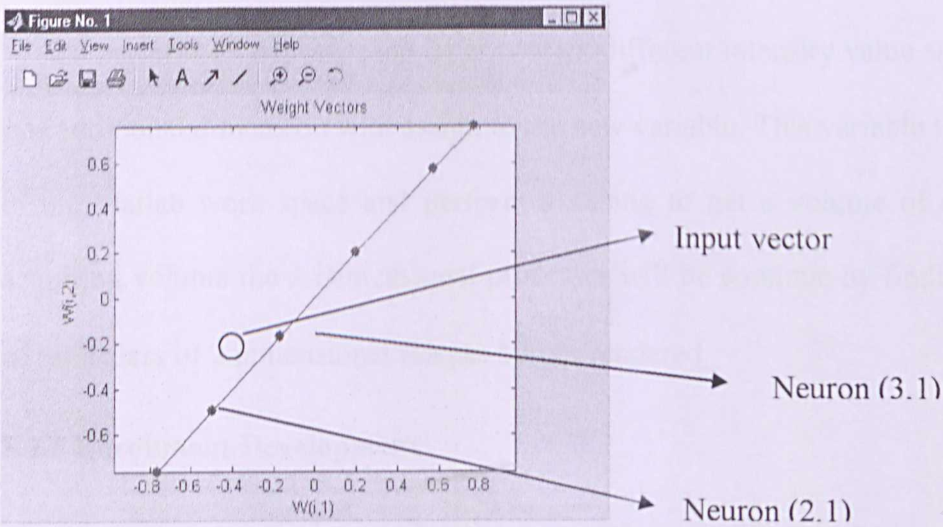


Figure 5.6: Simulation process

The white spot is the input vector that located in the middle of 2 neurons. For example, assumed that the neuron will be in matrix (2,1) and (3,1). But in the actual process, neuron will be unknown location in the space. The Self –organizing map neural network decided which neuron would be responding to the input vector based on nearest location and similarity of input vector to the neuron class. After known which neuron responding, input vector would located in one class by changing the value to certain value decided. Segmentation can be perform by looking different value in the image after classification by self-organizing map neural network.

5) Acquiring an output and change to 3-dimensional matrix.

From an output each class calling back by it similarity in intensity value. It is selected by getting region of interest. From this selecting it will result different region in the image.

1) Perform 3 dimensional

To perform sequences, each layer content different intensity value stacked by giving one additional dimension with assign to the new variable. This variable then calling back in the Matlab work space and perform a slicing to get a volume of an image. After acquiring volume the 3 Dimensional processes will be continue by finding their contour of each class of 2 dimensional images before rendered.

5.2.0 Enrollment Development

This system consist of several part explained in previous sub-chapter that will be translate back to the Matlab code. This code usually presented the process in each section in the system. The first section in the system is the Training data section. It is can be described in the code below:

```
P = [1 0.8 0.6 0.4 0.2 0 -0.2 -0.4 -0.6 -0.8 -1;1 0.8 0.6 0.4 0.2 0 -0.2 -0.4 -0.6 -0.8 -1]

Tc=[ 1 2 3]

T=ind2vec(Tc)
```

The variable P is the training data that created in 2-by-11 matrix and consist of value between 1 and –1. The variable Tc is assign a test value of index. It will be use in the acquiring vector for variable T. The command “ind2vec(Tc)” means the index of Tc will be changed to vector that assign the value of vector to T. The aspect result will be as below:

T=

(1,1)

(2,1)

(3,1)

This test data that show the mimic of result from the simulation can be changed back to the index by twisted this command to "vec2ind".

```
net = newsom([ -1  1; -1  1],[ 6 ]);
```

For the next code is to created a new Self-organizing map neural network. With the command "net=newsom" the new self-organizing map neural network can be created. The network used in this system consists of 6-by-1 neuron arrange in 2 dimensions and have two input vector fall in range –1 to 1 respectively. This network can be describe by graph plotted with calling by command below:

```
plotsom(net.iw{1,1},net.layers{1}.distances)
```

This graph that already present in figure 5.4 show one gray spot in the middle of the graph.

```
net.trainParam.epochs = 1500;
net = train(net,P);
plotsom(net.iw{1,1},net.layers{1}.distances)
```

The network that just create trained using the training data by command above with the 1500 epochs. Before that training data are test in 1 epoch, 10 epochs, 100 epochs, 500 epochs, 1000 epochs and 1500 epochs to view the distribution of the gray spot means a neuron synapse weight. All the evaluation process describe as graph in figure 5.5.

```
a = sim(net,P)
```

This "sim" command used to simulate a network by using testing value at training data "P". Variable "a" can keep the value in vector as:

```
        a=

                (2,1)    1
                ....
                ....
                ....
```

It present the vector of neuron that respond to the input vector.

```
ac=vec2ind(a)
```

From an output variable "a", it changed to the index by using command "vec2ind(a)". The variable "ac" can have a value of index as below:

```
ac=

        2
```

if variable "a" having a value (2,1).

```
load c1.m
in=c1
```

The command "load" is load an image that already changed to the matrix of intensity value. Variable "c1" is the file name and this is assign to other variable defined by "in" in this system.

```
for i=1:74
   for j=1:94
      v=-1+(in(i,j)/128)
      in(i,j)=v;
   end
end
```

This is the "for loop" part for the changed intensity value of an image from value 0 to 255 to the value between 1 to –1. It works pixel by pixel in the row and column.

Variable "in(i,j)" show that pixel value of image "in" in the row i-th and column j-th. In this part used a normalization formula as describe in above subsection that new intensity value store in variable "v" and the assign back to image "in" with the same row and column.

```
figure
imshow(in)
```

This function showed a new image after performed normalization process.

```
for i=1:74
   for j=1:94
      pi=[in(i,j);in(i,j)]
      a1=sim(net,pi)

      % changed vector of a1 to index

      ac1=vec2ind(a1)

      % assign new intensity value to the image in(i,j) follow by class.

         if ac1==1
            in(i,j)=-1
```

```
                else if ac1==2
                        in(i,j)=-0.5
                else if ac1==3
                        in(i,j)=0
                else if ac1==4
                        in(i,j)=0.3354
                else if ac1==5
                        in(i,j)=0.6563
                else if ac1==6
                        in(i,j)=1
                    end
                end
            end
        end
    end
end
end
end
```

This is again the for loop that process an image assign in variable "in" pixel by pixel for the simulation process. Variable "i" is the row of pixel and varible "j" defined column for pixel. Then each pixel value input to the simulation process by assign them to variable "pi". This value assign as 2-by-1 matrix dimension with the same value of each pixel (i,j). This is because original code for this Self-organizing map neural network taken in Matlab demo script process a intensity location (matrix of intensity) not an intensity value. But this is a modified script that accept an intensity value behind an intensity location. The location used just to calling an intensity value like "in(i,j)" that refer to the intensity value in location i-th row and j-th column. This simulation also process as same to fine the nearest neuron to respond an input vector.

The next part on this huge for loop part is vector changing. It's use a command "vec2ind" that assign a value of index to variable "ac1". This command operate as same as explained above that changing a vector value to an index value. This command used for the purpose of simulation output that appear in vector value.

The last part of this for loop is the if statement that contents a different separate value of variable"ac1". When an output from simulation function changed to index it will have value that refer to the which neuron respond because a simulation function acquiring an output consist a vector value n-th-by-1 dimension. The most wanted value is n that presents n-th neuron respond to the input vector. This value then differentiate with an if statement and the pixel value that belong to each class of variable "ac1" will be assign a new value defined. For summarization, each class of "ac1" will be different in intensity value and can be determine by calling back an image with code below:

```
figure
imshow(in)
```

This figure showed the new image with new intensity value that defined by different class.

```
BW1=roicolor(in, -1 , -0.6)
BW2=roicolor(in, -0.5 , -0.1)
BW3=roicolor(in, 0 , 0.3353)
BW4=roicolor(in, 0.3354 , 0.6563)
BW5=roicolor(in, 0.6564 , 0.9)
BW6=roicolor(in, 0.91 , 1 )
```

This code is to extract the different class that consist of different intensity value. Then variable "BW" is assign to the variable "p" below to make a 3 dimensional image.

```
p(:,:,1)=BW1;
p(:,:,2)=BW2;
p(:,:,3)=BW3;
p(:,:,4)=BW4;
p(:,:,5)=BW5;
p(:,:,6)=BW6;
```

Now variable "p" is become a collection of slices that can be show as command "slice". This "p" data can be treat as volume.

```
slice(p,[],[],[1:6])
contourslice(p,[],[],[1:6])
```

The "contourslice" command used to display a contour plot of a slice of volume. The code as below:

```
axis ij
xlim(x)
ylim(y)
daspect([1,1,1])
```

was defined to create a contour plot with the same orientation and size.

```
phandles = contourslice(p,[],[],[1,2,3,4,5,6],8);
view(3); axis tight
set(phandles,'LineWidth',2)
```

This code above used to display an orientation needed and to improve the visibility of the contour line.

6. SYSTEM TESTING

6.0 Introduction to System Testing

The purpose of this chapter is to evaluate the processing power, memory and secondary storage required for the effective implementation of the algorithm. A simulated study was performed to gauge the effectiveness of the algorithm.

6.1 Content of Testing

There are several parts in the testing process. It is tested to ensure that system will be given the good performances and running manually.

6.1.0 Unit/component testing

In this system consist of several small part that calling other function or calling other files to included in this system. But most of the part that used for calling the function already builds in the Matlab software. The other part like "load " command is calling the file that used in the system. For this part the content in the file filling with the matrix of image intensity value. In this system image can't be calling throughout the image file. It must be read in Matlab workspace and the value copy to another file that calling back to the system by command "load".

6.1.1 Integration testing

This sub-chapter purpose to ensure that all components performed correctly without problem when integrated. In this system all of the units or components perform correctly when it run in the Matlab workspace. The Matlab programming code are integrate each their code in the matrix match. One component or unit can be integrated to another component if the matrix dimension is match between each other. For this system, Self-organizing map neural network receive the input in 2-by-1-matrix that converge to –1 to 1 value of intensity. To integrated in self-organizing map neural network the training data and input vector must be set same as the input to the network. So the training data and input vector created same matrix dimension as a network input. The simulation component also has an input dimension as a neuron input.

The second component that must be integrated each other is a simulation component and if statement component. The simulation output is in vector and if statement receiving an index or a value that can turn to logic equation. This problem solve by using the command in the Matlab that changed the vector to the index before a simulation output being used in the if statement.

The last part to integrate is an output image acquiring after classified by SOM and create 3 dimensional images. This is the process to making the sequences from 2 dimensional images. It is process to adding dimension to an image with 2 dimensional. For this purpose matrix x-by-y in 2 dimensional images must be same value before it been adding another one dimension. This is set after classified process with in selecting region of interest process, which is region of interest selected by a value defined in the certain value. The pixel that have a value that located out of the range in selected consider been a dark region. But the region of interest is white region. For one image can be various slices in different value selected. But each slice can have all number of pixels in the image with is 74-by-94-matrix dimension. So each slice can have a same dimension with the different region. This all slices can be a sequence image by defined which slice can be 1-st slice and other slice be n-th slice.

6.1.2 System Testing

This chapter concern about fully functional operation of the system as proposed. Before the application of the algorithm apply to the real-world data sets, the algorithm was tested using synthetic data, comprising constant intensity objects on a constant intensity background. The test begins with the some created image that has different intensity value decided that showed in figure 6.0.

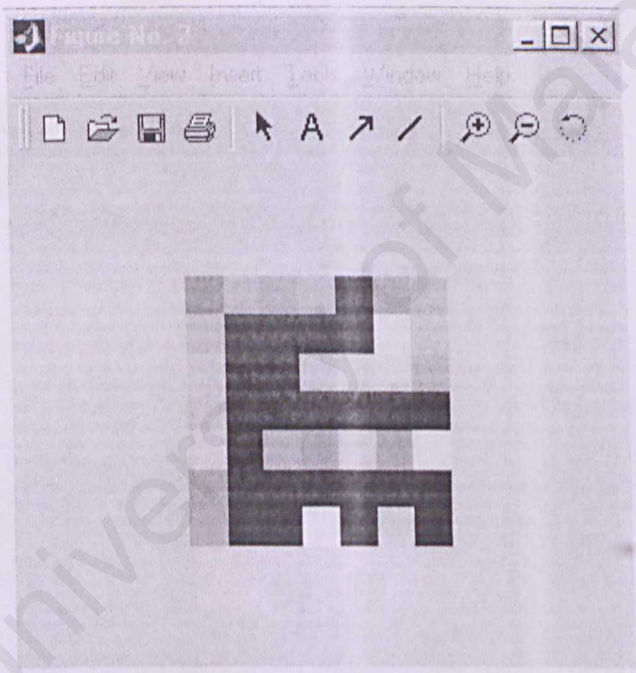


Figure 6.0: Image created with intensity value decided

Each pixel value simulated the different value at intensity in the image and region to be extracted. There are value in the dark color that showed the intensity value is 0 or less than 0. Otherwise the intensity value is bigger than 0 but not due to the 1. Each pixel has their unique intensity value. This color scheme also firstly assumed simulated the different meaning in the image. The dark color simulated the unwanted partitions in the human abdomen or even a blood. Beside that the other pixel color simulated the main heart elastic wall or other heart soft tissue. This Self-organizing map neural network wants to perform the classification and acquiring the different region on this image. For the beginning SOM network given 2 neurons to classify the different between the pixels simulated heart wall and other tissue. It perform well when it acquiring the result as present in figure 6.1. It changed this dark tissue to the white color and other tissue to the dark color. This algorithm then changed back to make sure that the color is right to the region it's present (white is soft tissue and dark is unwanted tissue or blood).

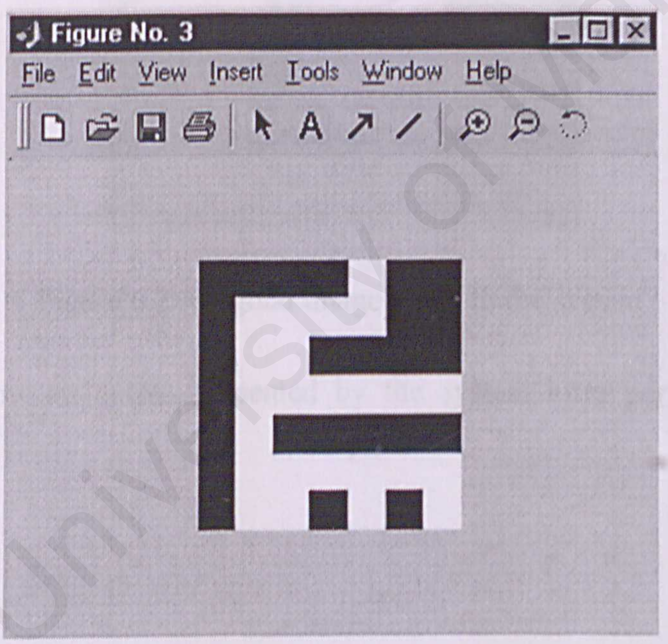


Figure 6.1: Tested image after classify in SOM network

The satisfactory of this classification process is to make the different between different values in the pixel. On this testing also can described the Self-organizing maps neural network can perform very well where there are many neuron created.

6.2 Result

The last part in this chapter is the result. Real-life image of human heart is used to ensure that the system perform well in the huge data. The first part to looking is the training scheme. The epochs of training is changed time after time to ensure that the neuron distribute perfectly in the training data. After 1500 epochs the neuron meet their maximum distribution in the training data as show in figure 5.5. The second part is looking an output acquired produced by the system on given input vector. Figure 6.2 show an image giving to the system.

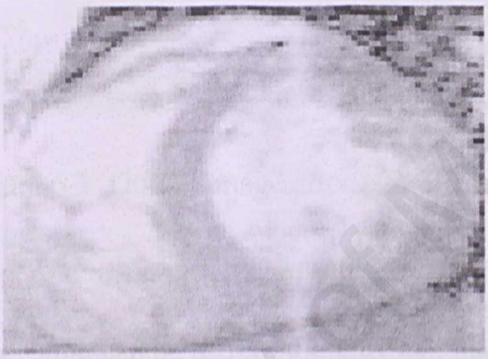


Figure 6.2: Original image given to the system

At figure 6.3 show an output presented by the system after perform classification operation.

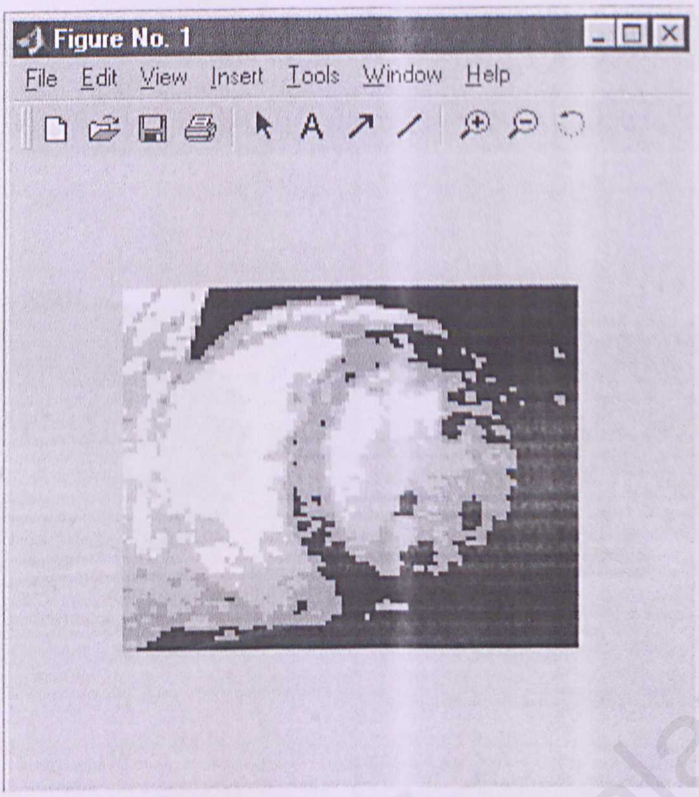
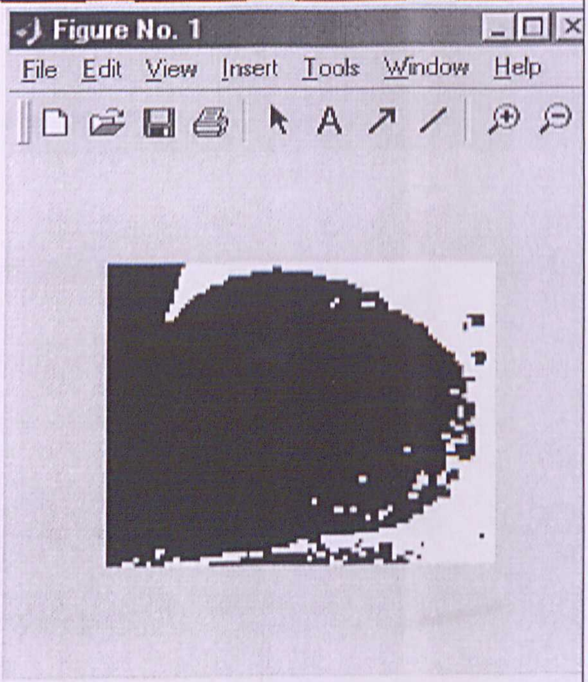


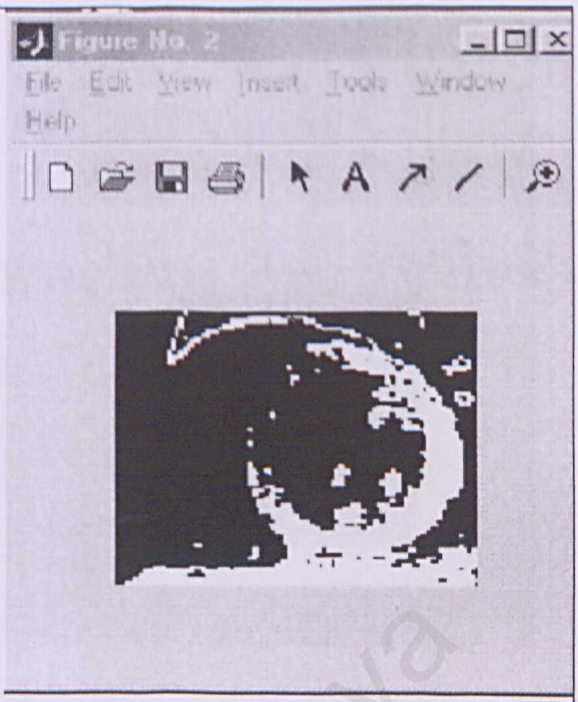
Figure 6.3: Output image after classification

This image consists of 6 different class of intensity value defined in each class. Each class of this intensity value can be differentiate by call back the value by performing region of interest operation as figure 6.4.



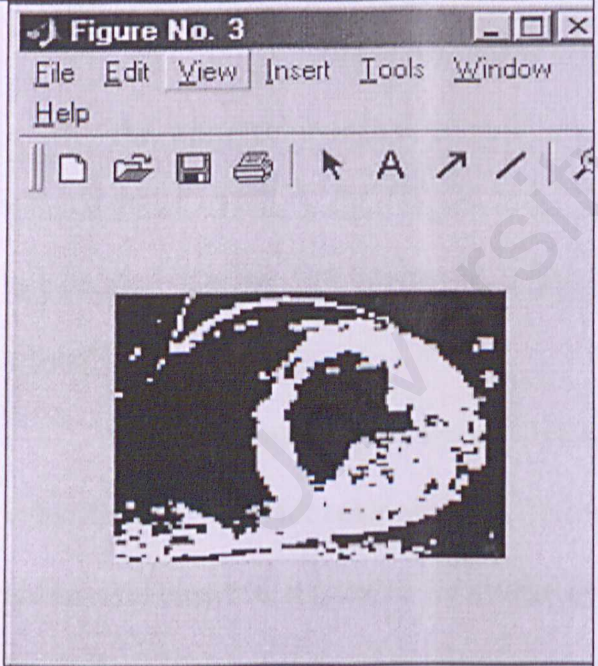
Cluster 1

Intensity value: -1 to –0.6



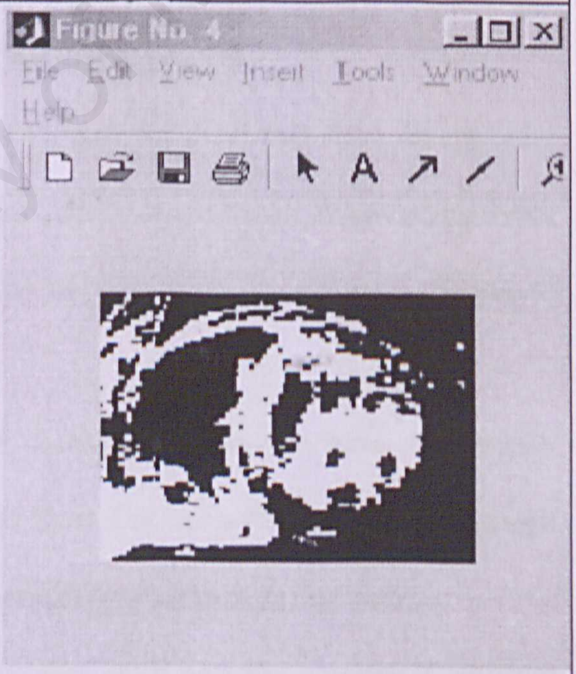
Cluster 2

Intensity value: -0.5 to –0.1



Cluster 3

Intensity value: 0 to 0.3353



Cluster 4

Intensity value: 0.3354 to 0.6562

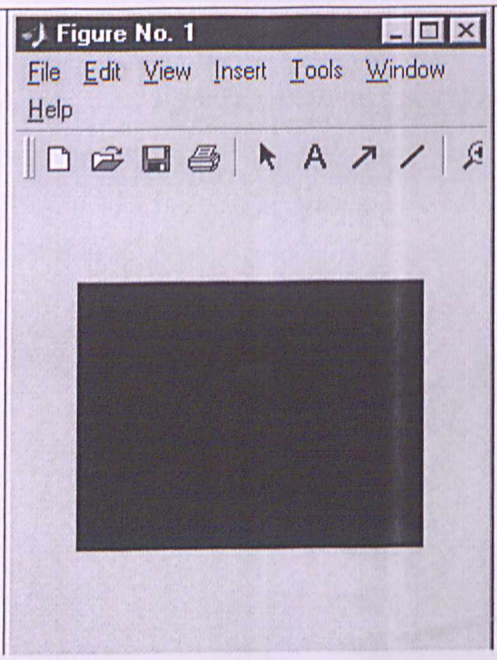
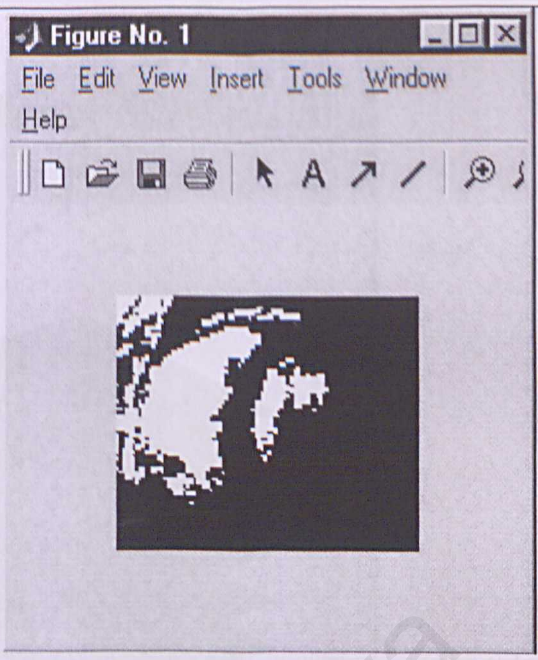
	
Cluster 5 Intensity value: 0.6563 to 0.9	Cluster 6 Intensity value: 0.91 to 1

Figure 6.4: Image after clustering process

The white color region on each image showed the pixel that have an intensity value defined. The dark color region is the pixel value that excluded. When going down stream from cluster 1till cluster 6, it showed the evaluation of different classes of clustering.

The white color region on each image showed the pixel that have an intensity value defined. The dark color region is the pixel value that excluded. When going trough cluster 1till cluster 6, it showed the evaluation of different classes of clustering.

To perform a volumetric, each image arranging into stack and the distance is set from one slice to another slice of 2 dimensional images as show in figure 6.5.

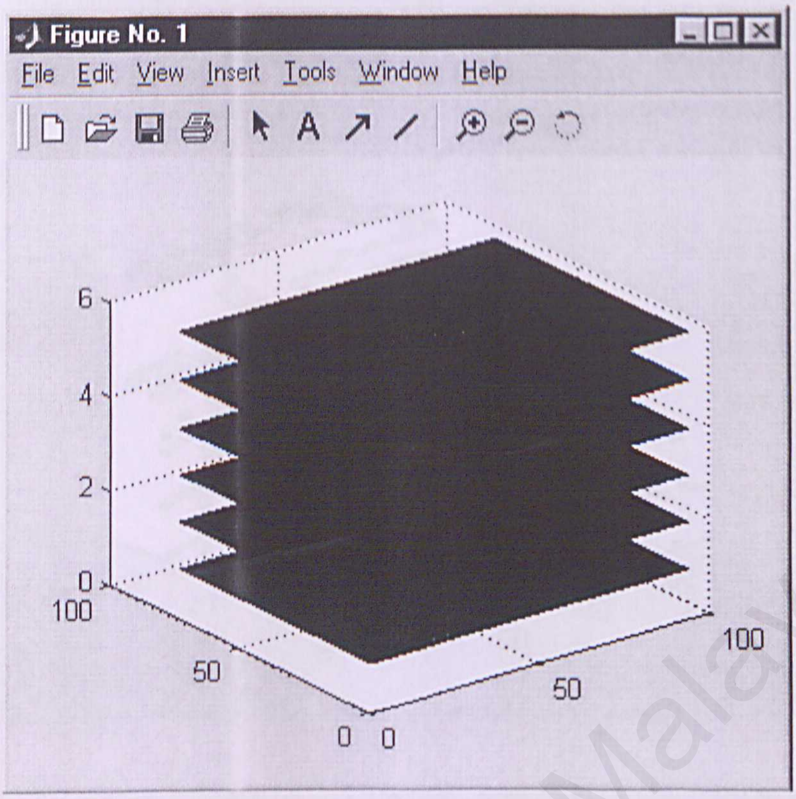


Figure 6.5: 2D image arranging in stack

Nevertheless the segmentation operation does not remove all of regions that not necessary used for 3D. Thus, the segmented image sliced again by selecting their contour as given a result in figure 6.6.

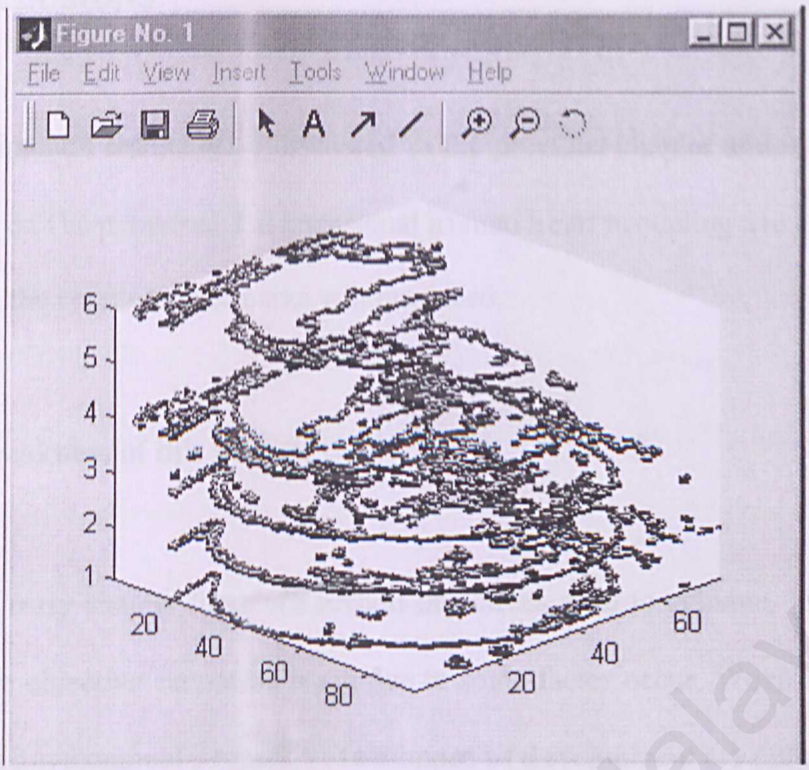


Figure 6.6: Image after contour selecting

This contour image then must be performing a rendering process to building a feature of human heart.

7 CONCLUSIONS

Experimental results were discussed in the previous chapter and in this chapter the weakness of the proposed 3 dimensional human heart modeling are analyzed and discussed and the concluding remarks will produced.

7.0 The weakness of this project

In the early chapter there are several objectives need to achieve. Even thought there are some objective cannot be reach due to some factor occur. When narrow down to the system development process, it is stopped at development a volumetric image. The rest of process cannot be proceed due to time limitation and too much time spending in process of understanding the behavior of self-organizing map neural network and also in the development phase. Two more part should be finish is rendered process and making an animation of heart beating process. But this project showed the useful path to achieve these two objectives and need a little bit research in the graphic and image area. This system cannot be one of the user-friendly systems because it is not provide any user interface layout. Nevertheless due to the system is trough-away kind like system; the interface is not too important part because the parameter and variable using in this system need to be changed in different time according to the used of system.

The problems also occur in high computational cost of space and time persists, despite the selection of a region of interest. The segmentation process for one image takes 1 hour and 15 minute in the Matlab platform using Microsoft environment. High computational power and data storage capacity is required for conversion of a biological model.

7.1 The Outcome of this Research

The performance of the proposed 3 dimensional human heart modeling algorithm is sensitive to various parameters such as the size of region of interest, location of region of interest and the self-organizing output class. This all parameters tested and changed from time to time to acquiring the maximum result satisfied.

7.2 Further Research Suggestion

From the weakness of this project, further research suggested in the graphic area and image section with in field of creating a movie to an image. This 2 area is the most important area to make a real-life like 3 Dimensional human hearts modeling with heart beating animation. Also this project can be making as a path to further real life organ modeling.

7.3 Concluding remarks

3 dimensional biological organ modeling has potential application in various medical fields e.g. traumalogy, neurosurgery planning, radiotherapy planning, medical research and education. It is a long process. When exploring from up-stream to down-stream it is consist of serial process that connected each other. Even though to produced the real-life like 3 dimensional model it is depend on acquiring an image from MR scanned. Otherwise it also depends on the various sensitive parameters that need to be

changed to produce a satisfied result as explain in the previous sub-chapter. Yet, this modeling system is not one of the exact models that will give a real-life like 3 dimensional models. It is because some of the segmented result not giving a full-satisfied result due to the pixel value that consists of nearly close value. This kind of value look likes to be the same but presenting 2 different regions in the image. Other factor that influences this modeling process is the quality of image acquiring by MR scanned image. The hardware requirement also one of the important components must be view when performing for conversion of a biological modeling.

Even though, hope this scholarly training will be one of the paths for the next extension research in this field.

REFERENCES

[1] F. Bloch, "Nuclear induction," *Physical Review*, vol. 70, pp. 460--473, 1946

[2] E. Purcell, H. Torrey, and R. Pound, "Resonance absorption by nuclear magnetic \ moments in a solid," *Physical Review*, vol. 69, pp. 37--38, 1946

[3] A. Garroway, P. Grannell, and P. Mansfield, "Image formation in NMR by a selective irradiative pulse," *J. Phys. C: Solid State Phys.*, vol. 7, pp. L457--L462, 1974.

[4] P. Lauterbur, "Image formation by induced local interactions: Examples employing nuclear magnetic resonance," *Nature*, vol. 242, pp. 190--191, 1973.

[5] J. Bezdek, L. Hall, and L. Clarke, "Review of MR image segmentation techniques using pattern recognition," *Medical Physics*, vol. 20, no. 4, pp. 1033--1048, 1993.

[6] L. Clarke, R. Vethuizen, M. Camacho, J. Heine, M. Vaidyanathan, L. Hall, R. Thatcher, and M. Silbiger, "MRI segmentation: Methods and applications," *Magnetic Resonance Imaging*, vol. 13, no. 3, pp. 343--368, 1995.

[7] P. van den Elsen, E. Pol, and M. Viergever, "Medical image matching - a review with classification," *IEEE Engineering in Medicine and Biology*, vol. 12, pp. 26--39, 1993.

[9]http://wwwex.physik.uni-ulm.de/lehre/PhysikalischeElektronik/Phys_Elektr/node364.html

[10] http://www.netdoctor.co.uk/health_advice/examinations/ctgeneral.htm

[11] Andreas Pommert, Micheal Bomans, Martin Riemer, Ulf Tiede and Karl Heinz Hohne,"Volume Visualization in Medicine: Technique and Applicatios", in Hans Hagen et al.(eds). Focus on Scientific Visualization, Springer-Verlag,Berlin,1993,pp.41-72.

[12] P.M.C Gieles," Images-guided surgery: digital imaging as a support to patient treatment", MedicalMundi, Vol 40,no.2,pp.54-63,1995.

[13] Ramin Shahidi, Rhea Tombroponlos and Robert P.Grzeszczuk," Clinical Application of 3-dimensional Rendering of Medical Data Set", proceeding of the IEEE,Vol.86, No.3, pp.555-568, March 1998.

[14] K.Kouris,"positron Emission Tomography", IEEE colloquium Tomography Reconstruction Technique Dig.,30 Group E4, May 1981..

[15] Habib Zaidi,"Medical Imaging: Current State and Future Perspectives". Communications Regional Symposium, On the Arab world and the Information Society, Tunisia,4-8May,1997,

ElectronicsVersion: Http://www.unesco.org/webworld/tunis/tunis97/com35/com35.html.

[16]………,………, Medical illustrations

Electronic version: http://yourmedicalsource.com/library/ctscan/CTS_whatis.html

[17] Marilyn McCord Nelson.W.T.Illingworth, A practical Guide to Neural Nets,Corporate and Professional Publishing Group.

[18] Kapur T(1999).Model based 3 dimensional medical Image Segmentations, Ph.D thesis, Artificial Intelligence Laboratory, Massachusetts Institute of Technology.

[19] Constantino Carlos Reyes-Aldosoro, Image Segmentations with Kohonen neural Network Self-Organising Maps, Instituto Tecnologico Autonomo de Mexico.

[20] Woo Chaw Seng, Magnetic Resonance Image Segmentation Using Wavelet Transform and Self-Organizing Map Neural Network, Thesis Master Degree of Computer Science, Faculty of Computer Science and Information Technology University of Malaya, session 1999.

[21] Robert Callen, The Essence of Neural Network, Prentice Hall Europe,1999.

[22] B.D,Ripley, Pattern recognition and neural Network, Cambridge University Press,1996

[23] Mohamad H. Hassoun, Fundamentals of artificial neural network, MIT press Cambridge,1995.

[24] Laurene Fausett, Fundamentals of neural network, Prentice Hall International,1994

[25] S.Y.Kung, Digital neural network, PTR Prentice Hall,1993.

APPENDICES

1) Whole project code

```
%% A One-dimensional Self-organizing Map

% classify 11 two-element vectors.

P = [1 0.8 0.6 0.4 0.2 0 -0.2 -0.4 -0.6 -0.8 -1;1 0.8 0.6 0.4 0.2 0 -0.2 -0.4 -0.6 -0.8 -1]

Tc=[ 1 2 3]

T=ind2vec(Tc)

%%

% Use a 6- by- 1 layer of neurons to classify the vectors above. We would

% like each neuron to respond to a different region, and

% neighboring neurons to respond to adjacent regions.Create a layer of 6

% neurons spread out in a 6 by 1 grid:

net = newsom([-1  1;-1  1],[6]);

%%

% This can visualize the network have just created with PLOTSOM.

%

% Each neuron is represented by a red dot at the location of its two weights.

% Initially all the neurons have the same weights in the middle of the vectors,

% so only one dot appears.

plotsom(net.iw{1,1},net.layers{1}.distances)

%%

% Now train the map on the 11 vectors for 1500 epoch and replot the network

% weights.
```

```
%
% After training, note that the layer of neurons has begun to self-organize so
% that each neuron now classifies a different region of the input space, and
% adjacent (connected) neurons respond to adjacent regions.
net.trainParam.epochs = 1500;
net = train(net,P);
plotsom(net.iw{1,1},net.layers{1}.distances)
%
% Now use SIM to classify vectors by giving to the network and
% seeing which neuron responds.
%
% The neuron indicated by "a" responded with a "1", so p belongs to that class.
a = sim(net,P)
% Changed from the vector to index for output "a"
ac=vec2ind(a)
% load an image matrix that want to classify
load c1.m
% assign an image to variable
in=c1
% Doing normalization- changed intensity value from 0 to 255, to 1 to -1
for i=1:74
   for j=1:94
      v=-1+(in(i,j)/128)
      in(i,j)=v;
```

```
    end
end
% show an image after normalization.
figure
imshow(in)
%% Now use SIM to classify image vectors by giving to the network and
% seeing which neuron responds.
%
% The neuron indicated by "a1" responded with a "1", so p belongs to that class.
for i=1:74
   for j=1:94
       pi=[in(i,j);in(i,j)]
       a1=sim(net,pi)
       % changed vector of a1 to index
            ac1=vec2ind(a1)
        % assign new intensity value to the image in(i,j) follow by class.
            if ac1==1
                   in(i,j)=-1
            else if ac1==2
                   in(i,j)=-0.5
            else if ac1==3
                   in(i,j)=0
            else if ac1==4
                   in(i,j)=0.3354
```

```
            else if ac1==5
            in(i,j)=0.6563
        else if ac1==6
            in(i,j)=1
          end
        end
      end
    end
  end
end
end
end
% show an image after classification-all of image
imshow(in)
% call back new intensity value of image.
BW1=roicolor(in,-1,-0.6)
BW2=roicolor(in,-0.5,-0.1)
BW3=roicolor(in,0,0.3353)
BW4=roicolor(in,0.3354,0.6563)
BW5=roicolor(in,0.6564,0.9)
BW6=roicolor(in,0.91,1)
% assign a value to variable p for making 3 dimensional sequences image
p(:,:,1)=BW1;
p(:,:,2)=BW2;
```

```
p(:,:,3)=BW3;

p(:,:,4)=BW4;

p(:,:,5)=BW5;

p(:.:,6)=BW6;

% doing slice and contourslice to an image

slice(p,[],[],[1:6])

contourslice(p,[],[],[1:6])

axis ij

xlim(x)

ylim(y)

daspect([1,1,1])

% making the 3D for the image

phandles = contourslice(p,[],[],[1,2,3,4,5,6],8);

view(3); axis tight

set(phandles,'LineWidth',2)
```

2) Self-organizing map parameter

SOM parameters value used (some of the value is the default value that already been set after using command "newson"):

SOM parameters	Parameter value
Ordering phase learning rate	0.9
Ordering phase steps	1000
Turning phase learning rate	0.02
Turning phase neighborhood distance	1
Epochs	1500
Topology function	Grid (gridtop)
Distance function	Link distribution (Linkdist)