

**Optimising  
CT Image  
Analysis  
&  
Visualisation**

2014

For the Clinical  
Environment



Keele  
University

CSC-30014:  
Computer Science  
Double Dissertation

Student ID:  
11005305/01

Submission Date:  
09/05/2014

Word Count:  
7,907



# Optimising CT Image Analysis & Visualisation

---

*For the Clinical Environment*

**CSC-30014: Computer Science Double Dissertation**

**Student ID: 11005305/01**

**Submission Date: 09/05/2014**

**Word Count: 7,907**

*“CT may be the first instance where analysis software is an essential component of medical diagnosis and comes between the physician and patient: the physician has no recourse but to trust the software.”*

*(Semmlow, 2004, pg 375)*



## Abstract

Prostate cancer alone afflicts over 40,000 men in the UK annually (Prostate Cancer UK, 2013). Up to this point the effectiveness of detecting and localising abnormalities has been through manual processes which are subject to human error. As a consequence the need to negate such errors has led to research towards an automatic localisation method specifically aimed at prostate cancer; highlighting the potential of such research to expand to localise other types of cancer. To this end the prototype application has been development to act as a research enabling tool which optimises the image analysis, which would be required to expand automatic localisation to different cancer types. This is achieved through the building of 3 modules, for the visualisation, in 2D & 3D, and reporting of internal prototype information.

## Acknowledgments

Special thanks must go to Dr Josep Sule-Suso who gratuitously gave up his time for this project. I must particularly thank Dr Sarah Clemerson without whose insight the quality of the semantic structure would have been much lower. My unending thanks must go to Aanna Barard, who has been working with me on this project since before its conception, helping with grammar, structure and acting as devil's advocate to really make me think. Penultimately, a particular thank you must go to my supervisor Dr KP Lam. Last but not least, I should like to thank the residents of 9 Poolfield Avenue for being 'supportive' and also for their seemingly unending supplies of cakes.

## Table of Contents

Abstract .....	5	Appendix 1: Evaluation Form (Atkin-Granville, 2012) .....	46
Acknowledgments .....	6	Appendix 2: Client Meeting Question Ideas .....	49
Table of Contents .....	7	Appendix 3: Project Meeting Minutes .....	50
List of Figures.....	8	Appendix 4: Project Assumptions.....	56
List of Tables.....	8	General .....	56
1 Introduction & Background .....	9	Hardware .....	56
1.1 Project Aim .....	12	Software .....	56
1.2 Project Objectives.....	12	Appendix 5: Deliverables in Detail.....	57
1.3 Project Stakeholders.....	12	Enhanced UI.....	57
1.4 Governance Levels.....	12	3D Visualisation .....	57
1.5 Report Structure .....	13	Report generation .....	57
2 Methodology .....	14	Appendix 6: Project Risks & Mitigation .....	58
2.1 Requirements Gathering - Client Meeting .....	14	Appendix 7: Viewer3D's Supported File-Types.....	59
2.1.1 Project Requirements.....	15	Appendix 8: CSV Output Example .....	59
2.2 Monitoring - Project Meetings .....	15	Appendix 9: VTK Data Types (Gasser, 2009).....	60
2.3 Assumptions .....	16	Appendix 10: Black-Box Evaluation Form Example .....	61
2.4 Deliverables .....	16		
2.5 Risk Management.....	16		
2.6 System Design .....	17		
2.7 Development Method – Prototyping Model .....	17		
2.8 Development Environments.....	19		
2.8.1 MATLAB and GUIDE.....	19		
2.8.2 VTK.....	20		
3 Implementation .....	21		
3.1 Currently Available Systems in MATLAB .....	21		
3.2 Mock-Up using MatVTK.....	23		
3.3 Application Prototype 0.0.1 .....	25		
3.4 Application Prototype 0.0.2 .....	26		
3.5 Application Prototype 0.0.3 .....	30		
3.6 Application Prototype 0.0.4 .....	31		
3.7 Application Prototype 0.0.5 .....	32		
3.8 Application Prototype 0.0.6 .....	33		
3.9 Application Prototype 0.0.7 .....	33		
3.10 Application Prototype 0.0.8 .....	34		
3.11 3D Visualisation Prototype .....	35		
3.12 Evaluation & Testing.....	37		
4 Project Summary & Future Recommendations .....	39		
5 References .....	41		

## List of Figures

Figure 1: Governance levels .....	12
Figure 2: MVC pattern relative to the user (RegisFrey, 2010) .....	17
Figure 3: The Evolutionary Prototyping with EPRAM model (Carter et al, 2001) .....	18
Figure 4: VTK structure (Schroeder, 2005).....	20
Figure 5: Viewer3D's greyscale 3D render .....	21
Figure 6: Viewer3D's multi-window mode .....	22
Figure 7: Viewer3D's ROI construction points .....	22
Figure 8: Viewer3D's ROI extrapolated from points .....	22
Figure 9: Viewer3D's DICOM information pop-up window .....	23
Figure 10: Application Mock-Up using MatVTK .....	24
Figure 11: Early implementation of boundary drawing.....	26
Figure 12: Boundary resolution from points cloud.....	27
Figure 14: Convex hull vs. Alpha shape (Graser, 2011).....	27
Figure 13: Convex hull (Matan, 2013).....	27
Figure 15: Alpha shape formation (Kumar, 2013) .....	28
Figure 16: A simple topographical vector structure and database (Indiana University, 2014) .....	29
Figure 17: Selected boundary for deletion .....	31
Figure 18: Contrast manipulation of 2D CT scan render.....	32
Figure 19: VTK window environment (Schroeder, 2005).....	35
Figure 20: VTK render pipeline (Schroeder, 2005).....	36
Figure 21: VTK volume render .....	36
Figure 22: VTK surface render of soft tissue and bone.....	36
Figure 23: Application Prototype 0.0.8 .....	39

## List of Tables

Table 1: Project stakeholders.....	12
Table 2: Deliverable success criteria .....	16



## 1 Introduction & Background

In the UK the most common cancer to afflict men is that of the prostate. In recent years the publicity surrounding the condition has grown with celebrity backers. In the UK over 40,000 men are diagnosed with the cancer every year and over 250,000 are currently living with the disease (Prostate Cancer UK, 2013). Many patients have Magnetic Resonance Imagery (MRI) or Computational Tomography (CT) scans either to monitor their condition or when their treatment is being planned (NICE, 2014).

Such imaging techniques produce large datasets, which far surpasses the working capacity of clinicians to analyse. The radiologist must search within, the large CT datasets to detect, localise and interpret potential abnormalities (Andriole et al, 2011). This is a difficult and time-consuming process, and human limitations can lead to errors being made in these tasks (Chakraborty, 2006; Eckstein, 2000; Palmer, 1995). These errors are false positive and negative judgments which, depending on the context can have a wide range of impacts on the health of a patient.

Andriole et al (2011) expresses the need for an integrated information system to relieve the pressure on clinicians' workflow to achieve safer and better quality care for patients whilst making radiologist more efficient and effective, multiple other authors state that such a system is desirable (Moise & Atkins, 2005; Boehm et al, 2004; Arenson et al, 2003). In other areas where large datasets are a problem, work is also being done to evaluate systems designed to assist in the handling of that data (Carter et al, 2008). Such an information system would require:

- Integrated advanced post-processing
- Integrating innovative HCI methods
- Embedded patient metadata
- Design informed by current best practices
- Considerations for the clinical environment

In such a system it would be beneficial to include 3D visualisation tools since it has been shown that such visualisations can help separate and differentiate objects that are normal from abnormal structures that have a similar appearance (Andriole et al, 2011). Currently there are a number of tools which offer such functionality for CT colonography, for example, V3D (Andriole et al, 2011). Other computer-aided-detection (CAD) systems exist for other cancer types such as the R2 ImageChecker system which highlights areas of a slide which potentially contain abnormalities which the radiologist can then re-examine (Andriole et al, 2011). Studies have shown that CAD can affect the accuracy of diagnoses since the systems carry assumptions about occurrence levels which result in high false positive rates (e.g. Horsch et al, 2008; Berbaum et al, 2007; Freedman, 2004). In studies on CAD effectiveness and confidence of radiologist when using it found that whilst the false-negatives slightly lowered confidence in the system, its use increased the number of true-positives made, and as a result the investigators concluded the such system are appropriate to uses since the benefits outweigh the risks (Freedman et al, 2010; Freedman et al, 2001; Kobayashi et al, 1996).

Modern CT imaging systems can produce isotropic voxel datasets which facilitate viewing in the three orthogonal planes, axial, sagittal, and coronal as well as in any custom angle (Andriole et al, 2011). This is possible in real time on workstation systems thanks to the advancement of computational hardware, specifically Graphic Processing Unit (GPU) technology, which enables the computationally intensive calculations required for the rendering process. Such volume rendering techniques produce realistic and useful representations for radiologists (Rubin et al, 1996). Despite problems internalising the spatial relationships and depth within 3D volume renderings that some radiologists have, which is mitigated to some degree by the use of transparency and colour. However, the ease of production means that it is the reference standard for 3D viewing (Ravenel & McAdams, 2003; Kuszyk et al, 1996). 3D slicer is an example of a software system which can run on a workstation and can perform such advanced post-processing easily. However, for optimum integration into the workflow environment of the radiologists such applications should be integrated into the PACS stations where the majority of image interpretation occurs (Prevedello et al, 2008; Prevedello et al, 2007; Andriole et al, 2006). Such integration enables real-time visualisation along with all of the features which are available through such as station for example non-image patient metadata.

When communicating the results back to the patient, this has traditionally been achieved through the uses of written prose describing the findings and their meanings, such methods requires the radiologist to compress a lot of quite complex information and that can introduce errors (Andriole et al, 2011). The inclusion of images in the reports, following the adage ' a picture speaks 1000 words' could ease this issue. An example of work done towards advance methods used to communicate with patients is Dr Josep Sule-Suso's work at the North Staffordshire NHS University Hospital, where 3D technologies have been used to help patients understand where their cancer is located, how the treatment impacts on other organs, and thus how some of the side effects come about (NSUH Charity, 2010). Building on best practices to improve patient care utilising a combination of technologies to improve the accuracy of diagnosis and localisation through the use of CAD, according to Andriole et al (2011) requires a paradigm shift towards combining image and non-image data along with workflow considerations such as audits, verifications, communication logs and reminders, forming a fully integrated information system for cancer treatments.

Against this backdrop academics and students at the University of Keele, have been documenting steps towards the creation of an objective automated system to detect and localize potentially cancerous abnormalities within large CT scan datasets (Lam et al, 2013; Atkin-Granville, 2012; Moss, 2010; Ossati, 2010). Currently the algorithm functions well for the detection of prostate cancer, however before testing can begin on other cancers the system needs to be developed to improve its integration into the working environment of radiologists. This project will first gather feedback from clinical staff on the current system within their environment. Usability changes will then be made to reflect their comments. Moreover data reporting and presentation will be improved through a 3D rendering of the data as well as the generation of reports for clinical staff, and researchers.

## 1.1 Project Aim

To construct a prototype application to facilitate the expansion and continuation of the research on the automated detection of cancerous abnormalities within CT scans, by gathering feedback from clinical staff on the current systems and thereupon building 3 modules, for the visualisation, in 2D & 3D, and reporting of internal prototype information.

## 1.2 Project Objectives

1. 2D visualisation with increased sensitivity to the working environment of clinicians' ensuring an efficient and user-friendly experience.
2. 3D visualisation to improve the assessment of demarcated zones within a CT scan dataset.
3. Report generation to provide data presentation.

The objectives of this project have been prioritised to ensure that certain requirements are met despite changes to the projects 'goalposts' part way through development. Thus this project may be completed without all of the initially stated objectives having been met.

## 1.3 Project Stakeholders

Table 1: Project stakeholders

Project Developer	Supervisor	Clinical Client
Benjamin Walker, Undergraduate Student, University of Keele	Dr KP Lam, School of Computing and Mathematics, University of Keele	Dr Josep Sule-Suso, Cancer Centre, University Hospital of North Staffordshire

## 1.4 Governance Levels

To ensure clear management of the project a governance structure has been setup so that the project priorities can be set in a hierarchical fashion by the stakeholders. The structure takes account of their level of engagement with the project and the priorities given to their desired outcomes.

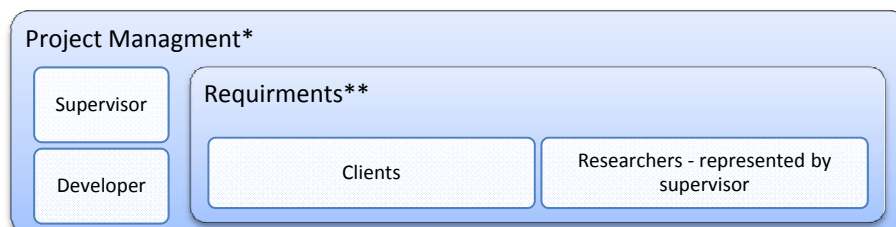


Figure 1: Governance levels<sup>1</sup>

<sup>1</sup> \*Project management can override requirement conditions for reasons of deliverability. Within this the supervisor's decisions out rank those of the developer. \*\*Ranking read left to right

## 1.5 Report Structure

Chapter 1 provides an overview of the background in which the project is situated with regard current literature on CAD and the reasoning and support for an integrated Cancer information system. It then sets out the project goals and the stakeholders involved, and the governance structure in which the stakeholders operate with the projects decision-making hierarchy.

Chapter 2 provides a detailed overview of the methodology the project employs. Beginning with outlining the collection of requirements and the monitoring system in place for ensuring the project stays on track and fulfils the stakeholder's wishes. This chapter also outlines the assumptions that have been made about the implementation of the project and further acts to scope the project by providing details of the deliverables. Risks and mitigation strategies are detailed in this section, followed by details of the system design and development methods and details of the development environments involved.

Chapter 3 details the stages of the projects implementation from the initial groundwork with tests of the currently available products through mock up stages and each increment of the prototyping development method. The chapter concludes with an overview of the testing and evaluation that have been carried out during the project.

Chapter 4 concludes the project with a summary of achievements and problems encountered then looks to the future with suggestion of improvements to be made to the system as well as lessons which can be learnt from the management of this project.

The Appendix materials included with this report act to support arguments made throughout and are referenced where appropriate within the main body of the text.

## 2 Methodology

The project began with access to a set of documentation from the previous Keele projects which looked into a multi-scale boundary algorithm. From this documentation it was possible to draw the preliminary requirements. The requirements that could be gathered from Atkin-Granville's (2012) evaluation form (See Appendix 1) included:

- Better boundary drawing, specifically the ability to modify mistakes easier
- Inclusion of both keyboard shortcuts and tooltips

The scope of the project could also be partly extrapolated from the literature surrounding CAD as stated in the introduction chapter. To focus these coarse requirements the project undertook a meeting with the clients, both academic and clinical, to ascertain how the projects output would assist in the continuation of research as well as providing an overview of the specific context & restraints of the project within the treatment planning lifecycle.

### 2.1 Requirements Gathering - Client Meeting

This informal meeting was with Dr KP Lam and Dr Josep Sule-Suso and was lightly structured working through a set of questions (See Appendix 1). The discussions revolved around previous work Keele students had done, and the areas which Dr Sule-Suso felt needed improving on. This discussion confirmed the requirements gathered from Atkin-Granville's (2012) work. During the meeting the Vert System which the NSUH Charity (2010) describes was demonstrated and the methodological thinking behind its use explained. In this demonstration 3D anatomical models on organs and cancerous structures were rendered. However the system required glasses and was unsuitable for use in the localisation tasks of the radiologists.

For this session a mock-up using currently available products was demonstrated to both stakeholders who walked through a series of different tasks which would be performed by a radiologist and the discussion of ways in which a workstation system could assist them. From this it was attained that the ideal would be automated boundary generation for all organs and potential cancer structures, which radiologists would then check over manually and enter error margins. This target was outside of the project remit due to time limitations and due to other stakeholder requirements. As such a probable solution was put forward, the integration of boundaries drawn in 2D into 3D along with usability enhancements from the Atkin-Granville (2012) system.

The technique discussed by Schneider (2007) in *Generating Fast Feedback in Requirements Elicitation* was used during the meeting. Since it saves one or two interview cycles, given the limitations on time for both the developer and the restricted availability of the clinical staff that are the source of the majority of the requirements, having a fast method for eliciting the requirements of the project is essential.

### 2.1.1 Project Requirements

The following are the project requirements as gathered from the client meeting.

- A Multi-Window application to enable flexible and personalised use of screen real-estate.
- Tooltips, keyboard shortcuts and onscreen instructions where suitable to ensure a low learning curve.
- 2D visualisation of DICOM images.
- Freehand boundary drawing, which can be constructed of multiple segments and must be editable.
- Image zooming and panning.
- Scroll wheel movement through scan slices.
- Contrast manipulation.
- Interfacing between Matlab, Java and C++ library for maximum portability.
- 3D volume rendering of CT scan DICOM files.
- Interpolation of a 3D surface from boundary points drawn in the 2D visualisation.
- Utilising the Matlab Report Generator to automatically produce reporting from within the application.
  - The ability to include system variables and CT scan DICOM metadata.
  - In a number of different formats, HTML, PDF, RTF, Microsoft Word, and XML.
  - Customisable and consistent reports using Setup Files and Style Sheets.

## 2.2 Monitoring - Project Meetings

Project meetings were designed to support the progression of the project and revolved around the resolution of technical issues, the defining of goals and ensuring the robustness of the projects scope management. The meetings were held weekly with the project supervisor Dr KP Lam, results of such meetings were document in the minutes (see Appendix 3) and normally contained goals for the next meeting and possible directions for advancement for the project to take.

## 2.3 Assumptions

The project makes a number of assumptions about the context in which it is to be implemented. These assumptions are detailed in Appendix 4 and have been made in conjunction with the project supervisor and have been deemed appropriate and proportionate.

## 2.4 Deliverables

The three objectives of this project translate to three deliverable elements of the system which it is this projects aim to develop. The deliverables have an attached success criteria which is detailed in Table 2 and against which the evaluation of the deliverable will be evaluated. For a more detailed description of the deliverable, its tests, and evaluation see Appendix 5.

Table 2: Deliverable success criteria

Deliverable	Success Criteria
Enhanced UI	Effective use by clinical staff with limited (~0-15 min) training.
3D Visualisation	Comparisons with the localisation results of clinical staff, and use of their expert knowledge to assess whether the formation shown in 3D is an accurate representation of the 2D depictions.
Report generation	Producing a content match with the existing reports on which the specification is based.

## 2.5 Risk Management

According to the Standish Group International (2009), mismanagement of software projects commonly results in project failure. The management of risk is a core part of any project management, and Otniel et al (2012) highlights in their review of the literature a number of methods which can be employed to prepare for these risks and appropriately manage around them. Since risk relates to any “events, occurrences and actions” (Otniel et al, 2012, pg 1014) that can prevent a project from reaching its goal, this project has taken steps to decrease the probability and impact of negative effects (Project Management Institute, 2008). At the most general level an appropriate amount of contingency time has been built into the project to allow for the occurrence of a number of risk events. For details of the perceived risks which this project could face and what mitigation action is being taken see Appendix 6.



## 2.6 System Design

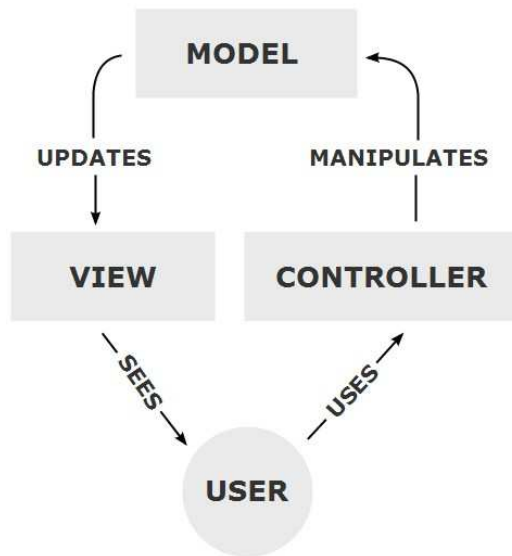


Figure 2: MVC pattern relative to the user (RegisFrey, 2010)

The Object Orientated (OO) paradigm for programming began with the vision of Doug Englebart who saw the computer as an extension of the human mind with the ultimate goal of capturing the end user's mental model in code (Reenskaug & Coplien, 2009). The Model-View-Controller pattern was developed by Trygve Reenskaug in 1988 as a method for providing good human-computer interaction through a Graphical User Interface (GUI) (Model, 2012). OO and MVC complement each other and thus this project has implemented the three objects required to fulfil the MVC pattern, as shown in Figure 2 (Burbeck, 1997). This approach

was chosen since MVC provides a well tested means through which an interactive and intuitive GUI can be implemented with the minimum risk to the projects viability.

## 2.7 Development Method – Prototyping Model

Carter et al's (2001) paper *Evolving Beyond Requirements Creep: A Risk-Based Evolutionary Prototyping Model* is used as a basis for the development methodology for this project.

Evolutionary prototyping offers a couple of benefits for this project particularly the creation of improved user interfaces but also the uncovering of missing and/or unknown requirements.

The Evolutionary Prototyping with Risk Analysis and Mitigation (EPRAM) model (See Figure 3), that Carter et al (2001) describes, is a horizontal prototyping model, which incorporates the risk and compliance assessment of Anton and Earp (2001) and the risk identification and ranking process of Baskerville & Stage (1996), to form a comprehensive risk mitigation strategy. An iterative process invites requirements creep which, in this project, is managed through risk analysis and mitigation.

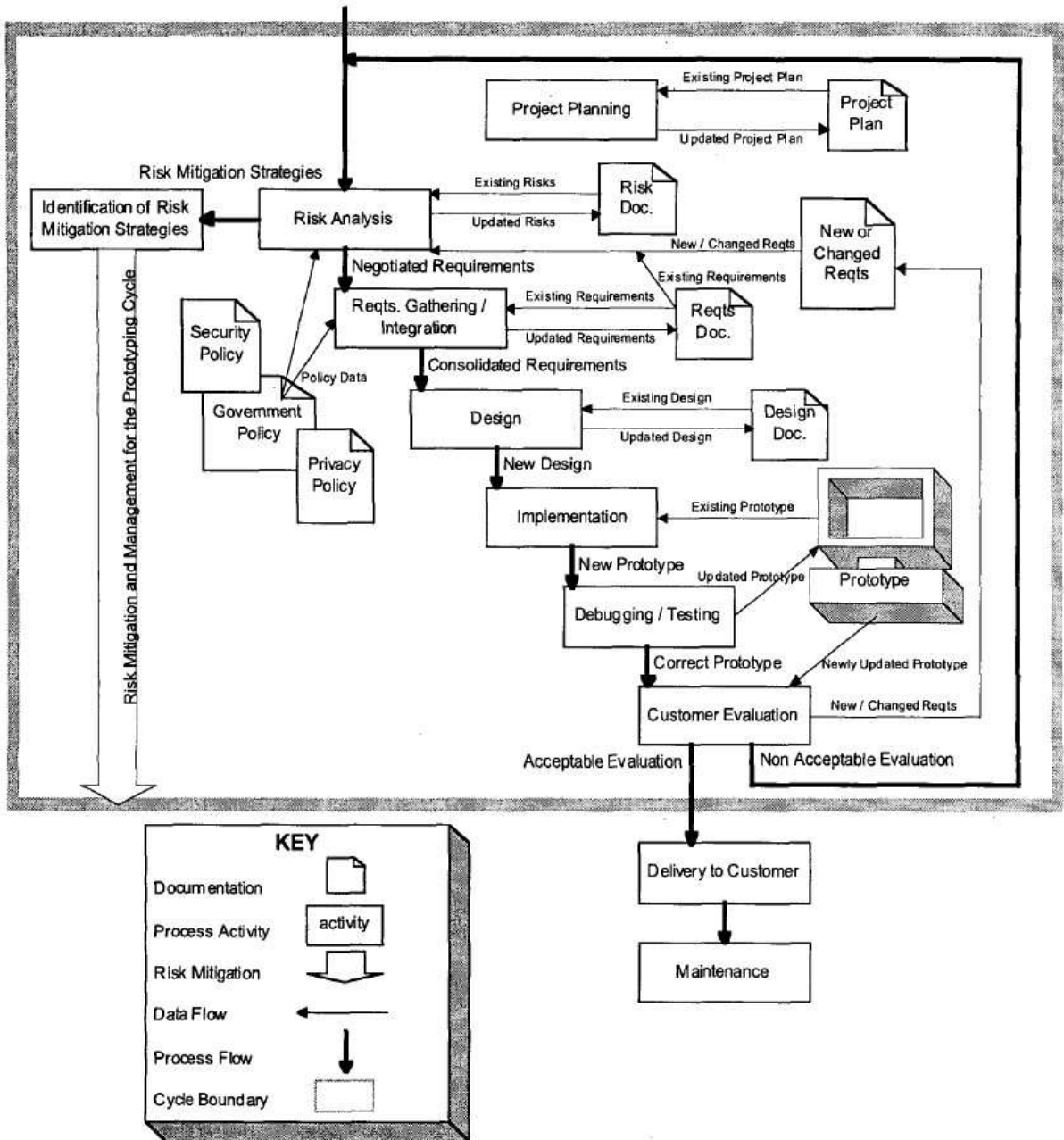


Figure 3: The Evolutionary Prototyping with EPRAM model (Carter et al, 2001)

## 2.8 Development Environments

### 2.8.1 MATLAB and GUIDE

MATLAB was chosen as the primary development environment for the system since the demarcation algorithm produced by Atkin-Granville (2012) was also coded in this language. Through its toolboxes MATLAB provides a number of powerful functionality for image processing, handling externalities such as Java classes, and generating reports from within the application. As already stated the project is using OO conventions for its code and the MVC pattern for its GUI. This is possible within the MATLAB environment, although the support is quite poor (McGarrity, 2008).

GUI construction can be done in two ways within MATLAB. The first of which is directly creating the GUI programmatically, this method was briefly investigated. But it was found not to be conducive to the rapid evolutionary prototyping environment in which the project is operating, and thus the second approach was adopted, MATLAB's Graphical User Interface Development Environment (GUIDE). However, since GUIDE is designed to be used in a procedural fashion its compatibility with OO and MVC is limited, as such this project utilises the modified MVC pattern that Schwarz (2013) describes. Where the individual GUIDE GUI's are wrapped within the view object, enabling the main program to operate the MVC pattern, whilst the `View` class provides the abstraction away from GUIDE's underlying procedural GUI implementation.

The project must be aware of the deployment of the application, as such ensuring an easy transfer from the development system to the next was desirable. MATLAB provides the ability to compile a program into a package which can be distributed to other systems. This means that the application can be sent into the clinical environment as a simple one file install, since MATLAB operates independent of the operating system to provide a consistent end user experience (Smith, 2006).

## 2.8.2 VTK

The Visualisation Toolkit (VTK) was chosen as the library in which the 3D visualisation for this project was to be done after experimentation and research into using MATLAB's own 3D rendering. Despite MATLAB's integration with Open GL the rendering of the datasets that this project required appeared to take too long to be considered functional within the intended deployment environment. As such other avenues were explored resulting in the adoption of VTK as the rendering library.

VTK is open-source, a freely available software system for 3D computer graphics, image processing and visualization written in the C++ language. The VTK is extensively used in medical research where 3D visualisation is required, since it is effective in terms of cost and performance (Wee et al, 2011). VTK is also a core part of the National Alliance for Medical Image Computing's (NAMIC) roadmap initiative for future computing tools. Forming a part of its 3DSlicer application (NAMIC, 2014).

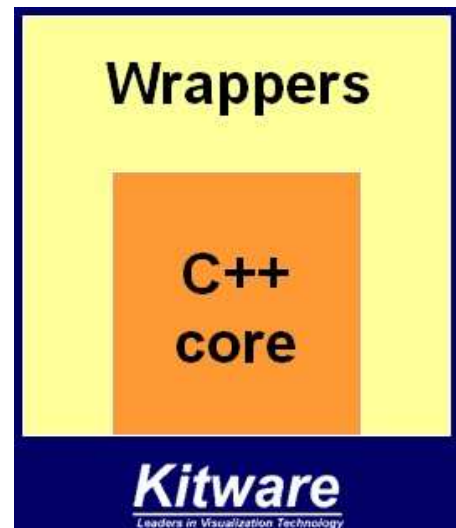


Figure 4: VTK structure (Schroeder, 2005)

Initially VTK was implemented through the adoption of MatVTK, a direct linking of the C++ library to MATLAB through the use of MEX code (Birngruber, 2009). The inflexibility of this method resulted in the use of the compatibility which MATLAB provides for the inclusion of Java classes within its application (MathWorks, 2014). Since VTK can be compiled into a Java packaged using the Java Native Method Interface, the library was then able to be utilised in full since a java class could be written to handle the interface between MATLAB and the VTK java wrapper.

### 3 Implementation

This chapter details the implementation section of the project. Each prototype will be discussed along with the initial pre-client meeting mock-ups and experimentation. The key decisions made about elements of the implementation will be stated and the reasoning behind them made clear.

#### 3.1 Currently Available Systems in MATLAB



Figure 5: Viewer3D's greyscale 3D render

MATLAB's FileExchange hosts a number of applications written in MATLAB code which are free to use, before the client meeting was held exploration had begun into what options already existed which could complete the objectives of the project. One application which fitted the objectives and requirements well was viewer3D. The application was however quite slow to produce its 3D renders (See Figure 5), even when Open GL was activated and despite including a "compile c-code" option to boost speed the application failed to

provide a real-time render of the sample dataset (Kroon, 2011). Despite that flaw the application provided support for numerous file type (see Appendix 7), segmentation, distance measurements, and Region Of Interest (ROI) selection in 2D slices, most of which were required as a part of this project (Kroon, 2011). As a result Viewer3D is used as an exemplar to build the 2D visualisation elements of the project. The following functionality is included within the project design since the viewer3D proved its benefit to usability.

- Contrast manipulation
- Windowing

The inclusion of contrast manipulation is supported by Abdallah & Siddig (2013), Zavaletta et al (2007), and Lerman et al (2006) who says that since the distribution of grey levels in a typical DICOM CT scan image means only 12.5% of them are used to represent soft tissue, which is the target of the radiologists attention, such poor distributions "reduces the overall contrast and the texture differences between individual organs, and poses a serious visualization problem since radiologists need clear visual representations of organs to produce proper diagnoses" (Lerman et al, 2006, pg 1).



Figure 6: Viewer3D's multi-window mode

The Windowing which is shown in Figure 6 is a design element which is incorporated into the project, all be it in a slightly altered form. The viewer3D windows are simply the partitioning of the main window into segments, in the project this will be altered such that the tools and renders are each in their own window so that the user can view only what is needed at that time. This maximises the screen space for the render which is the main area of interest to the radiologist.

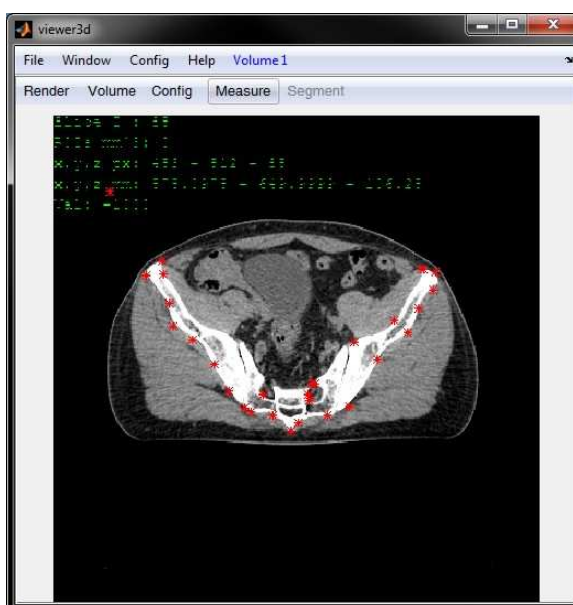


Figure 7: Viewer3D's ROI construction points

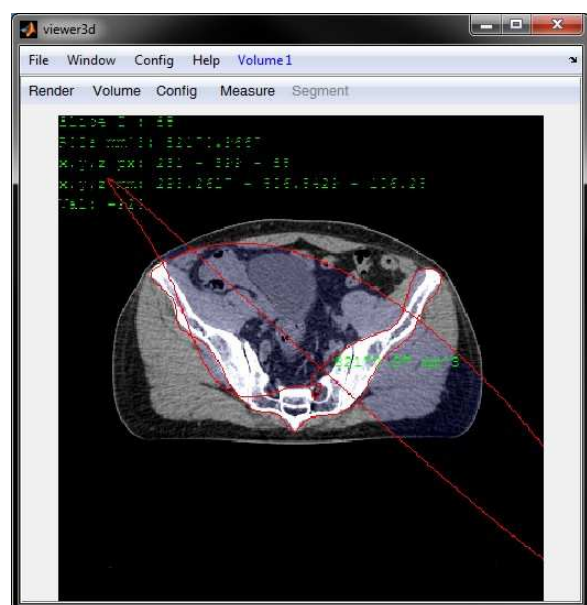
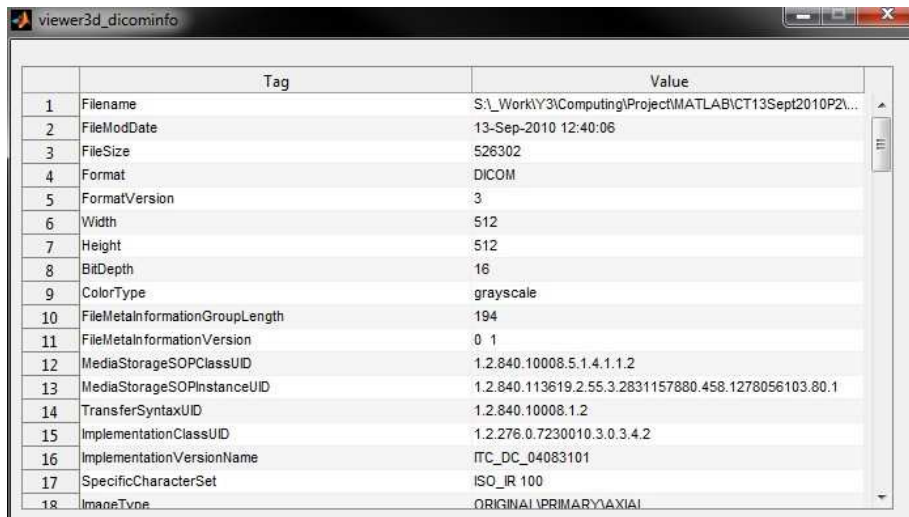


Figure 8: Viewer3D's ROI extrapolated from points

The creation of ROI boundaries in Viewer3D is done by the placement of single points in sequence (see Figure 7) which is then extrapolated to generate a curved ROI boundary. This boundary is highly inaccurate (see Figure 8) and thus completely inappropriate for use this project. However, the point manipulation, dragging construction points around, is a usability feature which is to be included.



	Tag	Value
1	Filename	S:\_Work\Y3\Computing\Project\MATLAB\CT13Sept2010P2\...
2	FileModDate	13-Sep-2010 12:40:06
3	FileSize	526302
4	Format	DICOM
5	FormatVersion	3
6	Width	512
7	Height	512
8	BitDepth	16
9	ColorType	grayscale
10	FileMetaInformationGroupLength	194
11	FileMetaInformationVersion	0 1
12	MediaStorageSOPClassUID	1.2.840.10008.5.1.4.1.1.2
13	MediaStorageSOPInstanceUID	1.2.840.113619.2.55.3.2831157880.458.1278056103.80.1
14	TransferSyntaxUID	1.2.840.10008.1.2
15	ImplementationClassUID	1.2.276.0.7230010.3.0.3.4.2
16	ImplementationVersionName	ITC_DC_04083101
17	SpecificCharacterSet	ISO_IR 100
18	ImageType	ORIGINAL\PRIMARY\AXIAL

Figure 9: Viewer3D's DICOM information pop-up window

Figure 9 shows the data carried within a single DICOM file or slice of the CT scan. The majority of this data is duplicated and thus in the project implementation only a single slice worth is held to reduce the memory requirements. However, the data held within the file is essential for correctly rendering the data and is to be utilised in the reporting objective of the project.

### 3.2 Mock-Up using MatVTK

As already stated in the methodology chapter, the client meeting used the results of the exploration into currently existing applications to produce a simplistic version of the application using predominantly premade components from Matlab's file exchange. This enabled the client to visualise what could be possible, which is an element that is encouraged by Schneider's (2007) methodology since it increases the speed of requirements elicitation.

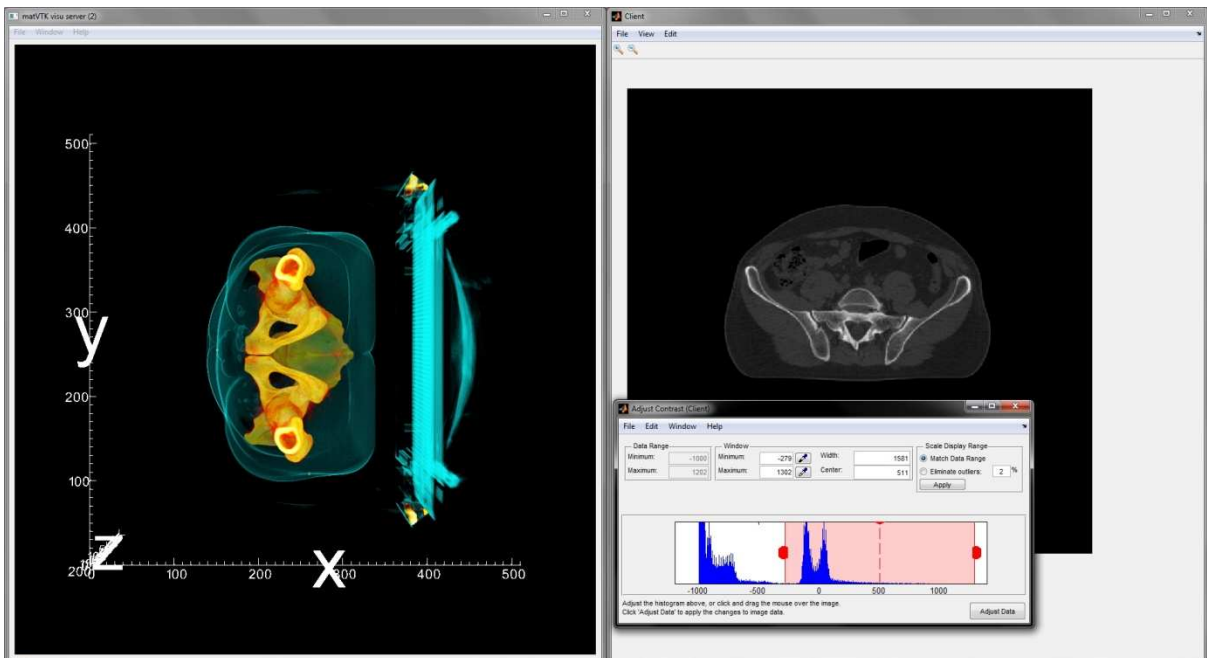


Figure 10: Application Mock-Up using MatVTK

This mock up had basic pan, zoom and rotate functionality in its 2D and 3D renders. It also included contrast manipulation through MATLAB's Image Processing Toolbox (IPT) (See Figure 10). Whilst this mock-up functioned well enough for demonstration purposes it wasn't a stable enough platform from which to build the subsequent prototypes. The mock up used VTK through the MatVTK package which Birngruber (2009) had created. This package provides improved rendering speed and quality over Veiw3D. However, the methodology which is required to implement MatVTK means a separate C++ exe was required to act as a localhost server for the actual VTK rendering (Birngruber, 2009). This hindered the portability of the program and was one of the reasons why MatVTK was not used in the subsequent prototypes. The other reason was the inflexibility that the server application and the MATLAB MEX code provided for 3D rendering. Since MatVTK isn't perfect for the projects goals a more flexible implementation method for VTK was required, so subsequently the prototypes developed were constructed on the basis that the VTK library would be integrated through the use of its Java wrapper and MATLAB's inbuilt Java support.



### 3.3 Application Prototype 0.0.1

This initial prototype was constructed using the MVC pattern as described by Schwarz (2013), providing the proof of concept for the underlying MVC design which underpins much of the later application components. To this end this prototype consisted of the classes which constitute the MVC pattern, the `Model`, `Control` and `View`. To test the MVC structure, loading of DICOM data was implemented using MATLAB's IPT's `dicomread` function. This is the chosen method of implementation since the MATLAB method has been shown to be a robust way of reading the data from a DICOM file into the MATLAB environment. It would be impractical and beyond the scope of this project to implement a separate load function since the DICOM standard is complex (NEMA, 2011). MATLAB's IPT also offers a `dicominfo` function which reads the non image data included in a DICOM file. This function was utilised when reading the DICOM data into the application, in order to provide the correct dimensions for the 3D volume i.e. the scans height, width and depth (or number of slices). Since at this stage in development the application is only intended to provide proof of concept for the MVC pattern in MATLAB with the GUIDE GUI, the display of only a single slice was implemented. In this way the prototype shows that user input could be taken by the system control GUI via the `View`, processed by the `Control` which could trigger changes in the `Model` and as a result trigger the `View` to display a second window rendering the 2D slice with data from the `Model`. This proved that the MVC pattern operated under the constraints of the MATLAB and GUIDE environment.

### 3.4 Application Prototype 0.0.2

One of the main features which the application required was the ability to create hand drawn boundaries around ROI. Atkin-Granville (2012) investigated a few different methods of capturing the boundary from user input, firstly individual points chosen to form the boundary, and after further consultation with the clinical staff, a type of free hand pixel drawing. Learning from this the application utilises a freehand drawing method, powered by `imfreehand`, which is then used as the basis for the construction of a polyline, using `impoly`, to provide a intuitive method for subsequent editing. To improve the usability of the freehand drawing element of the boundary construction process, the user is able to construct a boundary out of multiple freehand segments which are automatically connected together (See Figure 11). This allows for greater accuracy, since if a mistake is made only a single segment must be deleted and not the entire boundary. The deletion of the last drawn segment is tied to a keyboard hotkey for efficiency. This method however suffers from the drawback that all segments must be drawn in the same direction, i.e. clockwise or anti-clockwise. This is due to an assumption made, about the order of the points in the data structure being related to their position in the boundary, when generating the polyline.

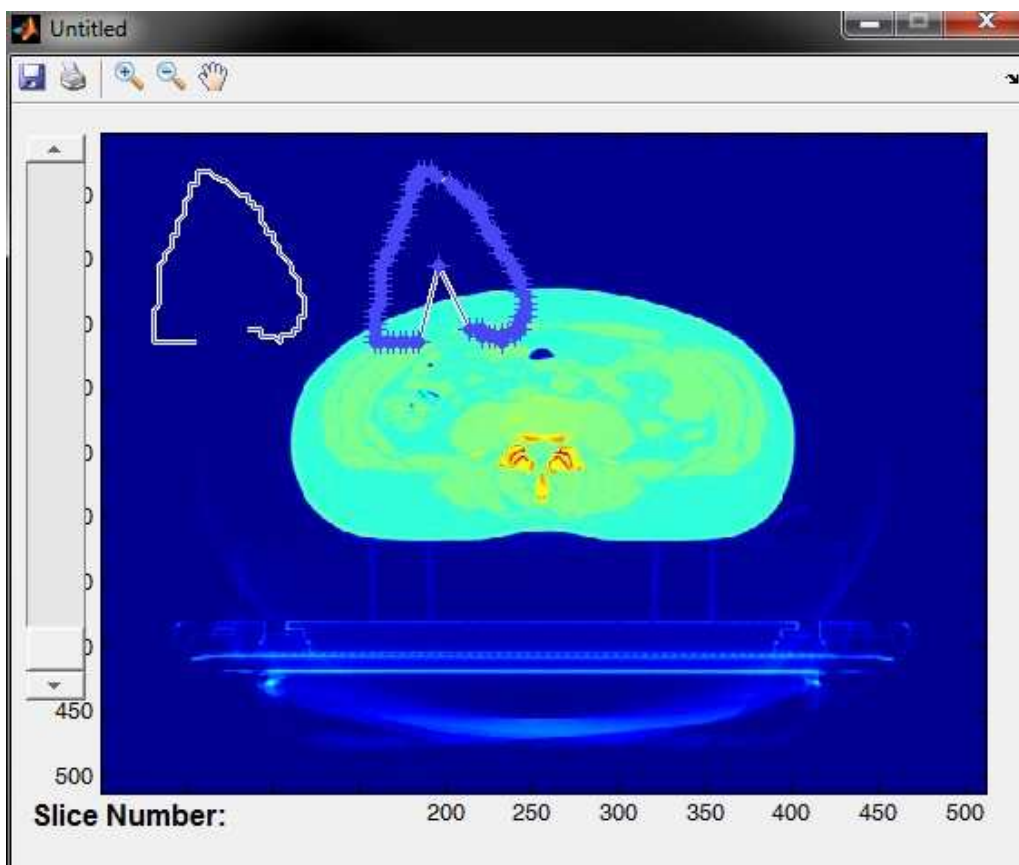


Figure 11: Early implementation of boundary drawing

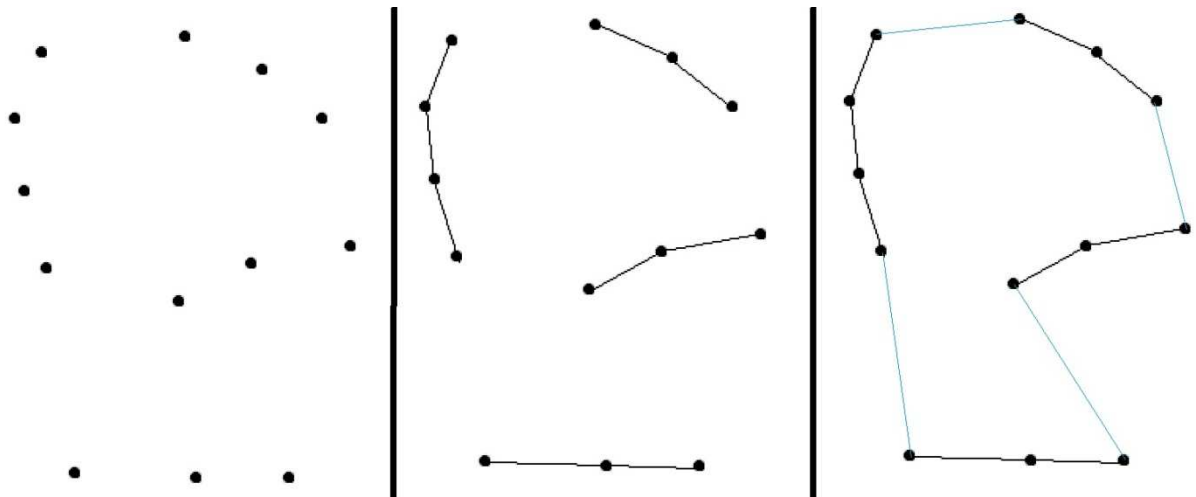


Figure 12: Boundary resolution from points cloud

Figure 12 shows a simplified example of how a boundary is constructed in this prototype. The far left set of point represents at an abstract level the points which make up a boundary. It is possible to derive a convex hull from these points using a well defined computational method. However, as Atkin-Granville (2012) found in his project, a convex hull only provides a crude shape for the points

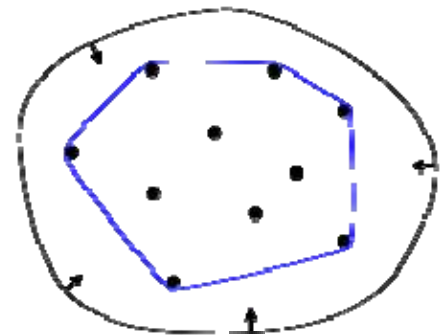


Figure 13: Convex hull (Matan, 2013)

it is looking for the smallest boundary which contains all the points. So a method for capturing the shape which the points form in fine detail is needed.

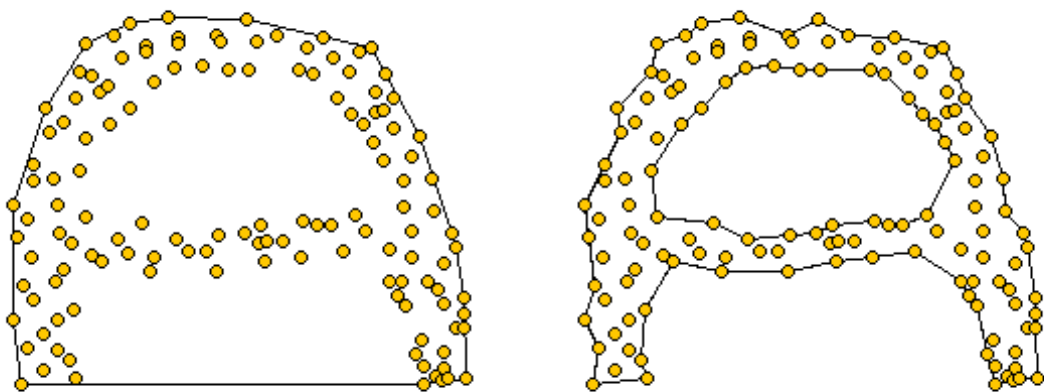


Figure 14: Convex hull vs. Alpha shape (Graser, 2011)

The idea of an alpha shape is that it is the intuitive 'shape' formed by a set of spatial point data (See Figure 15). Unlike an iso-surface an alpha shape is a concrete geometric entity which is unique to that particular set of points.

*Given a finite point set  $S$ , and a real parameter alpha, the alpha shape of  $S$  is a polytope which is neither necessarily convex nor necessarily connected. The set of all real numbers alpha leads to a family of shapes capturing the intuitive notion of "crude" versus "fine" shape of a point set. For sufficiently large alpha, the alpha shape is identical to the convex hull of  $S$ . As alpha decreases, the shape shrinks and gradually develops cavities. These cavities may join to form tunnels and voids. For sufficiently small alpha, the alpha shape is empty. (BioGeometry, 2002)*

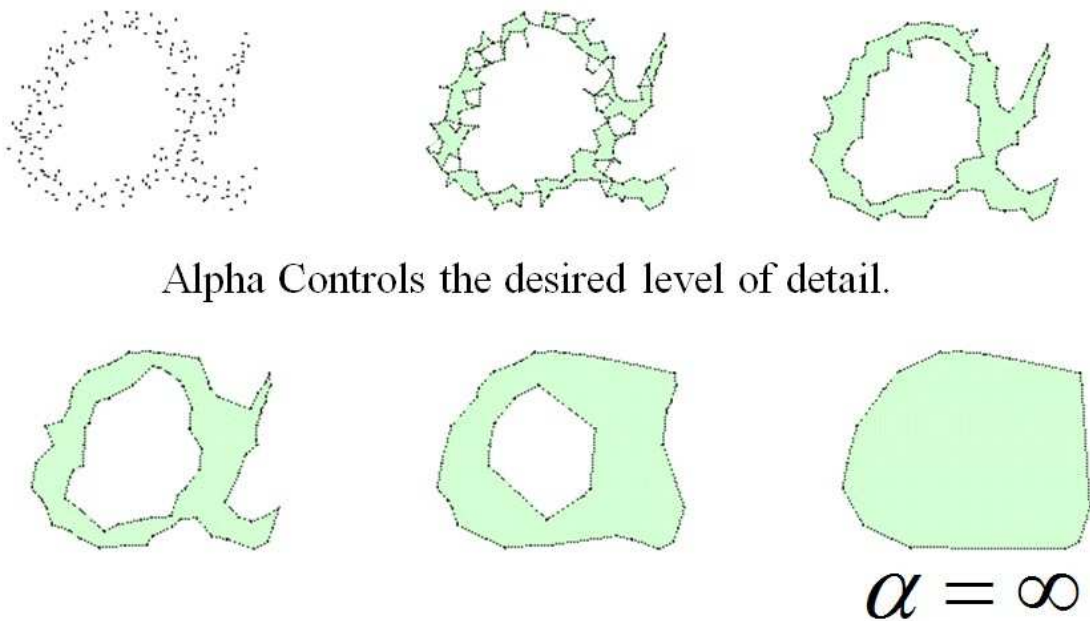


Figure 15: Alpha shape formation (Kumar, 2013)

The generation of alpha shapes such as the one pictured on the right in Figure 14, requires a moderately complex algorithm which still requires user input to adjust the output, so that the correct shape is defined (Duckham, 2008). This method could have been useful in this prototype had it not been for the freehand segment drawing which provides some topographic information about the points (See the middle image of Figure 12). Since this information is known it is reasonable that the prototype makes the assumption about the ordering of the points, since this creates a more accurate boundary than generating an alpha shape based on the point cloud (See the far right image of Figure 12).

The MATLAB IPT methods are used for the initial boundary drawing, since they provide a high level of accuracy and in the case of `impoly` handle, internally, the manipulation of the boundary data structure. Which means the manipulation of boundary points is automatically represented in the boundary data structure. Within the data model the boundaries are stored as simple 'spaghetti' data, which is an ordered collection of x, y coordinate pairs. This method was chosen since it is the output format of the objects used to represent the boundaries by the ITP functions used. So the re-instantiation of the boundaries later is straightforward.

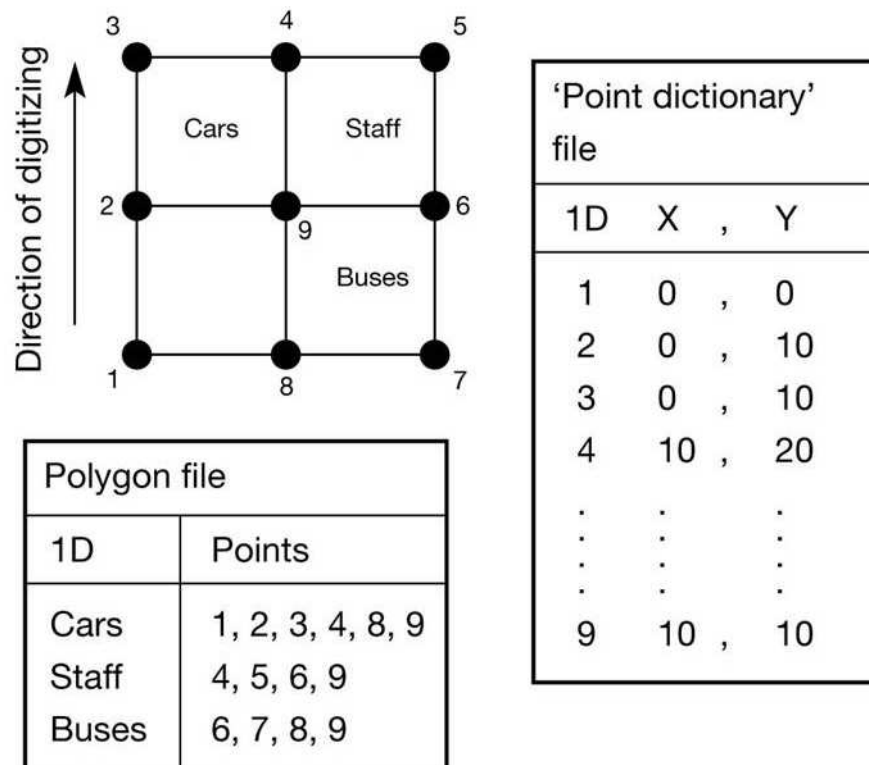


Figure 16: A simple topographical vector structure and database (Indiana University, 2014)

However, other methods for storing the boundary data had been investigated before this method was decided upon. Given that the points which make up the boundaries are inherently topographically related, the investigation looked to the data structures used by Geographic Information Systems (GIS) for inspiration. The structure illustrated in Figure 16 shows a common structure which was considered for storing the boundary data. In this example each point which makes up the boundary is stored in a relational database table, with the 2D polygon (the boundary), being constructed from a subset of those points. This enables additional information to be stored with the boundary data. Using this method it would also be possible to calculate topographic information such as adjacency. This method was rejected since the complexity of maintaining a database was something which in this prototype wasn't appropriate, since portability is a key concern. If a database had been used then MATLAB's database toolbox would have been used and this is designed to function with an external database server or file, this creates external dependencies which would hamper the portability of the prototype.

### 3.5 Application Prototype 0.0.3

The deletion of boundaries is included in this implementation of the prototype. This is done by tagging a boundary with a string which recorded its global ID. The following syntax was used "b\_#" where # is the ID number of the boundary. This enabled multiple boundaries to be created and manipulated independently since they could be individually identified. The deletion of selected boundaries is also implemented using the tags since the selection of an active boundary is made possible. The ability to have multiple boundaries present on a single slice is important since anomalies can occur in multiple locations and all must be localised by the radiologist. The active or selected boundary is shown by using the colour red to make it visually distinct for the user (See Figure 17).

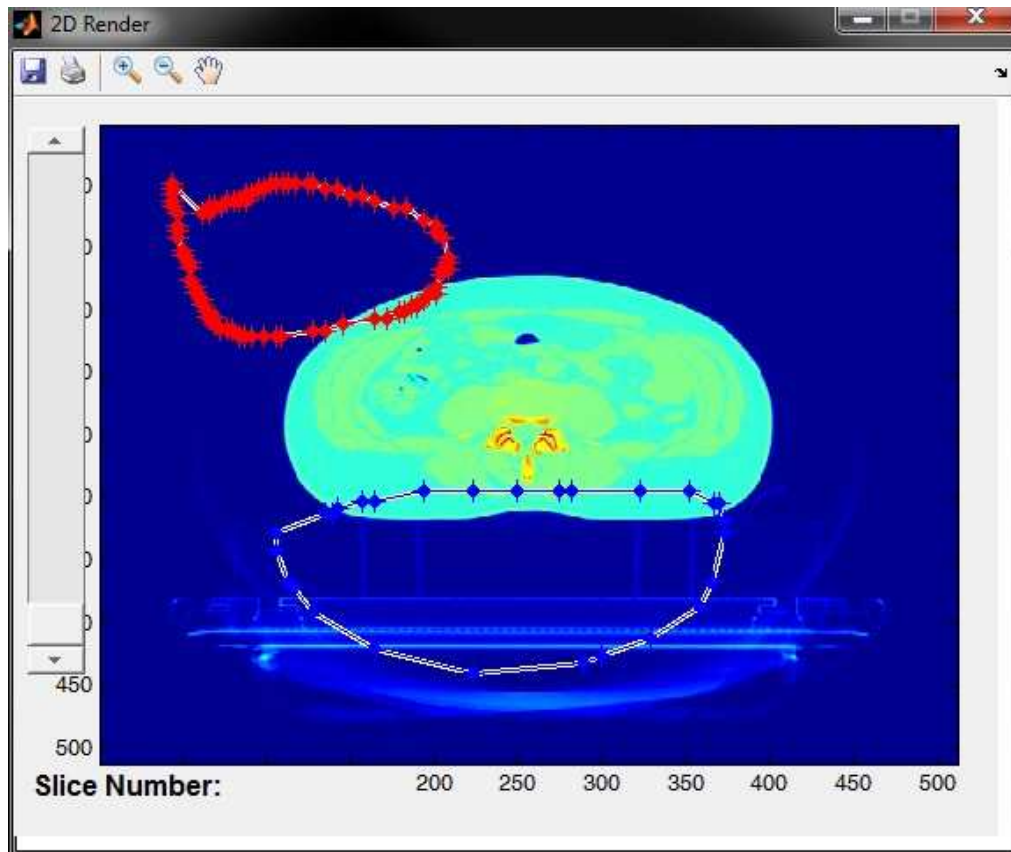


Figure 17: Selected boundary for deletion

### 3.6 Application Prototype 0.0.4

Both the evaluation of Atkin-Granville's (2012) work and the client meeting brought up the manipulation of contrast as a tool which would be useful to the clinical staff. Lerman et al's (2006) work on contrast manipulation for CT scans provides a number of different methods for automatic contrast enhancement with regard to soft tissue in CT scans; unfortunately their work doesn't provide a clear 'winner' out of the methods they evaluated. Since the evaluation of these methods is beyond the scope of this project, automatic contrast enhancement is not included within this prototype. However, the functionality for a user customised contrast enhancement for a single slice is implemented. This is achieved using the powerful contrast adjustment tool MATLAB provides in the IPT, `imcontrast`. Using this method a GUI is automatically generated which displays a value histogram for the current slice (See Figure 18). The user can then manipulate a 'window' generated by the maximum and minimum data values, to see in real time the contrast adjustment of the slice. The adjustment can then be 'saved' to the application. In this way the prototype includes a highly intuitive and usable implementation of contrast manipulation, despite the need for direct user involvement in the adjustment process.

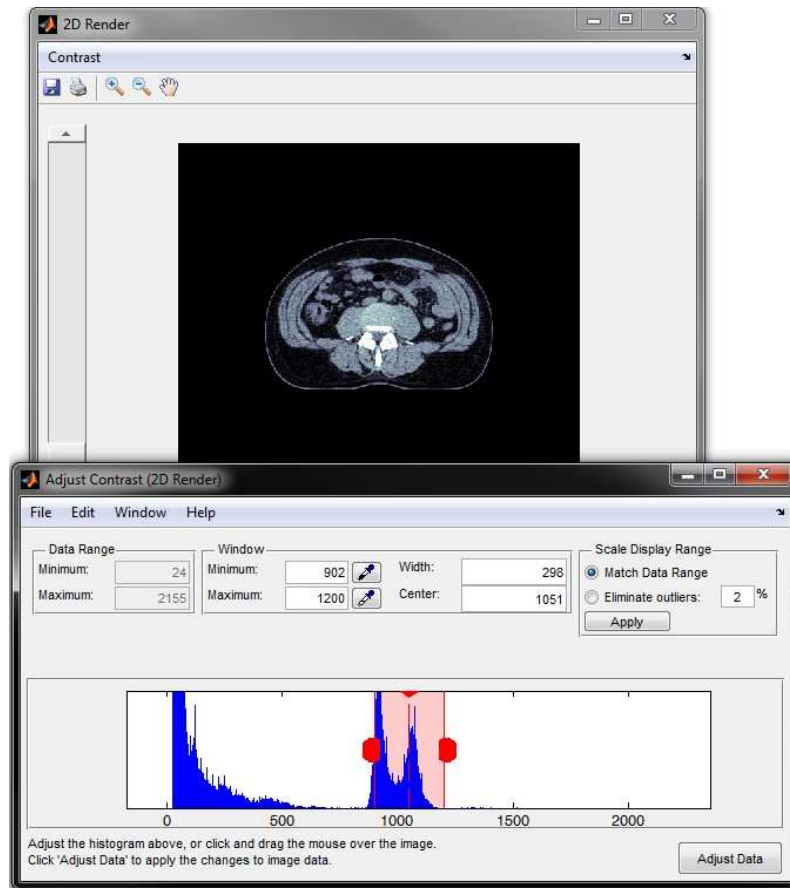


Figure 18: Contrast manipulation of 2D CT scan render

### 3.7 Application Prototype 0.0.5

Since a DICOM image consists of multiple slices this prototype implements the display of consecutive slices in the 2D GUI. The movement between the images is achieved using the scroll bar and user input via a text box. The scroll bar can be used to jump one slice or ten depending on where clicked, and the text box both displays the current slice and enables the user to jump directly to a specific slice. These methods were chosen since they provide a natural way to move through an image set. The client meeting with Dr Sule-Suso, brought up this as the method currently used by radiologist so its inclusion in this prototype should remove any learning curve for the radiologists.



### 3.8 Application Prototype 0.0.6

This prototype implemented the saving of the data held in the data model. This was achieved using MATLAB's `uisave` function, since this enables the user to choose the save location and the file name saved under. Using `uisave` meant that only a workspace variable could be saved, so the state of the entire data model is saved into a `.mat` file. This includes the entire image volume, the DICOM info, and boundary data. The use of MATLAB's file type means that the data is encoded using a compression algorithm which results in a relatively slow file IO, since saving and loading are one off events during the life time of the prototype this is an acceptable cost for the reduction in file size.

There were a number of alternate methods considered for saving, including generating masks from the boundaries; this would have been a quick and efficient saving method since only slices with boundaries would have to be saved. However, such a saving method wouldn't include any changes that had been made to the DICOM contrast, resulting in the need for the user to duplicate their actions. If masks had been used then such changes to contrast could have been saved separately but this would have required a separate file, which is an added complexity for the user and is ultimately unnecessary since single file saving is possible.

### 3.9 Application Prototype 0.0.7

This prototype fixed a serious bug relating to the saving of boundaries. The `imply` function stores its point data within a wrapper object, which is tied to the 2D GUI. When the save use case occurs only the pointers to these objects were written to file. The fix simply required changing the data model so that it extracted the point data from the object and stored that along with its tag. One adverse effect of this change is that when a slice, with a boundary, is navigated to in the 2D GUI, a new wrapper object has to be created; this is a small amount of processing but necessary.

This prototype also includes the exporting of the boundary data in a txt file for use in other applications. The file is constructed as a simple CSV, including the tag, slice number, and point's data in each record (See Appendix 8 for an example). This was chosen since it is a simple format to save into and likely has high compatibility with other applications.

Changes were also made to reduce the memory used by the Model both during runtime and when saved to reduce system load. This was achieved by only storing one copy of the DICOM information since it is almost identical across every slice, other than slice ID and position data. Given this, only the first slice's information is retained within the data model after loading. Holding all the information in memory during loading is done due to a quirk of the DICOM format. The assumption had been made that the DICOM filenames dictated the slice order. However, it is clear from scrolling through the slices this isn't the case. The first attempt at a solution was to use the slice ID's from the DICOM information however, they also didn't match the correct order, and so all DICOM information is retained so that the slices can be ordered using the absolute position of each slice. Once this has been achieved all but the first slice's information is dropped.

As stated this quirk was picked up after scrolling through the slices, this prototype enabled the use of the mouse wheel for scrolling, the additional speed this offers was something which Dr Sule-Suso requested during the client meeting but it is what also enabled the spotting of the mixed up slice order since the images didn't scroll smoothly, during testing, as would be expected of a correctly ordered set.

### **3.10 Application Prototype 0.0.8**

This prototype version implemented the final objective of the project, reporting from within the application. This is achieved using MATLAB's Report Generator Toolbox (RGT) since it allowed for multiple output types and has direct access to the workspace variables of the application. Other approaches were investigated but they required external 'local servers' to be running to process the information output by this application and thus impacted on portability. MATLAB's RGT however, isn't an ideal solution since it is lacking an easy way for the user to modify the styling and content of the report as was desired by the requirements of the project. This compromise was made since the reporting features of the application were aimed more towards the research stakeholders. Once the automatic segmentation and other features are added to the program in the future the ability to output internal system variables and capture systems states would be a sought after feature. Given that context the lack of a 'user friendly' method of styling the report isn't as troubling since the 'bare bones' of the application will be exposed anyway, and the technical expertise will be available to customise the report through MATLAB's native GUI.

Additions also made in this version made changes to the usability of the application. Numerous tool tips and error messages were added to inform the user of mistakes and reasons for actions failing. The boundaries GUI had the instructions for use added to remove the need for an instruction booklet. A few of the longer processes had process bars added to indicate the completion to the user ensuring that the application wasn't terminated part way through an operation by mistake. All of these additions work towards the first objective of a user friendly interface.

Finally this prototype implemented a slight change to how boundaries were processed within the application, following a similar principle to that used in UNIX's VI Editor, a screen-oriented text editor. The storing of the boundary data is changed to use a method similar to the edit buffer, which VI uses (IEEE & The Open Group, 2013). In this way the edits the user makes are only committed to the data model when a boundary is completed, for this prototype the commit is automatic and there is no roll back functionality since it was decided, at this stage lacking user confirmation, that having to separately commit each modification would be a usability issue for time starved radiologists.

### 3.11 3D Visualisation Prototype

As has been previously stated the VTK library is the engine used for 3D rendering by this project. The VTK library provides an environment through its wrapper to open windows on the users machine (See Figure 19). This is used to fulfil multi-window requirement which was set out in the client meeting. Furthermore, the pipeline structure of VTK's rendering process (See Figure 20) is used to transform the DICOM images imported, as VTK image data (See Appendix 9), to produce the output 3D renders as seen in Figure 21 and Figure 22.

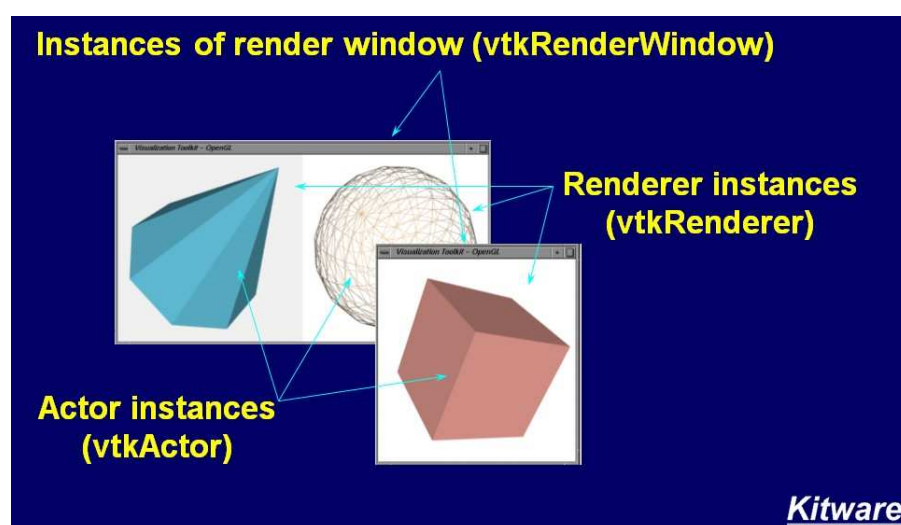


Figure 19: VTK window environment (Schroeder, 2005)

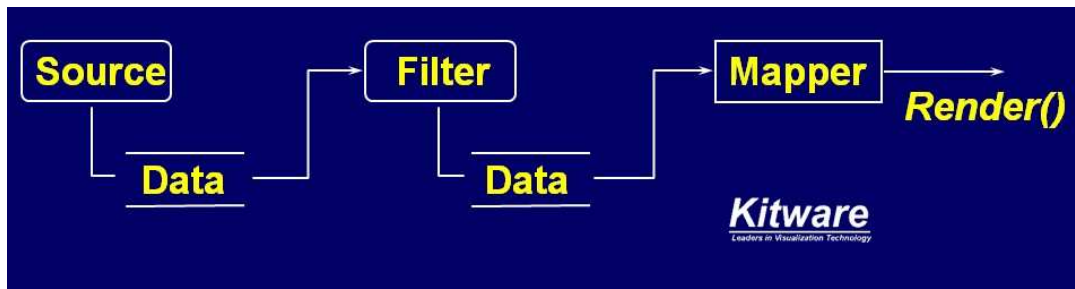


Figure 20: VTK render pipeline (Schroeder, 2005)

Two main methods of 3D rendering were attempted for this project. Firstly, direct volume rendering was used to visualise the voxel data of the DICOM, using ray-casting on the GPU. This provides a direct representation of the CT scan image however; the information contained within the render results in too complex of an image for use in this application. Secondly, iso-surface rendering was attempted, where a threshold value of radio-density, preset to that of soft tissue, is used to construct a 3D image using edge detection. This method can include reference points such as bones which can be added simply by including a second radio-density value displayed in a different colour, whilst also removing superfluous information such as muscles. Unfortunately, the level of noise and uncertainty surrounding the soft tissue (See Figure 21 and Figure 22) was such that instead of fully integrating 3D visualisation into the prototype implementation, time was spent investigating methods which could reduce the noise.

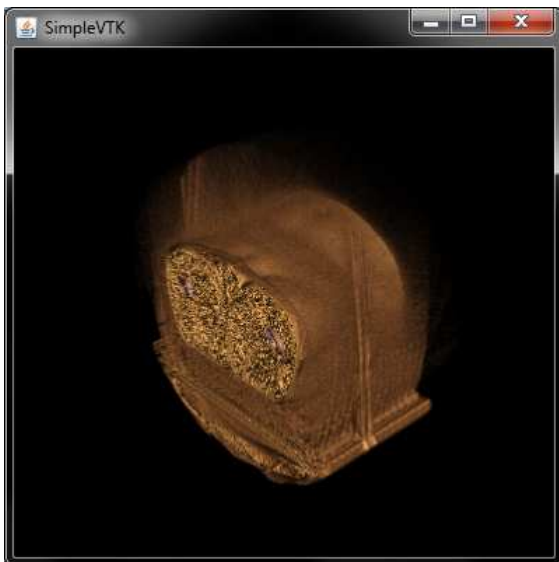


Figure 21: VTK volume render

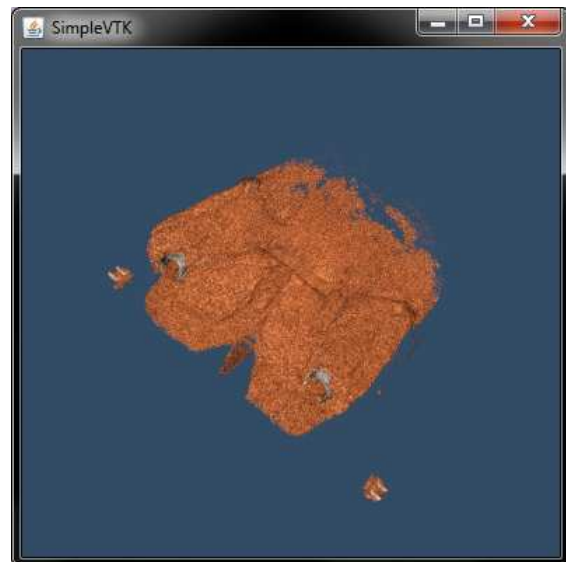


Figure 22: VTK surface render of soft tissue and bone

Numerous methods of noise reduction were looked at. For example, Zhen & Guoli's (2010) *Application of Hybrid Filter in CT Image Processing Based on Visualization Toolkit*, before focusing in on wavelet based methodologies since they had support from the literature, within which a number of different methods have been suggested (e.g. Reddy, 2013; Jumah, 2013; Borsdorf et al, 2008; 2006). After a project meeting with Dr Lam it was decided not to attempt to implement such noise reduction techniques since as the quote from Semmlow (2004) on the title page of this report indicates diagnosis made from CT imagery is reliant on the quality of the computationally reconstructed image. Adjustments of this nature should occur during initial CT image reconstruction, where all raw data is present and can be utilised, ensuring the quality of the output image.

### **3.12 Evaluation & Testing**

Carter et al's (2001) ERAM model describes testing and evaluation occurring with the stakeholders involvement at the end of each prototyping cycle so that "they may judge whether or not it meets their expectations" (Carter et al, 2001, pg 96). Within the context of this project such a meeting is impractical due to the pressures on Dr Sule-Suso's time; as a result the project meetings act as both project management, testing, and evaluation sessions. Since the project meeting is held with just the project supervisor Dr Lam, it is a possibility that this could bias the evaluation in favour of the non-clinical goals. However, due to Dr Lam's long professional history with Dr Sule-Suso and this research topic, it is fair to assume that Dr Lam can act as a proxy for Dr Sule-Suso's comments. During these meetings the current prototype is demonstrated, progress discussed, and evaluated against the project aims. As a result of these meetings new requirements are generated and risk assessed, once that has been done the project repeats the prototyping cycle once more until no new requirements are found. The review of risk is in line with Carter et al's (2001) work since as is stated "this iterative approach all but invites requirements creep, which [should be] manage via risk analysis and mitigation" (Carter et al, 2001, pg 96).

Testing during the prototype's development is done in the white-box format, since the developer knows the application logic. This testing is done to ensure that the accepted test cases function as expected. For this project this was applied to units, which in the OO programming style are considered as an interface, from an entire class down to a single method (Xie et al, 2007). In this way the testing ensures that all core function points of the prototype are covered (Cornett, 2014). Black-box testing is done using users who attempt to use the prototypes functionality without knowledge of the underlying code (Patton, 2005). This project used such testing to check GUI usability and to test unexpected use cases (See Appendix 10 for example evaluation form). The black-box testing in this project was done by non-clinical staff and thus doesn't test the application's suitability to the clinical work environment. To remedy this omission a questionnaire is planned for Dr Sule-Suso to complete after viewing a prototype demo, this method was chosen since it is sensitive to the restraints on Dr Sule-Suso's time.

However, if an evaluation meeting with Dr Sule-Suso could be organised, then this evaluation and testing procedure would correspond with informal A/B testing as described by Brodovsky & Rosset (2011). Where a baseline group is compared to a number of single variable test samples, the informality in this instance is from the lack of statistical analysis undertaken. Such an A/B testing regime would provide useful information with regard to possible future directions for the prototype application.

## 4 Project Summary & Future Recommendations

In summary this project has fulfilled its aim to create a prototype application which is sensitive to the working environment of clinicians' in order to provide for the expansion and continuation of the research on the automated detection of cancer abnormalities within CT scans. This was achieved using feedback from previous projects and using the knowledge of clinical staff, gathered both in person and from literature on radiologist's workstations and their working environment. As a result three modules have been prototyped to fulfil the objectives of the project, a slice viewer and boundary creation program coupled with a report generator and 3D visualisation of the same dataset.

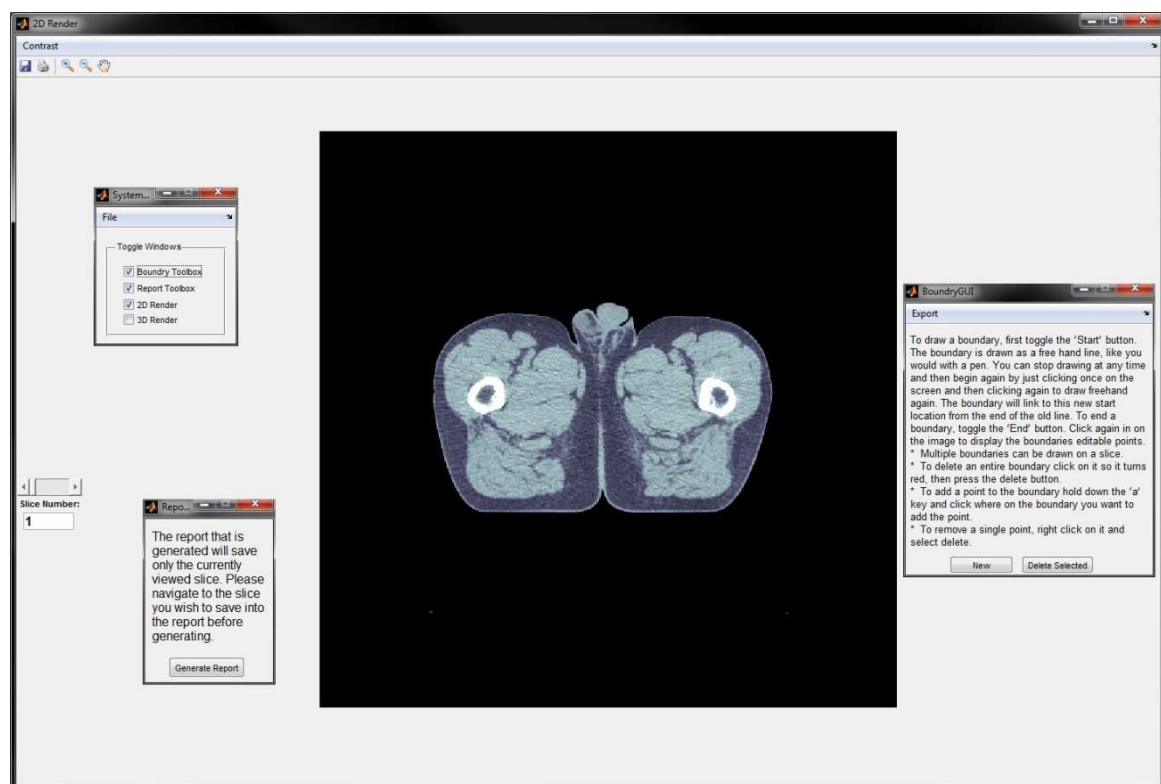


Figure 23: Application Prototype 0.0.8

Although this project can be considered to have met the stated objectives, within the remit of the initial project framework however, time constraints and practicality proved a limitation on complete implementation of two of the projects functional requirements. In the case of 3D visualisation it was fully realised conceptually however, because of time restraints it was not fully integrated into a prototype. This could be overcome with additional time to deliver more prototyping cycles. Further, the report generation was not as practical as initially thought, due to limitations of the implementation method.

In consideration of future recommendation a number of key areas from the project have been highlighted.

- Firstly, the inclusion of Lam et al's (2013) multiscale algorithm, for automatic boundary detection, would enhance the quality and amount of research data available to the research group. This would be providential in future detection and localisation of cancer abnormalities.
- Secondly, enhancements to the report generation in such a way that it is more accessible to non-technical users. This would enable all users to access the functionality of this element of the application, consequently this would develop more in depth summaries and provide the ability to calculate within the application, reducing the time needed externally for research analysis.
- Thirdly, the implementation of 3D changes over time would enable tracking of treatment progress and could be use as an educational tool for patient empowerment and knowledge, and further would be inline the ethos of patient centred care, along the same lines as Dr Sule-Suso's project using the VERT system (NSUH Charity, 2010).
- Fourthly, other Human Computer Interaction (HCI) tools, such as a stylus and pad, would provide a more natural engagement for the radiologist and assist in motor coordination when using the boundary drawing tools, resulting in greater accuracy and speed. This is an enhancement which would require no changes to the application code since such hardware integration can be handled by the operating system's drivers.

In hindsight although the project has met its requirements, if undertaken again a key element of the development process could be changed to provide a more beneficial outcome. This would be to rebuild with a less restrictive programming language such as C++ which would enable the use of VTK libraries without a native interface; meaning processing speed improvements and stronger adherence to MVC and OO, consequently also enabling distribution without reliance on third party applications such as MATLAB.



## 5 References

- ABDALLAH, Y.M.Y. and SIDDIG, M., 2013. Contrast Improvement of Chest Organs in Computed Tomography Images using Image Processing Technique. *Asian Journal of Medical Radiological Research*, vol. 1, no. 1, pp. 39.
- ANDRIOLE, K.P., BARISH, M. and KHORASANI, R., 2006. Advanced image processing in the clinical arena: issues to consider. *Am Coll Radiol*, vol. 3, no. 4, pp. 296.
- ANDRIOLE, K.P., WOLFE, J.M., KHORASANI, R., TREVES, S.T., GETTY, D.J., JACOBSON, F.L., STEIGNER, M.L., SITEK, A. and SELTZER, S.E., 2011. Optimizing Analysis, Visualization, and Navigation of Large Image Data Sets: One 5000-Section CT Scan Can Ruin Your Whole Day. *Radiology*, vol. 259, no. 2, pp. 346.
- ANTON, A.I. and EARP, J.B., 2001. Strategies for Developing Policies and Requirements for Secure Electronic Commerce Systems. In: *Recent Advances in Secure and Private E-Commerce* Kluwer Academic Publishers.
- ARENSON, R.L., CHAKRABORTY, D.P., SESHADRI, S.B. and KUNDEL, H.L., 2003. The Digital Imaging Workstation. *Journal of Digital Imaging*, vol. 15, no. 2, pp. 142.
- ATKIN-GRANVILLE, C., 2012. *Visual Cues for CT Image Segmentation*. Keele University.
- BASKERVILLE, R.L. and STAGE, J., 1996. Controlling Prototype Development Through Risk Analysis. *MIS Quarterly*, vol. 20, no. 4, pp. 481.
- BERBAUM, K., CALDWELL, R., SCHATZ, K., THOMPSON, B. and FRANKEN, E., 2007. Does computer-aided diagnosis for lung tumours change satisfaction of search in chest radiology?. *Acad Radiol*, vol. 14, no. 9, pp. 1069.
- BioGeometry., 2002. *Alpha shapes introduction*. 2002, Available from: <http://biogeometry.duke.edu/software/alphashapes/index.html>.
- BIRNGRUBER, E., DONNER, R. and LANGS, G., 2009. MatVTK - 3D Visualization for Matlab. *The MIDAS Journal - Systems and Architectures for Computer Assisted Interventions*.
- BIRNGRUBER, E. *MatVTK Volume Render [Image]*. Available from: <http://www.cir.meduniwien.ac.at/team/birngruber/matvtk/>.
- BOEHM, T., HANDGRAETINGER, O., LINK, J., PLONER, R., VOELLMY, D.R., MARINCEK, B. and WILDERMUTH, S., 2004. Evaluation of radiological workstations and web-browser-based image distribution clients for a PACS project in hands-on workshops. *Eur Radiol: Computer Applications*, vol. 14, pp. 908.
- BORSODORF, A., RAUPACH, R., FLOH, T. and HORNEGGER, J., 2008. Wavelet based Noise Reduction in CT-Images using Correlation Analysis. *Ieee Transactions on Medical Imaging*, vol. 27, no. 12, pp. 1685.
- BORSODORF, A., RAUPACH, R. and HORNEGGER, J., 2006. Wavelet Based Noise Reduction by Identification of Correlations. *Pattern Recognition Lecture Notes in Computer Science*, vol. 4174, pp. 21.
- BRODOVSKY, S. and S. ROSSET. A/B Testing at SweetIM: The Importance of Proper Statistical Analysis *IEEE 11th International Conference Data Mining Workshops (ICDMW)*. Vancouver, BC, 2011.
- BURBECK, S., 1997. *Applications Programming in Smalltalk-80(TM): How to use Model-View-Controller (MVC)*. illinois.edu: illinois.edu. March 1997, Available from: <http://st-www.cs.illinois.edu/users/smarch/st-docs/mvc.html>.
- CARTER, K.W., MCCASKIE, P.A. and PALMER, L.J., 2008. SimHap GUI: An intuitive graphical user interface for genetic association analysis. *BMC Bioinformatics*, vol. 9, pp. 557.

- CARTER, R.A., A.I. ANTON, A. DAGNINO and L. WILLIAMS. Evolving Beyond Requirements Creep: A Risk-Based Evolutionary Prototyping Model *Fifth IEEE International Symposium on Requirements Engineering*. Toronto, 2001.
- CHAKRABORTY, D.P., 2006. ROC curves predicted by a model of visual search. *Phys Med Biol*, vol. 51, no. 14, pp. 3463.
- CORNETT, S., 2014. *Code Coverage Analysis*. bullseye.com: Bullseye Testing Technology. 2014, Available from: <http://www.bullseye.com/coverage.html>.
- DUCKHAM, M., KULIK, L., WORBOYS, M. and GALTON, A., 2008. Efficient generation of simple polygons for characterizing the shape of a set of points in the plane. *Pattern Recognition*, vol. 41, no. 10, pp. 3224.
- ECKSTEIN, M., THOMAS, J., PALMER, J. and SHIMOZAKI, S., 2000. A signal detection model predicts the effects of a set size on visual search accuracy for feature, conjunction, triple conjunction, and disjunction displays. *Percept Psychophys*, vol. 62, no. 3, pp. 425.
- FREEDMAN, M., 2004. State-of-the-art screening for lung cancer (part 1): the chest radiograph. *Thorac Surg Clin*, vol. 14, no. 1, pp. 43.
- FREEDMAN, M. and OSICKA, T., 2010. Perceptual effect of CAD in reading chest radiographs. In: E. SAMEI and E. KRUPINSKI eds., *The Handbook of medical image perception and techniques* Cambridge: Cambridge University Press, pp. 290.
- FREEDMAN, M., T. OSICKA and S. LO. Methods for identifying changes in radiologists behavioural operating point of sensitivity-specificity trade-offs within an ROC study of the use of computer-aided detection of lung cancer *Proceedings of SPIRE: medical imaging 2001 - image processing and performance*. Bellingham, 2001.
- GRASER, A., 2011. *A Closer Look at Alpha Shapes in pgRouting*. September 2011, Available from: <http://anitagraser.com/2011/09/25/a-closer-look-at-alpha-shapes-in-pgrouting/>.
- GASSER, R., 2009. *VTK Data Types [Image]*. bu.edu: Boston University. September 2009, Available from: <http://www.bu.edu/tech/about/research/training/online-tutorials/vtk/#VISPIPE>.
- HORSCH, k., GIGER, M. and METZ, C., 2008. Prevalence scaling: applications to an intelligent work-station for diagnosis of breast cancer. *Acad Radiol*, vol. 15, no. 11, pp. 1446.
- IEEE and The Open Group., 2013. *The Open Group Base Specifications Issue 7 IEEE Std 1003.1, 2013 Edition*. opengroup.org: The Open Group. 2013, Available from: <http://pubs.opengroup.org/onlinepubs/9699919799/utilities/vi.html>.
- Indiana University., 2014. *Vector data structure [Image]*. 2014, Available from: [http://www.indiana.edu/~gisci/courses/g338/images/vector\\_data\\_structure.jpg](http://www.indiana.edu/~gisci/courses/g338/images/vector_data_structure.jpg).
- JUMAH, A.A., 2013. Denoising of an Image Using Discrete Stationary Wavelet Transform and Various Thresholding Techniques. *Journal of Signal and Information Processing*, vol. 4, pp. 33.
- KOBAYASHI, T., XU, X., MACMAHON, H., METZ, C. and DOI, K., 1996. Effect of a computer -aided diagnosis scheme on radiologists performance in detection of lung cancer nodules on radiographs. *Radiology*, vol. 199, no. 3, pp. 846.
- KROON, D., 2011. *Viewer3D*. FileExchange: MathWorks. January 2011, Available from: <http://www.mathworks.co.uk/matlabcentral/fileexchange/21993-viewer3d>.
- KUMAR, P., 2013. *Alpha shapes [Image]*. compgeom.com: April 2013, Available from: [www.compgeom.com/~piyush/talks/alpha.ppt](http://www.compgeom.com/~piyush/talks/alpha.ppt).

- KUSZYK, B., HEATH, D., BLISS, D. and FISHMAN, E., 1996. Skeletal 3D CT: advantages of volume rendering over surface rendering. *Skeletal Radiol*, vol. 25, no. 3.
- LAM, K., J. SULE-SUSO, R. BHANA and A. MOLONEY. On Evaluation of a Multiscale-based CT Image Analysis and Visualisation Algorithm *6th International Conference on Biomedical Engineering and Informatics*. China, 2013.
- LERMAN, R., D.S. RAICU and J.D. FURST. Contrast enhancement of soft tissues in Computed Tomography images *SPIRE 6144 Medical Imaging: Image Processing*. San Diego, 2006.
- MATAN, A., 2013. *Concave Hull [Image]*. [gis.stackexchange.com](http://gis.stackexchange.com): [gis.stackexchange.com](http://gis.stackexchange.com). July 2013, Available from: <http://gis.stackexchange.com/questions/1200/what-are-definition-algorithms-and-practical-solutions-for-concave-hull>.
- MathWorks., 2014. *Bringing Java Classes into MATLAB Workspace*. [mathworks.co.uk](http://mathworks.co.uk): MathWorks. 2014, Available from: [http://www.mathworks.co.uk/help/matlab/matlab\\_external/bringing-java-classes-and-methods-into-matlab-workspace.html](http://www.mathworks.co.uk/help/matlab/matlab_external/bringing-java-classes-and-methods-into-matlab-workspace.html).
- Mathworks. *MATLAB Compiler [Image]*. Available from: <http://www.mathworks.co.uk/products/compiler/description2.html>.
- MCGARRITY, S., 2008. *Introduction to Object-Oriented Programming in MATLAB*. [mathworks.co.uk](http://mathworks.co.uk): MathWorks. 2008, Available from: <http://www.mathworks.co.uk/company/newsletters/articles/introduction-to-object-oriented-programming-in-matlab.html>.
- MODEL, M., 2012. *Model View Controller History*. [c2.com](http://c2.com): [c2.com](http://c2.com). May 2012, Available from: <http://c2.com/cgi/wiki?ModelViewControllerHistory>.
- MOISE, A. and ATKINS, M.S., 2005. *Designing Better Radiology Workstations: Impact of Two User Interfaces on Interpretation Errors and User Satisfaction*. *Journal of Digital Imaging*, vol. 19, no. 2, pp. 109.
- MOSS, C., 2010. *Intelligent Image Analysis for Radiotherapy Treatment Planning*. Keele University.
- NAMIC., 2014. *NA-MIC-Kit*. [na-mic.org](http://na-mic.org): NA-MIC. January 2014, Available from: <http://www.na-mic.org/Wiki/index.php/NA-MIC-Kit>.
- NEMA., 2011. *The DICOM Standard*. [nema.org](http://nema.org): Medical Imaging & Technology Alliance. August 2011, Available from: <http://medical.nema.org/standard.html>.
- NICE., 2014. *NICE clinical guideline 175 Prostate cancer: diagnosis and treatment*. NICE.
- NSUH Charity., 2010. *Cancer specialist focuses on 3D technology*. [uhnscharity.org](http://uhnscharity.org): NSUH Charity. 2010, Available from: <http://www.uhnscharity.org.uk/index.php?mact=News,cntnt01,detail,0&cntnt01articleid=49&cntnt01origid=73&cntnt01returnid=82>.
- OSSATI, M., 2010. *Segmentation of the prostate on pelvic CT images*. MSc ed. Keele University.
- OTNIEL, D., NICOLAE, B. and CLAUDIU, B., 2012. RISK MANAGEMENT APPROACHES AND PRACTICES IN IT PROJECTS. *Annals of the University of Oradea, Economic Science Series*, vol. 21, no. 1, pp. 1014.
- PALMER, J., 1995. Attention in visual search: distinguishing four causes of a set size effect. *Curr Dir Psychol Sci*, vol. 4, no. 4, pp. 118.
- PATTON, R., 2005. *Software Testing (2nd Edition)*. Indianapolis: Sams.

PREVEDELLO, L., K.P. ANDRIOLE, R. GILL, I. OLIVA and R. KHORASANI. Post-processing tools for radiologists: understanding the principles, obstacles and utility of embedded advanced processing into the routine workflow in thoracic radiology *Radiological society of North America scientific assembly and annual meeting program*. Oak Brook, 2007.

PREVEDELLO, L., K.P. ANDRIOLE and R. KHORASANI. advanced post processing tools in radiology: what do we know, what should we know, and how can we utilize them in evidence-based approach? *Radiological Society of North America scientific assembly and annual meeting program*. Oak Brook, 2008.

Project Management Institute., 2008. *A guide to the project management body of knowledge (PMBOK)*. Newton Square, PA: Project Management Institute.

Prostate Cancer UK., 2013. *Prostate cancer*. prostatecanceruk.org: Prostate Cancer UK. Jan 2013, Available from: <http://prostatecanceruk.org/information/prostate-cancer>.

RAVENEL, J. and MCADAMS, H., 2003. Multiplanar and three-dimensional imaging of the thorax. *Radiol Clin North Am*, vol. 41, no. 3, pp. 475.

REDDY, D.S., VARADARAJAN, S. and GIRIPRASAD, M.N., 2013. 2D-DTDWT Based Image Denoising using Hard and Soft Thresholding. *International Journal of Engineering Research and Applications*, vol. 3, no. 1, pp. 1462.

REENSKAUG, T. and COPLIEN, J.O., 2009. *The DCI Architecture: A New Vision of Object-Oriented Programming*. artima.com: Artima Developer. March 2009, Available from: [http://www.artima.com/articles/dci\\_vision.html](http://www.artima.com/articles/dci_vision.html).

RegisFrey., 2010. *MVC-Process [Image]*. wikipedia.org: wikipedia.org. Available from: <http://upload.wikimedia.org/wikipedia/commons/a/a0/MVC-Process.svg>.

RUBIN, G., NAPEL, S. and LEUNG, A., 1996. Volumetric analysis of volumetric data: achieving a paradigm shift. *Radiology*, vol. 200, no. 2, pp. 312.

SCHNEIDER, K., 2007. Generating Fast Feedback in Requirements Elicitation. In: *Requirements Engineering: Foundation for Software Quality*, pp. 160.

SCHROEDER, W.J., 2005. *VTK [Image]*. ohio-state.edu: Kiteware. April 2005, Available from: [http://www.cse.ohio-state.edu/~crawfis/cis694L/Slides/Vtk\\_VisualizationSystems.ppt](http://www.cse.ohio-state.edu/~crawfis/cis694L/Slides/Vtk_VisualizationSystems.ppt).

SCHWARZ, C., 2013. *MVC in Matlab*. myunscriptedblog.blogspot.co.uk: myunscriptedblog.blogspot.co.uk. February 2013, Available from: <http://myunscriptedblog.blogspot.co.uk/2013/02/mvc-in-matlab.html>.

SEMMLOW, J.L., 2004. *Biosignal and Biomedical Image Processing: MATLAB-Based Applications*. USA: CRC Press.

SMITH, S.T., 2006. *MATLAB: Advanced GUI Development*. Indianapolis: Dog Ear Publishing.

Standish Group International., 2009. *CHAOS Summary 2009 report*. standishgroup.com: .

WEE, L.K., CHAI, H.Y. and SUPRIYANTO, E., 2011. Surface rendering of three dimensional ultrasound images using VTK. *Journal of Scientific & Industrial Research*, vol. 70, pp. 421.

XIE, T., K. TANEJA, S. KALE and D. MARINOV. Towards a Framework for Differential Unit Testing of Object-Oriented Programs *Proceeding AST '07 Proceedings of the Second International Workshop on Automation of Software Test*. IEEE Computer Society Washington, 2007.

ZAVALETTA, V.A., B.J. BARTHOLMAI and R.A. ROBB. Nonlinear histogram binning for quantitative analysis of lung tissue fibrosis in high-resolution CT data *Medical Imaging 2007: Physiology, Function, and Structure from Medical Images, 65111*, 2007.

ZHEN, C. and L. GUOLI. Application of Hybrid Filter in CT Image Processing Based on Visualization Toolkit *Proceedings of the Third International Symposium on Computer Science and Computational Technology*. Jiaozuo, P. R. China, 2010.

## **Appendix 1: Evaluation Form (Atkin-Granville, 2012)**

### **Do the zoom and panning features provide enough control and flexibility?**

Please include details such as the suitability of having separate buttons (toggles) for zoom in, zoom out and panning.

The features regarding zoom and panning allow good control and easy access to the areas of interest. I think having separate buttons for these options makes their use easier.

### **Does the boundary drawing tool provide enough control and flexibility? Please comment on details such as the pixel painting functionality as opposed to a drag-handle approach.**

The boundary drawing tool together with the zoom features allows good control and flexibility. However, removing some of the boundary points requires zooming the image which requires further time as it is not always easy to remove a point without the zooming first. I think that a “undo” hotkey would be useful to erase a point that has been mistakenly placed.

### **Is the tool fast enough? For example, does the interface react in a fluid way? Please describe areas where it is not.**

The tool is in my opinion fast enough. However, there are areas where improvements are needed such as erasing boundary points (mentioned above) and filling as the software does not always provide the correct filling for the boundary points identified.

### **Would you prefer the ability to choose where the output files are saved, or is the current system where the location is determined automatically preferable?**

I think it is always better to be able to choose where the files have to be saved. Each user might have his/her own preferences and in some cases files have to be saved in USB. The user should be able to choose where to save rather than rely on the system to save the files.

### **Do you prefer not to have a separate erase mode versus a “built-in” erase mode?**

One of the problems mentioned above is erasing boundary points. I think it would be easier to have a separate erase button that would allow not only erase boundary points but also areas within the planning of the treatment. It is not uncommon for doctors to have to erase a given area. Having such facility would speed the planning of treatment.

**How useful are the contrast and high-detail zoom view features?**

I think the contrast and high-detail zoom features are good. If I am not mistaken, that will depend up to certain extent on the quality of the CT Scan image.

**Please comment on the use of keyboard hotkeys for features such as boundary filling as opposed to toolbar buttons (such as zoom).**

The use of keyboard hotkeys is also useful as it speeds up the whole process. It is however always important to have the toolbar buttons in case a user forgets a given keyboard hotkey.

**With regards to boundary shapes, is it required for the tool to handle boundaries with complex boundaries, for example:**



Definitely yes. Tumours come in different sizes and shapes and therefore, it is required to be able to draw complex boundaries.

**Should the drawn boundary be shown on all slides, or only the slide it was drawn on?**

Due to different tumour shape and sizes, the boundaries might change from one CT slice to another. Therefore, boundaries do not necessarily need to be reproduced for each CT Slice. However, there should be a button (copy and paste) that the user might wish to use in order to copy a given boundary from 1 CT slice to the next one if he/she thinks tumour size and shape might be similar in 2 consecutive slices.

Also, there should be a facility to then modify easily the boundary if the user thinks it might need reshaping slightly.

**Please comment on the preferred workflow regarding similar texture matching and drawing the boundary. For example, should the user draw a rough boundary, then the system suggests similar areas and the user draws the boundary again with this new information?**

It is not always easy to draw the exact boundary of a given tissue or tumour. Therefore, I think that once the user has drawn the boundaries, there should be the possibility for the software to provide a boundary based on pixel homogeneity. That might in the future help the user to then modify the boundary and be more accurate. However, research is required first to assess whether homogeneity in contiguous pixels correlates with better and more accurate boundary drawing.

**Do you have any other comments?**

I think this software is an excellent step towards better understanding tissue/tumour boundaries in CT Scans which will help in furthering the research in this area.



## Appendix 2: Client Meeting Question Ideas

Tasks performed?

- Walk through in bullets of each

What tools used to make boundaries

- Add/remove points
- Paint like
- Colour
- Weight of lines
- Creation of 3D surface from points

What is presented in a normal report?

- Patient data
- Images
- Own comments

Tools used most often?

- Contrast
- Hotkey used currently?
- Zoom in/out: +/-
- X/Y/Z slice: Ctrl + X/Y/Z
- Tools would like to see

Help documentation?

- Seperate
- Toot-tip

Where the most time is generally spent when using such a program

## Appendix 3: Project Meeting Minutes

**From:** Ben Walker [<mailto:v6r37@students.keele.ac.uk>]

**Sent:** 24 March 2014 6:07PM

**To:** 'Ben Walker'; 'Dr KP Lam'

**Subject:** RE: Project Meeting Minutes Ben Walker

KP,

I don't have Joseph's email.

### Meeting summery

- work to integrate 3D views into application
- work to finish the usability of 2D GUI – VI Unix like features
- work to finish the report GUI

### Meeting summery

- work to finish the 2D GUI – with focus on boundary saving/loading
- 

**From:** Ben Walker [<mailto:v6r37@students.keele.ac.uk>]

**Sent:** 04 March 2014 1:56 AM

**To:** 'Ben Walker'; 'Dr KP Lam'

**Subject:** RE: Project Meeting Minutes Ben Walker

Hi KP,

### Meeting summery

- work to integrate 3D views into application – Demo issue fixed from meeting, 2 dll missing
  - work to finish the usability of 2D GUI
-

**From:** Ben Walker [<mailto:v6r37@students.keele.ac.uk>]

**Sent:** 17 February 2014 5:51 PM

**To:** 'Ben Walker'; 'Dr KP Lam'

**Subject:** RE: Project Meeting Minutes Ben Walker

Hi KP,

### **Meeting summery**

Send poster draft first thing on Tuesday for review

- Work on adjusting the boundary so that the points are collected by freehand but the polygon is constructed from the points with no assumption on the direction in which the segments of freehand were drawn
  - Work on Java wrapper over the MatVTK for VTK integration with matlab
- 

**From:** Ben Walker [<mailto:v6r37@students.keele.ac.uk>]

**Sent:** 15 February 2014 5:07 PM

**To:** 'Ben Walker'; 'Dr KP Lam'

**Subject:** RE: Project Meeting Minutes Ben Walker

Hi KP,

### **Meeting summery**

- Shown poster textual and notes so far, got advice on content
  - Work on Java wrapper over the MatVTK for VTK integration with matlab
-

**From:** Ben Walker [<mailto:v6r37@students.keele.ac.uk>]

**Sent:** 03 February 2014 5:55 PM

**To:** 'Ben Walker'; 'Dr KP Lam'

**Subject:** RE: Project Meeting Minutes Ben Walker

Hi KP,

#### **Meeting summery**

- Have the majority of the 2D rendering gui functional and end-user polish for end of week 3
  - reduce poly-line resolution to make editing easier
  - enable deletion of past boundaries
  - save boundaries to file
- Work on Java wrapper over the MatVTK for VTK integration with matlab

#### **Meeting summery**

- Work on getting boundary capture working
  - Work on Java wrapper over the MatVTK for VTK integration with matlab
-

**From:** Ben Walker [<mailto:v6r37@students.keele.ac.uk>]

**Sent:** 07 December 2013 5:11 AM

**To:** 'Ben Walker'; 'Dr KP Lam'

**Subject:** RE: Project Meeting Minutes Ben Walker

Hi Kp,

Would it be possible to have the boundary algorithm code to work on over Xmas, I'm thinking of integrating it into the system so everything is accessible and integrated in one place.

### **Meeting summery**

- Organised Data collection for next meeting
  - CT Scan and associated boundaries from both Medical staff and Algorithm
- Discussed Class diagram
  - Construct shell system to test out the structure (look to examples from APP of MVC)
  - Look to the possible implementation in Java over MATLAB for better OO support
- Christmas holiday plans
  - The production of a complete plan and the majority of modules functioning in un tested state
  - Begin planning and write up of project report

### **Client Meeting summery**

- Shown Vert System and Anatomical models
    - Presented on laptop basic idea for system
  - Ideal would be automated boundaries for all organs and cancer for people to check over manually and enter error margins for
  - Probable solution put forward the integration of boundaries drawn in 2D into 3D and the adaption of past exemplar boundaries and algorithmically generated boundaries for display in 3D
-

**From:** Ben Walker [<mailto:v6r37@students.keele.ac.uk>]

**Sent:** 18 November 2013 11:15 AM

**To:** Dr KP Lam; Ben Walker

**Subject:** Re: Project Meeting Minutes Ben Walker

Hi Kp,

#### Meeting summery

- Create a mock up system which can import KP dataset and renders to the matVTK client into 3D rendering of DICOM for next meeting
    - **Present on laptop**
  - If possible in time to impliment 3D boundry render
- 

**From:** Ben Walker [<mailto:v6r37@students.keele.ac.uk>]

**Sent:** 4 November 2013 12:00PM

**To:** Dr KP Lam; Ben Walker

**Subject:** Re: Project Meeting Minutes Ben Walker

Hi Kp,

#### Meeting summery

- Look into 3D rendering of DICOM for next meeting
    - **Present on laptop:** a sample using MATLAB volume rendering from FX
  - Produce an example set of Q for medical staff on working environment etc to build UI requirments up from
  - **REMINDER:** arrange meeting with Joseph at the new Cancer Centre early on in the project
-

**From:** Ben Walker [<mailto:v6r37@students.keele.ac.uk>]

**Sent:** 3 October 2013 3:42 PM

**To:** Dr KP Lam; Ben Walker

**Subject:** Re: Project Meeting Minutes Ben Walker

Hi Kp,

#### Meeting summery

- Schedule a regular meeting for Monday's 1130-1200
  - **NB:** Needs changing as clash with optional lecture (1100-1200) I wish to attend
- Minutes system discussed and established
- Discussion of evaluation paper relating to the project
- **TARGET:** Read, understand explore around the paper for next meeting
- Idea mooted of using Blender in conjunction with MATLAB to facilitate more professional 3D image rendering
- Suggestion that a focus on GUI Design might be a sensible direction to take the project
  - Eg. Better implementation of MVC
- **REMINDER:** arrange meeting with Joseph at the new Cancer Centre early on in the project

## Appendix 4: Project Assumptions

### General

- **Language:** The system will only be used by people proficient in English, and who understand context specific terminology.
- **Security:** The system will be running within a secured environment and not require additional security measures.
- **Data-Types:** For each input there will be one data/file type. E.g. CT scans in the DICOM data format.
- **Output:** Outputs will be simple text files (RTF or HTML) in the Unicode text format.

### Hardware

- **Storage:** Enough space to store the data files, system and potential outputs.
- **Screen:** A resolution of 1920x1080, and 32 bit colour.
- **Processing:** Enough power to provide reasonable response times under regular system load.

### Software

- **Operating System:** Windows 7.
- **Supporting Environment:** Java 7.
- **Drivers:** Suitable drivers installed for attached peripherals.



## Appendix 5: Deliverables in Detail

### Enhanced UI

#### *Deliverables*

- A user friendly user interface (UI) based on elicited requirements from all stakeholders
- A report documenting test and evaluation against success criteria

#### *Tests*

- Ensure all help documentation is accurate.
- Hand UI to unrelated user to see how intuitive it is to use.

#### *Evaluation*

Prototypes taken to interviews with clinical staff, and feedback will be obtained at each stage. The final version to be handed to a clinical staff member who should be able to utilise its functionality in correspondence with its success criteria.

### 3D Visualisation

#### *Deliverables*

- A system module for rendering the 3D visualisation
- A report documenting test and evaluation against success criteria

#### *Tests*

- A simple small manually created dataset renders correctly
- A real data set renders correctly

#### *Evaluation*

The project developer should compare the 3D render to the 2D data set from which it has been created not find any glaring differences. The 3D render should also be assessed for accuracy against the 2D version by members of clinical staff so that confidence can be had as to the accuracy of its anatomical representation.

### Report generation

#### *Deliverables*

- A report generation module based on elicited requirements from all stakeholders
- A report documenting test and evaluation against success criteria

#### *Tests*

- Prints accurately
- Saves accurately
- Metadata imported uncorrupted

#### *Evaluation*

The project manager should compare all generated reports to the requirements given by all stakeholders. Stakeholders should also be given example reports to assess for compliance with their existing examples and to their expectations given the set requirements.

## **Appendix 6: Project Risks & Mitigation**

### **Unavailable for interviews**

This is a situation which cannot be left unaddressed due to the paralyzing effect it would have on the project. Email conversations will be a back-up in the case that an interview cannot take place. This will be a slower and less efficient method for requirements elicitation, thus requiring extra time.

### **Previous system code illegible**

If the previous system code hasn't been adequately documented, clarifications will need to be sought. Having the supervisor who understands the previous systems well and having the contact details of the previous developer is mitigation to this risk.

### **Failure to learn MATLAB's language**

The purchase of MATLAB specific texts, online video/tutorials, access to a supervisor proficient in the language and coming from a programming background helps to mitigate this risk.

### **Poor time management skills**

As this project isn't being carried out in isolation, other projects could interfere causing delays. Regular progress meetings with the supervisor and the option to drop objectives based on priority. Outside of the project time-management meetings are scheduled to reduce interference. Any deviation from the outlined project timeline can be detected and dealt with in the aforementioned meetings.

### **Development environment failure**

Files will be stored in two locations.

## Appendix 7: Viewer3D's Supported File-Types

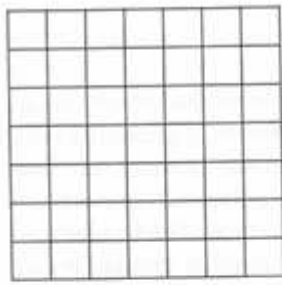
- Matlab Files (.mat)
- Dicom Files (.dcm , .dicom )
- V3D Philips Scanner (.v3d )
- GIPL Guys Image Processing Lab (.gipl )
- HDR/IMG Analyze (.hdr )
- ISI Files (.isi )
- NifTi (.nii )
- RAW files (.raw , .\* )
- VMP BrainVoyager (.vmp )
- XIF HDllab/ATL ultrasound (.xif )
- VTK Visualization Toolkit (.vtk )
- Insight Meta-Image (.mha, .mhd )
- Micro CT (.vff )

## Appendix 8: CSV Output Example

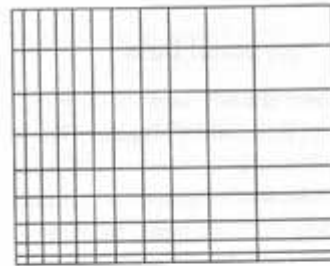
Tag, Slice Number, [x y point pair; x y point pair; x y point pair; x y point pair;]

b\_1,1,[28 23;49 137;137 76]

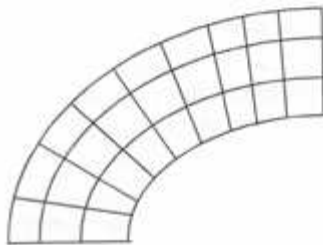
## Appendix 9: VTK Data Types (Gasser, 2009)



(a) Image Data



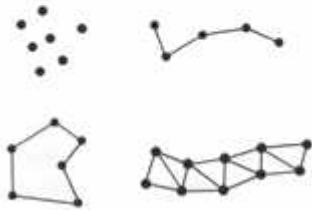
(b) Rectilinear Grid



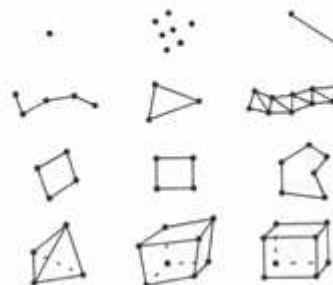
(c) Structured Grid



(d) Unstructured Points



(e) Polygonal Data



(f) Unstructured Grid

## **Appendix 10: Black-Box Evaluation Form Example**

### **How easy was it to work out what the different parts of the prototype did, and how the tools function?**

The program helpfully guided me through the programs application of drawing multiple boundaries on the C.T slices. The functionality was occasionally to be desired as if you mistakenly move through a slide to the next the program can occasionally freeze or remove the boundaries; though the program did occasionally recover these.

A problem that occurred after saving one slice with a boundary marked out on it was when starting the program a new the previously marked out slide was the only accessible slide and the boundaries were thereafter unworkable- However this may have been due to me saving the slide in the incorrect place and this action then caused the program to crash after loading.

### **Is the prototype fast enough e.g. does the interface react smoothly?**

#### **Please describe areas where it is not.**

The reporting system does not make you aware that it freezes all of the program which can cause slight confusion and also means that the user occasionally ( especially if it is me) will click and attempt to continue work which slows the program down further and can cause errors.

Otherwise the program worked well and speedily.

### **Where there any parts of the application which did things you didn't expect?**

The program blew my mind and I was not expecting this.

The program operated well with attempts to break it and dealt well with my purposely stupid actions even correcting my actions when I drew two boundaries at the same time without ending the program.

### **Do you have any other comments?**