The candidate confirms that the work submitted is their own and the appropriate credit has been given where reference has been made to the work of others.

I understand that failure to attribute material which is obtained from another source may be considered as plagiarism.

(Signature of student) ______________________________
Summary

The time a busy surgeon has is vital, and pre-radiotherapy planning can consume valuable minutes, even hours, of this time, bearing in mind that segmentation of the airways alone can take 15 minutes for one patient. This project looks into automating the airway segmentation part of this process. Previous attempts at airway segmentation will be discussed, along with the closely related field of vessel segmentation. I have developed an automated method for airway segmentation and this report will analyse this method for effectiveness. Finally there will be suggestions for future work in this area.
Acknowledgements

I would like to thank my supervisor Derek Magee, for his wise guidance in weekly meetings and throughout the project, as well as my assessor Mark Everingham for his feedback.

I would also like to thank Jonathan Sykes at St. James’ Hospital for providing the data and useful information on how the planning process is currently managed, and also Kevin Franks for his professional insight.

Finally I’d like to thank my girlfriend, whose dedication for her own dissertation was a true inspiration throughout.
# Contents

1 **Introduction** 1

1.1 Project Aim 1

1.2 Minimum Requirements 1

1.3 Project Management 2

1.3.1 Original Plan 2

1.3.2 Revised Plan 2

1.3.3 Progress Review 3

2 **Background Research** 4

2.1 Airway Segmentation 4

2.2 Vessel Segmentation 7

2.3 Segmentation Methods 8

2.3.1 Region Growing Methods 8

2.3.2 The Watershed Method 9

2.3.3 Connected Components Analysis 9

2.4 Shape Analysis 9

2.4.1 Model-Based Segmentation 10

2.4.2 Model-Based Pruning and Skeletonisation 10

2.5 Other Pruning Methods 11

2.5.1 Erosion 11

2.6 Post Pruning 12

2.6.1 Dilation 12

2.7 Airway Contouring 12
2.7.1 Edge Detection 12
2.7.2 Boundary Tracing 13
2.8 Evaluation Methods 13
2.8.1 Dice Coefficient 13
2.8.2 Hausdorff Distance 14

3 Design of Airway Segmentation Method 15
3.1 Reading the Data 15
3.2 Segmentation 15
3.2.1 Thresholding 15
3.2.2 3D Connected Components Analysis 16
3.3 Airway Component Selection 16
3.4 Modelling 16
3.4.1 Model Design 16
3.4.2 Skeletonisation 17
3.4.3 Model Fitting 17
3.5 Contouring 18
3.5.1 Scanning 18
3.5.2 Boundary Tracing 18
3.6 Writing the Results 18

4 Development of Airway Segmentation Method 19
4.1 Iteration One 19
4.2 Iteration Two 21
4.3 Iteration Three 21
4.4 Iteration Four 22
4.5 Iteration Five

4.6 Iteration Six

5 Evaluation

5.1 Description of Results

5.1.1 Dataset One

5.1.2 Dataset Two

5.1.3 Dataset Three

5.2 Evaluation of Results

5.2.1 Dataset One

5.2.2 Dataset Two

5.2.3 Dataset Three

5.2.4 Overall Evaluation of Results

5.3 Overall Evaluation of Project

6 Further Work

Bibliography

A Personal Reflection

B Technical Acknowledgements
Chapter 1

Introduction

1.1 Project Aim

To help surgeons save time by automatically highlighting airways in CT images prior to radiotherapy. This is currently done manually; I will create a program which automates this process, and whilst this doesn't necessarily mean the program should be able to perform faster than a surgeon, it must allow the surgeon to be doing other things whilst it runs. Also, through being done automatically, the process can be much accelerated, meaning that images of all 12 phases of the breathing cycle can be processed, currently manual segmentation is only done on one phase of 4DCT.

1.2 Minimum Requirements

1) A method for segmentation of airways in CT images.

2) A software implementation of the chosen method.

3) A review of possible methods for solving the problem.
1.3 Project Management

1.3.1 Original Plan

Figure 1 shows my original plan for time management over the course of the 12 week semester (ignoring the Easter break, which would be classed as contingency to be used if I was behind in the development). I planned to perform iterative development, each step building on what had been done before, with the minimum requirements of the project being met by iteration three, and built upon through four and five. However, this plan was created under the assumption that iterations four and five would be dealing with applying shape analysis and model fitting to the segmentation, an approach which changed after my progress meeting. Details of this change of approach can be found in chapter three.

1.3.2 Revised Plan

The schedule was largely followed for the first half, up to the mid-project report in week 6 and progress meeting in week 7, however with the change in approach at this time, along with the realisation that more than enough time had been set aside for evaluation, the second half of the semester turned out quite differently. Figure 1.2 (on the next page) shows a better picture of how the second half of the semester worked out.
As shown by this (condensed) Gantt plan, I have taken out the time allotted for evaluation as this became more of a fluid part of the testing process and the start of the write up.

![Figure 2: Revised plan](image)

### 1.3.3 Progress Review

In review of how I feel that the project progressed in terms of the original plan I would say that I was naive with the amount of time set aside for the solution development, which was solved by the change in approach after week 7, and the slack time in the original schedule over Easter. I was also very generous in how much time was given to the evaluation, which became a dynamic part of other areas of the project, and so no dedicated time was really required.
Chapter 2

Background Research

2.1 Airway Segmentation

Airway Segmentation can mean a couple of different things. In the case of my project it means distinguishing the airway channel from the rest of the lungs based on a small amount of prior knowledge, and displaying this segmentation by means of drawing a line around the airway walls, for the purpose pre-radiotherapy planning. However it can also mean the segmentation and modelling of airway trees [1] for various possible purposes such as medical research, diagnosis of lung-related illnesses and monitoring effects of treatments. This paper outlines a method for generating a 3D model of the bronchial trees based on CT images and some prior knowledge relating to the curvature of the branches of the airways. The algorithm executes by firstly acquiring a basic segmentation of the lungs through thresholding. Next the program applies “marching cubes” to model the anatomical structure. “Marching cubes” produces a triangle mesh by computing iso-surfaces from discrete data. By connecting the patches from all cubes on the iso-surface boundary, it gives a surface representation. Next, from this surface representation, the algorithm uses Rusinkiewicz’ curvature estimation to approximate the curvature of the branches (See figure 3). Now the information provided by the curvature computation is compared to empirically predetermined criteria to distinguish vertices of the bronchial tree.

Figure 3: Curvature Visualisation
from non-airway regions (Figure 4 shows the results of this, where (a) is the complete result of marching cubes, (b) is the regions classified as airway and (c) is regions classified as non-airway). Finally the algorithm performs a “puzzle game” to match holes in the airways to pieces wrongly classified as non-airway matter. Whilst the paper has some merit in that the system used is capable of generating some impressive and indeed realistic looking 3D models of the inner workings of the lungs, the paper ultimately falls down in its inability to prove the accuracy of the method. The accuracy of the algorithm comes into question because of the inability to analyse it against a ground truth. From the text “It is extremely difficult if not impossible to manually and accurately trace and mark the 3-D airway trees depicted on CT images and use these as a gold standard for evaluation purposes.” Therefore they instead judge the results against a number of performance measures; the total tree length, the branch number, and the generation number. However it is possible (whilst unlikely) that two data sets could share this information but still represent two different structures. I found this paper very interesting and the possible uses for it could provide a deep insight into some of the mysteries of the lungs, however the methods used would not suit my application for two main reasons: Firstly, in radiotherapy planning surgeons are only concerned with the larger (upper) sections of the airways, because it is these areas which, if irradiated, would cause serious complications, whereas smaller areas are less crucial. Secondly this method creates a visualization of the airways as opposed to providing an actual “model” to be used for pruning.

Other research projects [2], [3] relate more closely to my application. Both methods aim to achieve the same key elements as my method, namely; the ability to accurately segment the first three/four generations of the airways (generations referring to the amount of splits in the trees), negation of the problem of leakage (when the algorithm ‘leaks’ out of the airway and wrongly identifies lung-parenchyma as airway) and most importantly elimination of the need for manual operators. “Segmentation and quantitation of the primary human airway tree” [2] uses a 8-connected region growing technique to follow the downward branches of the tree, the detection of upward branches works in the same way, with the use of some prior
knowledge to determine the slices from which to begin an upward search. The key problem
with this method is ‘leakage’, and whilst this paper is slightly dated and so the coined term
isn’t necessarily mentioned, the solving of the issue has been addressed using ray profiles.
Ray profiles are cast from points which are designated as being possible centroids of the
airway, the profile of a ray cast through a section of airway will show a small variation
coefficient as the inside of the airway canal is solidly black with a smooth transition to the
brighter airway walls. The profile of a ray cast through the parenchyma will show a larger
variation coefficient because of the bright spots created by blood vessels. “Intrathoracic
Airway Trees: Segmentation and Airway Morphology Analysis from Low-Dose CT Scans” [3]
is a more recent research article (published 2005) which approaches the leakage issue in a
more direct way. The method uses segmentation through ‘fuzzy connectivity’ focused on a
relatively small region of interest which follows the airway as it is being segmented. The
region of interest is cylindrical and it adapts its size, orientation etc to the branch being
segmented. The advantages of this region of interest are increased speed and the possibility
to detect leakage as it occurs. The actual detection of leaks is achieved through the
knowledge that a leaked segmentation result exhibits a ‘spongy’ structure, i.e. it is full of
holes. A topology-preserving thinning method is applied and the size of the resulting
topological kernel is used as a leak detector. The methods used in this paper are very
impressive in theory; however the likely accuracy achievable by the geometric region of
interest is questionable and the leakage detection works on the premise that the leaked
structure is “almost always” of a spongy nature, implying the possibility that this is not always
the case. I believe this is why the paper claims that acquiring a ground truth provided by an
expert is unfeasible, which, for the number of generations to which their algorithm can
distinguish, is not the case. I can say this because I have a ground truth against which to test
my results, and my application works to the same generation depth on the same type of
image data.

Four years subsequent to “Segmentation and quantitation of the primary human airway tree”
link to my application. The paper looks into a method segmenting the lungs, as opposed to
the airways, from 3D CT images, as well as segregating the right lung from the left. The
method begins by extracting the lung regions from the CT images through grey-level
thresholding. The algorithm then uses a dynamic search to establish the ‘junctions’ at which
the lungs have connected to become a single component (Figure 5 Shows how this can
occur). The reason this caught my eye is that we foresaw leakage leading to the airway
being joined to the lungs (in terms of connected components analysis) as a probable issue. I
considered numerous different methods for differentiating the lungs from the main airway,
including this one and two others (modelling and erosion) which have been discussed in sections 2.4 and 2.5 respectively. Although this method was appealing and showed good results against a manually acquired ground truth, I will discuss in 4.5 why I decided to go with erosion.

2.2 Vessel Segmentation

The area of research into airway segmentation is linked closely to that of blood vessel segmentation; this is due to the similarities in structure. Both are tubular in shape, with an outer wall encasing the lumen and both can be thought of as displaying a tree-like structure, only ever separating into 2 branches per split. Lots of research has been done into the study of blood vessels, and similarly to the problem of airway segmentation the solutions can be broken down by two clear distinctions: segmenting the vessels or modelling the vessels.

Wilkinson et al [5] posed the idea of a moving window based thresholding method for segmenting the blood vessels, in this case in human brain. The moving window approach echoes that used by Tschirren et al [3] describes previously, although in this case the window is not so much a geometric object which follows vessels, as there would be far too many of them to do so, instead the window moves over the entire image and exhibits a dynamic thresholding system, “Robust Automatic Threshold Selection (RATS)”. This method is taken from “Threshold selection based on a simple image statistic” [6]. Kittler et al here propose a novel method (novel at the time, being from 1985 this paper is slightly dated, however it is well cited in subsequent papers) for automatic threshold selection based on statistics which can be acquired without histogramming the grey levels of the image. Whilst a dynamic thresholding system is a novel and interesting idea, I don't feel it is required for my application. This is only really important when the program is required to look at different areas in the body i.e. the abdomen and the brain etc. Whereas I am looking at CT images
focused on the lungs only, in which airways are uniformly coloured in greyscale (they are black). With respect to the moving window approach [5] I feel this also would be unnecessary in my application as it would make more sense to assess the image volume as a whole in 3D than to sequentially scan through smaller parts.

2.3 Segmentation Methods

2.3.1 Region Growing

As discussed previously, many of the existing airway/vessel segmentation methods make use of, or at least mention, the idea of region growing segmentation techniques, the idea of growing an airway/vessel until it meets a boundary, usually determined by a sharp decrease/increase in contrast (in terms of greyscale) Hojjatoleslami et al [7] sticks to this base principal, however unlike other methods they define two different types of contrast; average contrast and peripheral contrast. To define these they introduce two more terms, Current Boundary (CB) is the set of pixels adjacent to the current region and Internal Boundary (IB) is the boundary produced by the set of connected outermost pixels of the current region. To define the two previous terms; average contrast is the difference between the grey level average of the current region and the average of its CB pixels, and the peripheral contrast is the difference between the grey level average of the CB and the average of the IB. This information is used in the final segmentation results. “The unique feature of the proposed approach is that in each step at most one candidate pixel will exhibit the required properties to join the region, which makes the direction of the growing process more predictable”. The results of this method are shown by distinguishing certain parts of the brain, with apparent success, and judging by these results the method would work equally well in 3D (which would be the best approach to my data) as it does here in 2D.

Revol-muller et al [8] propose a method that, whilst it can work in 2D, is designed to be used in 3D which makes use of an assessment function (similar to the use of a fitness function when talking about search methods). I feel this method would work well for my application; however I would consider it to be a bit of an over-complication better suited to more demanding segmentation tasks, for reasons that will be further explored in section 2.3.3.
2.3.2 The Watershed Transform

The watershed transform, or watershed method, is a more modern example of the region growing segmentation paradigm. Intuitively the watershed idea can be thought of in terms of its meaning in geography [9] in that imagining the image as a landscape the watersheds would be the dividing lines. The watershed transform is rarely applied to the original image, but to its (morphological) gradient [10]. This produces watersheds at the points where the greyscale level between two areas is at its most inconsistent, as is commonly desired in image segmentation, and indeed is the case in my application. In the beginning I saw the watershed transform as an acceptable method for segmenting the airways; however, much like the other region growing techniques, this also became obsolete through its unnecessary over-complication of the task at hand.

2.3.3 Connected Components Analysis

Connected components analysis (CCA) scans a binary image and groups its pixels into components based on pixel connectivity, i.e. neighbouring pixels, if they share similar intensity, are grouped and labelled. Extracting and labelling of various disjoint and connected components in an image is central to many automated image analysis applications, and it works equally well in 3D as well as 2D, which makes it perfect for my application. Whilst this method works very well for extracting the airway component from volumetric CT data, it shares one fatal problem with the methods discussed in sections 2.1 and 2.2, leakage. In the small number of datasets I had it was found that in some cases CCA would distinguish the airway from both lungs as 3 separate components, however in other cases it would group both lungs and the airway into one component, or possible even just the airway and one lung. The solution to this would be pruning; discarding the unwanted parts wrongly classified as airway, whilst retaining the structural integrity of the airway itself.

2.4 Shape Analysis

Shape analysis, in terms of image segmentation, can be split into two areas, based on their differing approach; either model-based segmentation, where model-fitting is done first and the image is segmented based on this information, or model-based pruning, where the image is already segmented to some degree, and model fitting is used to distinguish the key areas of interest.
2.4.1 Model-Based Segmentation

Model-based segmentation [11] is the method of using a pre-determined model, applied to an image, to which the model adapts to fit the example in the image, which it then contours (the notion of a ‘contour’ is discussed in section 2.7). Clearly the Philips example given here is a very high level professional instance of the method being used in practice, but it provides a good demonstration of how this can be used. Existing model-based algorithms can be categorised by the type of learned information they use [12] i.e. ‘no learned information’, ‘learned shape models’, ‘learned appearance models’ and ‘learned shape and appearance models’. Here the term ‘no learned information’ refers to methods like the active contour [13] or “snakes” approach, which isn’t really a model-based approach in these sense for which I would use them. The three other methods listed imply the need for volumes of training data, which for my project simply isn’t feasible, ergo the use of shape analysis would like be used in the post-segmentation pruning process.

2.4.2 Model-Based Pruning and Skeletonisation

Model-based pruning based on model fitting is the idea of removing undesired parts of a given image (or segmentation of an image) leaving only the desired image context. One approach to model fitting uses skeletonisation, or thinning, as a first step. Lam et al [14] provide a very thorough overview of thinning in general, the paper sets out to be a comprehensive survey of the area of thinning, and also skeletonisation more generally, and whilst is a little dated it has been heavily cited in subsequent papers. It describes thinning as the process of removing layers of pixels on the boundary until a skeleton of singular pixel width remains, and goes on to split the wide range of thinning methods into three categories; sequential thinning, parallel thinning and non-iterative thinning. Sequential methods iterate through the set of pixels, flag certain pixels as boundary pixels and at the end of each of the iterations delete the flagged pixels. This method avoids sequentially deleting an entire branch in a single iteration. Parallel thinning methods examine pixels for deletion based only on the results of the previous iteration, unlike sequential which bases on all previous iterations, meaning they are called parallel because they are designed to be run on parallel machines/processes for better efficiency. Finally, non-iterative thinning methods are methods that are not considered to be pixel based; they instead produce the centre line directly in one pass by using, for example, the minimum distance transform. The output of a distance transform, when used on binary image, is a greyscale image similar to the original
however the greylevel intensities of the pixels represent their distance to the nearest boundary. Therefore, the minimum of this distance transform corresponds to the point furthest from any boundary, i.e. the centre. I believe this to be the way that a human mind would perform thinning.

Model fitting takes the information left over from the skeletonisation process and attempts to fit a pre-designed model to that data. This may be through random search, heuristic search or dynamic programming (amongst other possibilities). The nature of these methods is that they are based on probability, which for the purpose of my application creates a dangerously random element; ergo a more predictable method of pruning is desirable.

2.5 Other Pruning Methods

2.5.1 Erosion

Erosion, whether on binary images or greyscale images, is the process of eroding away the boundary regions of foreground pixels [15] and can be equally effective when used in 3-dimensions as well as 2-dimensions. Erosion takes two arguments to return an eroded image, the original image and a structuring element. The structuring element [16] consists of a pattern specified as the coordinates of a number of discrete points relative to some origin and when erosion is carried out, the origin of the structuring element is typically translated to each pixel position in the image in turn, and then the points within the translated structuring element are compared with the underlying image pixel values. Image erosion can be used either as a method of thinning (see previous section) or, as its use in my application, to break apart the image in certain locations to split an image into distinct parts. To compute the erosion of an input image, each foreground pixel is considered as an input pixel, onto which the origin of the structuring element is superimposed. If for every pixel in the structuring element, the corresponding pixel in the image underneath is a foreground pixel, then the input pixel is left as it is. If any of the corresponding pixels in the image are background, however, the input pixel is also set to background value. It is important to work out that the structuring element need not be a flat structure (2-dimensional) it can be considered as a ‘ball’. This means the input pixel is not only compared to those around it but also to those above and below it.
2.6 Post Pruning

2.6.1 Dilation

The very nature of erosion means that some image data is lost, and whilst some of this data is lost through choice the side effect is that the remaining (desired) image has also been eroded. This is where the need for the reverse process, dilation, comes in. Dilation has the opposite effect of erosion, in that it gradually enlarges the boundaries of regions of foreground objects [17], and like erosion it requires a structuring element to determine the nature of the dilation. Therefore if dilation is performed on an eroded image, taking the same structuring element, it stands to reason that the relating image will be equivalent to the original; “Dilation is the dual of erosion i.e. dilating foreground pixels is equivalent to eroding the background pixels”.

2.7 Airway Contouring

Contouring is defined as the construction of contour lines through a matrix of points each having a known value of some variable, which implies the idea of following a boundary to generate a set of points lying on that boundary. However I felt that, whilst this is a possible method for achieving this, I also wanted to look into using edge detection. This was because of multiple artefacts, per slice, needing to be outlined.

2.7.1 Edge Detection

The theory of edge detection is the act of locating changes in intensity levels [18] based on some pre-determined threshold. The main benefit of this method, in terms of my application, is that when more than one disjoint element of the airway is present this will not cause a problem, which it may have done when using a boundary tracing method. Within the area of edge detection one of the methods of choice is the Gaussian filter method [19]. The benefit of a Gaussian-based approach over earlier approaches such as Prewitt or Sobel edge detectors is that both Prewitt and Sobel use local gradient operators which can only detect edges of certain orientations, and struggle with noisy images. However whilst the Gaussian filter approach to edge detection is a desirable method it still suffers from problems such as edge displacement and false edges, which would not be acceptable in my application, hence the move towards a boundary tracing technique.
2.7.2 Boundary Tracing

Ghuneim [20] defines the Moore-neighbourhood as the set of 8 pixels which a vertex or edge with a given input pixel. This idea of neighbourhood is used as the basis of a boundary tracing method. Given a starting pixel known to be on the boundary of an object the algorithm visits all the Moore-neighbours of that pixel (travelling in pre-designated direction, either clockwise or anti-clockwise) and moving onto the next pixel it determines to be a part of the foreground, terminating when it returns to the starting pixel. I feel this method would be perfectly suited to my application as it will be very possible to find a starting pixel, and the method will be guaranteed to accurately follow the outline of the image as long as the image is in binary format.

2.8 Evaluation Methods

Evaluating the results generated for the coordinates of the contours of the airway is made easier by the fact that I have access to a ground truth. A ground truth, in the medical field also known as a Gold Standard, is defined as any standardised clinical assessment, method, procedure, intervention or measurement of known validity and reliability which is generally taken to be the best available, against which new tests or results and protocols are compared [21]. This means I can compare the results of my algorithm to real results given by a professional (a surgeon in this case). However, this does mean an appropriate comparison method must be decided on.

2.8.1 Dice Coefficient

Sampat et al [22] in a comprehensive study of many different similarity coefficients comment on the Dice Coefficient being commonly used in medical image studies to determine the overlap of two images. The calculation of Dice is given by:

\[
\frac{2a}{2a + b + c}
\]

Where: \(a\) is the number of corresponding pixels of value 1 in both images, \(b\) is the number of pixels taking the value of 1 in the first image only, \(c\) is the number of pixels taking the value of 1 in the second image only. When applied to my results this will mean \(a\) is the number of coordinates that exist in both the ground truth set (A) and my result set (B), \(b\) is the number of coordinates that exist only in A and \(c\) is the number of coordinates that exist only in B.
When using this similarity measure, Dice = 1 means perfect overlap, Dice = 0 means exactly no overlap. Sampat quotes that Dice > 0.7 is generally considered excellent.

This is very closely related to the Jaccard coefficient, which is given by:

\[
\frac{a}{a + b + c}
\]

Clearly the only difference made by Dice is that more weight is given to the number of coordinates existing in both sets (in the example of my application). I feel that the problem with the Dice coefficient is that there is no measure of the distance between the two sets of data. I feel this would be a bad measure for my application as a result could be almost identical (say 1 pixel out for every coordinate, still indicating a very small distance) but the result would still be classified as Dice = 0.

2.8.2 Hausdorff Distance

The Hausdorff distance [23] from A to B is defined as:

\[
h(A, B) = \max \{ \min \{ d(a, b) \} \}
\]

Where \(a\) and \(b\) are members of the sets \(A\) and \(B\) respectively, more generally, the distance between \(A\) and \(B\) can be shown by:

\[
H(A, B) = \max \{ h(A, B), h(B, A) \}
\]

This more general form, in my opinion, is a closer representation of what the human mind would as a comparison of the similarity of two complex polygons. This means it would be much better suited to my application.
Chapter 3

Design of Airway Segmentation Method

This section will consider the original design for the implementation of my application; the subsequent chapter will follow the iterative process through development.

3.1 Reading the Data

At the start of the project we met with our contact at St. James’ Hospital, Jonathan Sykes. It was his suggestion that pointed towards a method for reading Dicom data into matlab, CERR [24]. CERR (Computational Environment for Radiotherapy Research) is a platform for developing and sharing research results using radiation therapy treatment planning information. This performed the task of converting Dicom series information into a structured data format which matlab could understand. My only qualm with the software was its need for user interaction, something that didn’t completely suit the aim of my application.

3.2 Segmentation

3.2.1 Thresholding

What CT images actually show in a scan is the density values of each part of the body translated into intensity values in the image. Air, in the airways and lungs, is much less dense than the surrounding tissue, therefore a simple threshold (if chosen correctly based on some simple trial-and-error tests) can separate these areas nicely. Bearing in mind that all the data my application would handle would be from the same scanner (or at least the
same type of scanner, as they are from the same hospital) then a threshold that works on one dataset should work equally well on all others.

3.2.2 Connected Components Analysis

As described in section 2.3.3 connected components analysis (CCA) works by grouping connected pixels into labelled regions, and works equally well in 3D as well as 2D. Because of the nature of the airway, being a connected entity when considered from a 3D perspective, CCA guarantees to segment the airway from the other tissue (and indeed the background) whilst losing no important information.

3.3 Airway Component Selection

Moving from the relatively simple task of removing the background component (which is simple as it is always the largest component) to the more considered task of selecting the component containing the airway. I planned to use a heuristic, based on some prior knowledge, to determine the correct component. This heuristic was to be based on two things: firstly location of the component in the first slice in which it appears (the airway would appear in the centre). Secondly the size of the component in terms of z, the number of slices in which it appears, as the airway component would exist in more slices than the others. Once component selection has been completed the rest of the algorithm (airway contouring etc) will deal only with the specified part of the image volume.

3.4 Modelling

After CCA and component selection the airway may or may not be connected to one or both of the lungs. This is because of the problem of leakage during the CCA phase. I planned to solve this through modelling of the data.

3.4.1 Model Design

The design of the model is described by an industry standard which is followed when the process is done manually, as shown figure 2.1.
This figure shows the generations of bronchi which are segmented when the process is conducted manually. Further generations are considered too small to be concerned with during radiotherapy. Over this image I have drawn a graph representation (circles depicting vertices and lines showing the edges between them) of what the model fitting would search for a match to. However, prior to the model fitting skeletonisation is required to minimise the search space.

Figure 6: Bronchial Model

3.4.2 Skeletonisation

Skeletonisation, as described in section 2.4.2, is the process of thinning an image (in this case a 3D image) to an eroded version of that image. I planned to perform skeletonisation through an iterative thinning method. This is a type of method where, with each pass of the algorithm the outer-most pixels of the image are removed until it is determined that the remaining image is only one pixel in width. Whilst this is not an exact science (often the image remaining will not be exactly one-pixel wide, more likely a small group of pixels) it is a useful method for narrowing the space to be searched during the model fitting.

3.4.3 Model Fitting

The starting point for the model is a simple choice; it is plainly the pixel which appears as part of the airway in the uppermost slice. From here the search will move down the thinned airway component assessing at each point whether a split is likely to exist. The probability of a split will be mainly based on the number of neighbours the pixel has on a given slice, compared to its previous and subsequent slices. In theory this number will be higher in the case of a splitting point. Once the search has found the desired number of splits (as denoted by figure 2.1) the remaining points not in the fitted model will be discarded. What is left will
be a graphical representation of the airway; this will then be related to the original segmentation (after CCA) to determine the image of the airway to be contoured.

3.5 Contouring

In the very beginning I planned to use edge detection for returning the points lying on the boundaries of the airway components; however it became clear that this would not always guarantee adequate results. It became clear that a boundary tracing method would be better suited.

3.5.1 Scanning

Boundary tracing requires a starting point; therefore I planned to use a method of ‘scanning’ to move through each slice, row-by-row, until an element of the airway is located. At this point the algorithm would use the current pixel as a starting point for the boundary trace.

3.5.2 Boundary Tracing

A description of how boundary tracing works can be found in section 2.7.2. This method would guarantee a contour with a constant width of a single pixel (something that could not be guaranteed with edge detection) and it also assures that the contour will be a complete polygon. The coordinates of the boundary trace can be plotted in matlab to produce visual results, something I will use in the evaluation in section 5; however this is not the desired outputting method to meet the aim of this project. To achieve that, the information has to written to a Dicom file.

3.6 Writing the Results

The results must be written to a Dicom RT file, as this is the standard format read by the computer in radiotherapy planning. As the datasets I have been supplied already contain the contour information generated manually I planned to use this existing file as a template, and simply overwrite the image data produced by my application.
This section will follow the iterative development process of the application, describing and explaining any changes which occurred since the design stage.

4.1 Iteration One

The first iteration of the development included reading the data into matlab and performing connected components analysis (CCA) on the data volume. In section 3.1 CERR is discussed as a possible method for reading the data, however I felt that this had a serious drawback, the need for user interaction. I wanted to make the program as automated as possible, therefore I moved from this method to one which could achieve this. The final approach implements a loop through the Dicom file structure, based on the file ID location of the image data (this ID is known, and is always the same). This image data is then read into a volumetric set slice-by-slice. The only negative effect of this is that the slice order of the series is reverse, however this is easily managed.
The second part of iteration one combines CCA with thresholding, to generate a set of labelled components. At this stage in the development the airway component is chosen manually, by finding its label number through a volume viewing tool. Figures 7 and 8 show images of the same slice before and after applying a threshold. Figure 9 shows the image of the same slice after connected components analysis has been applied.
4.2 Iteration Two

This iteration involved moving from manual component selection to an automated approach. As discussed in section 3.3 this would involve creating a heuristic based on certain aspects of the component. After discussions with Jonathan Sykes it became apparent that the airway would always appear in the first slice of the series, because obviously the airway continues up to the mouth but the scan stops around the top of the shoulders. The lungs, however, are guaranteed not to appear in this first slice, meaning a search within the first slice for the largest component (in all three directions) was a simple heuristic which worked perfectly. This information is then used to create a new volume of data containing only the airway component. Figure 10 is of the same slice as the previous three, showing the binary image of the slice after component selection has been executed.

![Figure 10](image)

4.3 Iteration Three

Iteration three was the contouring of the airways in image data. In the design section it was discussed that this would be done by boundary tracing, and that an initial point for this trace would be found through scanning. During the development, however, certain problems arose with the scanning approach when two instances of the airway appeared on the same slice (something that happens after the splits). It arose that instances could become hidden behind each other as the scan would not pass through components already contoured, leading to an approach which required scanning in each of the four directions on every slice. This became too computationally demanding both in terms of space and time. Therefore a better approach was suggested, the use once again of CCA. This time, however, the CCA would be performed on a 2D basis on each slice to distinguish the different airway instances. Each instance would then be placed into its own image. A simple search for the first
occurrence of a foreground pixel to serve as the arbitrary starting pixel was then performed before boundary tracing could begin. A more detailed description of boundary tracing can be found in section 2.7.2, but simply put the algorithm executes on each by assessing its 8 neighbours in clockwise order until it finds the next outermost foreground pixel. This continues until it returns to the starting point. This method guaranteed a closed polygon one pixel in width on the outline of the airway.

### 4.4 Iteration Four

By this point in the development process the project met the minimum requirements laid out in section 1.2, therefore it seemed a good point to deal with writing the output back into the Dicom file format. As discussed in section 3.6 I planned to use the existing Dicom RT as a template, upon which to simply change the data points which needed changing. Other than difficulties with the nature of Dicom, such as the need to piece together specific strings of information before evaluating them later, this approach worked successfully.

### 4.5 Iteration Five

Iteration five is where the largest difference between design and implementation came to fruition. As discussed in the design section the original plan was to use model fitting to prune parts of data that, through leakage, were wrongly classified as airway. This approach failed for reasons such as skeletonisation not being an exact science, meaning that using this to create a graph-like representation of the data has serious precision issues. Because of this it was suggested that an erosion-dilation method would be a suitable alternative (more information on these techniques can be found in sections 2.4.1 and 2.6.1). The purpose of

![Figure 11](image-url)
dilation is split the airway from the lung information connected through leakage in the CCA process. This is done through using a 3D elliptically-shaped structuring element which is significantly larger in the z direction than the x and y. The effect of this is that shallower connections are more likely to be broken whilst less intrusive erosion is performed on the longer airway (figure 11 shows the results of the erosion process). After erosion airway component must be performed again to correctly re-establish the image data relating to the airway, which is done through the exact same method as is described in iteration two, section 4.2. As described in section 2.6.1 dilation must be performed after erosion (and in this case also after the component selection) to retrieve the original image data. My application uses an identical structuring element as the erosion for the dilation, meaning the image returns to something that, if not pixel-perfectly identical, is an accurate representation of the original.

4.6 Iteration Six

Iteration six, the final part of the development process, is the validation of the method. I had 3 sets of data to use throughout development, each containing 12 separate series of images corresponding to the different phases of the breathing cycle of each of the 3 patients. This created a good sized test bed to work out the bugs of my algorithm. Main things I assessed were that the program ran with no errors on all sets, that the program correctly distinguished the correct component as airway and that the erosion process effectively removed all unwanted information.
Chapter 5

Evaluation

5.1 Description of Results

This section will describe the general competence of the method performed on the three datasets. I will also analyse the Hausdorff distance between the results of the method on the known ground truth.

5.1.1 Dataset One

Figure 12: Slice 145 (top slice)
Figure 13: Slice 135

Figure 14: Slice 125
Figure 15: Slice 115

Figure 16: Slice 105
As you can see in figures 12 to 15 the segmentation is relatively simple. In figure 16 the airway is beginning to split, as is the segmentation, and in the following figures you will see how the algorithm follows the split airway.

*Figure 17: Image (a) of slice 95*

*Figure 18: Image (b) of slice 95*
As you can see from figures 17 and 18 the segmentation has split into multiple components. Whilst I have separated the data here for the purpose of closer examination, when the data is written to the Dicom file it will rejoined.
In figures 19 and 20 it can be seen that already on the right-hand lung the airways are too small to be concerned with, which is also the case for the left-hand lung shortly after slice 75.

5.1.2 Dataset Two

Figure 21: Slice 155

This slice is not the top slice of dataset two; instead the first 25 slices have been omitted. This is because this dataset (for an unknown reason) scans much farther up the neck. Whilst segmentation has been successful up to the top of the series (as will be shown in the evaluation in section 5.2) I deemed the images not interesting enough to show.
Figure 22: Slice 145

Figure 23: Slice 135
Figure 24: Slice 125

Figure 25: Slice 115
Figures 21-25 show the contours following the main airway, as in the previous dataset. Figures 26-28 are images of the same slice, showing how there are 3 instances of the airway visible. The splitting has occurred a similar distance down into the lungs as dataset one (roughly slice 105) due to the similarities in the lung structures, even though these datasets are from two different patients. As before I have separated the results into different images, showing the different branches of the airway.

*Figure 26: Image (a) of slice 105*
Figure 27: Image (b) of slice 105

Figure 28: Image (c) of slice 105
By the time the segmentation has reached the next interval at slice 95 (as shown in figure 29) it can be seen that the algorithm is no longer concerned with the branches on the left-hand lung as they have become too small. This is the last interval at which segmentation has occurred in this dataset.

Figure 29: Slice 95
5.1.3 Dataset Three

Figure 30: Slice 145

Figure 31: Slice 135
Figure 32: Slice 125

Figure 33: Slice 115
As with the previous two datasets we can see between the top slices and around slice 105 (figures 30 to 34 in the case of dataset 3) the segmentation follows the main airway up to its first split. By the next interval, slice 95 (figures 35 and 36), the airway has split.
These results show segmentation is a relatively simple process up to where the splits begin, roughly around slice 105. Therefore I would like to show further results from the second dataset from around this slice to where the contours end, however they will be at 3-slice intervals as opposed to 10-slice intervals. Figures 37-47 proceed from slice 106 to slice 91 showing all instances of segmentation.
Figure 37: Image (a) of slice 106

Figure 38: Image (b) of slice 106
Figure 39: Image (c) of slice 106

Figure 40: Image (a) of slice 103
Figure 41: Image (b) of slice 103

Figure 42: Image (a) of slice 100
Figure 43: Image (b) of slice 100

Figure 44: Slice 97
Figure 45: Image (a) of slice 94

Figure 46: Image (b) of slice 94
5.2 Evaluation of Results

In this section I will use the Hausdorff distance to evaluate the difference between the ground truths I have been given for these datasets to the results which my program has produced. I will assess this per dataset on a slice-by-slice basis, as I feel this will give the best impression of the accuracy of the results through the generations of the airways.

5.2.1 Dataset One

Figure 48 on the following page shows the analysis of dataset one. The range of the differences is 26.8590, the average is 18.3633 and the standard deviation is 5.9334. Although the range of the results is a little larger than I would like the average is acceptable. Also the small standard deviation shows a relatively pleasing uniformity of results. The distances are steadily up and down throughout the slices, putting to one side some concerns I had that the results would become less and less accurate down through the smaller airways after the split points.
<table>
<thead>
<tr>
<th>SLICE</th>
<th>HAUSDORFF</th>
</tr>
</thead>
<tbody>
<tr>
<td>145</td>
<td>21.1680</td>
</tr>
<tr>
<td>143</td>
<td>21.1450</td>
</tr>
<tr>
<td>141</td>
<td>20.4880</td>
</tr>
<tr>
<td>139</td>
<td>21.0740</td>
</tr>
<tr>
<td>137</td>
<td>21.4180</td>
</tr>
<tr>
<td>135</td>
<td>17.4410</td>
</tr>
<tr>
<td>133</td>
<td>15.4410</td>
</tr>
<tr>
<td>131</td>
<td>13.1840</td>
</tr>
<tr>
<td>129</td>
<td>15.1370</td>
</tr>
<tr>
<td>127</td>
<td>8.4650</td>
</tr>
<tr>
<td>125</td>
<td>6.4650</td>
</tr>
<tr>
<td>123</td>
<td>3.5120</td>
</tr>
<tr>
<td>121</td>
<td>18.1130</td>
</tr>
<tr>
<td>119</td>
<td>21.0660</td>
</tr>
<tr>
<td>117</td>
<td>23.0430</td>
</tr>
<tr>
<td>115</td>
<td>25.9960</td>
</tr>
<tr>
<td>113</td>
<td>12.3710</td>
</tr>
<tr>
<td>111</td>
<td>15.3480</td>
</tr>
<tr>
<td>109</td>
<td>16.3240</td>
</tr>
<tr>
<td>107</td>
<td>27.9960</td>
</tr>
<tr>
<td>105</td>
<td>24.9960</td>
</tr>
<tr>
<td>103</td>
<td>17.9730</td>
</tr>
<tr>
<td>101</td>
<td>12.4410</td>
</tr>
<tr>
<td>99</td>
<td>17.6050</td>
</tr>
<tr>
<td>97</td>
<td>21.4650</td>
</tr>
<tr>
<td>95</td>
<td>23.9260</td>
</tr>
<tr>
<td>93</td>
<td>25.8790</td>
</tr>
<tr>
<td>91</td>
<td>16.1370</td>
</tr>
<tr>
<td>89</td>
<td>15.3480</td>
</tr>
<tr>
<td>87</td>
<td>15.6290</td>
</tr>
<tr>
<td>85</td>
<td>17.4650</td>
</tr>
<tr>
<td>83</td>
<td>21.5590</td>
</tr>
<tr>
<td>81</td>
<td>30.3710</td>
</tr>
</tbody>
</table>

*Figure 48: Information on results of dataset one*
5.2.2 Dataset Two

<table>
<thead>
<tr>
<th>SLICE</th>
<th>HAUSDORFF</th>
</tr>
</thead>
<tbody>
<tr>
<td>170</td>
<td>12.1130</td>
</tr>
<tr>
<td>168</td>
<td>14.0430</td>
</tr>
<tr>
<td>166</td>
<td>14.0430</td>
</tr>
<tr>
<td>164</td>
<td>18.9260</td>
</tr>
<tr>
<td>162</td>
<td>19.8790</td>
</tr>
<tr>
<td>160</td>
<td>25.7620</td>
</tr>
<tr>
<td>158</td>
<td>23.7620</td>
</tr>
<tr>
<td>156</td>
<td>19.7850</td>
</tr>
<tr>
<td>154</td>
<td>15.7850</td>
</tr>
<tr>
<td>152</td>
<td>12.7850</td>
</tr>
<tr>
<td>150</td>
<td>15.6680</td>
</tr>
<tr>
<td>148</td>
<td>13.6680</td>
</tr>
<tr>
<td>146</td>
<td>12.6680</td>
</tr>
<tr>
<td>144</td>
<td>10.6680</td>
</tr>
<tr>
<td>142</td>
<td>10.6680</td>
</tr>
<tr>
<td>140</td>
<td>10.6680</td>
</tr>
<tr>
<td>138</td>
<td>9.6680</td>
</tr>
<tr>
<td>136</td>
<td>8.6680</td>
</tr>
<tr>
<td>134</td>
<td>9.6210</td>
</tr>
<tr>
<td>132</td>
<td>8.6210</td>
</tr>
<tr>
<td>130</td>
<td>8.5980</td>
</tr>
<tr>
<td>128</td>
<td>8.5980</td>
</tr>
<tr>
<td>126</td>
<td>9.5270</td>
</tr>
<tr>
<td>124</td>
<td>9.5270</td>
</tr>
<tr>
<td>122</td>
<td>9.5270</td>
</tr>
<tr>
<td>120</td>
<td>10.5040</td>
</tr>
<tr>
<td>118</td>
<td>9.5040</td>
</tr>
<tr>
<td>116</td>
<td>8.5040</td>
</tr>
<tr>
<td>114</td>
<td>14.3870</td>
</tr>
<tr>
<td>112</td>
<td>33.1450</td>
</tr>
<tr>
<td>110</td>
<td>24.8320</td>
</tr>
<tr>
<td>108</td>
<td>42.0980</td>
</tr>
<tr>
<td>106</td>
<td>39.0270</td>
</tr>
<tr>
<td>104</td>
<td>34.9800</td>
</tr>
<tr>
<td>102</td>
<td>35.8630</td>
</tr>
<tr>
<td>100</td>
<td>15.1370</td>
</tr>
<tr>
<td>98</td>
<td>84.7070</td>
</tr>
<tr>
<td>96</td>
<td>73.9410</td>
</tr>
<tr>
<td>94</td>
<td>78.9650</td>
</tr>
<tr>
<td>92</td>
<td>77.0820</td>
</tr>
<tr>
<td>90</td>
<td>9.1910</td>
</tr>
</tbody>
</table>

Figure 49: Information on results of dataset two
Figure 49 shows Hausdorff results for the second dataset. The overall range of the distances is 76.2030, the average is 22.5637 and the standard deviation is 20.7824. Whilst the average is only slightly higher than the previous dataset the range and standard deviation are far less than acceptable, and this is because of the concern I mentioned in section 5.2.1. In the first dataset the accuracy drop-off towards the lower slices was not too drastic, in this dataset however it is a big problem. For the first 56 slices the range is 17.2580, the average is 12.9705 and the standard deviation is 4.6976; an ideal analysis of the data if the form had continued to the end. I will suggest reasons for this in section 5.2.4.

5.2.3 Dataset Three

Figure 50 on the next page shows the Hausdorff results for dataset three, and they are by far the results I am most happy with. It shows the lowest range, 14.5066, and whilst it has a slightly higher average than dataset one, 20.1745, it has the lowest standard deviation, 4.3077. The results are very uniform with no notable outliers. Beyond this, however, the most pleasing fact is that the data shows no discernable drop off in the lower generations of the airways, unlike the second dataset.
<table>
<thead>
<tr>
<th>SLICE</th>
<th>HAUSDORFF</th>
</tr>
</thead>
<tbody>
<tr>
<td>145</td>
<td>14.1680</td>
</tr>
<tr>
<td>143</td>
<td>18.1450</td>
</tr>
<tr>
<td>141</td>
<td>19.4880</td>
</tr>
<tr>
<td>139</td>
<td>20.0740</td>
</tr>
<tr>
<td>137</td>
<td>18.4180</td>
</tr>
<tr>
<td>135</td>
<td>20.9709</td>
</tr>
<tr>
<td>133</td>
<td>25.3863</td>
</tr>
<tr>
<td>131</td>
<td>28.0656</td>
</tr>
<tr>
<td>129</td>
<td>24.4233</td>
</tr>
<tr>
<td>127</td>
<td>14.4650</td>
</tr>
<tr>
<td>125</td>
<td>15.4650</td>
</tr>
<tr>
<td>123</td>
<td>14.5120</td>
</tr>
<tr>
<td>121</td>
<td>23.6493</td>
</tr>
<tr>
<td>119</td>
<td>23.6493</td>
</tr>
<tr>
<td>117</td>
<td>21.6397</td>
</tr>
<tr>
<td>115</td>
<td>24.2964</td>
</tr>
<tr>
<td>113</td>
<td>16.5350</td>
</tr>
<tr>
<td>111</td>
<td>13.5590</td>
</tr>
<tr>
<td>109</td>
<td>14.3240</td>
</tr>
<tr>
<td>107</td>
<td>24.2964</td>
</tr>
<tr>
<td>105</td>
<td>23.3757</td>
</tr>
<tr>
<td>103</td>
<td>23.3334</td>
</tr>
<tr>
<td>101</td>
<td>24.9260</td>
</tr>
<tr>
<td>99</td>
<td>24.4650</td>
</tr>
<tr>
<td>97</td>
<td>20.9260</td>
</tr>
<tr>
<td>95</td>
<td>14.7230</td>
</tr>
<tr>
<td>93</td>
<td>14.8160</td>
</tr>
<tr>
<td>91</td>
<td>19.7930</td>
</tr>
<tr>
<td>89</td>
<td>23.1714</td>
</tr>
</tbody>
</table>

*Figure 50: Information of results on dataset three*
5.2.4 Overall Evaluation of Results

Overall I am relatively pleased with the results my application has shown.Datasets one and three show good consistency and a fairly low average Hausdorff distance. Looking back on the images provided through section 5.1 I believe the reason for the difference in results is that my contours lay (consistently) slightly inside the walls of the airway. I feel the best way to approach a solution for this would be to look into the thresholding. I wanted to choose a single threshold for all three datasets, and because of this it had to be quite conservative. A dynamic thresholding approach may provide better results between varying datasets as levels of contrast and brightness can fluctuate between datasets.

With respect to the poor results seen in relation to dataset two, I feel that the reason for this may have been because this dataset required the most erosion to separate the airway from the lungs. Although dilation was performed to return the remaining airway information to its former shape I believe that the higher level erosion broke away part of the airway at the points where leakage had occurred. I chose an erosion approach because I felt it was an acceptable alternative to model-fitting, and whilst there is a chance model-fitting could have suffered from this same issue, it would have been a more capable approach for solving it.

5.3 Overall Evaluation of Project

In this section I will evaluate how my final implementation meets my original aims and requirements in terms of efficiency/speed, ease of use and adaptability to real world application.

The program takes just less than 2 minutes to run on a single dataset, the most time demanding processes being the erosion and dilation. I believe that this is an acceptable running time as the initiating of the algorithm is the only user interaction required. My initial aim was to reduce the time surgeons devoted to segmentation. The running of my algorithm theoretically requires no surgeon’s time at all as it can be run by anyone. At its current level of consistency, however, it would require a surgeon to check the accuracy of the results prior to radiotherapy. If the method could be refined and the results extensively tested then it is perfectly feasible that a lot of time could be saved.
In terms of ease of use the program requires no user interaction once the algorithm is started; assuming user could establish the name of the Dicomdir file and the name of the folder containing the Dicom series. I have discussed in chapter 6 how ease of use could be further improved through the addition of a graphic user interface.

Finally I would like to comment on the adaptability of my application to the real world problem I set out to solve. In the very beginning I commented that unless the results were very close, if not identical, to the ground truth then the program would not be accepted. This is because of the fact that radiotherapy could a dangerous procedure if this information was incorrect. As shown through section 5.2 the results are not identical to the ground truth, and whilst I would have accepted a Hausdorff distance of between 5 and 10 for the purposes of this project, an overall average of 20.3672 indicates that the program needs improvement before implementation in real radiotherapy planning could be considered.
Chapter 6

Further Work

At the start of this project I outlined two possible extensions for the application, I have also subsequently considered a third:

• Creating a GUI for the system.
  o I felt this would be a good idea to make the program more user friendly, bearing in mind that the users will be surgeons and not computer scientists.
  o A possible implementation would have to allow the user to select the desired dataset for the input and allocate a destination for the results. It could also give the user the option of a visualisation of the results, to allow the user to check for errors. However, the only user interaction should be at the very beginning and very end of the process.

• Analysing multi-volume segmentations
  o Using a multi-volume approach to the segmentation of the airways could shed more light on how much each of the different parts of the chest cavity move during the patients breathing.
  o Whilst my application can handle the segmentation of multiple volumes the further work here would come from the analysis and evaluation of this extra information for the purpose of determining how much the airway moves. This knowledge can mean that the margins for the radiotherapy laser can be set more accurately.

• Segmentation of more than just the airway.
  o During the planning process a surgeon not only outlines the airway but also many other things including: the tumour itself, a 2cm area of interest outside the airway and the oesophagus and pericardium.
  o Research into the segmentation of these areas, as well as the airway, could lead to the complete automation of the radiotherapy planning process, saving surgeons even more valuable time.
Bibliography


Appendix A

Personal Reflection

This project has been the largest single body of work which I have ever undertaken; something which I imagine is the case for most other final year students. I hope through outlining the good, the bad and the difficult times this project has produced it will give future students a better understanding of what it means to devote a full semester to a huge single task.

Starting at the beginning, before the beginning in fact, the choice of the project to undertake is a huge one to make. I felt from the start that I wanted to do something in vision field of computer science, and the medical context was also something that greatly interested me, and enjoying the subject matter is vital when embarking on something of this scale. Aside from the subject matter I was also keen to have Derek as my supervisor. I knew that a good working relationship with the supervisor would help greatly, and having been in Derek’s lectures before, I highly rated his motivational teaching style.

Moving onto what is possible the single most important thing to think about over the course of the project, time management. It was suggested to me at the start of the project that the time at the start of the semester, when it doesn’t feel like there is much work to be done, is often time you reflect upon in the end and feel like it could have been better used. Personally I would agree that this has certainly been the case. With hindsight I would have gotten into the development more thoroughly much earlier on, giving more time to refine towards the end as opposed to feeling pressured to just get results. Whilst I believe the program I have produced meets, and slightly surpasses, my minimum requirements I do accept that given better time management it could have been better. As well as time management, it is equally important to keep motivated throughout, and over a 12+ week period self-motivation can
prove difficult, therefore to remain motivated it was vital for me to have a firm interest in the subject matter.

I would also likely to note on the difficulties of searching through research articles, journals and conferences to find relevant and useful information, and suggest to future students that they take this area seriously. It is important to allow a good portion of time for finding quality research, and don’t underestimate the usefulness of the literature search seminar at the start of the semester!

Another decision to be made by future students is the method used for the write up, and I would add that this is a decision that should be made very early on. All suggestions made in the beginning were to use LaTEX; however because I was less experienced using this tool I decided it was a learning curve I couldn’t afford to make time for and went with Microsoft Word. I felt this was a positive decision; I have had very few, if any issues over the course of the write up, whilst I have heard reports from those who are who have struggled with LaTEX.

In closing I would add that I have thoroughly enjoyed the experience of my final year project. It has been demanding and challenging in more ways than just academically, but I will look back over the semester as a whole with a feeling of satisfaction.
Appendix B

Technical Acknowledgements

This section will outline any sections of code used in my implementation that were not written by myself (not including built-in matlab functions)

- The function parseDicomdir() was used to read the Dicom information into matlab in a format that matlab can understand. This function was found on the Matlab Central File Exchange: [http://www.mathworks.de/matlabcentral/fileexchange/27740-dicomdir-parser/content/parseDicomdir.m](http://www.mathworks.de/matlabcentral/fileexchange/27740-dicomdir-parser/content/parseDicomdir.m).

- A small section of code was suggested by Derek Magee, and although this did not translate into the final implementation word-for-word it was heavily influenced. These sections were the loop which uses the result of parseDicomdir() to sort the image data into a volumetric series.

- Finally I used another function from the Matlab File Exchange to execute the Hausdorff distances between my results and the ground truths: [http://www.mathworks.com/matlabcentral/fileexchange/26738](http://www.mathworks.com/matlabcentral/fileexchange/26738).