THREE-DIMENSIONAL (3D) RECONSTRUCTION OF COMPUTER TOMOGRAPHY CARDIAC IMAGES USING VISUALIZATION TOOLKIT (VTK)

MOHAMMAD ZAHID BIN ZAMALUDIN

SUBMITTED IN FULFILMENT OF THE REQUIREMENTS FOR THE MASTERS IN BIOMEDICAL ENGINEERING

FACULTY
UNIVERSITY OF MALAYA
KUALA LUMPUR

2021

UNIVERSITY OF MALAYA

ORIGINAL LITERARY WORK DECLARATION

Name of Candidate: Mohammad Zahid b. Zamaludin

(I.C/Passport No:

Matric No: 17202168/1

Name of Degree: Masters in Biomedical Engineering

Title of Research Report ("this Work"): **Three-Dimensional (3d) Reconstruction of Computer Tomography Cardiac Images Using Visualization Toolkit (VTK)**

Field of Study:

I do solemnly and sincerely declare that:

(1) I am the sole author/writer of this Work;
(2) This Work is original;
(3) Any use of any work in which copyright exists was done by way of fair dealing and for permitted purposes and any excerpt or extract from, or reference to or reproduction of any copyright work has been disclosed expressly and sufficiently and the title of the Work and its authorship have been acknowledged in this Work;
(4) I do not have any actual knowledge nor do I ought reasonably to know that the making of this work constitutes an infringement of any copyright work;
(5) I hereby assign all and every rights in the copyright to this Work to the University of Malaya ("UM"), who henceforth shall be owner of the copyright in this Work and that any reproduction or use in any form or by any means whatsoever is prohibited without the written consent of UM having been first had and obtained;
(6) I am fully aware that if in the course of making this Work, I have infringed any copyright whether intentionally or otherwise, I may be subject to legal action or any other action as may be determined by UM.

Candidate's Signature Date: 13/1/2021

Subscribed and solemnly declared before,

Witness's Signature Date:

Name:

Designation:

ABSTRACT

Cardiac study proves to be important to diagnose and treat cardiovascular disease that accounts for 20% of death in the world. Three-dimensional (3D) images of the heart are important to understand its real structure, functions and physical abnormalities occurring in the tissue. The objective of this paper is to reconstruct 3D images from 2D CT Scan slices of the cardiac region of interest using Visualisation Toolkit (VTK). The Visualization Toolkit (VTK) is non-propriety software with state-of-the-art tools for 3D rendering that is easily customizable programming. 3D images are reconstructed using surface rendering technique and volume rendering technique. Volume rendering shows best rendering quality compared to surface rendering when reconstructing cardiac 3D images. Artery structure are easily identifiable using volume rendering in VTK whereas surface rendering image yielded false positive or false negative images. VTK capabilities are also compared RadiAnt DICOM Viewer. The 3D image rendered match the quality of RadiAnt 3D image.

ABSTRAK

Kajian jantung terbukti penting untuk mendiagnosis dan merawat penyakit kardiovaskular yang menyumbang 20% kematian di dunia. Gambar tiga dimensi (3D) jantung adalah penting untuk memahami struktur, fungsi dan kelainan fizikal sebenar yang berlaku di dalam tisu. Objektif kertas kajin ini adalah untuk menyusun semula gambar 3D dari potongan CT Scan 2D dari kawasan jantung yang menarik dengan menggunakan Visualization Toolkit (VTK). Visualization Toolkit (VTK) adalah perisian sumber terbuka dengan alat canggih untuk membina imej 3D yang mudah disesuaikan dengan pengaturcaraan. Gambar 3D direkonstruksi menggunakan teknik 'surface rendering'(SR) dan teknik 'volume rendering'. 'Volume rendering' menunjukkan kualiti imej terbaik berbanding 'surface rendering' (VR) ketika menyusun semula gambar 3D jantung. Struktur arteri mudah dikenalpasti menggunakan rendering isipadu dalam VTK sedangkan gambar rendering permukaan menghasilkan gambar positif palsu atau negatif palsu. Keupayaan VTK juga dibandingkan dengan RadiAnt DICOM Viewer. Imej 3D yang dihasilkan setanding dengan kualiti imej 3D RadiAnt.

ACKNOWLEDGEMENTS

First and foremost, I would like to express my deep and sincerest gratitude for my research supervisor Ir. Dr. Lai Khin Wee, Department of Biomedical Engineering, Faculty of Engineering, University of Malaya for providing invaluable guidance throughout the research. His vision, motivation and advice has inspired me to conduct this research. I am forever beholden to his insight that ultimately lead to the completion of this paper.

I would also like to thank my sponsors Ministry of Health Malaysia for allowing me to go through this journey of learning to upgrade my knowledge. Their willingness to provide financial support and sabbatical leaves spoke greatly on their commitment for continuous professional development. The knowledge gain will surely help to improve and contribute to the healthcare service of the nation.

I am forever indebted to my father, my sisters and my brother for their continuous support throughout my journey completing this paper. No amount of money can ever repay their kindness for accommodating my needs during this period. I am thankful for the endless love shown by my daughter Sofea Zahra that help to keep my spirit up when the going gets though. I am also overwhelmed by the memory of my late mom and my late wife that ultimately shape the man who I am today.

The world is currently battling a worldwide pandemic and it does not look like it will over soon. Circumstances are not ideal especially when trying to complete research during this period. Many challenges were faced but ultimately task can be done with drive and focus. The hope is humanity will triumph in facing COVID19 and life can get back to where it was or go on with the new ‘normal’ as soon as possible.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS AND ABBREVIATIONS

CAT	:	computerized axial tomography
CAD	:	coronary atherosclerosis
CTA	:	computerized tomography angiography
DCE-US	:	dynamic contrast-enhanced ultrasound
ECG	:	electrocardiographic
EBCT	:	Electron Beam Computer Tomography
HU	:	Hounsfield Unit
IV	:	interpolated value
IVUS	:	intravascular ultrasound
MC	:	marching cube
MSCT	:	multislice CT
ROI	:	region of interest
SSD	:	Surface shaded display
3D	:	three-dimensional
TCC	:	transitional cell carcinoma
2D	:	two-dimensional
VTK	:	Visualization Toolkit
VR	:	volume rendering

LIST OF APPENDICES

CHAPTER 1: INTRODUCTION

1.1 Introduction

Cardiovascular disease (CVD) accounts for more than 20% of death in the world (Roth et al., 2018). To provide accurate diagnosis and di In cardiac imaging, Computed tomography (CT) can be used to detect conditions such as congenital heart disease, coronary atherosclerosis, valves injury, blood clots or tumors in the heart. CT utilizes a passing of x-ray beam through the patient from multiple directions and reconstructs x-ray collected in the detectors as cross-sectional images in axial plane known as slices. The images of the specific organ or region of interest (ROI) are in two-dimensional (2D) slices with the series of slice image would make up the entire ROI. The advent of CT has certainly assisted clinician in diagnostic and treatment of CVD. Technological advancement such as fast gantry rotation and electrocardiographic (ECG) gating help CT to provide better quality cardiac images with minimal artifacts (Becker, Jakobs, et al., 2000; Fink et al., 2011)

Motion artifacts are the biggest challenges when acquiring high quality images for diagnostic in the cardiac region. The fact is human heart constantly in motion to pump oxygenated and deoxygenated blood throughout the body. The presence of artifacts further intensified with irregular cardiac cycles stemming from uneven rhythms or premature beats (Kalisz et al., 2016). Conventional CT early on is considered inadequate for any cardiac related imaging because of longer scan times. Improvements have been made in the field of cardiac imaging with CT as a non-invasive coronary imaging modalities (Sun, 2010). Introduction 4-slice CT scanner in 1998 represents technical evolution in cardiac imaging. The development of scanning techniques significantly improves diagnostic accuracy of multislice CT (MSCT). This was demonstrated by the emergence of CT scan from 16-slice to the more advance 256-slice and 320-slice systems (Sun, 2010). MSCT allows extensive volume coverage with high-resolution image. ECG triggering or gating also an important factor for cardiac coronary CTA to ensure sufficient temporal resolution (Ronald Mikolich, 2012). ECG gating technique acquire images at a specific point in one (1) heartbeat or R-to-R wave. Motion artifacts occurs with rapid heart rate or the rhythm is erratic even with ECG gating (Lawler et al., 2005). Administration of beta (β) blockers can help slow patients heart rate to approximately 60 beats per minute (bpm) and limit motion artifacts by increasing the time cardiac cycle spent in diastole (Lawler et al., 2005). A high quality 2D slice cardiac images can be achieved with the current advancement in CT technology.

However, the inherent limitation of a 2D images representing a 3D volumetric structure cannot be ignored. X-Ray Angiography are generally used to project cardiovascular images with the help of contrast agent in 2D. The stenosis or lesion may be underrated by the 2D X-ray angiography images, and the precise coronaries 3D geometry cannot be reconstructed. CT generates 2D image slices in sequence and depending on scanning protocol it can be separated by gaps. By stacking the 2D images in sequence, the picture elements (pixels) in 2D will create volume elements (voxels) to produce 3D images (Luccichenti et al., 2005). Visualization of volume are done through ray casting where built-in virtual rays from the screen's pixels to the volume. Pixel values of the flat panel screen are attained from voxels data along the projection rays to show depth perception associated with 3D images. Interpolation are carried out when volume voxels does not intersect the virtual lines. Different method such as surface shaded display and volume rendering can be used to define pixel values from voxels data. Surface shaded display (SSD) or surface rendering technique reconstruct anatomy surface's by assigning pixel's value nearest to the screen above chosen threshold. No densitometric data is provided and the threshold configure the similarity between resulting image and actual anatomy respectively (Luccichenti et al., 2005). In volume rendering (VR), pixels value is determined from densitometric information provided by the voxels which in this case attenuation in Hounsfield Unit (HU).

3D reconstruction of medical imaging data using Visualization Toolkit (VTK) has been explored in multiple studies (Dong et al., 2013; Hongjian, 2009; Nugroho et al., 2017; Wee et al., 2011; Wheeler et al., 2018; Xu et al., 2013). VTK is an open-source system with graphic library for 3D modeling, image processing, graphics, volume rendering and data visualization. It has been extensively used in 3D computer graphics, image processing and visualization. The application of marching cube (MC) algorithm in VTK has shown promises when reconstructing 3D medical images from 2D CT Scan images (Hongjian, 2009). MC have relatively higher efficiency of reconstruction compared to Contour reconstruction due to its low rendering time (Xu et al., 2013). The MC technique developed linkages between slices then conceive a table of 15 cases which represents the triangle topology independently. 3D medical data would then be processed in in scan line order and linear interpolation is used to determine triangle vertices between adjacent slices (Cirne & Pedrini, 2013). Lai et al (2011) proposes a simplified triangle mesh and improved storage structures by reducing storage amount that improve image reconstruction and image display speed (Wee et al., 2011).

1.2 Problem Statement

A conventional computed tomography images are 2D slices at axial plane. A cardiologist must transform the tomographic images mentally to form a 3D impression. Its time consuming and inefficient which may lead to misdiagnosis and incorrect therapy delivery. Estimation of organ and lesions such as tumor volume are being done through assumption of an idealized shape when using a 2D images. This practice will lead inaccuracy and operator variability. Patient anatomy and organ position also present some challenges for diagnostic using a 2D images. Visualization are hindered by overlapping organ preventing accurate diagnosis and therapeutic interventional procedures

1.3 Aim & Objective

The aim of this research paper is to explore the use of VTK as a post processing tool to reconstruct a 3D cardiac images from 2D images slices acquired using CT Scanner. To achieve this, the following objectives are set:

(a) To visualize cardiac in 3D from 2D CT scan slices using reconstruction algorithm in VTK

(b) To analyze the reconstructed image between surface rendering and volume rendering method

(c) To compare 3D reconstruction result between VTK and Radiant DICOM Viewer

CHAPTER 2: LITERATURE REVIEW

2.1 Introduction

In this literature review various topic related to application of CT Scan and 3D reconstruction related to cardiac imaging. Section 2.2 reviews the history of CT Scanner in Radiology including technology for cardiac imaging. Protocol and application for cardiac imaging are discussed in Section 2.3. Section 2.4 discussed 3D reconstruction method especially for surface rendering and volume rendering with the advantages of 3D vs 2D images for diagnostic and therapy. Finally, Section 2.5 explain VTK program and its visualization pipeline mechanisms t reconstruct 3D images

2.2 CT Scanner in Cardiology

The initial CT Scan invented in 1971 by Sir Godfrey N. Hounsfield was able to produce only a single slice image up until 1981 where systems with detector arrays of four slices and fast rotation time of 500ms were invented. Cardiac application has greatly

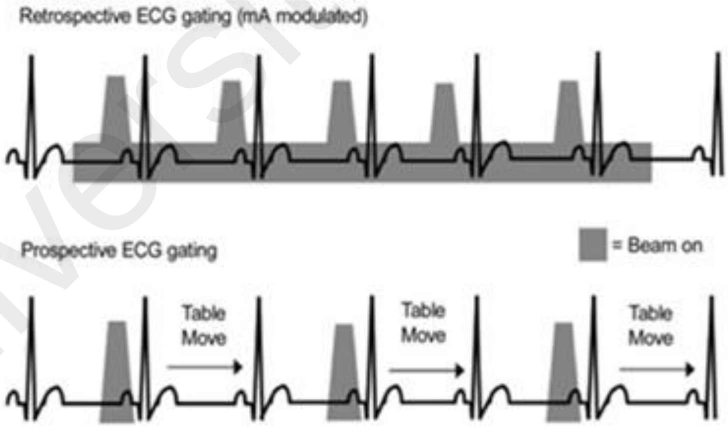

Figure 2-1: Retrospective ECG gating vs Prospective ECG gating

benefitted from the multi-slice acquisition and fast rotation time. Compared to a single slice system, the new technology performed eight times better in terms of scan time (Becker, Knez, et al., 2000) . An elevated heart rate is the most common cause of motion artifacts. For cardiac CT techniques that acquire and reconstruct data over multiple R-R intervals, a slow and regular heart rate is imperative for excellent image quality. The cardiac rest period averages 120 *ms* and ranges from 66 to 330 *ms*. A temporal resolution of less than 250 *ms* for diastole and less than 50 ms for systole required at low heart rate. Higher temporal resolution (i.e., a smaller number) required for higher heart rates (Kalisz et al., 2016). Cardiac motion during scanning can reduce image quality and impinge diagnostic accuracy on coronary CTA and calcium-scoring studies.

ECG triggering or ECG gating can reduce the motion artifacts by synchronizing a digital ECG real time record of the patient and synchronized it with the image acquisition process. It requires a digital ECG reading to be recorded and synchronized with the acquisition of the image data. Prospective electrocardiographic (ECG) gating is a use forecast of R wave timing, step-and-shoot non-spiral acquisition without table motion during imaging, and unique cone-beam reconstruction. Meanwhile, cardiac CT with retrospective ECG gating uses a backward-looking measurement of R wave timing, spiral scanning during table motion, and more traditional cone-beam reconstruction (Shuman et al., 2008). With ECG gating, there is a motion artifact if the heart rate is fast or the rhythm is irregular (Becker, Jakobs, et al., 2000). However, a non-ECG-gated routine CT of the heart with thin collimation and fast gantry rotation speed may be enough for the study in select cases of patients with high heart rates (Lawler et al., 2005). However, ECG-gated reviews are the optimal method for clinical practice in most patients.

2.3 CT Scanner Cardiology Application

2.3.1 Coronary calcium detection and quantification.

The presence of coronary atherosclerosis (CAD) in symptomatic patients are mainly detected through coronary calcifications. Agatston et al was the first to introduce quantitative assessment of coronary calcium (Agatston et al., 1990). On non-contrast cardiac CT images, calcification identified as areas of hyper attenuation of at least 1 mm2—with > 130 Hounsfield units (HU) or ≥ 3 adjacent pixels (Lawler et al. , 2005).

The Agatston method is the most widely used as a reference for population databases and reports involving risk stratification. It uses the weighted sum of lesions with a density above 130 HU, multiplying the calcium area by a constant related to maximum attenuation of plaque: 130–199 HU, factor 1; 200–299 HU, factor 2; 300–399 HU, factor 3; and ≥ 400 HU, factor 4 (Neves et al., 2017). Therefore, the slice thickness and the interval must follow the original protocols to reduce the variation in noise. The result is the maximum attenuation of the plaques, allowing the initially published scores to reproduced.

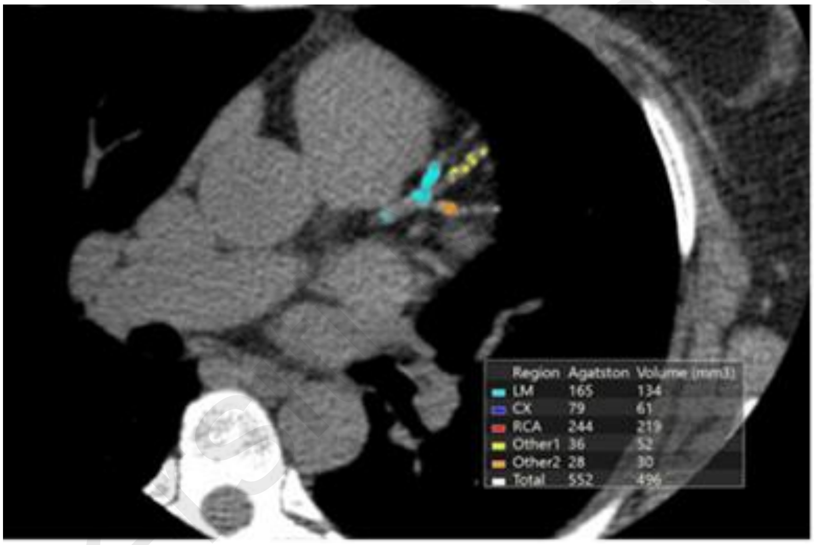

Figure 2-2: CT Coronary artery image with calcium scoring based on Agatston method

The radiation exposure should be ‘as low as reasonably achievable’ or ALARA since the identification and quantification of coronary calcium is used as a screening tool in potentially healthy subject. Electron Beam Computer Tomography (EBCT) was commonly used for coronary calcium screening due its low radiation exposure. However, using prospective ECG triggering when scanning with multi detector CT scanner produce similar radiation exposure with comparable image quality (Becker, Ohnesorge, et al., 2000). Reproducibility of calcium measurement are also improved when applying spiral acquisition with small slice increment. The option to increase radiation in CT to achieve better quality image and minimize image noise are helpful when dealing with obese patient.

2.3.2 Coronary CT Angiography (CTA).

Images of the coronary arteries typically acquired during the end-diastolic phase, when there is almost no cardiac motion. Radiocontrast material is injected intravenously to accomplish opacification of the coronary arteries. Images are acquired after time delay to allow the contrast media to propagate and reach the coronary arterial circulation (Ronald Mikolich, 2012). Motion artifacts can render the image unacceptable for diagnostic. The use of β-blocker administered orally to the patient prior to scanning can help to ensure satisfactory image quality (Becker, Ohnesorge, et al., 2000). CTA can be used to evaluate changes of coronary artery wall. Coronary atherosclerosis can happen over time and in most severe cases can lead to stenosis or prone to rupture. Compared to intravascular ultrasound (IVUS), CTA able to project non-calcified plaques rather than IVUS where it will be shown as soft tissue (Becker, Ohnesorge, et al., 2000).

2.4 3D Reconstruction

3D post processing of patient image in volumetric data format has been established as an essential diagnostic tool. It has effectively delineated anatomic feature, facilitate

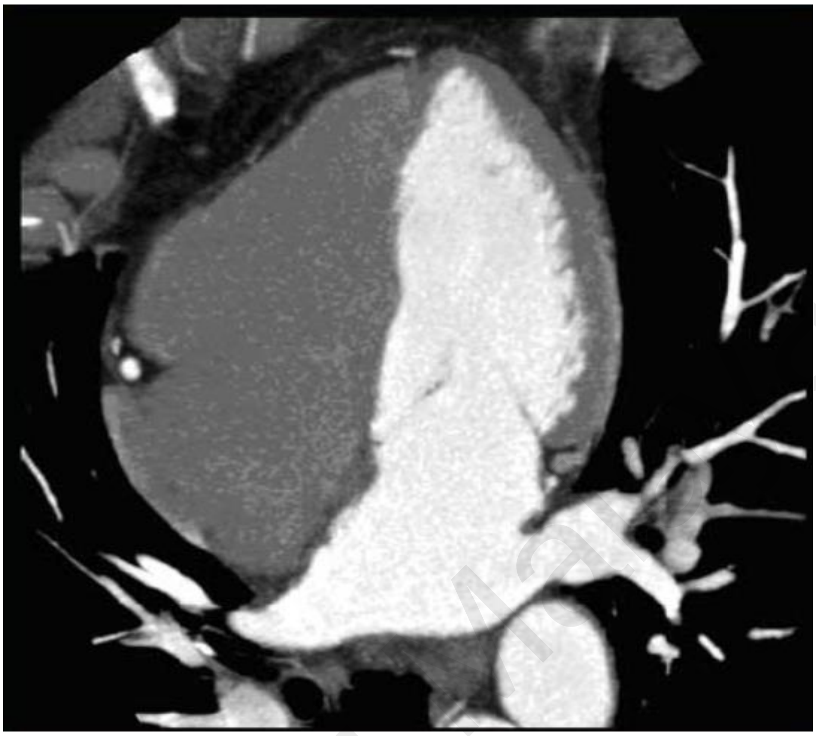

Figure 2-3: Cardiac CTA images where contrast is filling the left ventricle during scanning.

classification of changes in pathology and reporting of quantitative data for devising a treatment plan (Pierce et al., 2012). For CT Scan, 3D images are reconstructed using multiple image slices in 2D. The quality of the 2D image slices are crucial to ensure the best 3D images can be produced. This is particularly hard for cardiac where the heart is in constant motion pumping oxygenated blood all over the body. Motion artifacts are common and the use of β- blocker, ECG gating and faster gantry rotation that leads to faster scan time as previously discussed can help to eliminate the issue.

CT Scan generates serial 2D images with gap separation depending on the scanning protocols like pitch selection. Interpolation is used to fill these gaps to gather a volume of continuous data. This volume data in 3D is known as volume elements (voxels) compared to 2D images where it is assembled by picture elements (pixels). Interpolation determine the missing value from established surrounding point such as nearest neighbor and, linear association between two adjacent points or cubic convolution from four or more points. The values assigned are called interpolated value (IV) (Luccichenti et al., 2005). Voxels volume that projected to a flat screen does not corresponds properly to the pixels in the screen. To enhance depth perception, ray tracing or ray casting is performed where projection rays from a hypothetical observer point are generated from the screen's pixels value and convert it to volume. The value of the pixel from which a ray hit an object can be determined by the type of reconstruction technique used such as shaded surface display (SSD) and volume rendering (VR). Ray casting is a volume rendering algorithm while ray tracing is used in surface rendering algorithm with in-built iteration. In ray casting, the projection ray passes through the volume rather than selecting the closest point like in ray tracing. The color and opacity of the image are interpolated at constant located points. These interpolated values produced the color at the pixel on the render window (Khan et al., 2018).

2.4.1 Surface shaded display (SSD)

SSD or surface rendering (SR) set the image pixels value corresponds to the value closest to the screen. A threshold is selected to determine how much reconstructed images respects the actual ROI anatomy. No densitometric data is provided by an SSD image. Therefore, calcifications of vascular structures and the contrast material can be represented in the same way when proper threshold are not selected. A depth-encoded shading scale makes the closer voxel to appear brighter enhancing the depth perception (Luccichenti et al., 2005).

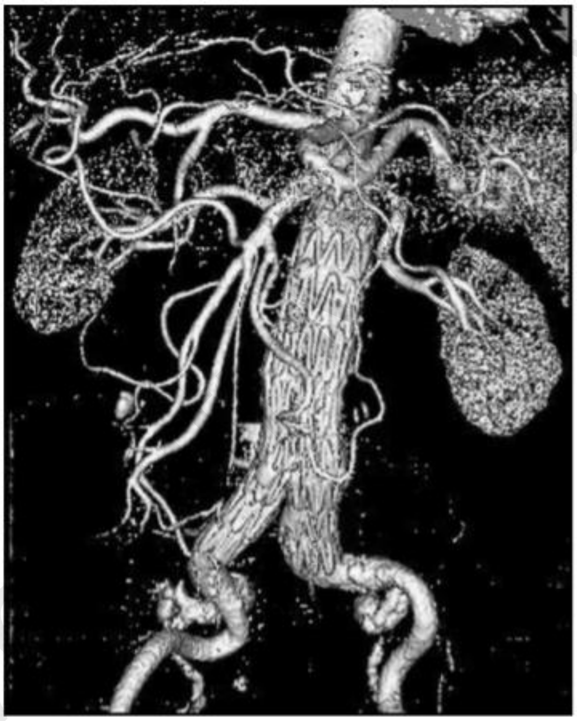

Figure 2-4: Surface shaded display of aortic branches of renal.

2.4.2 Volume rendering (VR)

While the pixel's value in SSD depends on the distance of the volume from the screen, VR depends on the pixel's value encasing densitometric information. VR provides both densitometric and spatial information on the reconstructed images through three main steps including sampling, classification and composing. volume sampling is implemented for further integration, assignment of color and opacity is done through classification and composing is the arrangement of samples as defined in 3D volume (Khan et al., 2018). An opacity function curve correlates the voxel value with its opacity. The pixel value of the screen will be obtained from the contribution of the opacities of the points along the ray. Only voxels, the values of which lay within selected interval, are represented. The

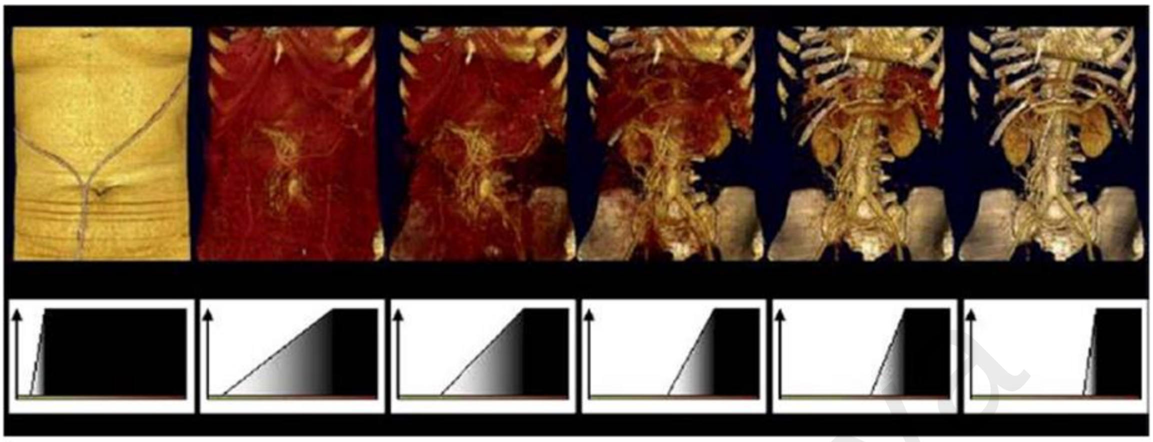

Figure 2-5: Volume rendering image by varying the window level

voxels that are outside of this interval are transparent. In CT, voxel value corresponds to the attenuation in Hounsfield Units (HU). The shape of the curve defines the visibility of the structures in keeping with their attenuation (Figure 2-5). The pixel value can be represented through grey or a color scale, which may enhance the depth perception and the densitometric information.

2.4.3 Advantage of 3D vs 2D image for diagnostics and therapy

The reconstruction of 3D images for diagnostics and therapy was explored by various ROI study and have shown to be advantageous. In evaluation of ureteral tumors, 3D imaging of the gastrointestinal tract allowed detection of transitional cell carcinoma (TCC) otherwise unseen (Raman et al., 2013). Axial images in 2D often less perceptible to subtle difference of two of caliber structures. VR images help to emphasize understated structures and narrowing sites. VR images characterize both ureter structure entirely in a single image plane resulting in easier detection of subtle sites and proximal dilatation. VR images can also be referenced with the source 2D images in axial plane to check for signs of subtle wall thickening or normal enhancement involving small portion of ureteral circumference. (Raman et al., 2013).

3D images improve depth perception and spatial orientation compared to 2D. In laparoscopic surgery, 3D vision system has shown statistically significant improvement of time and error rate when performing advanced laparoscopic skills (Byrn et al., 2007). The improved depth perception offered by 3D allow greater recognition and differentiation which leads to enhance visual performance and motor skills. The overall time required to perform using 3D images was 24.6% shorter than using 2D images (Buia et al., 2017). Ultimately it has the potential to lower operation times and patient morbidity.

Quantitative measurement such as segment length can be done in 3D. Although 2D axial CT images still remain the standard reference in most of the pre-operative situations such as measurements of aortic wall calcification, 3D CT images proven to be practical for visualization of stent graft and the arterial branches relationship (Sun, 2006). VR technique was able to exhibit all the anatomical structures using all the volumetric dataset. Measurement of aortic wall size can be done to monitor non-calcified atherosclerosis since the voxel represent each structure proportionately. This reduces the need for invasive procedure for screening and checkup compared to Angiography system.

Reproducibility of studies in 3D is superior compared to 2D. A 3D dynamic contrast-enhanced ultrasound (DCE-US) study of liver metastasis shows that 3D measurements between pairs of repeated acquisitions can be repeated and reproduced with was excellent result (El Kaffas et al., 2017). 3D imaging manages the challenges of finding the same viewing plane in longitudinal applications by capturing images volumetric data instantaneously. This can yield biased quantitative results since tumors are highly heterogeneous and susceptible to sampling errors in 2D due to plane-to-plane perfusion variation.

2.5 Visualization Toolkit (VTK)

VTK consists of a C++ class library and several interpreted interface layers including Java and Python. VTK supports a wide variety of visualization algorithms including scalar, vector, tensor, texture, and volumetric methods. VTK has an extensive information visualization framework, has a suite of 3D interaction widgets, supports parallel processing, and integrates with various databases and GUI toolkits. VTK is cross-platform and runs on Linux, Windows, Mac and Unix platforms. The essence of VTK lies on its implementation as a C++ toolkit, requiring users to build applications by combining various objects into an application. The system also supports automated wrapping of the C++ core into Python, Java and Tcl, so that VTK applications may also be written using these programming languages.

In VTK image processing, VTK image data is categorized into type such structural point, unstructured points, structured grid, unstructured grids, polygon data and line grid (Wee et al., 2011). Different filter will be designated for processing each structure type. However, most filter have similar primary process flow (Fig. 2-6). The basic class in VTK library is vtkObject, which provides basic flow for visualization method. Source data is applied to develop data layer. vtkSource is a derived class from vtkObject and is parent class of vtkFilter, which is a derived class from vtkSource. vtkFilter implements raw data processing through filters, transform it into module that can be directly processed using an algorithm model. vtkMapper is also a derived class of vtkObject; it will undergo various filtering process and be translated into geometric data as interface in connecting original data and image data. Mapper function is to achieve a mapping from data to graphics. Actor plays role of visual realization using Mapper results to show the resulting image. Actor can manipulate image display properties through vtkproperty to make s realistic image to be presented. Next step is to display an image on computer screen; this task is accomplished by Render & Render Window. Render

Window Interactor is used to allow use to interact with the image. Rendered image can be rotated by mouse to see images from all angles. All the elements discussed are presented in VTK and to be used in pipeline mechanism. It can process most of structural data type and provides a various type of data modification or processing for several corresponding classes. In this research, vtkSource class object creates a data source; then, vtkFilter class or its derived class processes data source object. It follows by vtkMapper class or its derived classes mapped pixel data, and then vtkActor class or a derived class represents a drawing entity. Finally, image is rendered by vtkRenderer. By using VTK for 3D reconstruction developments, visualisations of 3D image can be achieved.

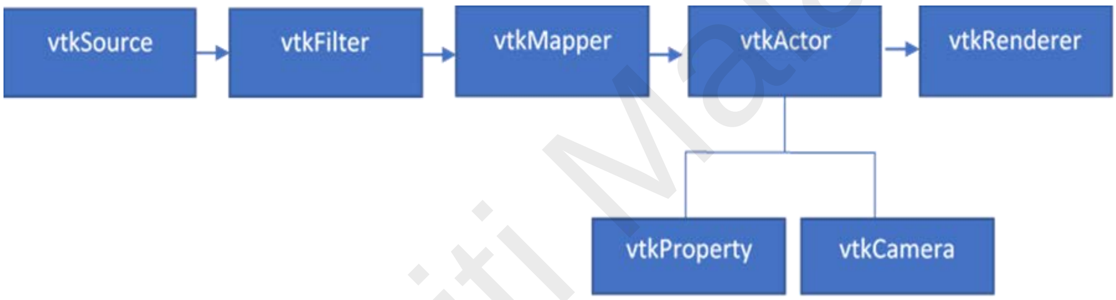

Figure 2-6: Basic Flow of Visualisation Toolkit (VTK) pipeline mechanism

CHAPTER 3: METHODOLOGY

3.1 Introduction

This chapter describes in detail the methodology of reconstructing a 3D cardiac image from 2D CT Scan. Section 3.2 describes the 2D CT Scan images used for this research. The software and hardware used is defined in Section 3.3 and Section 3.4. Section 3.5 explained the 3D rendering process workflow with the coding for surface rendering technique, volume rendering technique, length measurement in VTK as well as RadiAnt 3D imaging process is detailed in Section 3.6.

3.2 CT Image Acquisition in 2D Slices

The image for the 3D reconstruction in this paper is 2D image slices acquire using CT Scan with CTA protocol. There are 395 image slices of 512 x 512-pixel size with 0.3mm slice width. Patient heart rate was recorded at 87bpm to 89bpm during scanning. Images file are in DICOM format and reformatted into JPEG with grayscale color. Images then renamed in order of img001 to img395.

3.3 Software

*(a) **Visualisation Toolkit 5.6.1***

The Visualization Toolkit (VTK) is open source software for manipulating and displaying scientific data. It comes with state-of-the-art tools for 3D rendering, a suite of widgets for 3D interaction, and extensive 2D plotting capability. The source codes written for the 3D reconstruction are in C++ and based on VTK pipeline mechanisms using the built-in library.

*(b) **CMake 2.8.2***

CMake is a meta build system that uses scripts called CMakeLists to generate files for a specific environment. It is used for managing the build process of software using a compiler-independent method. The software able to support directory hierarchies and applications that depend on multiple libraries. For this research paper, CMake is used to compile the source code and build the binaries file to reconstruct the 3D image.

(c) ***Microsoft Visual Studio 2008***

Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to build the solution and run debug to generate the 3D reconstructed images of CT scan.

(d) ***RadiAnt DICOM Viewer (64-Bit)***

RadiAnt is a PACS DICOM viewer for medical images. The software can display studies obtained from various imaging modalities in DICOM format. It has basic tools for manipulation and measurement of images such as zooming and panning, brightness adjustment, window setting alteration, and segment length measurement. The software can also volume render a 3D image from a set of 2D slices CT scan images. The trial version use in this paper are free for usage for 3 month and have the same functionality with the paid version.

3.4 Hardware

All 3D reconstructions are run through ILLEGEAR Raven SE laptop with Intel® Core™ i5-103000H CPU @2.50GHz CPU, Windows 10 64-Bit Operation System, Memory RAM of 16.0GB, and NVIDIA GeForce GTX 1650 GPU card.

3.5 3D Rendering Process

CT Scan image slices are converted from DICOM into JPEG format with grayscale coloring before being read by VTK using *vtkJPEGReader* function. The rendering process adhered to the 3D rendering pipeline mechanisms specified in VTK User Guide 2010 shown in figure 3.3-1.

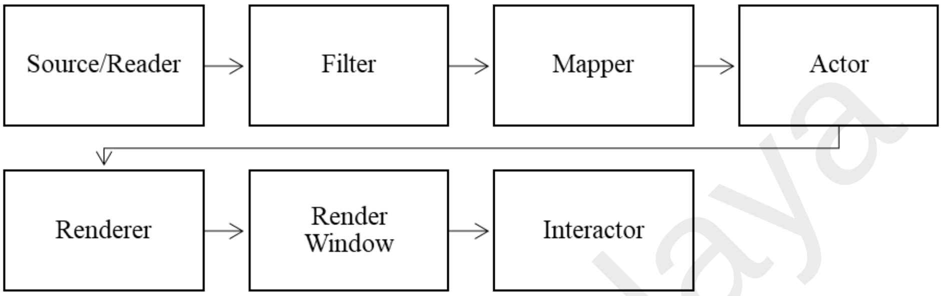

Figure 3-1: VTK rendering pipeline mechanisms

(a) ***Source/Reader***

Source algorithms build data by reading or constructing one or more data objects (reading objects vs procedural source objects). vtkJPEGReader is used to read the source since the 2D CT scan images are in JPEG format. The inheritance diagram for vtkJPEGReader can be seen in Figure 3-2.

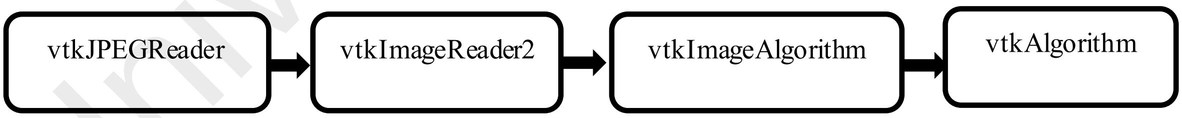

Figure 3-2: Inheritance diagram for *vtkJPEGReader*

(b) ***Filte*r**

Filters consume one or multiple objects data and generate it on defined output. It modifies the 2D images into a 3D model that is smooth and clear. Visualization pipelines can contain loops, but the output of a filter cannot be directly connected to its input.

(c) Mapper

Mappers (or in some cases, specialized actors) take the data and convert it into a visual representation that is displayed by the rendering engine.

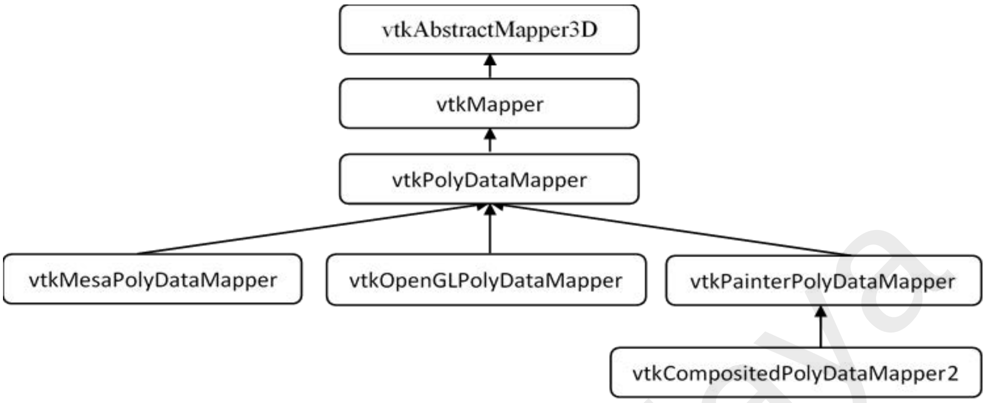

Figure 3-3: Inheritance Diagram for vtkPolyDataMapper

(d) Actor

An actor is the most common type of vtkProp3D. Like other concrete subclasses of vtkProp3D, vtkActor serves to compile main attributes for rendering such as surface properties (e.g., ambient, diffuse, and specular color), representation (e.g., surface or wireframe), texture maps, and/or a geometric definition (a mapper).

(e) Renderer

The vtkRenderer is responsible for managing the rendering process for the scene. Multiple vtkRenderer objects can be used together in a single vtkRenderWindow. These renderers may render into different rectangular regions (known as viewports) of the render window or may be overlapping.

(f) Render Window

The vtkRenderWindow provides a connection between the operating system and the VTK rendering engine. Platform specific subclasses of vtkRenderWindow are responsible for opening a window in the native windowing system on the computer and managing the display process.

(g) ***Interactor***

The vtkRenderWindowInteractor is responsible for processing mouse, key, and timer events and routing these through VTK's implementation of the command / observer design pattern. A vtkInteractorStyle listens for these events and processes them in order to provide motion controls such as rotating, panning and zooming. The vtkRenderWindowInteractor automatically creates a default interactor style that works well for 3D scenes.

3.6 Coding

This section describes the programming code to reconstruct a 3D cardiac ROI from a 2D image slices. Various visualizing results were achieved with different code in this paper. For the 3D reconstructed images to be viewed, volume rendering technique, and surface rendering technique will be applied.

3.6.1 Surface Rendering

The volume rendering technique shall include related VTK libraries to allow reconstruction command to be executed. The renderer, the render window and the interactor are created to allow 3D object rendered in the specified window and enable mouse interaction to zoom and flip the object.

```
vtkSmartPointer<vtkRenderer> aRenderer =
  vtkSmartPointer<vtkRenderer>::New();
vtkSmartPointer<vtkRenderWindow> renWin =
  vtkSmartPointer<vtkRenderWindow>::New();
renWin->AddRenderer(aRenderer);

vtkSmartPointer<vtkRenderWindowInteractor> iren =
  vtkSmartPointer<vtkRenderWindowInteractor>::New();
iren->SetRenderWindow(renWin);
```

A series of 2D images from cardiac CTA are converted into JPEG to allow vtkJPEGReader to read the image. The coding to read the image dataset is same for both surface rendering and volume rendering. The source file of image data is set through SetFilePrefix scan the file root name: img in series. SetDataExtent is configured to allow image size of 512x 512 pixels and 245 image slices to be included in the reconstruction. Only 245 image slices are chosen for faster reconstruction time and avoid memory error. Data spacing between image slices is set through SetDataSpacing.

```
vtkJPEGReader *v16 = vtkJPEGReader::New();

v16->SetDataExtent(0,511,0,511,1,245);
v16->SetFilePrefix("C:\\Users\\Zahid\\Desktop\\Research
Project\\Project Paper Raw Data (Cardiac)\\cardiacangioLIANTO1\\img");

v16->SetFilePattern("%s%d.jpg");
v16->SetDataSpacing (1, 1, 1);
v16->SetDataOrigin(0.0, 0.0, 0.0);
v16->Update();
```

An isosurface, or contour value allow only ROI surfaces to be rendered. The value of 150 is chosen since its correspond to the value of artery surface in the cardiac ROI. Once developed, a vtkPolyDataNormals filter is used to create normals for smooth surface shading during rendering.

```
vtkSmartPointer<vtkContourFilter> skinExtractor =
  vtkSmartPointer<vtkContourFilter>::New();
skinExtractor->SetInputConnection(v16->GetOutputPort());
skinExtractor->SetValue(0, 150);

vtkSmartPointer<vtkPolyDataNormals> skinNormals =
  vtkSmartPointer<vtkPolyDataNormals>::New();
skinNormals->SetInputConnection(skinExtractor->GetOutputPort());
skinNormals->SetFeatureAngle(60.0);

vtkSmartPointer<vtkPolyDataMapper> skinMapper =
  vtkSmartPointer<vtkPolyDataMapper>::New();
skinMapper->SetInputConnection(skinNormals->GetOutputPort());
skinMapper->ScalarVisibilityOff();

vtkSmartPointer<vtkActor> skin =
  vtkSmartPointer<vtkActor>::New();
skin->SetMapper(skinMapper);
    skin->GetProperty()->SetDiffuseColor(1, 1, 1);
```

An outline command provides context around the data.

```
vtkSmartPointer<vtkOutlineFilter> outlineData =
  vtkSmartPointer<vtkOutlineFilter>::New();
outlineData->SetInputConnection(v16->GetOutputPort());

vtkSmartPointer<vtkPolyDataMapper> mapOutline =
  vtkSmartPointer<vtkPolyDataMapper>::New();
mapOutline->SetInputConnection(outlineData->GetOutputPort());

vtkSmartPointer<vtkActor> outline =
  vtkSmartPointer<vtkActor>::New();
outline->SetMapper(mapOutline);
outline->GetProperty()->SetColor(0,0,0);
```

To create and initial view of the data, SetFocalPoint and SetPosition form a vector position and direction of the camera.

```
vtkSmartPointer<vtkCamera> aCamera =
  vtkSmartPointer<vtkCamera>::New();
aCamera->SetViewUp (0, 0, -1);
aCamera->SetPosition (0, 1, 0);
aCamera->SetFocalPoint (0, 0, 0);
aCamera->ComputeViewPlaneNormal();
aCamera->Azimuth(30.0);
aCamera->Elevation(30.0);
```

Actors are added to the renderer. An initial camera view is created. The Dolly() method moves the camera towards the FocalPoint, thereby enlarging the image.

```
aRenderer->AddActor(outline);
aRenderer->AddActor(skin);
aRenderer->SetActiveCamera(aCamera);
aRenderer->ResetCamera ();
aCamera->Dolly(1.5);
```

Background color is set by configuring the RGB value, Rendering window is set according to the image pixel size.

```
aRenderer->SetBackground(.2, .3, .4);
renWin->SetSize(512, 512);
```

3.6.2 Volume Rendering

The volume rendering technique shall include related VTK libraries to allow reconstruction command to be executed. The renderer, the render window and the interactor are created to allow 3D object rendered in the specified window and enable mouse interaction to zoom and flip the object

```
vtkSmartPointer<vtkRenderer> ren =
  vtkSmartPointer<vtkRenderer>::New();
vtkSmartPointer<vtkRenderWindow> renWin =
  vtkSmartPointer<vtkRenderWindow>::New();
renWin->AddRenderer(ren);
vtkSmartPointer<vtkRenderWindowInteractor> iren =
  vtkSmartPointer<vtkRenderWindowInteractor>::New();
iren->SetRenderWindow(renWin);
```

A read command is used to read a series of 2D image slices that composes the volume. SetDataExtent configured to the slice dimensions which is 512 x 512 pixels and number of slices included in this reconstruction are 300. The reader uses the FilePrefix in combination with the slice number to construct filenames using the format FilePrefix.%d and scan the file root name: img in series.

```
vtkJPEGReader *v16 = vtkJPEGReader::New();
v16->SetDataExtent(0,511,0,511,1,300);
v16->SetFilePrefix("C:\\Users\\Zahid\\Desktop\\Research
Project\\Project Paper Raw Data (Cardiac)\\cardiacangioLIANTO1\\img");

v16->SetFilePattern("%s%d.jpg");
v16->SetDataSpacing (1, 1, 1);
v16->SetDataOrigin(0.0, 0.0, 0.0);
v16->Update();
```

A ray-cast mapper is used to display volume by ray-cast alpha compositing. A compositing function is needed to do the compositing along the ray. The color transfer function maps voxel intensities to colors. The voxel intensities are a function of different tissue densities. The color are based on RGB setting for each voxel intensity.

```
vtkSmartPointer<vtkVolumeRayCastCompositeFunction> rayCastFunction =
 vtkSmartPointer<vtkVolumeRayCastCompositeFunction>::New();
 vtkSmartPointer<vtkVolumeRayCastMapper> volumeMapper =
 vtkSmartPointer<vtkVolumeRayCastMapper>::New();
```

```
volumeMapper->SetInput(v16->GetOutput());
volumeMapper->SetVolumeRayCastFunction(rayCastFunction);
```

```
vtkSmartPointer<vtkColorTransferFunction>volumeColor =
  vtkSmartPointer<vtkColorTransferFunction>::New();
volumeColor->AddRGBPoint(50,    1.0, 0.2, 0.2);
volumeColor->AddRGBPoint(150,  1.0, 0.5, 0.0);
volumeColor->AddRGBPoint(200, 1.0, 1.0, 1.0);
```

The opacity transfer function is used to control the opacity of different tissue types. For CTA the emphasis is to show the artery in cardiac ROI.

```
vtkSmartPointer<vtkPiecewiseFunction> volumeScalarOpacity =
  vtkSmartPointer<vtkPiecewiseFunction>::New();
volumeScalarOpacity->AddPoint(50,  0.05);
volumeScalarOpacity->AddPoint(150, 1.00);
volumeScalarOpacity->AddPoint(200, 1.00);
```

The gradient opacity function is used to decrease the opacityin the "flat" regions of the volume while maintaining the opacity at the boundaries between tissue types. The gradient is measured as the amount by which the intensity changes over unit distance.

```
vtkSmartPointer<vtkPiecewiseFunction> volumeGradientOpacity =
  vtkSmartPointer<vtkPiecewiseFunction>::New();
volumeGradientOpacity->AddPoint(0,   0.0);
volumeGradientOpacity->AddPoint(90,  0.5);
volumeGradientOpacity->AddPoint(100, 1.0);
```

Filters such colours is also applied using the vtkVolumeProperty where it attaches the colour and opacity functions to volume. For a high-quality rendering to be achieved, the interpolation is set to linear. ShadeOn is used to turn on directional lighting, which enhance the appearance of the 3D rendered view. However, it should be noted that the quality of the shading depends on how accurately the volume can be calculated. Hence, the gradient estimation will be poor for data with noise.

The impact of the shading is influence by the SetAmbient, SetDiffuse, and SetSpecular. By decreasing the SetAmbient value, and increasing the other two, the shading of the 3D image will decrease whereas increasing the SetAmbient value and decreasing the SetDiffuse and SetSpecular would increase the impact of shading.

```
vtkSmartPointer<vtkVolumeProperty> volumeProperty =
  vtkSmartPointer<vtkVolumeProperty>::New();
volumeProperty->SetColor(volumeColor);
volumeProperty->SetScalarOpacity(volumeScalarOpacity);
volumeProperty->SetGradientOpacity(volumeGradientOpacity);
volumeProperty->SetInterpolationTypeToLinear();
volumeProperty->ShadeOn();
volumeProperty->SetAmbient(0.3);
volumeProperty->SetDiffuse(0.7);
volumeProperty->SetSpecular(0.7);
```

To control positions and oreintation of volume in world coordinates, vtkVolume is used. The volume is finally added to the renderer with AddViewProp.

```
vtkSmartPointer<vtkVolume> volume =
  vtkSmartPointer<vtkVolume>::New();
volume->SetMapper(volumeMapper);
volume->SetProperty(volumeProperty);

  ren->AddViewProp(volume);
```

3.6.3 Measurement of structure length

This programming code enables the measurement of length between two points in the rendering window. The size of one pixel is defined as 0.264583333 *mm* based on the raw image size is 512 x 512 pixel with the assumption of 96 dot per inch (dpi).

```
double milimeter;
milimeter=length * 0.264583333 * 2;

char text[120];
sprintf( text, "The length of two points is %5.5f cm", milimeter);
textMapper->SetInput( text );
textActor->SetPosition(10, 10);
           textActor->VisibilityOn();


style->OnRightButtonDown();
       break;
case vtkCommand::LeftButtonReleaseEvent:
       if (MouseMotion == 0)
       {
           int *pick = iren->GetEventPosition();
           picker->Pick((double)pick[0], (double)pick[1], 0.0, ren1);
       }
       style->OnLeftButtonUp();
       break;
   case vtkCommand::MouseMoveEvent:
       MouseMotion = 1;
       style->OnMouseMove();
       break;
```

A read command is used to read a series of 2D image slices that composes the volume. SetDataExtent configured to the slice dimensions which is 512 x 512 pixels and number of slices included in this reconstruction are 395. The reader uses the FilePrefix in combination with the slice number to construct filenames using the format FilePrefix.%d and scan the file root name: img in series.

```
vtkJPEGReader *reader = vtkJPEGReader::New();

reader->SetDataExtent(100,511,100,511,1,395);
reader->SetFilePrefix("C:\\Users\\Zahid\\Desktop\\Research
Project\\Project Paper Raw Data (Cardiac)\\cardiacangioLIANTO1\\img");

reader->SetFilePattern("%s%d.jpg");
reader->SetDataSpacing (1, 1, 1);
reader->SetDataOrigin(0.0, 0.0, 0.0);
reader->Update();
```

The opacity transfer function is defined through a vtkPiecewiseFunction. The objective in designing the opacity transfer function is to reveal as much as possible of the internal structures of each dataset. Opacity transfer function value of 1024+400 correspons to the internal pulmonary artery.

```
vtkPiecewiseFunction *opacityTransferFunction =
vtkPiecewiseFunction::New();
      opacityTransferFunction->AddPoint(1024+50,0.0);
      opacityTransferFunction->AddPoint(1024+400,0.8);
```

The vtkColorTransferFunction add colours to the rendered image. AddRGBPoint is used to set the value of (R, G, B) of the 3D image.

```
vtkColorTransferFunction *colorTransferFunction =
vtkColorTransferFunction::New();
colorTransferFunction->AddRGBPoint(1024+50.0, 0.0, 0.0, 0.0);
colorTransferFunction->AddRGBPoint(1024+400.0, 0.8, 0.8, 0.0);
```

To represent the properties of volume rendering, vtkVolumeProperty is used. The brighness of the surrounding the render object configured through SetAmbient. SetSpecularPower set the contrast of the image. The volume rendering process is developed using vtkVolumeRayCastMapper. The vtkPlane command is used to project corss-sectional view plane across the image.

```
vtkVolumeProperty *volumeProperty = vtkVolumeProperty::New();
volumeProperty->SetColor(colorTransferFunction);
volumeProperty->SetScalarOpacity(opacityTransferFunction);
volumeProperty->ShadeOn();
volumeProperty->SetInterpolationTypeToLinear();
volumeProperty->SetAmbient(10);
volumeProperty->SetDiffuse(8);
volumeProperty->SetSpecular(1);
volumeProperty->SetSpecularPower(80);


vtkVolumeRayCastCompositeFunction *compositeFunction =
vtkVolumeRayCastCompositeFunction::New();

vtkPlane *plane=vtkPlane::New();

vtkVolumeRayCastMapper *volumeMapper = vtkVolumeRayCastMapper::New();
volumeMapper->SetVolumeRayCastFunction(compositeFunction);
volumeMapper->SetInputConnection(readerImageCast->GetOutputPort());
volumeMapper->AddClippingPlane(plane);

volume = vtkVolume::New();
volume->SetMapper(volumeMapper);
volume->SetProperty(volumeProperty);
volume->PickableOff ();
```

A filter is then applied to create a an outline around the 3D render image. the outline will create a box in 3D. The color of the box are set in RGB configuration.

```
vtkOutlineFilter *outline = vtkOutlineFilter::New();
      outline->SetInputConnection( readerImageCast->GetOutputPort() );
vtkPolyDataMapper *outlinemapper = vtkPolyDataMapper::New();
      outlinemapper->SetInputConnection( outline->GetOutputPort() );
vtkActor * outlineactor = vtkActor::New();
      outlineactor->SetMapper( outlinemapper );
      outlineactor->GetProperty()->SetColor( 0.0, 1, 1);
```

As a guide to annotate between two point of measurement, vtkSphereSource command is used to generate a sphere with a radius of 5 mm. Two sphere are generated in the rendering window for the 2 point of length measurement using the same line of coding. The color of the sphere are set using RGB setting.

```
vtkSphereSource *sphere1 = vtkSphereSource::New();
     sphere1->SetRadius(5);
vtkPolyDataMapper *sphereMapper1 = vtkPolyDataMapper::New();
     sphereMapper1->SetInput(sphere1->GetOutput());
```

```
     sphereMapper1->GlobalImmediateModeRenderingOn();
sphereActor1 = vtkActor::New();
     sphereActor1->SetMapper(sphereMapper1);
       sphereActor1->GetProperty()->SetDiffuseColor(1,0,0);

vtkSphereSource *sphere2 = vtkSphereSource::New();
     sphere2->SetRadius(5);
vtkPolyDataMapper *sphereMapper2 = vtkPolyDataMapper::New();
     sphereMapper2->SetInput(sphere2->GetOutput());
     sphereMapper2->GlobalImmediateModeRenderingOn();
sphereActor2 = vtkActor::New();
     sphereActor2->SetMapper(sphereMapper2);
       sphereActor2->GetProperty()->SetDiffuseColor(1,0,0);
```

A line connecting the 2 point are also created using vtkLineSource where the resolution of the line is set aat 200 and the color is green by setting the RGB vallue.

```
line=vtkLineSource::New();
        line->SetPoint1(0, 0, 0);
        line->SetPoint2(0, 0, 0);
        line->SetResolution(200);
vtkPolyDataMapper *lineMapper = vtkPolyDataMapper::New();
        lineMapper->SetInput(line->GetOutput());
lineActor = vtkActor::New();
        lineActor->SetMapper(lineMapper);
        lineActor->GetProperty()->SetDiffuseColor(0,1,0);
```

A cell picker will shoot a ray into a 3D scene and return information about the first object that the ray hits. It returns point and cell information, plus the normal of the surface that was intersected at the pick position. For volumes and images, it also returns coordinates for the point and the cell that were picked.

```
PickCommand* pickObserver = PickCommand::New();
picker = vtkCellPicker::New();
picker->AddObserver( vtkCommand::EndPickEvent, pickObserver );
```

The resulting pick is displayed in the render window using the vtkTextMapperCommand. The properties of text can be configured using tprop command where font type, size, boldness and color can be set.

```
textMapper = vtkTextMapper::New();
vtkTextProperty *tprop = textMapper->GetTextProperty();
  tprop->SetFontFamilyToArial();
  tprop->SetFontSize(20);
  tprop->BoldOn();
  tprop->SetColor(1, 0, 0);
textActor = vtkActor2D::New();
```

```
     textActor->VisibilityOff();
     textActor->SetMapper(textMapper);
```

To allow the usage of mouse and keyboard, vtkInteractorStyleTrackBallCamera is used control the camera.

```
vtkInteractorStyleTrackballCamera *style =
vtkInteractorStyleTrackballCamera::New();
vtkCallbackCommand * pickerCommand = vtkCallbackCommand::New();
     pickerCommand->SetClientData(style);
     pickerCommand->SetCallback(PickerInteractionCallback);
     style->AddObserver(vtkCommand::LeftButtonPressEvent,
pickerCommand);
     style->AddObserver(vtkCommand::MouseMoveEvent, pickerCommand);
     style->AddObserver(vtkCommand::LeftButtonReleaseEvent,
pickerCommand);
     style->AddObserver(vtkCommand::RightButtonPressEvent,
pickerCommand);

    iren->SetInteractorStyle(style);
    iren->SetPicker(picker);

      sphereActor1->VisibilityOff();
      sphereActor2->VisibilityOff();
      lineActor->VisibilityOff();

  vtkTIPWCallback *myCallback = vtkTIPWCallback::New();
  myCallback->Plane = plane;
  myCallback->Volume = volume;

  vtkImplicitPlaneWidget *planeWidget = vtkImplicitPlaneWidget::New();
      planeWidget->SetInteractor(iren);
      planeWidget->SetPlaceFactor(1.25);
      planeWidget->GetPlaneProperty()->SetOpacity  ( 0.1 ) ;
      planeWidget->GetOutlineProperty()->SetColor(0,0,1);
      planeWidget->SetOrigin(xx/2,yy/2,zz/2);
      planeWidget->SetInput((vtkDataSet *)readerImageCast-
>GetOutput());
      planeWidget->PlaceWidget();
      planeWidget->On();
      planeWidget-
>AddObserver(vtkCommand::InteractionEvent,myCallback);

vtkCamera *cam1 = ren1->GetActiveCamera();
   cam1->Zoom(1.5);
```

The background color can be changed using RGB value using SetBackground and the window size can be set using the SetSize command. SetKeyCode define the function of ‘i’ to turn the visibility of the plane.

```
ren1->AddVolume(volume);
ren1->AddActor(sphereActor1);
ren1->AddActor(sphereActor2);
ren1->AddActor(lineActor);
ren1->AddActor2D(textActor);
```

```
ren1->AddActor( outlineactor );
ren1->SetBackground(0.2, 0.3, 0.4);
renWin->SetSize(600, 600);
renWin->Render();
iren-> SetDesiredUpdateRate(99);
iren->SetKeyCode('i');
```

3.6.4 3D Reconstruction using RadiAnt DICOM Viewer

RadiAnt DICOM Viewer is used to read the 2D image slices in DICOM format. In 2D, images can also be viewed in sagittal and coronal plane as well as the existing axial plane view. 3D images can be reconstructed automatically where object manipulation such as zoom, rotate and window level can be varied. Length measurement can also be done in 2D and 3D images.

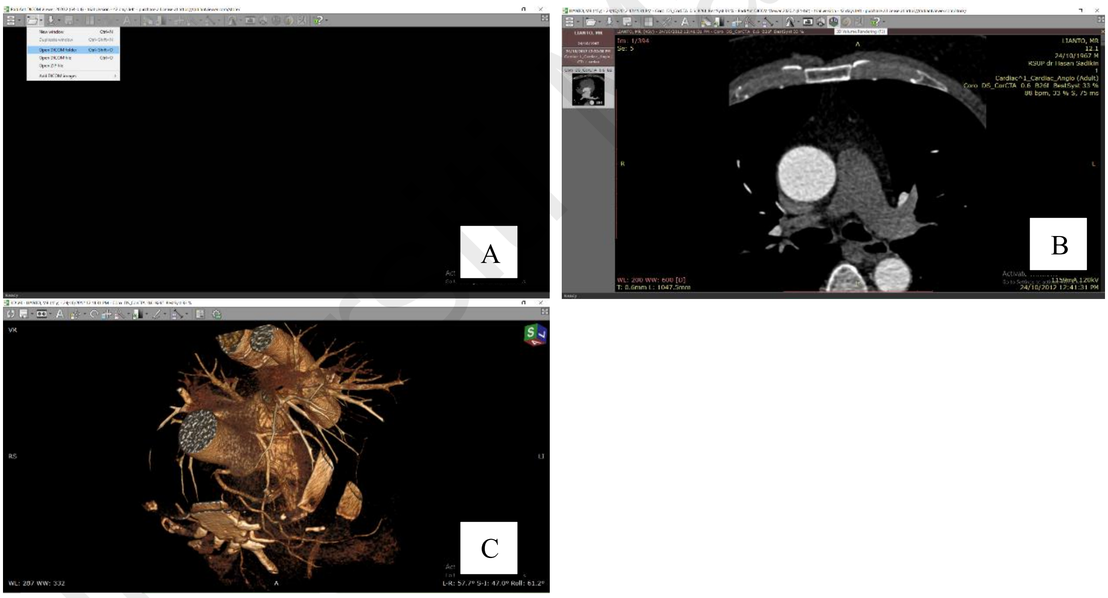

Figure 3-4: Reconstruction 3D image through volume rendering method. A) Scan the folder containing series of DICOM files. B) Select 3D volume rendering on the taskbar. C) Resulting 3D image from the DICOM files

CHAPTER 4: RESULTS

It was possible to display the heart image in 3D using VTK in the desired plane. The VTK libraries and programming allow the reconstruction of 3D cardiac images from 2D CT Scan Slice image performed with CTA protocol. The surface rendering and volume rendering technique applied to reconstruct the images yield different result since the number of datasets considered for each technique varies. This section will show the result of both rendering technique as well as the quality of VTK reconstruction compared RadiAnt DICOM Viewer which is readily available in the market.

4.1 3D reconstruction image using SR and VR technique

The image shown in figure 4-1 shows the heart reconstructed in 3D using surface rendering technique. There is only the surface of the artery visible and no discerning volume data in the image. Only 245 of 394 available image slices is used to reconstruct the image hence image only shows the 1/3 of the cardiac scan. Since surface rendering only able to visualize only one property of the heart, the program is focused on rendering the artery and all other structure that have the same attenuation will be illuminated.

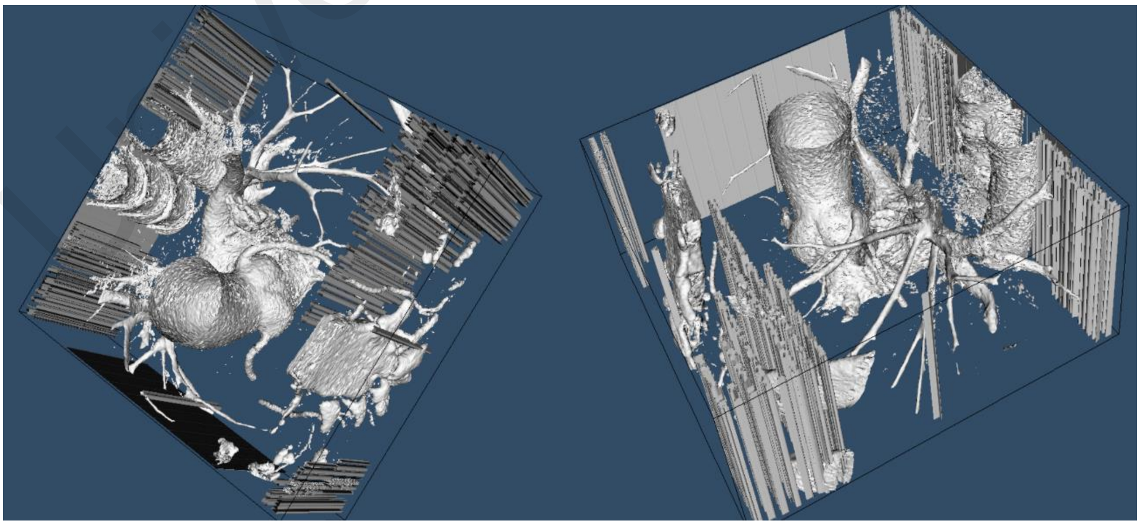

Figure 4-1: Surface rendering image of cardiac

The 3D image shown (Fig. 4-2) is generated by using the volume rendering technique.

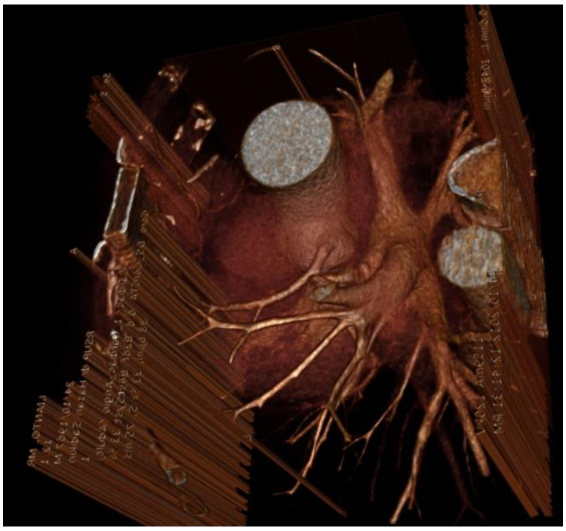

Figure 4-2: 3D images reconstructed using volume rendering technique

The image consists of volumetric data generated through reconstructing the 2D image slices. The vascular structure inside the heart such as aorta, superior vena kava, left pulmonary artery and right pulmonary artery are easily identified in the rendering.

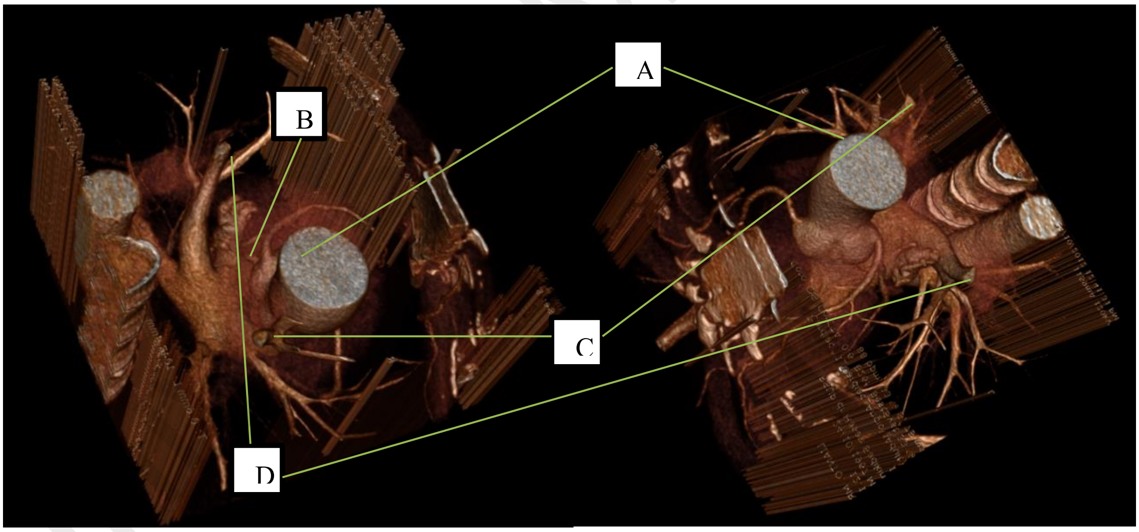

Figure 4-3: Volume rendering image showing the vascular and soft structure with A) aorta B) superior vena kava C) left pulmonary artery D) right pulmonary artery

Both rendering techniques have yielded a 3D image from a series 2D image slices of the cardiac ROI. However, the quality of image using volume rendering is better compared to image using surface rendering. The outer layer of the artery to has shown to be uneven with only the aorta appear to have hollow volume while the other artery appears to be solid object. Few arteries have also shown breakage in the surface rendering image. Volume rendering images shows a better structure of the heart artery with identifiable vascular structure. Assigning different color to different image intensity allow differentiation between adjacent tissue.

The reconstruction time for both rendering also varies. The number of 2D slices image needed to reconstruct was set at 100 for both techniques. The time taken was from the program start debugging and the 3D image to appear. The image rendering time are significantly faster in VR compare to SR (Table 4.1). The average rendering time for SR is 52.39±0.24s while the average rendering time for VR is 2.63±0.11s for the same number of 2D image slices.

Table 4.1: Rendering time of Surface Rendering vs Volume Rendering

Rendering Technique	Rendering Time 1 (s)	Rendering Time 2 (s)	Rendering Time 3 (s)	Average Rendering Time (s)
SR	52.36	52.17	52.64	52.39±0.24
VR	2.76	2.58	2.55	2.63±0.11

4.2 Comparing the 3D reconstruction using VTK vs RadiAnt DICOM Viewer

The VTK and RadiAnt was able to reconstruct 3D images form the same set of 2D images. For VTK, images were converted to JPEG whereas RadiAnt able to read the images direct from DICOM format. RadiAnt offer volume rendering as its 3D reconstruction technique. It was done automatically through selecting the relevant taskbar. Image reconstruction was instant and compared to VTK reconstruction where

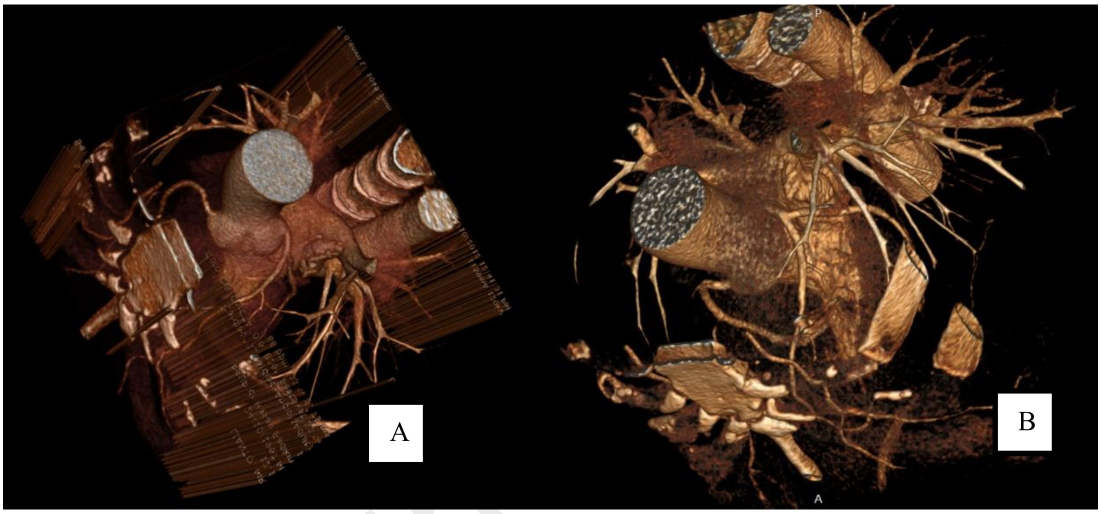

Figure 4-4: A) 3D VR image using VTK. B) 3D VR image using RadiAnt.

there is a noticeable time delay between running debug and the render window start to appear. Both images have shown (Fig. 4.4) identifiable and discerning vascular structure similar in nature. Other tissue in the cardiac ROI such as sternum are visible in both. RadiAnt was able to crop out the patient info from the image whereas VTK reconstruct it within the image. This cause an unobstructed view when using RadiAnt compared to VTK whereas image need to be rotated to find a better viewing angle. This stems from RadiaAnt ability reading DICOM images where filtering out the patient information is possible.

In VTK, opacity level can be changed to show the outer structure of the heart. This function is similar to adjusting the window level in RadiAnt. The outer layer of both images shows the heart outline with Radiant able to show better texture and shading (Fig 4-5).

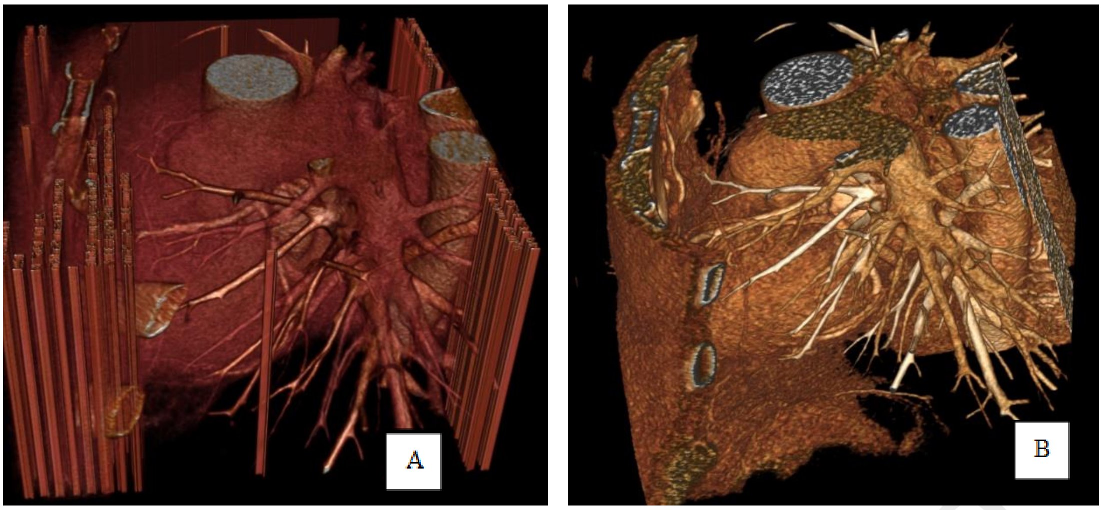

Figure 4-6: A) The effect of changing opacity function to show the outer layer of the heart in VTK. B) The effect of varying window level in RadiAnt

Figure 4-5: Aorta diameter measurement using RadiAnt.

VTK and RadiAnt allow for measurement between two points in the image. For the purpose of comparison aorta diameter was measured for both VTK and RadiAnt 3D images. Measurement in RadiAnt is done by selecting 'Measurement and tools' in the task bar and clicking the two points where the length needs to be obtained in the image. While in VTK, the two point can be selected directly in the render window represented with red sphere and the value appears on the bottom left corner of the render window.

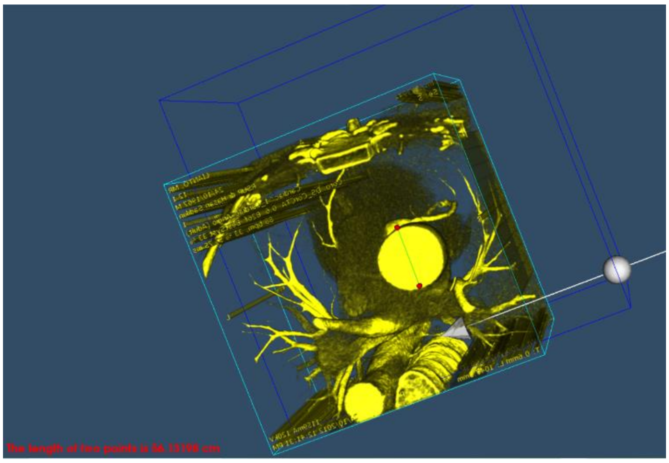

Figure 4-7: Aorta diameter measurement using VTK

For RadiAnt the diameter is 3.76 *cm* (Fig. 4-6) while VTK measured it at 56.14 *cm* (Fig. 4-7). The difference can be due to the pixel length assumption in the VTK code since the pixel length data cannot be obtained. While measurement differs in both rendering, VTK has the capability to incorporate the functionality of measurement inside its coding.

VTK allows user to have a cross section view of the 3D images. Figure 4-8 shows that object can be dissected using movable red outline plane where the gray arrow act as a guide to the angle of the plane in relation to the image. This is important for physician to be able to see the inside of the heart cavity or artery for a more thorough examination.

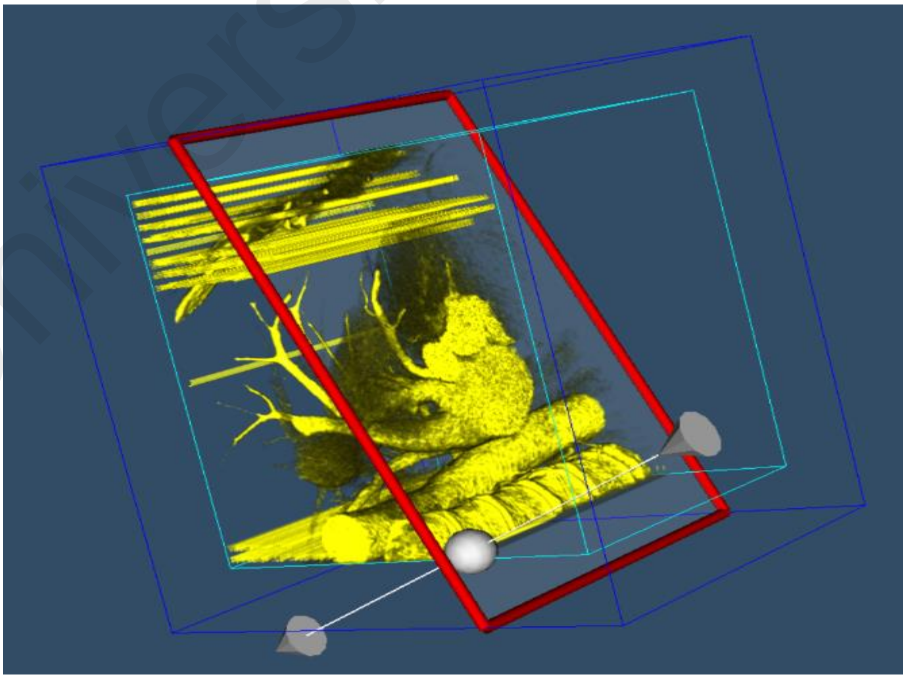

Figure 4-8: Image dissection through cross plane using VTK

The plane movement are not independent to the image meaning any movement of the image such as rotation will results in the movement of the cross-section plane as well.

In RadiAnt, viewing the image can be done through multiplanar rendering (MPR). However, no 3D image is rendered during MPR which makes viewing gauging

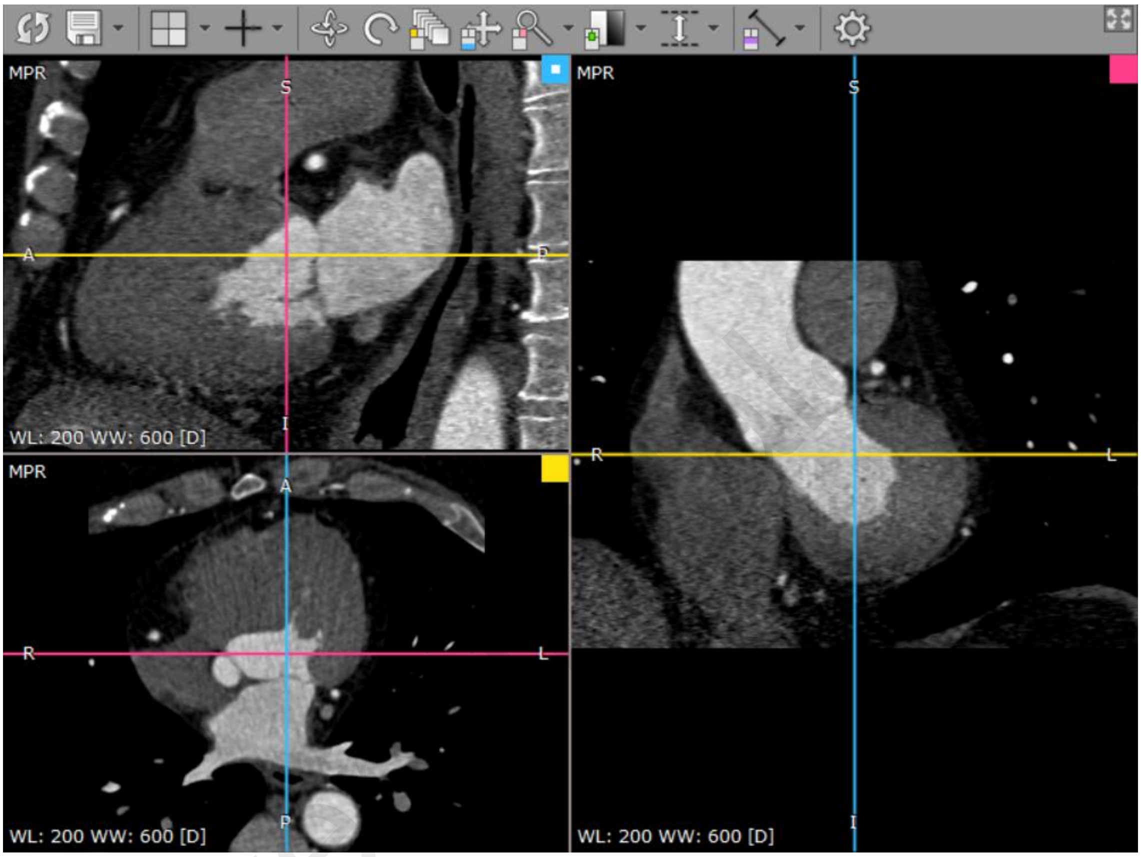

Figure 4-9: RadiAnt MPR view at sagittal, coronal and axial plane.

orientation harder for the untrained eye. Figure 4-9 shows the MPR image viewed using RadiAnt at sagittal, coronal and axial plane. Overall image rendering quality between VTK and RadiAnt are comparable with each other. Although the RadiAnt have a built-in feature to assist user to view 3D images, the potential and customization in VTK cannot be overlook. The ability of VTK to reconstructs 3D images from 2D image slice CT Scan are proven in this paper. The 3D image rendering can be improved with better volume color and opacity mapping that corresponds with the different pixel or voxel intensity. However, resulting reconstruction image are largely dependent to the temporal and spatial resolution of the 2D image slice acquisition.

CHAPTER 5: DISCUSSION

In this paper, it has been demonstrated that VTK have the capability of reconstructing 3D cardiac images from 2D CT Scan image slices with surface rendering and volume rendering method. The result of both techniques can be seen in Figure 4-1 and Figure 4-3 where the overall quality of the rendering is much better using volume rendering than surface rendering for cardiac imaging. Surface rendering offers identifiable 3D images with good depth perspective. However, it uses only uses limited data available to perform the reconstruction. Selection of threshold to represent the surface can be time-consuming and reconstruction must be performed to check the image quality making it difficult and arduous task. The difficulty lies when surface of object is not adequate in structures and without well contrast anatomy. Approximation of surface can introduced false-positive such as spurious surfaces or false-negative such as holes in the surface particularly in presence of small or poorly defined features where soft tissue definition is limited (Cutroneo et al., 2016).

One of the advantages of surface rendering is fast and interactive manipulation of images (Ooijen, 2003), this does not hold true for cardiac image reconstruction. Reconstruction time for surface rendering was poor compared to volume rendering (52.39±0.24s vs 2.63±0.11s). This does not share the same result with other study (Rodt et al., 2006) where surface rendering provide time efficient means of 3D visualization. The lead to only 245 2D image slices were selected for rendering to avoid longer reconstruction time.

Reconstruction of cardiac image in 3D using volume rendering technique yielded better image quality. 3D volume rendering considers all data in the volume and display the composite result according to the voxel value. Cardiac structures can be visualized by segmentation of different structures density through multiple view plane. Volume rendering also allows part of the dataset to be removed through sectioning or varying opacity to reveal the anatomical intracardiac structure. 3D reconstruction in this paper (Figure 4-2) shows better image quality with distinguishable cardiac structure. More coronary arteries are visible in VR compared to SR images. Previous studies (Ooijen, 2003) have shown volume rendering visualizes higher percentage of artery (89-96%) compared to surface rendering technique (53%-70%). This is particularly true for artery with diameter range of 2mm to 3mm. However, determination of optimal settings can be a challenge when using volume rendering technique. A multiple set of voxel value need to be represented accordingly to ensure good contrast between structures or arteries. Ideal color would create good depth perception and shows exact arteries and vessels pathway and location. Selecting which data to visualize become a challenge since there are adjacent organ than share the same pixel intensity as the vessels such as esophagus. Furthermore, the effect is exacerbated when image with low spatial resolution is used for the reconstruction.

VTK is a powerful tool for 3D rendering. It has shown to be able to reconstruct images using surface rendering and volume rendering technique. Image rendered can rotate, zoom and measured for diagnostic purpose. This is important so physician can get a better view at different plane. All the features are incorporated through the programming code and the built-in libraries of VTK. It is easily customizable, and image are comparable with on the market 3D reconstruction software such as RadiAnt DICOM Viewer. Since the 2D images was acquired using CTA protocol, arteries structure is more prominent compared to other part of the heart. RadiAnt was able to vary window level to remove layers of structure by using mouse. This operation akin to 'peeling-off' outer part of the heart to get a better view inside. VTK achieved this by varying opacity level at each voxel segmentation in the program. For ease of usage, it is advisable to incorporate mouse control opacity function for VTK in the future. Another feature is structure measurement where different result was obtained in VTK (56.14cm) and in RadiAnt (3.76cm) for aorta diameter measurement. This can be due to the assumption of 1-pixel length is equivalent 0.2645 mm based on 96 dpi in the image. The program code for this part need to be improved further to ensure correct measurement can be made in the rendering window.

The program code in VTK reads the 2D images in JPEG format. Therefore, DICOM format images needed to convert into JPEG before reconstruction in VTK compared to RadiAnt where DICOM images can be directly used. This process makes the overall reconstruction process longer. Although quality of both images is comparable, RadiAnt was able to remove the patient information on the image making viewing the 3D object unobstructed. 3D image in VTK need to be angled to allow clear view path. Removing the patient information in VTK can be achieved with filtering however this would result in removal some part of the images.

RadiAnt only able to reconstruct images using volume rendering and multiplanar reformatting therefore comparison of surface rendering technique between the two cannot be achieved. One of the advantages of surface rendering is fast and interactive manipulation of images (Ooijen, 2003), this does not hold true for cardiac image reconstruction. Reconstruction time in VTK for surface rendering was poor compared to volume rendering (52.39±0.24s vs 2.63±0.11s). This does not share the same result with other study (Rodt et al., 2006) where surface rendering provide time efficient means of 3D visualization. The lead to only 200 2D image slices were selected for rendering to avoid reconstruction time delay. Therefore, further improvement on the coding need to be done to achieve highly efficient surface rendition in VTK.

The 3D image quality depends to the quality of the 2D image slices. Temporal resolution is crucial CT Scan ability to resolve fast-moving object such as the heart. Irregular and fast cardiac motion would result in artefacts which ultimately affect the 3D reconstructed image (et al Hyochol Ahn 2017, 2017). Temporal resolutions in CT can be improved by using dual source CT or partial scan reconstruction. The other factor that limits the quality of 3D image is spatial resolution of CT Scan. Spatial resolution refers to the ability to differentiate between object with different density. Structures in proximity can be distinguished with one another with high spatial resolution. Spatial resolution can be improved using small detector element or high definition CT. The difference in x-ray attenuation especially in soft tissue is small resulted in low contrast resolution in CT. Administration of intravenous contrast to structures such as coronary artery improves the contrast resolution in cardiac imaging especially in CTA protocol. Consideration must be taken to all these properties when acquiring 2D CT Scan images to ensure good quality image.

Overall, this research limitation lies on the quality of images and the protocol used to acquire it using CT Scan. High quality CT Scan cardiac images are difficult to acquire due to the technology limitations such as low temporal resolutions. MRI and Angiography are the better alternatives when doing coronary cardiac imaging however it suffers from long scan time and its invasive procedure. For future studies, the usage of VTK can be explored by using dataset from multiple modalities to improve the 3D reconstruction of cardiac. The added information can surely help to achieve a more detailed rendering with fine segmentation and contrast to show structures of the heart.

CHAPTER 6: CONCLUSION

In conclusion, VTK is a powerful program that can reconstruct a 3D cardiac image form 2D CT-Scan slice. Image were reconstructed using surface rendering technique and volume rendering technique successfully. Volume rendering are the preferred choices compared to surface rendering since it was able to capture all dataset in the displayed voxel. Surface rendering uses thresholding method that can lead to false positive or false negative in the presence of low tissue definition. Volume rendering image in VTK are comparable to RadiAnt DICOM viewer. Cardiac arteries are discernible between one another with the structure resembles an actual human heart. Features such as measurement and cross-section plane can be integrated within VTK programming code to allow higher diagnostic functionality.

In the future, the fusion of VTK 3D image from CT scan with other modality can be explored to increase better quality rendering. Different modalities have such as MRI or Angiography has resolution properties that would help to improve image rendering quality.

REFERENCES

Agatston, A. S., Janowitz, W. R., Hildner, F. J., Zusmer, N. R., Viamonte, M., & Detrano, R. (1990). Quantification of coronary artery calcium using ultrafast computed tomography. *Journal of the American College of Cardiology*, *15*(4), 827–832. https://doi.org/10.1016/0735-1097(90)90282-T

Becker, C. R., Jakobs, T. F., Aydemir, S., Becker, A., Knez, A., Schoepf, U. J., Bruening, R., Haberl, R., & Reiser, M. F. (2000). Helical and single-slice conventional CT versus electron beam CT for the quantification of coronary artery calcification. *American Journal of Roentgenology*, *174*(2), 543–547. https://doi.org/10.2214/ajr.174.2.1740543

Becker, C. R., Knez, A., Ohnesorge, B., Schoepf, U. J., & Reiser, M. F. (2000). Imaging of noncalcified coronary plaques helical CT with retrospective ECG gating. *American Journal of Roentgenology*, *175*(2), 423–424. https://doi.org/10.2214/ajr.175.2.1750423

Becker, C. R., Ohnesorge, B. M., Schoepf, U. J., & Reiser, M. F. (2000). Current development of cardiac imaging with multidetector-row CT. *European Journal of Radiology*, *36*(2), 97–103. https://doi.org/10.1016/S0720-048X(00)00272-2

Buia, A., Stockhausen, F., Filmann, N., & Hanisch, E. (2017). 3D vs. 2D imaging in laparoscopic surgery—an advantage? Results of standardised black box training in laparoscopic surgery. *Langenbeck's Archives of Surgery*, *402*(1), 167–171. https://doi.org/10.1007/s00423-016-1526-9

Byrn, J. C., Schluender, S., Divino, C. M., Conrad, J., Gurland, B., Shlasko, E., & Szold, A. (2007). Three-dimensional imaging improves surgical performance for both novice and experienced operators using the da Vinci Robot System. *American Journal of Surgery*, *193*(4), 519–522. https://doi.org/10.1016/j.amjsurg.2006.06.042

Cirne, M. V. M., & Pedrini, H. (2013). Marching cubes technique for volumetric visualization accelerated with graphics processing units. *Journal of the Brazilian Computer Society*, *19*(3), 223–233. https://doi.org/10.1007/s13173-012-0097-z

Cutroneo, G., Bruschetta, D., Trimarchi, F., Cacciola, A., Cinquegrani, M., Duca, A., Rizzo, G., Alati, E., Gaeta, M., & Milardi, D. (2016). In vivo CT direct volume rendering: A three-dimensional anatomical description of the heart. *Polish Journal of Radiology*, *81*, 21–28. https://doi.org/10.12659/PJR.895476

Dong, H., Xia, L., Zhang, J., & Cai, A. (2013). *Medical Image Reconstruction Based on ITK and VTK*. 642–645. https://doi.org/10.1109/CSA.2013.155

El Kaffas, A., Sigrist, R. M. S., Fisher, G., Bachawal, S., Liau, J., Wang, H., Karanany, A., Durot, I., Rosenberg, J., Hristov, D., & Willmann, J. K. (2017). Quantitative three-dimensional dynamic contrast-enhanced ultrasound imaging: First-in-human pilot study in patients with liver metastases. *Theranostics*, *7*(15), 3745–3758. https://doi.org/10.7150/thno.20329

et al Hyochol Ahn 2017. (2017). 乳鼠心肌提取 HHS Public Access. *Physiology & Behavior*, *176*(10), 139–148. https://doi.org/10.1016/j.jcct.2009.07.003.What

Fink, C., Krissak, R., Henzler, T., Lechel, U., Brix, G., Takx, R. A. P., Nance, J. W., Abro, J. A., Schoenberg, S. O., & Schoepf, U. J. (2011). Radiation dose at coronary CT angiography: Second-generation dual-source CT versus single-source 64-MDCT and first-generation dual-source CT. *American Journal of Roentgenology*, *196*(5), 550–557. https://doi.org/10.2214/AJR.10.5153

Hongjian, W. (2009). *3D Medical CT Images Reconstruction based on VTK and Visual C ++*. 1–4.

Kalisz, K., Buethe, J., Saboo, S. S., Abbara, S., Halliburton, S., & Rajiah, P. (2016). Artifacts at cardiac CT: Physics and solutions. *Radiographics*, *36*(7), 2064–2083. https://doi.org/10.1148/rg.2016160079

Khan, U., Yasin, A. U., Abid, M., Awan, I. S., & Khan, S. A. (2018). A Methodological Review of 3D Reconstruction Techniques in Tomographic Imaging. *Journal of Medical Systems*, *42*(10). https://doi.org/10.1007/s10916-018-1042-2

Lawler, L. P., Pannu, H. K., & Fishman, E. K. (2005). MDCT evaluation of the coronary arteries, 2004: How we do it - Data acquisition, postprocessing, display, and interpretation. *American Journal of Roentgenology*, *184*(5), 1402–1412. https://doi.org/10.2214/ajr.184.5.01841402

Luccichenti, G., Cademartiri, F., Pezzella, F. R., Runza, G., Belgrano, M., Midiri, M., Sabatini, U., Bastianello, S., & Krestin, G. P. (2005). 3D reconstruction techniques made easy: Know-how and pictures. *European Radiology*, *15*(10), 2146–2156. https://doi.org/10.1007/s00330-005-2738-5

Neves, P. O., Andrade, J., & Monção, H. (2017). Coronary artery calcium score: current status. *Radiologia Brasileira*, *50*(3), 182–189. https://doi.org/10.1590/0100-3984.2015.0235

Nugroho, P. A., Basuki, D. K., & Sigit, R. (2017). 3D heart image reconstruction and visualization with marching cubes algorithm. *2016 International Conference on Knowledge Creation and Intelligent Computing, KCIC 2016*, 35–41. https://doi.org/10.1109/KCIC.2016.7883622

Ooijen, P. M. A. Van. (2003). Using Electron Beam CT : Surface. *Most*, *January*, 223–226.

Pierce, L., Raman, K., Rosenberg, J., & Rubin, G. D. (2012). Quality improvement in 3D imaging. *American Journal of Roentgenology*, *198*(1), 150–155. https://doi.org/10.2214/AJR.10.5253

Raman, S. P., Horton, K. M., & Fishman, E. K. (2013). MDCT evaluation of ureteral tumors: Advantages of 3D reconstruction and volume visualization. *American Journal of Roentgenology*, *201*(6), 1239–1247. https://doi.org/10.2214/AJR.13.10880

Rodt, T., Bartling, S. O., Zajaczek, J. E., Vafa, M. A., Kapapa, T., Majdani, O., Krauss, J. K., Zumkeller, M., Matthies, H., Becker, H., & Kaminsky, J. (2006). Evaluation of surface and volume rendering in 3D-CT of facial fractures. *Dentomaxillofacial Radiology*, *35*(4), 227–231. https://doi.org/10.1259/dmfr/22989395

Ronald Mikolich, J. (2012). Cardiac computed tomographic angiography and the primary care physician. *Journal of the American Osteopathic Association*, *112*(5), 267–275. https://doi.org/10.7556/jaoa.2012.112.5.267

Roth, G. A., Abate, D., Abate, K. H., Abay, S. M., Abbafati, C., Abbasi, N., Abbastabar, H., Abd-Allah, F., Abdela, J., Abdelalim, A., Abdollahpour, I., Abdulkader, R. S., Abebe, H. T., Abebe, M., Abebe, Z., Abejie, A. N., Abera, S. F., Abil, O. Z., Abraha, H. N., … Murray, C. J. L. (2018). Global, regional, and national age-sex-specific mortality for 282 causes of death in 195 countries and territories, 1980–2017: a systematic analysis for the Global Burden of Disease Study 2017. *The Lancet*, *392*(10159), 1736–1788. https://doi.org/10.1016/S0140-6736(18)32203-7

Shuman, W. P., Branch, K. R., May, J. M., Mitsumori, L. M., Lockhart, D. W., Dubinsky, T. J., Warren, B. H., & Caldwell, J. H. (2008). Prospective versus retrospective ECG gating for 64-detector CT of the coronary arteries: Comparison of image quality and patient radiation dose. *Radiology*, *248*(2), 431–437. https://doi.org/10.1148/radiol.2482072192

Sun, Z. (2006). 3D multislice CT angiography in post-aortic stent grafting: A pictorial essay. *Korean Journal of Radiology*, *7*(3), 205–211. https://doi.org/10.3348/kjr.2006.7.3.205

Sun, Z. (2010). Multislice CT angiography in cardiac imaging: Prospective ECG-gating or retrospective ECG-gating? *Biomedical Imaging and Intervention Journal*, *6*(1), 1–8. https://doi.org/10.2349/biij.6.1.e4

Wee, L. K., Chai, H. Y., Supriyanto, E., Group, B. E., & Ilmenau, T. U. (2011). *Surface rendering of three dimensional ultrasound images using VTK*. *70*(June), 421–426.

Wheeler, G., Deng, S., Toussaint, N., Pushparajah, K., Schnabel, J. A., Simpson, J. M., & Gomez, A. (2018). Virtual interaction and visualisation of 3D medical imaging data with VTK and unity. *Healthcare Technology Letters*, *5*(5), 148–153. https://doi.org/10.1049/htl.2018.5064

Xu, T., Wei, N., & Jiang, T. (2013). *Comparision of Surface Reconstruction Method for CT based on VTK*. *Icaiees*, 215–218.