AUTOMATED SELECTION AND ENTRY OF COMPUTED
TOMOGRAPHY DATA IN FINITE ELEMENT
MODELLING OF THE HUMAN FEMUR

DAWNA RAE GREENING

INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

NOTE TO USERS

This reproduction is the best copy available

UMI

AUTOMATED SELECTION AND ENTRY OF COMPUTED TOMOGRAPHY DATA IN FINITE ELEMENT MODELLING OF THE HUMAN FEMUR

BY

©DAWNA RAE GREENING, B.SC., B.ENG.

A THESIS SUBMITTED TO THE SCHOOL OF GRADUATE
STUDIES IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF
MASTER OF ENGINEERING

FACULTY OF ENGINEERING AND APPLIED SCIENCE
MEMORIAL UNIVERSITY OF NEWFOUNDLAND
AUGUST, 1998

ST. JOHN'S   NEWFOUNDLAND   CANADA

ABSTRACT

Researchers in the Biomedical Engineering Centre (BEC) at Memorial University of Newfoundland are studying the structural and biomechanical properties of the human femur in order to better model and test hip prostheses. The advent of computed tomography (CT) technology has allowed researchers to visualize and use 3D anatomical data for such studies. In an attempt to generate realistic and useful 3D computer models of the human femur, BEC researchers have been using geometric and densimetric data from CT imagery to construct finite element models. Initial construction methods required manual input and manipulation of the data. This proved tedious and labour intensive.

The aim of this project was to reduce user involvement and develop a software package to automate the model building process. By simplifying the procedure, it is hoped that users will be able to build models quickly and easily with minimal training in modelling software. Software modules were developed using several languages and run on a personal computer networked to a UNIX system. The software reads and translates CT data, detects and tracks the inner and outer edges of the femoral cortical bone, and selects equidistant points along these edges which define the geometry. This geometric data, combined with densimetric data in the form of CT numbers, is used to generate input files for the ANSYS finite element modelling package, which generates a solid volume model of the femur used for further

testing.

With restricted access to specimens and CT scanning facilities, testing of the software was limited throughout the course of this project. In view of the small test base, the results must be interpreted with caution, but preliminary results are encouraging. Compared with manual procedures used prior to this project, user involvement and the time needed to generate a model are dramatically reduced. The models are smoother in appearance and are easier to mesh. Direct measurement comparisons between true edges, and manually and automatically selected model edges indicate that the automated models are equal, if not superior, to those generated using the manual technique.

ACKNOWLEDGMENTS

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

BEC	Biomedical Engineering Centre
CCAE	Centre for Computer Aided Engineering
CT	Computed Tomography
FEM	Finite Element Modelling
GUI	Graphical User Interface
LUB	Least Upper Boundary
MUN	Memorial University of Newfoundland
PC	Personal Computer
ROI	Region of Interest
SAE	Sum of Absolute Error
SAPE	Sum of Absolute Percentage Error
VOI	Volume of Interest

CHAPTER 1

INTRODUCTION

1.1   Motivation for Research

With an ever-growing population of people over the age of 50, the number of hip fractures

is expected to double in the next 50 years [22]. Instances of hip disease, such as arthritis and

osteoporosis, are also expected to increase dramatically. The current treatment for many of

these hip ailments involves surgery, wherein the injured or diseased bone is removed and

replaced with a metallic hip implant.
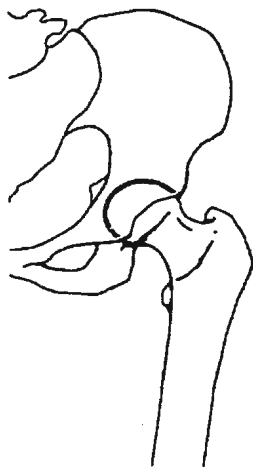
Figure 1.1*
Normal left hip joint

*Figures based on illustrations
in *Hip Hip Hooray Journal*,
Spring 1998
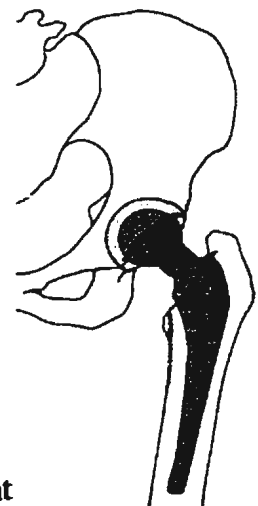
Figure 1.2*
Total hip
replacement
showing implant

1

There are many types of implants available, and selection among these is presently left to the best judgement of medical professionals. The data available to that team is often drawn from generalized, non-comparative research results, with limited data as it relates to a particular patient. Both doctor and patient would benefit if a method were developed which would allow comparison of computer models of various hip implants in actual femurs, and thereby aid in the selection of an implant prior to invasive surgery.

Other researchers have generated generic computer models of the femur. Where models have been developed from CT data, the model building has required considerable computing power and data storage, and few models have been validated by comparison with physical tests on the specific femur being modelled. The goal of this project was to develop procedures to produce useful computer models of specific femurs, with modest manual effort and modest computing resources. Working towards this goal, researchers in Memorial University of Newfoundland's Biomedical Engineering Centre (MUN BEC) are currently studying the structural and biomechanical properties of the human femur in order to better model and test the interaction between hip implants and specific, actual femurs. Computer models being developed using finite element modelling (FEM) techniques are being validated through corresponding physical testing. If valid computer models can be generated which accurately simulate physical testing, then computer-modelled hip implants and femur models can be used to perform virtual tests, and simulated testing of *in vivo* specimens will be possible. This virtual surgery and implant testing will provide the medical team with

2

valuable, patient-specific data prior to surgery.

1.2   Current Method Used To Model Femurs

Another graduate student in the BEC, Mr. Paul Smith, is developing finite element models

of the human femur and obtains bone dimensions and density data from CT scans, using the

density values to provide mechanical property data for the model. The aim of his research

is to model specific femurs, reproducing their mechanical behaviour with sufficient accuracy

for research and clinical use.

The basic principle of computed tomography is that a series of stackable, 2D image slices are

collected by rotating an X-ray tube and detector array in a full circle around a patient, as

illustrated in Figure 1.3. The width of the X-ray beam
determines the thickness of the slice or the tissue
depth to be scanned. The amount of attenuation of the
X-ray beam as it passes through that tissue volume is
recorded by the detector, and using a series of angular
projections and mathematical techniques, a 2D cross-
sectional slice of image pixels is produced. The
resulting gray scale image represents the distribution
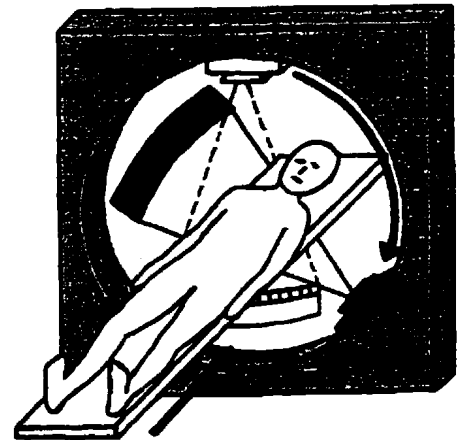
Figure 1.3
Operation of CT scanner

3

of X-ray attenuation within the tissue slice, and each gray scale value is a CT number obtained by volume averaging over the slice depth. Orthopaedic research has shown that these CT numbers increase with increasing bone density, and therefore, elastic modulus. A series of such slices can be treated as a 3D structural representation of CT numbers. Thus, digital CT data provides both geometric and densimetric information which can be further processed through image processing and 3D modelling techniques.

Using this CT scan data, the finite element method is being used to model the complex geometry of the femur, and study stress and strain distribution patterns. In the finite element method, a complex shape, such as a femur, is represented by a series of simpler shapes or finite elements. By combining the simpler models of each individual element, a model of the entire shape can be obtained.

NODES
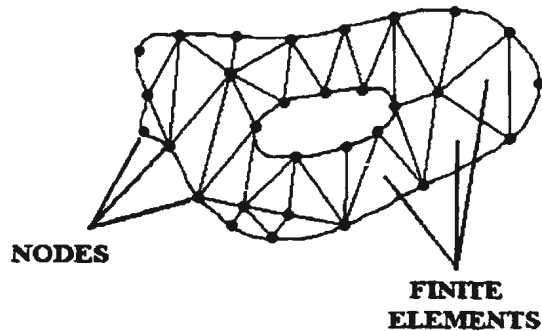
FINITE ELEMENTS

Figure 1.4
Basic finite element model

The finite element modelling process requires that a series of nodes, or keypoints, be defined describing the geometry of the object to be modelled, and that each of the resulting elements

4

be assigned material properties.

Therefore, the first step in building a finite element model of the femur is to obtain keypoints outlining its geometry, and to quantify a material property of the femur. While any boundaries can be used to define a geometry, the keypoints used in this project were the 3D coordinates of the points marking the outer surface of the femur and the inner surface of the medullary (marrow) cavity running through the center of the shaft. The material property quantified was bone density. Other researchers working in this area have successfully used CT data to obtain details of bone geometry, and have related CT numbers to density. In developing a similar technique in the BEC, a manual, user-intensive procedure was used to extract this information from the CT imagery. This manual method required that points be manually selected, their coordinates extracted to data files, and recombined with relevant CT numbers to form an input file for the finite element modelling package. However, research progress was slow due to difficulties in extracting this data from the CT files and using this information to build finite element models.

1.3   Shortcomings of the Manual Method

The manual procedure had several shortcomings.

1)   The CT data files, as obtained from local hospitals, are not in a standard image format and are difficult to read. BEC researchers had difficulty reading the CT files, and were unable to access individual CT numbers. This information was needed if material properties were to be assigned to the model's finite elements.

2)   The method used to generate keypoints was slow and labour intensive. For each CT image corresponding to a bone slice, users had to view and manually place points marking the edges of the cortical bone, as shown in Figure 1.5. This was done using a point placement tool from an image processing package downloaded from the internet (ImageTool for Windows V1.28, ©The University of Texas Health Science Center in San Antonio.) This process was tedious and time-consuming, and user judgement defined where the points were best placed, making this method prone to user bias and human error. Another problem was that the graphical point placement routine used to mark the
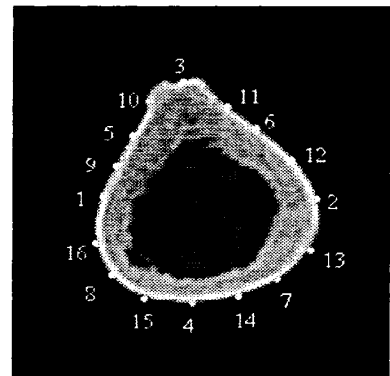
Figure 1.5
Manual point selection -
The numbering indicates
the order in which the
points were selected

6

points was not intended for this purpose, and did not permit correction of individual points. If an error was made, all points for that slice had to be reentered. As points were selected, their coordinates were automatically placed into a data file. When all points were identified for all slices, these data files had to be manually combined, and edited into a format compatible for input into the finite element modelling software package. This cut-and-paste technique was also laborious.

3) If a smooth and accurate model is to be constructed, a large number of keypoints is needed. Fewer points lead to a more generalized representation of the surface, and structure definition is often lost. In order to capture the true geometry of the femur and any subtle irregularities, more points and slices should be used. The manual technique was somewhat prohibitive in the number of points that could be placed, simply due to the amount of time required and the tedious procedures which had to be followed. As an example,

$$
\begin{array}{ll}
1 \text{ Femur Scan } = & \sim \quad 90 \text{ slices/scan} \\
& \text{x} \quad 2 \text{ edges/slice} \\
& \underline{\text{x} \quad 16 \text{ points/edge}} \\
& 2880 \text{ points to be placed}
\end{array}
$$

At 5 seconds per point, it would have taken 4 hours just to select these points, assuming no errors were made.

4) Registration, or alignment of points between CT slices, can also be a problem. In

7

order to generate a smooth model, the points should be aligned longitudinally between slices, allowing smooth splines to be drawn through these points. Using human judgement, it was difficult to correctly place and align these points from one slice to the next, and the slightest shift in point registration could have lead to rumples which became visible only when the 3D model was constructed. If the flaws were serious enough to question the validity of the model, the user had to reenter the points for the appropriate slices, reconstruct the input file, and generate a new model.

While all of these problems could be overcome using the manual method, the time constraints and the opportunity for human error were a deterrent to model building within the BEC, and the restrictive nature of the manual procedure led researchers to recognize the need for automation. Due to the specialized nature of this project, off-the-shelf software was not available, and other research labs, who developed similar software, were reluctant to share their techniques. As such, a custom software package was needed to advance the research being conducted in MUN's Biomedical Engineering Centre.

1.4   Scope of Research and Organization of Thesis

With this goal in mind, this thesis describes the development and testing of a computer-assisted method to generate 3D models of the human femur. The main focus was the

8

generation of input data, obtained from CT scans, for the construction of finite element models. The new software package read CT data files and automatically generated files of keypoint coordinates, as well as the CT numbers from which the material properties used with each element were derived. It automated much of the modeling process, making it quicker and easier to generate finite element models of femurs. Existing computer resources were used for this project.

The thesis begins with a review of literature on computer techniques used in modelling bone from CT data and edge detection techniques in bone imagery. The search was restricted to topics directly relevant to the ongoing work, and leant itself towards answering specific questions:

- Who has conducted similar research on 3D modelling of femurs?
- How did they obtain CT data?
- Was specialized hardware used?
- Did they develop custom software or use specialized image processing and FEM packages?
- How did they read the CT data, distinguish between cortical and trabecular bone, detect edges, and select points?
- Once they had the points, how did they construct the 3D model?

Chapter 3 discusses development issues and outlines the methods used to automate the model building process. Select details of software development are discussed, but for proprietary

reasons, software listings are not included. Algorithmic details are limited, but sufficient detail is given to explain and validate the methodology. This is followed in Chapter 4 by a discussion of the preliminary software testing and results, and conclusions and recommendations are given in Chapters 5 and 6. References are cited in Chapter 7, with supporting material being given in the Appendices and Bibliography.

CHAPTER 2

REVIEW OF LITERATURE

2.1 Introduction

Universities and labs conducting orthopaedic research similar to the studies at the BEC have developed in-house software packages to model femurs. For proprietary reasons, this software is not generally available to the research community, forcing the BEC to develop its own in-house software package. As a fellow graduate student, Mr. Paul Smith, was already conducting research in this area, the development of such software was considered key if further models were to be generated and tested for inclusion with his research. As such, software development began almost immediately, based on automating the manual methodology that was being used. The literature search was undertaken concurrently with the software development and served to support the methodology being used.

Since the invention of CT technology in 1972, an abundance of papers have been written on

11

the use of CT imagery in various fields, ranging from reconstructive cranio-facial surgery to the microscopic structure of bone. Depending on the type of research, various types of CT scanners have been used, but similar processing techniques were generally used in all investigative studies. In reviewing the literature relating to this topic, the search was narrowed and the focus was placed on two key areas. This review limits the discussion to those papers which specifically described the use of CT data to model bone, and image processing techniques used to detect edges in biological specimens, the principal topics addressed by this thesis project.

2.2   Historical Background

Before the advent of computed tomography and FEM technologies, early reconstruction efforts usually involved the construction of wax models, or using artists' drawings to visualize the 3D effect. Sometimes, outlines of the structures would be traced onto transparent sheets and stacked to simulate a 3D object. Manually digitized points would often be taken directly from anatomic specimens, photographs, or video film using digitizing gantries or tablets. With advances in computer hardware and software, and the development of CT in 1972, the tedium of manually constructing such 3D models was lessened, and researchers began to develop new techniques to process true 3D digital images.

The research community had long recognized the need to automate the 3D modelling process [7,14,17,21,30,32,46]. By 1980, the rapidly expanding field of computer graphics enabled further research in image reconstruction techniques. Radiologists were particularly interested, as image processing algorithms offered the possibility of providing valuable information from multiple radiographic images, such as CT scans. The ready availability of CT data sparked numerous research projects to investigate its possible uses in modelling human bone.

2.3 Computer Techniques Used to Model Bone

Of the numerous papers reviewed, most used some form of contour data to represent bone structure. Several papers, however, proposed the use of generalized shapes, such as cylinders or ellipses, to describe femur geometry [12,50,54]. Instead of using actual contours of objects, more generalized shapes, such as cylinders, were used to describe structures. Dwyer et al. [12] described a paper by Soroka and Bajcsy in which they considered each planar contour to be an ellipse, and then structured the object as a series of elliptic cylinders. This generalized method was briefly considered for use in this project, but it was thought that such generalizations were unnecessary when actual contour points could be used to give a closer representation of the true contours. With specific edge definition possible, it seemed unnecessary to make such generalizations, particularly when detailed femur morphology was

13

required. Any model built using generalized contour outlines would not give the accuracy needed for comparison with actual physical tests. If a process were to be developed to fit general contours, such as a circle or ellipse, to a given bone contour, then that effort might just as well be applied to finding the true contour. Indeed, most researchers have agreed, and the focus has been placed on generating detailed and specific models.

The vast majority of the papers reviewed used some of the same techniques described in this project, however, the computing facilities and software packages used varied widely. Clearly, in the 1970's and 1980's, the personal computer (PC) had not developed to the stage where intense computational load was possible. Therefore, of the papers reviewed, most researchers utilized high power computer systems and specialized software packages to process the data. With the assistance of external research funding and support from computer hardware and software companies, many had access to state-of-the-art facilities. Some developed in-house software, and with the recent advances in PC technology, the personal computer finally became a feasible alternative [57]. As one of the objectives of this project was to use existing facilities in the BEC, lab PC's networked to the university's Centre for Computer Aided Engineering (CCAE) UNIX system were used. Software developed in-house incorporated modules from MATLAB® (The MathWorks, Inc., V.5.1.0.421) and ANSYS (Ansys, Inc., ©1996 SAS IP, Inc.,V5.3) software packages.

While most researchers were quick to detail the computing platforms used, and to give credit

to the developers of the image processing packages, they were generally reluctant to share any details of the algorithms used. Indeed, most of the literature described the data processing methodology in a single paragraph. However, using the proposed methodology of the BEC as a guideline, the following techniques were described.

A)   Transferring and Converting CT Data

Once the CT scan has been performed, the acquisition terminal must have processing capabilities, or as is most often the case, the data must be transferred to a separate processing system. This has proved to be a cumbersome task, as the usual method of transport involves copying the data to external media, such as 3½" diskettes. Several researchers [3,38,43,45,53] encountered the same difficulties, and reported having to use magnetic tape, or floppy disk [13,32,33,56]. Recognizing the number of large data files, Sutherland et al. [53] described having to convert the input data into a new data set which was dimensionally reduced. Extraneous data, of no use to the end user, was automatically removed from each of the image files using a thresholding technique, thereby saving storage space and quickening the processing. Müller and Rüegsegger [37] described a similar procedure in which volumes of interest (VOIs) were manually chosen by the user. Bandak et al. [4] chose to read the individual formatted files and combined the data into a single file in which a 3D array held all the data. In this way the data structure mirrored the actual object geometry. Of those papers reviewed, only Marom and Linden [31] reported having difficulty reading the

15

data and described having to convert the integer and floating point data into a format readable by the VAX/11 processing computer.

B)   Contour Detection

Contour detection, as used here, is a broad term encompassing several steps. It includes image enhancement, edge detection, contour closure, and boundary tracking. Image enhancement generally involves enhancing the contrast by normalizing a segment of the grey-level histogram. Pepino et al. [40] pointed out that the quality of CT images is "very high," and this preliminary contrast enhancement may not be necessary. To enhance the edges, they recommended the use of one of five different operators - Sobel, Robinson, Prewitt, Laplace, or Kirsch. These operators generated an image where the "contours are enhanced but not definitely detected." No comparison of these operators was made.

Before describing the contour detection techniques used, it should be noted that in most cases, *a priori* knowledge of the general shape and nature of a boundary or edge exists. Several authors [2,3,20] pointed out that such knowledge should be used to full advantage. Knowledge of femur morphology can be helpful in determining search patterns, termination criteria, and threshold levels, and can lead to more efficient edge detection.

Srihari [50] described the first step in the detection of edges as being that of isolating the

16

desired object from its background, which is more formally known as image segmentation. He described two different methods of segmentation, namely region-based and boundary-based. Using the region-based method, a thresholding technique was used whereby the pixel value was used to distinguish that pixel as being either object or background. The range of pixel values considered as being object, and the range considered background were predefined by the user, and a "decision function" compared each pixel with the range values and assigned a label accordingly. Srihari claimed that the thresholding technique was "effective when there is high contrast between object and background values and little clutter." Of the edge detection methods described, thresholding was by far the most common technique used by orthopaedic researchers [2,3,5,7,9,10,11,13,14,17,19,20,24,25,26,27,28, 30,33,34,39,41,44,45,46,47,50,51,52,53,54,55,56]. Rather than describe similar efforts by each of these groups, the literature review limits itself to innovative techniques and specific results and conclusions from these researchers.

Udupa [54] also described image segmentation and thresholding in his 1983 paper and noted that it was the usual approach. As an alternative to manually entering edge points, a border following algorithm was used to extract edges from each slice. He noted that this approach, while computationally easier than other methods, could not always reproduce many subtleties in the geometry of complex objects.

In 1996, Schwardt et al. [47] made passing reference to having segmented MRI images into

17

bone and background regions by applying local background correction followed by global thresholding. No further details were given.

Srihari [50] went on to explain that the alternative to thresholding, boundary-based segmentation, involved determining the edges by using directional derivatives to estimate gradients, or areas of abrupt gray level change, and then grouping these edges into boundary contours. This method is clearly more computationally intense, and no example was given to illustrate the difference in the two methods. In 1990, Englmeier et al. [13] described differentiating images by using a gradient filter, specifically the Roberts gradient, which produces a binary image of edges and background. Again, there was no discussion of results. Rubin et al. [42] mentioned using a contour detection program "using gradient and density techniques", and Müller et al. [36] claimed to have used a "sophisticated three-dimensional segmentation algorithm based on directional derivatives." Although details of these algorithms and their results were not given, Pepino et al. [40] gave an indication of possible problems when he stated "when density gradients are not great, manual or interactive edge detection may become necessary to avoid an excessive number of false positives and negatives."

Despite best efforts, the edges found using these gradient techniques are often incomplete, and have small gaps in the contours. Englmeier et al. [13] found that disturbances and contour irregularities sometimes existed, caused by blood vessel calcifications and contour

18

indentations. These contour irregularities must be addressed if a correct edge is to be found. The usual method has involved the postprocessing application of morphological operators, particularly erosion and dilation transformations. Erosion deletes pixels from an object's boundary, whereas dilation, does the opposite, and adds pixels to an object's boundary. Dilation followed by erosion can be used to close gaps in contours. Farman-Ara et al. [14] reported using a thresholding procedure, followed by a "series of 1 pixel dilations and erosions, until a closed edge was obtained." Similarly, Gao and Waite [17] also used "morphological thinning" to remove redundant contour points. The problem with applying generalized morphological operators to entire images is that the resulting edges are also stored in image format, in a raster or row-wise fashion. This resulting edge image must be searched to extract and link the edge points. Pepino et al. [40] mentioned that when the contour is found, "further calculations are needed to convert this frame, which has a 'raster' form, into a geometric or 'vector' form." This is often referred to as boundary tracking.

Other researchers, such as Artzy, Frieder and Herman [2], attempted to bypass this problem by avoiding generalized morphological operations. Instead, they used thresholding to select an initial seed pixel, and then used boundary tracking techniques to find the connected edge pixels. At whatever stage, boundary tracking, or some form of chaining, is necessary. One potential problem in edge traversal is that of infinite looping. In 1981, Artzy et al. used thresholding to define edge boundaries and addressed the problem of avoiding infinite looping when traversing edges. They used directed graphs to keep track of and minimize the

19

number of pixels previously visited when traversing the edges. By marking previously visited pixels, it was possible to avoid infinite looping during edge traversal.

Srihari [50] also discussed geometrical representations that gave details of shape. In particular, he described digital space curves as being connected sets of pixels where all but two pixels have exactly two neighbours in the set, and the two end pixels having just one neighbour. This method is also known as chain coding. He also explained that to isolate a single contour, one must track the connected pixels sequentially, forming a border by creating a circular list of points, where neighbouring points in the list represent adjacent points on the contour.

C)   Point Selection

Ideally, for best definition, all contour edge points should be used to define the femur surface, but computational restrictions prevent this. So, to approximate the femur edges on any CT slice, the data set must be reduced and a representative series of points selected which describe the full contour. The straight lines connecting these selected points should approximate the true contour. In determining how to select these points, researchers have taken different approaches. In 1977, Fuchs, Kedem and Uselton [16] described discretizing the edges by traversing each of the edges in a counterclockwise fashion, selecting a series of points on that curve. Like most of the papers reviewed, Fuchs did not specify the method

20

used to select these points. Several researchers, however, did outline the methodology. The most detailed discussion was presented by Kang et al. [25]. They outlined four methods of point selection.

a) One-to-One Matching - This method is suitable for simple geometries, and only if the number of pixels in contiguous slices is the same. In this project, these conditions are not met.

b) Direction Checking - Points which have changed direction are selected. This method usually produces different numbers of points for each slice, which does not lend itself towards easy FEM meshing.

c) Equiangular Selection - Equiangular lines are extended from the centroid of the image. Points are taken at the intersection of these lines with the edge. This method seems reasonable, but, for unclear reasons, was not used.

d) Segmentation by Equal Length - The total length of the edge is subdivided into equal length segments, and points are placed at the end of each segment. This was Kang et al.'s method of choice.

Using a completely different method, Sutherland, Bresina and Gayou [53] selected the points along an edge or contour by using a polygonization method wherein each contour was

21

described by "successively smaller polygons using an algorithm where points are selected by maximizing the distance between the contour and the perpendicular to the side of the preceding polygon." Hadiekker et al. [19] described a variation of the equiangular approach. Lines running outward from a central point intersected the cortex. The intersection points were determined using a wavelet-based edge detection algorithm.

From a finite element meshing standpoint, the alignment or registration of the points between slices is also important. This was confirmed by Adams et al. [1]. Improper alignment can lead to difficulties in splining through the points and cause rumples in the model. The registration between slices is "not obvious to the viewer [32]." Therefore, to ensure alignment, either naturally occurring landmarks are used, or, most often, artificial markers are introduced [29]. Another approach was to use a series of anatomically shaped alignment templates [48] to align the computer model after construction, but this required a best-fit judgement. Zachariah, Sanders, and Turkiyyah [58] stated that geometries from the bones were "aligned manually, with respect to each other, using display software written for this purpose (using Matlab)." Unfortunately, no further details were given.

Another potential problem in point selection arises when points are improperly placed, and do not lie on a cortical edge. This may result from improper thresholding. In such a case, Basu et al. [6] pointed out that "necessary corrective action" must be taken. Pepino et al. [40] recommended that visual verification be used, and if necessary, points should be manually

22

edited to "identify and separate any merged contours." To do this, they described a program which displayed the CT image and its contour on the screen. The operator then used a digitizer or mouse to draw new edge points. Vannier and Marsh [57] stated that the thresholding and point selection process should be "reiterated until a satisfactory simulation is achieved." The work of McNitt-Gray et al. [34] was innovative in that two editing tools were suggested. One was the "least upper bound (LUB) tool" which was used when the detected edge contour was not closed. The LUB tool searched through possible threshold values until it found the maximum threshold value which produced a closed contour. This was a useful tool that helped minimize human intervention. The second tool was a manual editor that allowed contour sections to be redrawn, similar to that previously described. None of the papers studied the effects of manual editing on the final model.

D)   Generation of Model

With representative points selected to describe the geometry of the femur, a 3D model must be constructed. The literature review shows that two distinct methods have been used - one using off-the-shelf finite element modelling packages such as ANSYS, and the other using custom-developed surface tiling software, as discussed by Dwyer et al. [12]. The ANSYS package was already being used by the BEC to generate models, so there was no need to write custom software to accomplish the same tasks, and this review concentrates on the methods using finite element modelling packages.

FEM software came into use as an orthopaedic modelling tool in the late 1980's, when advances in computing power allowed researchers to build detailed, complex structures. The approach used has been to define a series of keypoints describing the geometry, create a series of splines, or ribs, through these points, skin over the ribs to create areas, and finally, generate a solid volume representation. Bae et al. [3] recognized that a single spline function would not be able to generate a smooth model when the surface fluctuations of the femur were substantial. They concluded, along with others [9,35,40,58], that "B-spline functions (i.e. a composite spline) ... will render a smoother boundary." Gutkowski, Raftopoulos and Williams [18] used a similar tactic and generated a B-spline surface over a series of horizontal cubic splines. The splining operation was followed by a skinning operation and then solid model generation.

All points, lines, and surfaces are numerically labelled by ANSYS as they are created, and these labels must be used for subsequent commands. The interactive entry of these commands can be tedious and prone to error if labels are used improperly. This problem was partially addressed by Basu et al. [6] and Marom and Linden [31] who constructed an FEM input file directly from the extracted CT data.

E)   Validation

A number of techniques have been used to verify the accuracy of the detected edges. The obvious method is to conduct a visual comparison and this method was used by many of the researchers. An early visual comparison can point out obvious flaws, allowing further refinements to the model before continuing with the validation.

More quantitative methods have involved taking a series of linear measurements from both the actual bone or CT image and the corresponding model. Where computer methods were not available, these measurements were taken by using paper or acetate tracings [1,49,56]. The different modalities used to produce the various images make computer evaluation and comparison difficult, but it has been done. Farman-Ara et al. [14] mentioned using a "shape-independent surface-matching algorithm" to align the images prior to measurement. Rubin et al. [42] measured the radii of the bone and the model, and calculated an accuracy index as being the difference between the two.

Others have compared the area of the cortical bone with the area enclosed by the model contour. Farman-Ara et al. [14] had physicians draw reference contours for comparison with the detected edges. To evaluate the accuracy of the fit, they calculated the proportion of pixels exterior to the image overlap. Proportion values close to zero indicated a good fit. Unfortunately, the complexity of the shapes makes proportion calculation and area

25

determination difficult. Where computer methods were not available, the method used by Adams et al. [1] offered a solution. They cut out paper tracings of the areas, weighed them, and converted these weights into areas by weighing a known area of the same paper.

While not directly modelling femurs, Smith et al. [49] studied methods of measuring the cortical thickness of a human cadaver femur. Their interesting findings are noteworthy in a validation study in that they indicated that CT scans overestimate the thicknesses by 6%. This is due largely to resolution limits, which are dependent on several factors, such as X-ray beam width, reconstruction algorithm, and display pixel size. While recognizing this as a drawback, they concluded, after comparing several methods, that the ease of taking the measurements from CT scans makes it better suited for research purposes than other, more accurate, but tedious methods.

2.4   Conclusions

A variety of methods have been used to model bones, and most researchers in this field have had access to facilities using state of the art CT processing terminals, dedicated mini-computers, and specialized image processing and modelling software. The use of personal computers in this area has been very limited, but is becoming more widespread with advances in PC technology.

26

It seems limited testing has been done with respect to newly developed software, and comparisons between techniques, and validation through physical testing, if performed, have not been well documented in the literature. As well, algorithmic details are scarce, forcing researchers new to this field to make certain assumptions, and develop their own methodology in a piecemeal fashion.

CHAPTER 3

DEVELOPMENT OF METHODS FOR DATA PROCESSING AND MODEL BUILDING

Based on the literature review and discussions with researchers in the Biomedical Engineering Centre, the problem of developing automated modelling software was broken into a series of seven distinct processing steps.

- Read and translate the CT data files

- Extract regions of interest

- Determine threshold levels

- Find contours defining the edges

- Select representative edge points

- Verify correct point placement

- Generate an FEM input file to build the model

28

3.1   Acquisition and Translation of CT Data Files

3.1.1 Acquisition of CT Data

In the context of various research projects, Memorial University's Medical School periodically provides the Biomedical Engineering Centre with femurs obtained from donated cadavers. In preparation for scanning, the femur specimens are cleaned to remove as much soft tissue as possible, and are wrapped in formalin-soaked cheesecloth and stored in environmentally controlled, refrigerated facilities until testing can be performed.

Before a CT scan is performed, a mounting frame is made to hold the femur in the required orientation on the CT scanning couch, with the CT scan axis parallel to the femur's natural load axis. The frame holds the bone on each end and suspends it in air. The subsequent physical load tests to be performed on the femur are conducted such that the loading is oriented along the femur's natural load axis, which runs from the center of the head to the approximate midpoint between the condyles. This load axis is illustrated in Figure 3.1.

29
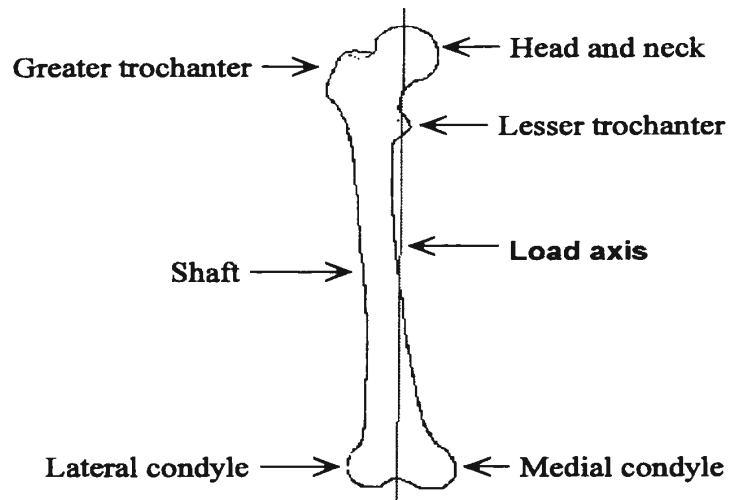
Figure 3.1   Femur morphology and load axis

Prior to physically testing the femur in a hydraulic test frame, Cerrobend™, a lead-bismuth alloy, is used to mold platens to fit on each end of the femur. These platens hold the bone in proper alignment in the test frame and ensure all loading is along the natural load axis.

Figure 3.3
Femur mounted between platens in hydraulic test frame

Figure 3.2   Molded platens

30

Since the computer models will be used to simulate physical tests, it was decided that the platens used in physical testing would be made prior to the CT scan, and be used to correctly position the femur on the CT scanning couch.

The platens held the femur such that when scanned, the scan lines ran perpendicular to the load axis of the femur. The molded platens were positioned on either end of the femur, and the femur placed in a supine position on the scanning couch, with the distal end, or condyles, toward the detector. As the platens were not attached to the femur, weights were placed against the outside edge of the platens to anchor the femur firmly between them. This was necessary to prevent any slippage as the CT couch travelled through the scanner.

Before starting a scan, the technician used a laser guide to position the starting slice. With the laser guide on, the edge of the platens were aligned parallel to the proposed scan lines, and therefore, perpendicular to the load axis of the femur. When viewed on the CT acquisition console, the femur was roughly centered in the field of view. A standard slice thickness of 5 millimeters was used. The starting slice was taken at the
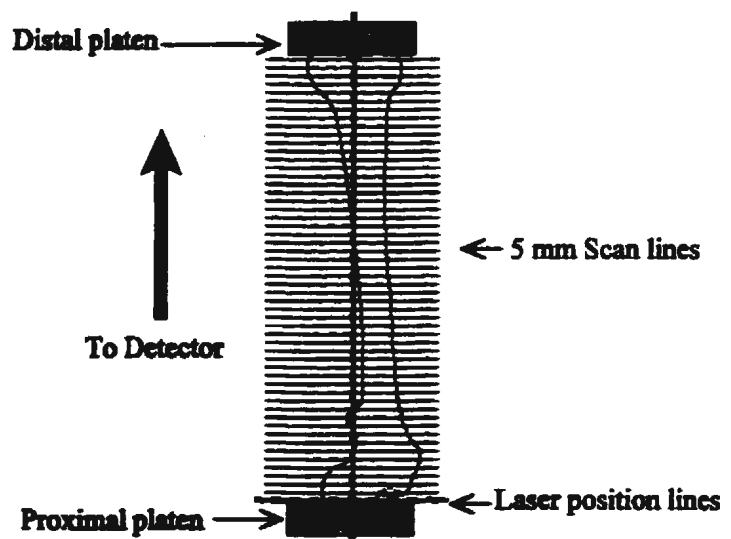
Figure 3.4   Alignment of femur on CT couch

31

point of the femur protruding just outside the platen. A starting point on the platen would give erroneous data because its metallic content would create severe artifacts in the CT imagery.

At present, the local hospitals perform scans using Toshiba Xpress/SX scanners with typical setup parameters as shown in Table 3.1.

Table 3.1 Typical Toshiba Xpress/SX CT scanner settings	
S/W Version : V4.5.0	Zoom: None
Scan Type: Normal CT	Image Size: 512x512 pixels
kV: 120 kV	Position: Supine/Head first
mA: 200mA or 400mA	View: From foot
Scan Time: 2.0s	Couch Slide: -45.0mm
Slice Type: Small field/ 5mm slice	Couch Height: +130mm
Filter Code: FC80 or FC11	Gantry Tilt: +0.0D
Image Filter: Original	

The local hospitals currently have limited data transfer capabilities, and after the scan was performed, all data had to be transferred to 3½" diskettes for transport back to the university. Each CT image is 512 x 512 pixels of 16-bit data, so each slice of the femur requires 500 KB of storage space. At 5 mm slice intervals, an average femur of 45 cm in length, yields 90 slices or 45 MB of data. After performing the scan, these 90 slices of raw, unprocessed data

were transferred to 45 diskettes by hospital CT technicians, and then transferred to computers in the Biomedical Engineering Centre, as described in Appendix A.

3.1.2 Translation of CT Data

At this stage in the preprocessing, a central data file was created which controlled the processing of the CT data. As the same data procedures were to be used on a large number of slices, this file prevented the user from having to repeatedly run the same program for a series of CT slices. The format of this control file is shown in Appendix B. This file contained all relevant processing information, such as scan name, slices to be processed, number of points to select, and file storage locations. By updating the information in this one file, different scans and slices could be processed. The file controlled the process looping for each of the software modules developed for this project.

Preliminary attempts at reading the CT data soon revealed a problem. The CT numbers retrieved from the BEC data files did not match those recorded by the hospital's CT scanner. Further investigation showed that the raw, 16-bit, signed CT data provided by the hospital had been stored in a byte-swapped format, with the most significant bytes being stored in the lower addresses. The processing computer, however, expected the least significant bytes to be in the lower addresses, causing the CT numbers to be interpreted incorrectly. As an example:

Toshiba 16-bit CT scanner recorded: $\qquad -1018_{10} = 1111\ 0011\ 1111\ 1010_2$

Processing computer read: $\qquad 1788_{10} = 0000\ 0110\ 1111\ 1100_2$

After studying the data representations, the byte-swapping pattern became apparent, as did the representation of negative numbers in two's complement format. A program was written to convert the CT data into a readable, decoded format matching that recorded by the CT scanner, and proper conversion was verified by comparing CT numbers displayed on the scanner acquisition terminal with those generated by the conversion software.

3.2 Extraction of Regions of Interest

With the CT data translated into a compatible and readable format, attempts to process the data soon showed that the image files were too large and cumbersome to work with. Computer memory quota restrictions, and processing and time constraints made it difficult to process all the data from a CT scan at one time. All slice data would have to have been compressed, and then selectively uncompressed, processed, and recompressed to allow further processing of other slices. As one of the goals of this project was to eliminate tedious and time-consuming procedures, another method was implemented. Instead of processing a small number of large data files, it was decided better to process a large group of small data

34

files.

The original CT images showed the full field of view, which was primarily air. Since this extraneous data was not required by this study, regions of interest (ROIs), or subimages containing the region of bone, were extracted, excluding all unnecessary image data.
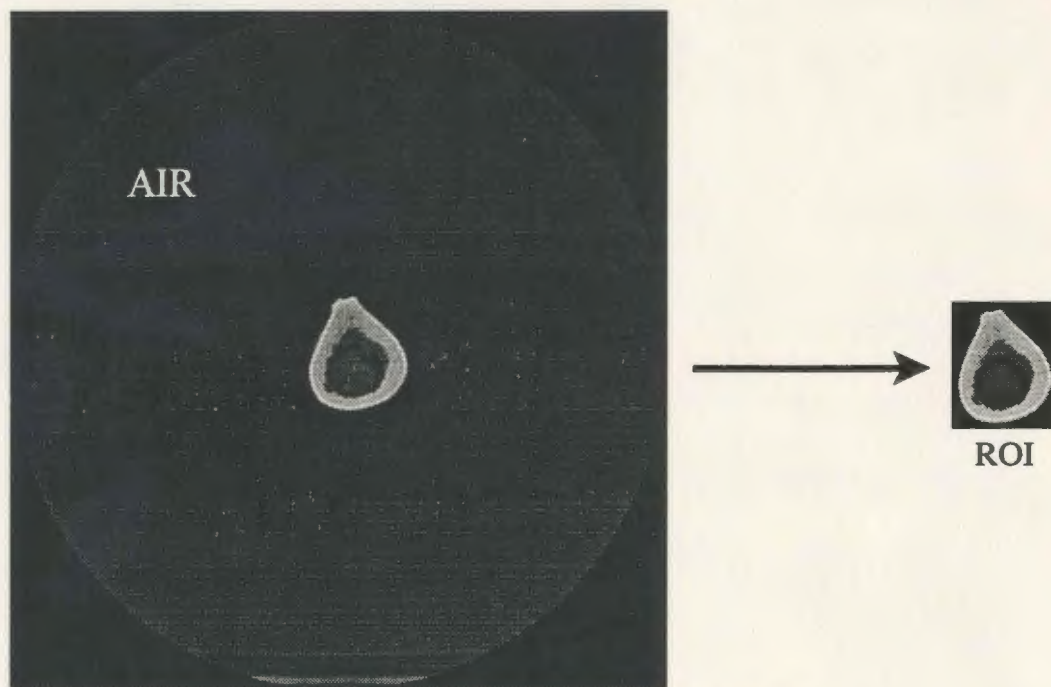
Figure 3.5   Full field of view and extracted ROI

Using basic knowledge of the CT values of bone and air, and using *a priori* knowledge of the central placement of the femur in the field of view, a quick scan of the data cropped the image. The CT value of air is approximately -1000, while that of cortical bone is >800. By reading each pixel along each row of the image, the X and Y coordinates of any pixel with

35

a CT value corresponding to possible bone were noted, and the extreme boundaries, marking the upper, lower, left and right edges of the possible bone region, were marked. To ensure that any borderline pixels on the periphery of this block, or region-of-interest, were also included, the region was padded by including extra pixels on each side. A pad size of five pixels was selected as being appropriate. All data within this region-of-interest was then written to a new, smaller data file. This step in the processing reduced the size of the data files by approximately 88%, and enabled faster processing.

3.3   Determination of Threshold Levels

The first step in developing automated point placement, was finding the edges along which to place the points. While the exact geometry required can vary from project to project, the research being conducted in the BEC required that two edges be defined - the outer edge of the entire femur, and the inner edge of the cortical bone surrounding the marrow cavity running through the shaft. These edges are illustrated in Figure 3.6.
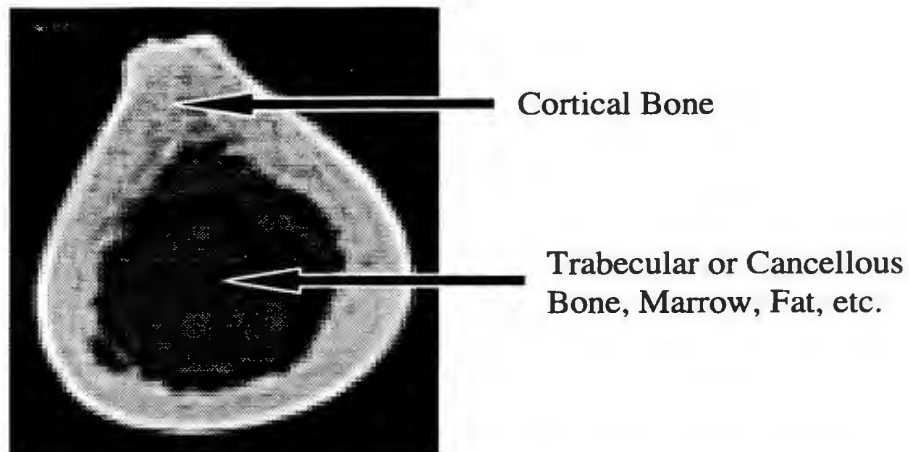
Figure 3.6   CT slice through shaft showing bone classification

Standard edge detection algorithms are readily available, and MUN's Centre for Computer Aided Engineering offers an image processing toolbox with the MATLAB mathematical analysis software package. The Image Processing Toolbox (Version 2.0, The MathWorks, Inc.) includes such standard edge detection routines as Sobel, Prewitt and Roberts. It was hoped that one of these off-the-shelf algorithms would be able to provide clear edge data. Unfortunately, testing showed each method to be ineffective in defining the inside edge of the cortical bone. Figures 3.7, 3.8, and 3.9 show the results for a CT slice from the shaft of the femur, where the cortical edge is best defined.
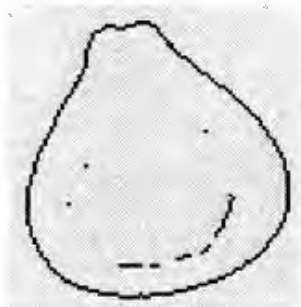
Figure 3.7   Sobel
edge detection
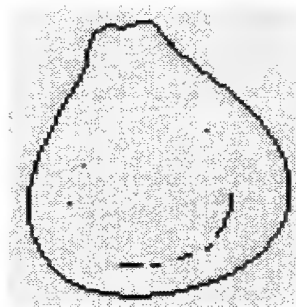
Figure 3.8   Prewitt
edge detection
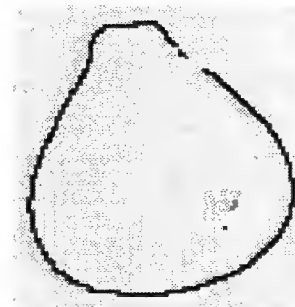
Figure 3.9   Roberts
edge detection

37

Clearly, these commonly used edge detection algorithms were able to successfully detect the exterior air/bone boundary, but had difficulty differentiating the inner edge between cortical and trabecular bone. This transition is often very subtle, and is difficult for even the human eye to clearly discern. This is shown in the linear profile of Figure 3.10, and the photograph of Figure 3.11, which shows cross sections of the femur, progressing upwards from the mid-shaft to the neck. The relatively sharp inner edge of the mid-shaft becomes less obvious as the honeycombed trabecular content increases towards the neck.
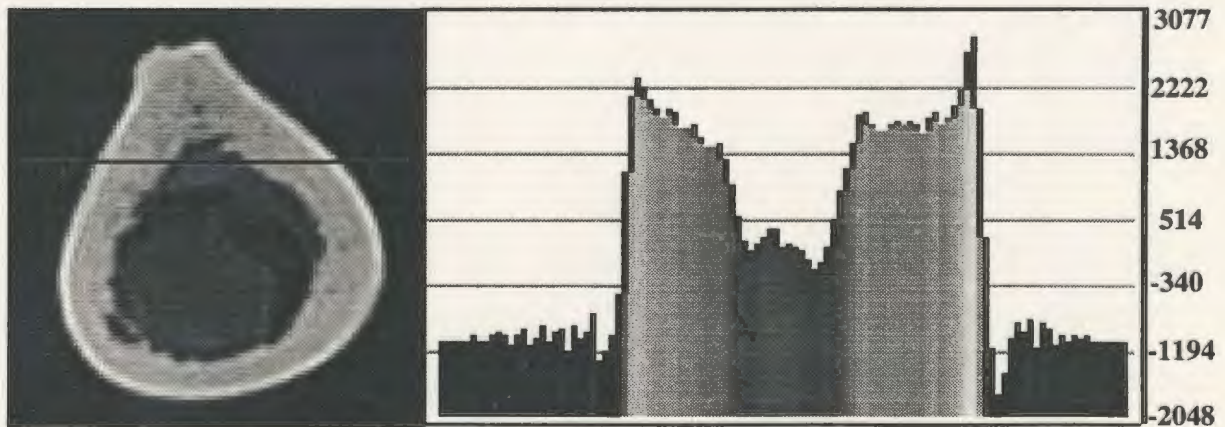
Figure 3.10   Linear profile through femur shaft - The sharp outer edge transition and more gradual inner edge transition are shown
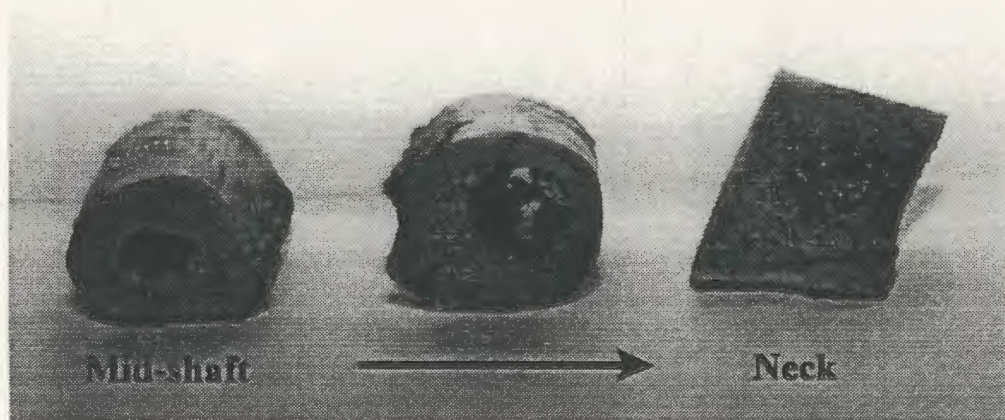
Figure 3.11   Femur cross sections

Many edge detection algorithms look for sharp gradients, which aren't always present in bone, making these edge detection techniques less than ideal. Most researchers have found that satisfactory edges can be found by using a thresholding technique, wherein users define a range of CT values which may be considered cortical bone, similar to the procedure used to extract the regions of interest. Using these limits, pixels can be marked as being either bone or non-bone. The problem, therefore, becomes how to select the appropriate threshold values.

Using readily available image processing software, selected CT slices can be viewed, color masks applied, and a range of CT values for the cortical bone can be approximated. Figures 3.12 and 3.13 illustrate the threshold selection procedure. The images were generated using OSIRIS Medical Imaging Software V.3.12, a package from The University Hospital of Geneva, available from the internet. The original image was opened using the OSIRIS software package, and a processing tool known as a color mask was selected. This tool allows the user to highlight a range of CT values, and to easily adjust the range by sliding the threshold level indicator until the desired pixel range is selected. While just one range was needed to define the high and low values of cortical bone, Kuhn et al. [27] and Peyrin et al. [41] found that two thresholds provided greater flexibility in fine-tuning the boundaries. A single threshold value can produce a sharp outer edge, but often results in a ragged, poorly defined inner edge. Therefore, two thresholds were used - one threshold value defined the

outer edge between air and cortical bone, and the other defined the inner edge between cortical and trabecular bone. The outer edge was thought of as being the dividing line between the surrounding air and the compact cortical layer. Therefore, the threshold was selected by manually adjusting the threshold level indicator until all pixels which were not considered to be marrow, fat, water, or air were highlighted. The low pixel value of the selected range became the outer edge threshold value, below which everything was considered to be marrow, fat, water, or air. Although subjective, this method produced a sharp outer edge. Figure 3.12 illustrates the selection of the outer edge threshold.
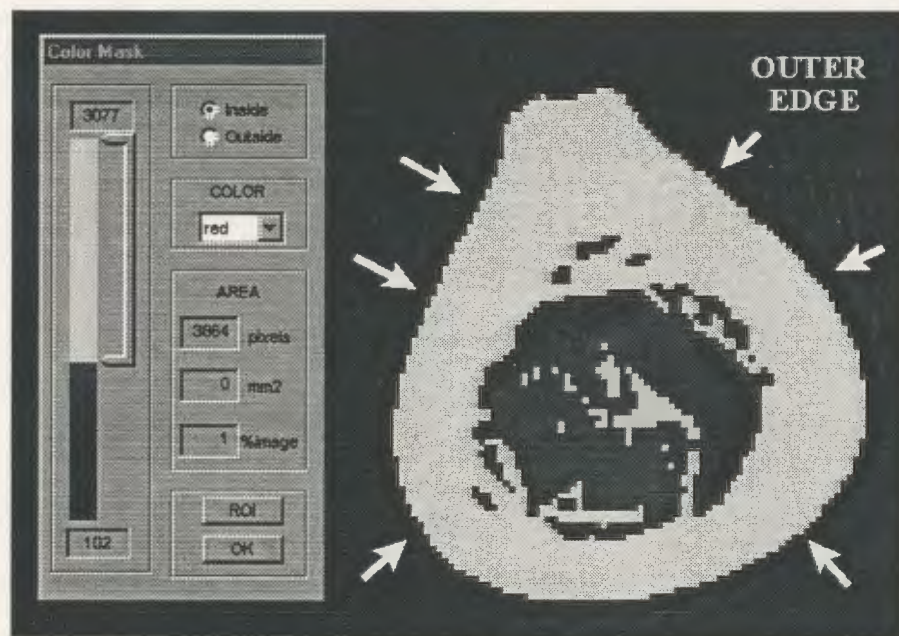
Figure 3.12 Selection of outer edge threshold

Pixel values < 102 were considered to represent marrow, fat, water, or air.

Similarly, the inner cortical edge, outlining the marrow cavity, was thought of as being the dividing line between the compact cortical layer and the spongy cancellous bone or marrow.

40

Therefore, the inner edge threshold was selected by adjusting the threshold level indicator until all pixels which were considered to be cortical bone, and produced a smooth, closed inner edge were highlighted. The low pixel value of the selected range became the inner edge threshold value, above which everything was considered to be cortical bone. Figure 3.13 illustrates the selection of the inner edge threshold.
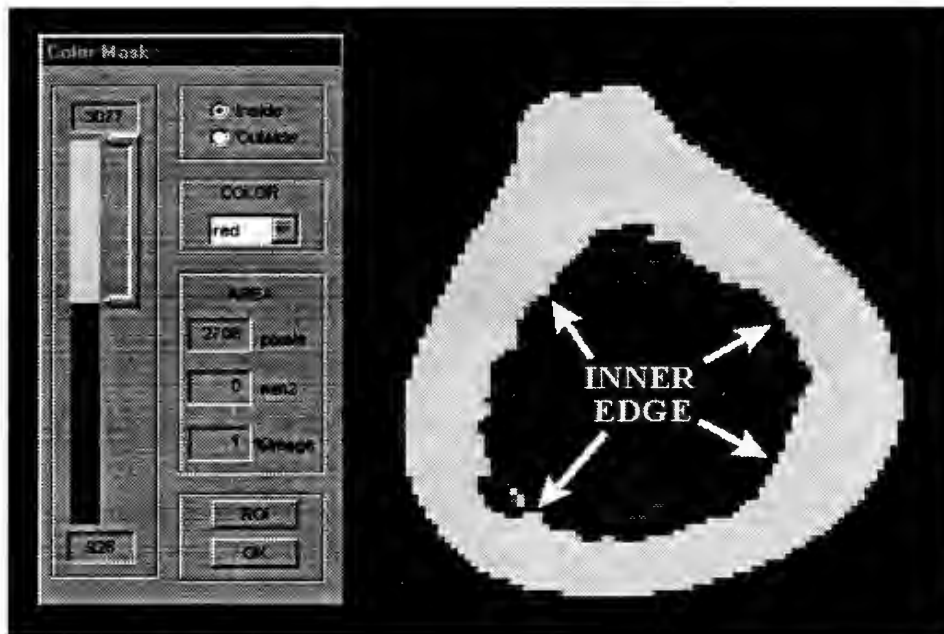
Figure 3.13   Selection of inner edge threshold

Pixel values > 926 were considered to represent cortical bone

There may be slight variation in these threshold values with any given bone, so several slices were viewed, and, if possible, threshold values were averaged to give a general value that could be used for the entire model. Testing showed these generalized threshold values worked well in most cases, but for certain slices, the threshold values were not suitable, and individual slice thresholds had to be set. This became evident only when the detected edges

41

were viewed. In addition to this, depending on the length of the model, two separate sets of threshold values were sometimes required – one set for shaft slices containing a marrow cavity, and a second set for those slices nearer the ends of the femur, where the marrow cavity is replace by trabecular bone and the range of CT numbers changes.

3.4   Detection of Edges

With the threshold values selected, MATLAB's Image Processing Toolbox was used to find and mark the edges. The procedure used, and illustrated in Figure 3.14, was as follows:

- An image mask was created which set all pixels corresponding to air to 0 (black), eliminating them from further morphological operations.

- The images were then converted to intensity images, by scaling all pixel values in the indexed image to values between 0 and 1.

- The outer edge of the cortical bone was found by applying density slicing using the first preselected threshold value.

- Small gaps in the edges were filled using a series of morphological operations.

- MATLAB's *bwperim* command was used to mark the edges, as shown in Figure 3.14(c). As shown, an inner edge was also found, but at this threshold level, it did not correspond to the edge required. Before saving this outer edge, the false inner edge

42

was removed by scanning each row in the image and deleting these interior points. The resulting clean outer edge is shown in Figure 3.14(d).
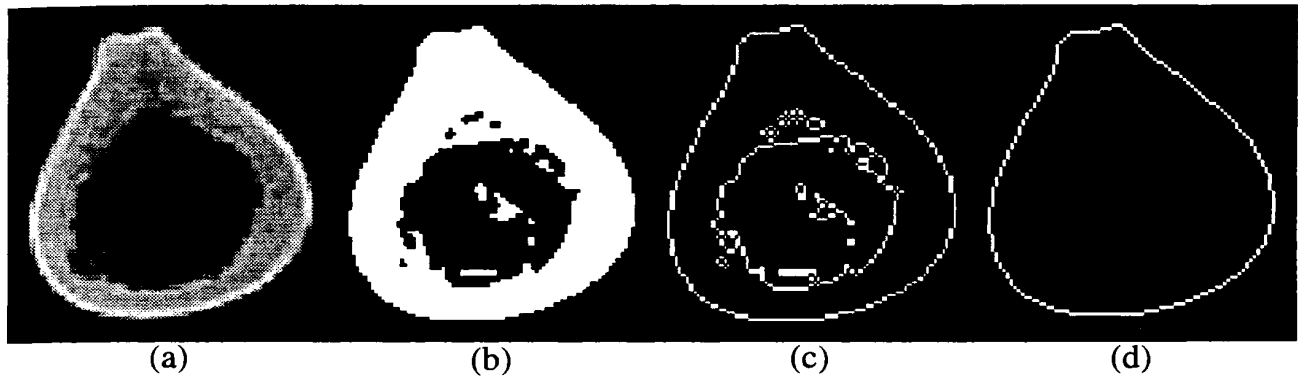
Figure 3.14   Steps in detection of outer edge

- Using the second preselected threshold value, a similar process was used to find the correct inner edge, as shown in Figure 3.15. The image was scanned and points exterior to the inner edge were removed.
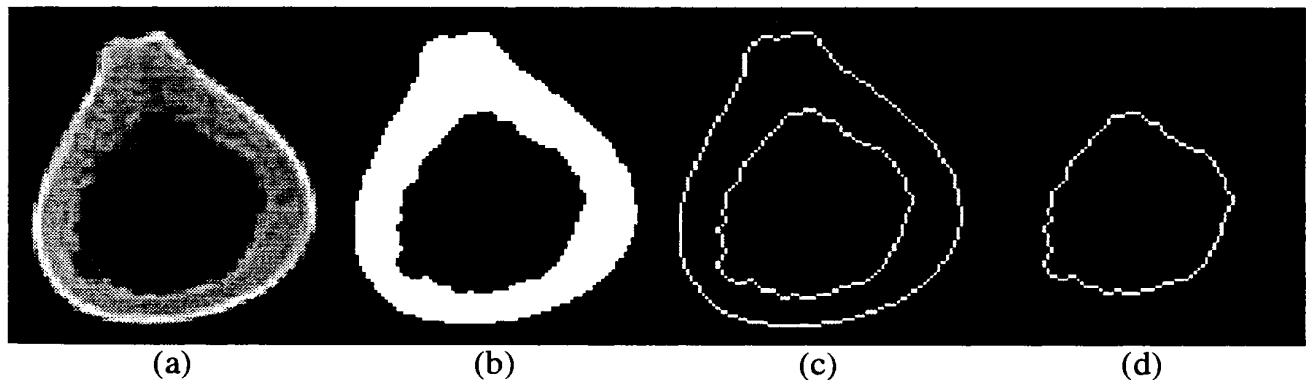
Figure 3.15   Steps in detection of inner edge

- These inner and outer edge perimeters were written to two separate data files which were used in further processing.

Two key assumptions were made when detecting the edge of the femoral cortical bone, the first being that the problem of bone bifurcation would not be addressed at this stage in the project. Depending on the alignment of the CT slices, it is possible for two distinct and separate regions of bone to be visible in one or more of the CT images. This occurs when a scan line passes

a)   through the head and neck of the femur, through a small pocket of air, and then through the top of the greater trochanter, or

b)   between the lateral and medial condyles.

These situations are illustrated in Figure 3.16
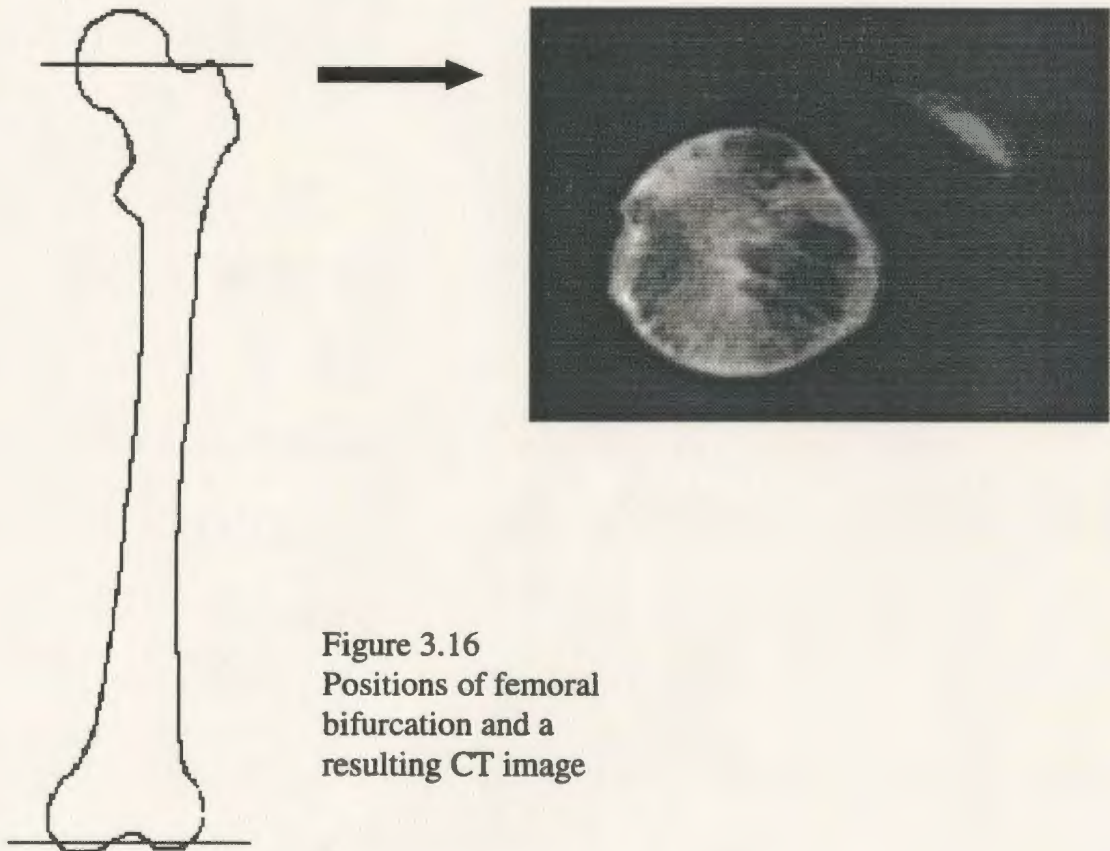
Figure 3.16
Positions of femoral
bifurcation and a
resulting CT image

It was decided that since the tip of the greater trochanter does not play a significant role in the loading of the femur and because the bifurcation is not easily meshed in the finite element model, the small tip of bone would be ignored for modelling purposes at this time. The bifurcation between the condyles was not a factor as the Cerrobend™ platen prevented scanning in that region. Other researchers [3,13,25] had difficulty with bifurcation issues as well, and their techniques may suggest possible solutions if this modelling restriction is addressed in future versions of the software.

The second, and related, assumption was that if multiple regions of bone were detected in a single image, the largest region would be considered the primary structure and would be used for model construction.

3.5   Selection of Edge Points

When building the model manually, the only criteria used to select edge points was that they be approximately equispaced around the perimeter, and that point alignment between slices be maintained as much as possible. This was done by viewing the previous slice, with its selected points, while manually placing the points for the current slice. The user used a series of visual checks and best judgement to approximate equal spacing between points, and point alignment between slices. With reduced reliance on user judgement, it was suspected that

45

point alignment between slices would improve with the use of an automated method of point placement.

When automating the method, several methods of point placement were considered, including equidistant spacing around the perimeter, intersection of equiangular lines with the perimeter, and increased point placement in areas with greater linear gradients. The use of equiangular lines most closely approximated the manual method being used, but this technique falls short in that it uses the same number of points to mark a straight line as it would to mark a rapidly changing curve. Using this method, proper definition of the edges is dependent on the number of equiangular lines used. Similar to the gradient method, equidistant point placement, by nature, places more points in areas of rapid change, simply because the length of the contour increases in these areas. Placement of equidistant points was the easier of the two methods to implement, and for this reason, it was decided that this method would be the first tested. If the results proved unsatisfactory, another technique would be used.

To improve point alignment between slices, point selection would begin from the same location on each slice, the vertical midpoint along the left side. In most cases, the shape of the bone is very similar in adjacent slices, therefore the midpoint does not shift significantly from one slice to the next. With this midpoint as an anchor, it was thought that all other selected points would also produce minimal shifts between slices, and lead to better overall

46

point alignment. Before selecting these points, however, further processing was necessary.

The edge perimeters found by MATLAB were stored in two separate data files, but because they were taken from image files, they were stored in a raster or row-wise fashion. As the edges were stored, each row was scanned, and the points were written in the order found. The edge points from row one were stored before the edge points in row two, and so on. This meant adjacent points along an edge were not necessarily stored in adjacent positions in the data file. Figure 3.17 shows the order in which MATLAB stored the edge contour points.
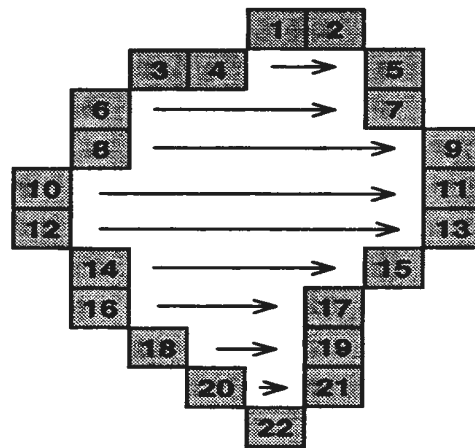
Figure 3.17
MATLAB edge storage - Points
were stored in numerical order

While all the points were recorded, the visual connection between them was lost. This connection was important in facilitating the selection of equidistant edge points.

If the edge points in the data files could be ordered to reflect the connectivity between them, then equidistant points could be easily found by selecting equally spaced points from the data files. Therefore, to facilitate point selection, the points in the edge files were sorted or re-ordered to indicate their connectivity. They were stored such that any two adjacent points identified pixels which were connected either horizontally, vertically, or diagonally. The increment in the X or Y coordinate between any adjacent points was at most one, as shown in Figure 3.18.
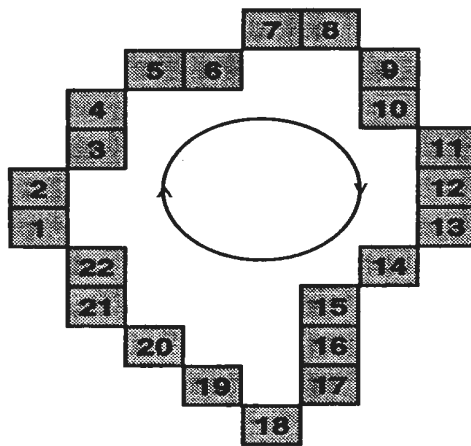
Figure 3.18
Desired edge storage

To accomplish this, a tracking algorithm was developed which conducted a search in a circular pattern and linked points around the perimeter, as they would be if one were to visually trace around the inner and outer perimeters.

The initial algorithm was similar to that used in chain coding where a boundary is

48

represented by a connected sequence of straight line segments, as described by Freeman [15]. The starting point was selected as being the midpoint along the left edge. Due to the circular nature of the bone edge, it was initially assumed that the cortical edge would trace upwards and to the right. Based on this *a priori* knowledge, the list of edge points was searched to see if the point immediately above the starting point was in the list, if it was not, the list was searched to see if the point to the immediate upper right was on the edge. This procedure was followed searching in each of the eight possible directions until a connected edge point was found. The search pattern is shown in Figure 3.19. Initially, the tracking software followed the same search pattern for each pixel, searching the eight adjoining pixels in the same order for the next connected point. The proper connections between pixels were established, but the program was very inefficient, taking 20 minutes to process each CT slice. Further refinements to the algorithm were sought.

3 X-1,Y-1	1 X,Y-1	2 X+1,Y-1
5 X-1,Y	C	4 X+1,Y
7 X-1,Y+1	8 X,Y+1	6 X+1,Y+1
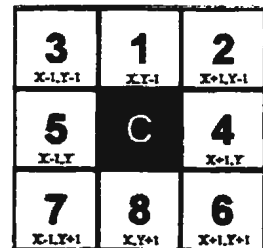
Figure 3.19
Initial boundary tracking search pattern - The search order from current pixel C is shown numerically

Knowing that the edges are basically circular in nature, connected points tend to follow from the same direction. Therefore, direction patterns were used to streamline the search. The pattern between the current end pixel (C), and the previously linked pixel (P) was analyzed, and one of eight search patterns was implemented, searching in up to seven directions. The order in which the search was conducted is indicated numerically in patterns A through H of Figure 3.20.

49
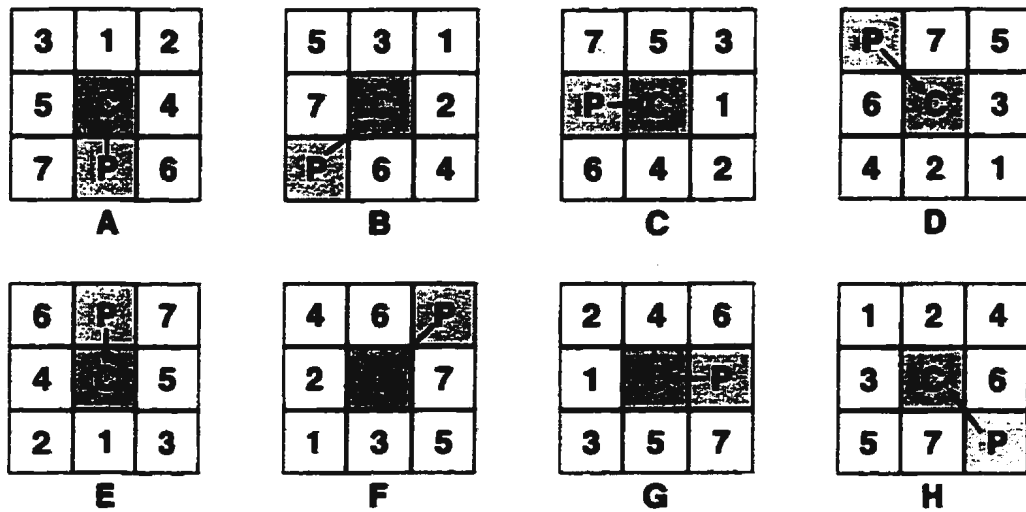
Figure 3.20 Refined boundary tracking search patterns

This improved method increased efficiency and reduced the search time dramatically from 20 minutes to approximately 2 seconds per slice.

The search was continued until the contour was closed, or until an infinite loop was detected. Although rare, an infinite loop could occur if a section of the cortical bone was thin enough to cause the contour lines to touch and form a loop. If this occurred, a small adjustment was made to the threshold value for that edge, and a new contour was found. This problem should be addressed in future versions of the software by implementing a backtracking scheme, similar to that used by Herman and Liu [20], or a threshold selection tool, similar to that proposed by McNitt-Gray et al. [34]. This would eliminate the need for manual intervention and threshold adjustment.

50

With the edge points re-ordered, equidistant points were selected by extracting every N^{th}

point from the data file, where

$$N = \frac{\text{Total number of points in the contour}}{\text{Number of points to be selected}} \qquad (3.1)$$

The user was able to specify the number of points to be selected on each of the edges. An

example of a ROI with automatic point placement is shown in Figure 3.22.

233 281
233 280
233 279
233 278
233 277
232 276
232 275
232 274
232 273
232 272
232 271
232 270
232 269
233 268
233 267
233 266
233 265
233 264
233 263
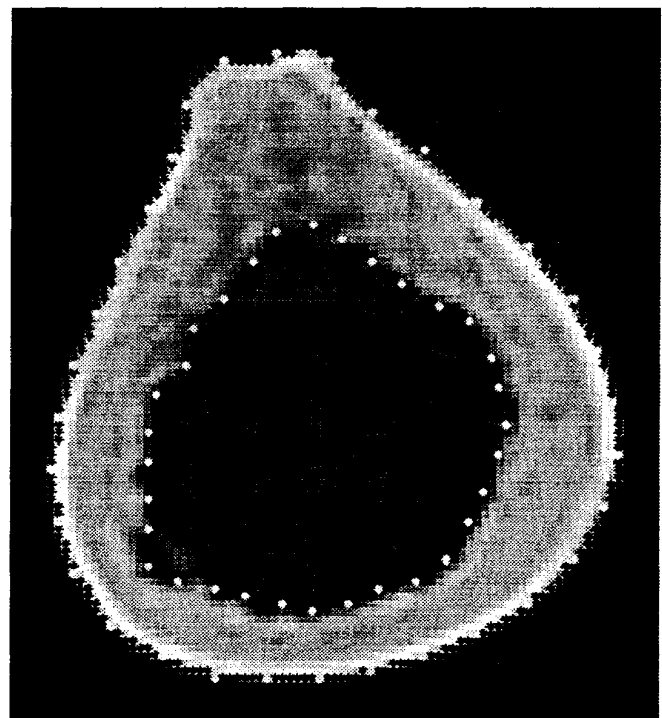
Figure 3.21
Partial data file showing selected,
equispaced pixel coordinates

Figure 3.22
ROI with edges and 64 selected points

51

3.6 Verification of Point Placement

After the edge points for each CT slice were automatically selected, the points were viewed and edited as needed. Occasionally, objects other than bone were retained in the field of view, for example, metallic strain gauges, plexiglass support structures or bed linens. Depending on the material properties of these objects, the CT numbers generated were sometimes similar to that of bone and interfered with the threshold techniques used in edge detection. This sometimes resulted in a false edge being generated. Figure 3.23 shows a CT slice of a femur with three strain gauges attached. Using the thresholding technique, the eighteen solder points were wrongly interpreted as cortical bone.
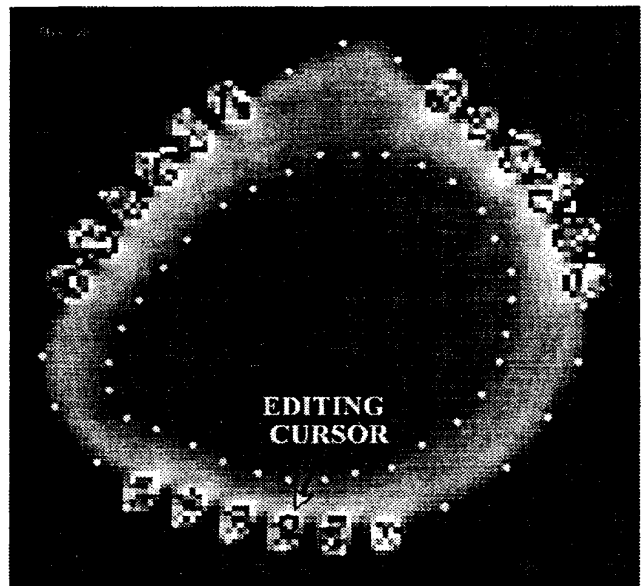
Figure 3.23
CT image of femur with attached strain gauges

The software package used to manually place the points did not allow individual point correction. If one point was incorrectly placed, all points for that particular slice had to be reentered. As this was seen as a major problem, a graphical point editor was developed in-house using MATLAB. Using this editor, a circular editing cursor was positioned over the

point to be moved, and the mouse was used to click and drag the point to its new location. As the mouse button was released, the edge point was redrawn, reflecting the true edge location, and the new coordinates were automatically updated in the edge file.

After each slice was viewed and any manual point adjustments made, the data points were ready for input to the finite element modelling package.

3.7 Generation of ANSYS Input File and Construction of Preliminary Model

Once all edge points were satisfactorily defined, the edge point coordinates were passed to the ANSYS finite element modelling package where a solid model was generated. Each of the edge points was input as being a keypoint. Splines were then generated through these keypoints, creating a series of 'ribs'. Areas were created by skinning over theses ribs, and a solid volume model was defined from these areas. These model entities are illustrated in Figures 3.24 to 3.27.

53
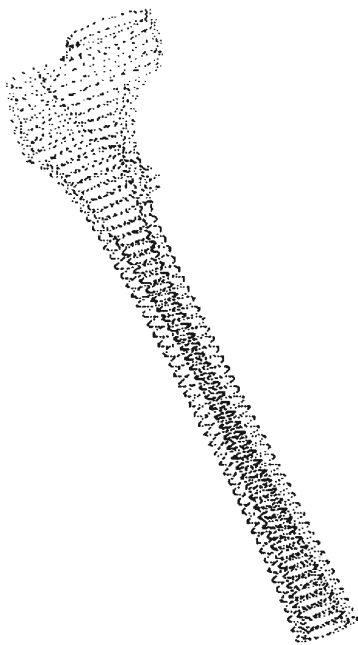
Figure 3.24
ANSYS display of generated
keypoints
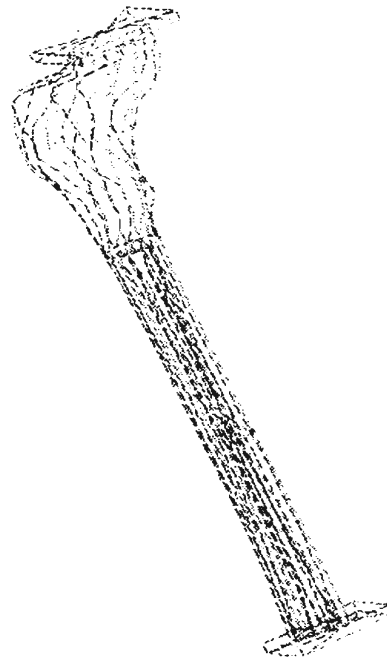
Figure 3.25
ANSYS display of generated
lines (with platens)

Figure 3.26
ANSYS display of generated
areas

Figure 3.27
ANSYS display of generated
volume

In ANSYS, each keypoint, line, area and volume is considered an entity and is automatically assigned a number as it is generated. When the model was generated manually, the user had to keep track of all entity numbers assigned by the program, and enter commands specifying which of them to process. Again, this was a tedious and time-consuming procedure, prone to human error, and required the operator to be familiar with ANSYS commands. This suggested an automated solution would be beneficial.

Knowing the number of CT slices, the number of edge points and the order in which they were selected and stored, an ANSYS input file could be automatically generated containing the commands needed to build a 3D volume representation of the femur. ANSYS commands could be generated to define the femur using keypoints, lines, areas and volumes without the user having to keep track of the various entity numbers and manually enter commands. A FORTRAN program was written which read the selected edge point coordinates from the data file, and converted them into keypoint input format. The program then generated ANSYS commands to assign keypoints, spline through inner and outer edge points, skin over these lines, and generate a solid volume model, as shown in Figures 3.24 through 3.27. To build the model, the user simply had to run ANSYS, read the input file, and watch the graphics window as the model was generated. If the 3D model was not satisfactory, the process became iterative, with corrections being made until a suitable model was obtained. At this stage, the platens used in the physical tests were also modelled on the computer and positioned on each end of the femur model, as shown in Figures 3.25 through 3.27. The

model was then available for meshing, interactive testing and analysis using ANSYS.

CHAPTER 4

EVALUATION OF NEW METHODOLOGY

In evaluating the new methodology, there were three primary goals:

1) Validate the algorithm and ensure that models are generated as expected.

2) Conduct preliminary user acceptance testing of the new software and obtain feedback.

3) Establish qualitative and quantitative methods of comparison in order to assess generated models.

It was expected that the software would be able to construct femur models as anticipated, and that users would find the new method quicker and the software easier to use. While testing methods would be established, it was recognized that valid inferences would require a larger number of specimens and more extensive testing.

The results and discussion of the initial evaluations are presented in sections 4.1 through 4.7.

57

4.1   Ability To Read CT Data

Prior to this project, researchers at the BEC were able to view CT imagery, but were unable to directly access the CT numbers. This is now possible through the application of a decoding routine which makes all geometric and densimetric data accessible for further research. The decoded values have been compared with those displayed directly on the hospital's CT acquisition terminal, and have been verified correct.

4.2   User Involvement

The tedious and time-consuming elements requiring human intervention have been greatly reduced. Once the data has been transferred to the processing computer, the user has only to set up a control file, select threshold levels, run the various programs, and specify the number of points to be used for edge definition. Until the various new software components have been combined into a more user-friendly package, the user is still required to enter commands to run each of the individual programs, but this is much easier and less involved than the former process. Edge points can be easily viewed and, if necessary, modified using a newly developed graphical point editor. Prior to the development of the automated edge detection routines, it was not feasible to manually build a model using more than 16 points

to define each edge. Manually placing large numbers of points on each slice was time-prohibitive and prone to human error. The automated method permits any number of points to be selected with no additional user input.

In its preliminary stages, model building also requires less user involvement. ANSYS data files are automatically generated and files no longer need to be constructed using the cut-and-paste technique. The user does not have to keep track of system assigned entity numbers, and a knowledge of ANSYS commands is not necessary to build this preliminary model.

Testing and user reaction have been limited, but very positive.

4.3 Time Savings

A significant time savings has been realized in point selection and input of ANSYS commands. To test the time savings between the manual and automated methods, a comparative test was performed. As it is the hope that this new software will allow users with little training to model femurs, an undergraduate engineering student, who had no experience in image processing, computer modelling, or finite element analysis, was asked to construct a model of a 14 centimeter section of a femur shaft. The CT scan had been

previously conducted, and the raw data files were available for analysis.

The manual model was constructed using the method described in Chapter 1. The student was shown how to use the ImageTool software package to place points marking the edges of the cortical bone, and how to combine the data points from each slice into a single data file suitable for input into the ANSYS package. The student practiced the procedure on test slices until comfortable with the software package and the steps required. As the manual model was built, the student recorded the time spent entering points and constructing the data file for ANSYS. Student comments regarding the procedure were also recorded.

Having completed the manual model, the same student was then asked to construct a model of the same section of femur using the automated method, as outlined in Chapter 3. The student was given a flowchart showing the steps required to construct a femur model, and was shown how to run the newly written software. Again, when comfortable with the procedures, the student followed the flowchart, and constructed the model using the automated method. Comments and completion times were again noted.

In each case, the simulation was kept realistic by using 16 points to define each edge of the cortical bone. In future discussion, this will be referred to as a 16 point model. The user tests showed that it took 10 hours and 45 minutes to manually generate this 16 point model. The same model was constructed in 1 hour and 45 minutes using the newly developed software

60

routines. Although there was a learning curve associated with each method, the orientation times required to become familiar with the manual and automated methods were very comparable and it is not expected that these processing times would lessen substantially with increased familiarity. This time savings will allow more models to be built in shorter time, and will permit the construction of larger, more detailed models.

4.4   Model Smoothness

Although the primary focus of this project was to develop a software package to extract data from CT imagery, some preliminary validation testing has been conducted. For the purposes of testing, three separate models were considered – that generated using the manual methodology with 16 points along each edge (16pt Manual), the automated method with 16 points (16pt Auto), and the automated with 32 points (32pt Auto). All models were built by the aforementioned undergraduate engineering student. Although difficult to show quantitatively, visual comparison between the three models shows that an increase in the number of keypoints used leads to smoother models and improved registration between the CT slices.

To rate the visual appearance of each of the three models, five observers, not previously involved in this project, were independently asked to rate the smoothness of the generated

models. Each of the observers was unfamiliar with computer modelling, image processing and finite element analysis, but had some knowledge of femur morphology. ANSYS plots showing an oblique profile of each model were displayed, and rotated to show all sides of the 3D models. Two views of each of the generated models are shown in Figures 4.1, 4.2, and 4.3. for visual comparison. The observers were asked to consider the following questions when rating the models: (1) are there any irregularities in the model, such as rumples in the surface or sharp edges, which would not appear to reflect the true physical characteristics of a femur, and (2) does the model smoothness come at the expense of natural bone contours. A perfect cylinder would be the smoothest model, but would not represent the true geometry of the bone. The observers were given as much time as needed to complete the ranking, were free to revisit each model, and were able to rotate the models to view any orientation. They were asked to subjectively rank the three in order of smoothness, 1 being the smoothest, 3 being the least smooth. Two of the observers rated the models a second time to provide a measure of reproducibility, and in each case, their ratings were the same. Ideally, a direct comparison should have been made with the original specimen to verify the true geometry. Unfortunately, not anticipating comparative testing, the bone had been disposed of, and was not available. The observer rankings are given in Table 4.1.

Figure 4.1 16pt Manual model

Rumples visible on exterior
surface and sharp join lines
between areas
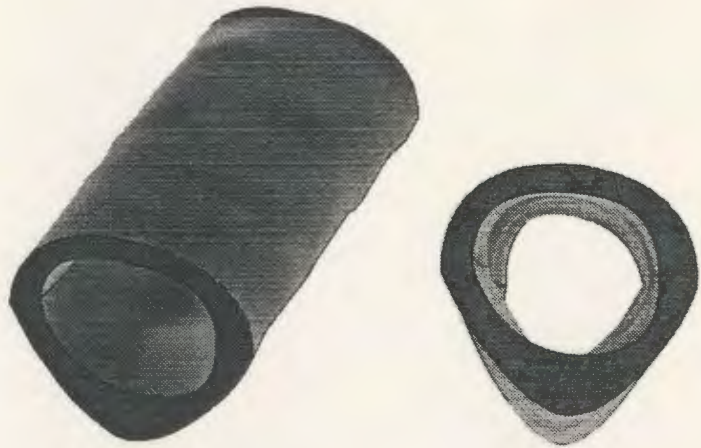
Figure 4.2 16pt Auto model

Smooth, but sharp join
lines between areas
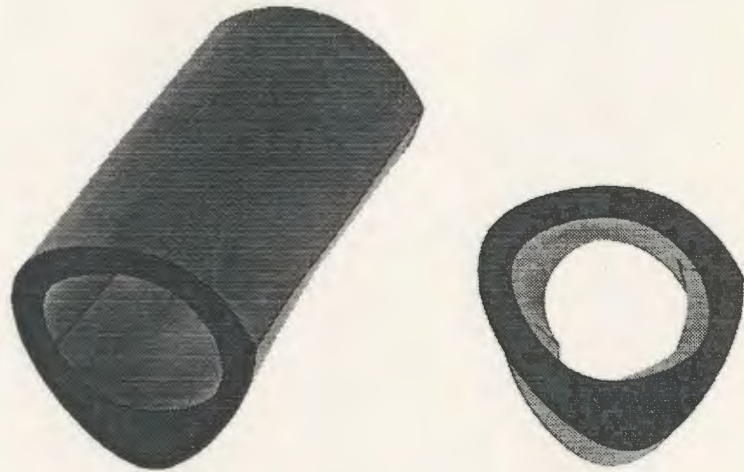
Figure 4.3 32pt Auto model

Smooth joins between
areas with no obvious
rumples
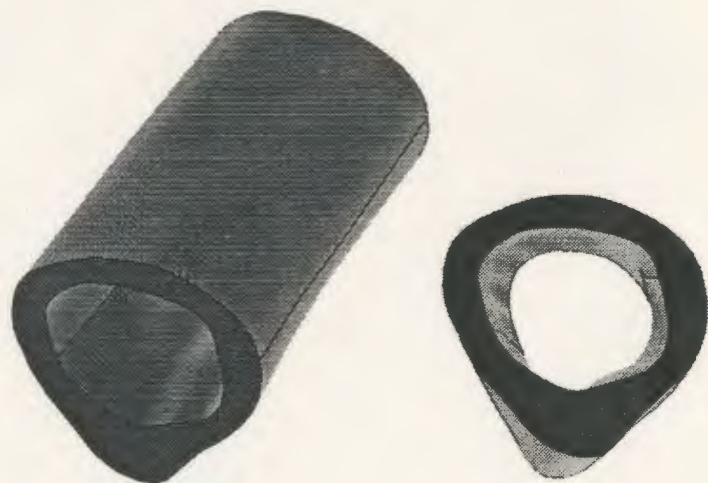
Table 4.1 Observer rankings of model smoothness			
	16pt Manual	16pt Auto	32pt Auto
Observer A	3	2	1
Observer B	3	2	1
Observer C	3	2	1
Observer D	2	3	1
Observer E	3	2	1

As expected, the 32pt Auto model was selected the smoothest by each of the observers. As the number of points selected increased, the greater the chance of proper point alignment, smoother splines, and fewer meshing problems. As a result, the 32pt Auto model is smoother in appearance than either of the 16 point models. The 16pt Auto model was judged smoother than the 16pt Manual model. It is suspected that the rumples in the 16pt Manual model were caused by irregularly spaced edge points, which in turn, caused rumples in the splines. Surface irregularities were also visible in the model where the graphical point editor was used to correct for the strain gauges, illustrating the sensitivity of any model to point registration and the difficulty in manually aligning the points between slices.

4.5   Comparison Through Direct Measurement

Again, the three separate models were considered - 16pt Manual, 16pt Auto, and 32pt Auto. Three different criteria were used to judge the accuracy of each model - overall visual appearance, direct slice measurement comparisons, and comparison of slice areas.

4.5.1 Overall Visual Comparison

Slices 005 and 065 from the generated ANSYS models were selected as test slices to be compared with their actual corresponding CT images. For comparison purposes, all measurements were taken from slice images displayed in the same orientation and scale. For each of the test slices, three ANSYS plots were produced, each showing the corresponding slice from each of the three models. For alignment purposes, four registration points were selected from the original CT slices. The points were selected at approximate midpoints along the left, right, top and bottom edges of the medullary cavity. The (X,Y) coordinates of these registration points were subsequently drawn on each of the model plots, allowing these images, produced using different modalities, to be scaled for proper alignment and comparison. As mentioned, the physical bone had been disposed of, making direct measurement impossible. Therefore, as a baseline, the true edges of the cortical bone were hand traced from the CT images using the observer's best judgement. Having some

knowledge of CT imagery, the same individual who manually placed the points for the 16pt Manual model was also selected to trace these true edges and four registration points onto a transparency film overlay.

For ease of measurement, after assuring proper alignment, the true edges were manually traced onto each of the model plots. When placing the transparency overlay of the true edge over each of the model plots, the four registration points were matched to ensured proper scaling and rotational alignment during the tracing process. The resulting plots and tracings are shown in Figures 4.5 and 4.6.
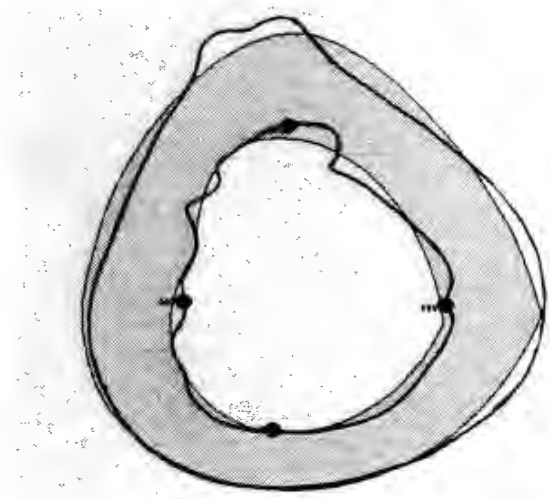
Figure 4.4
Model slice (shaded) with overlaid true edge and registration points

16pt Manual   16pt Auto   32pt Auto
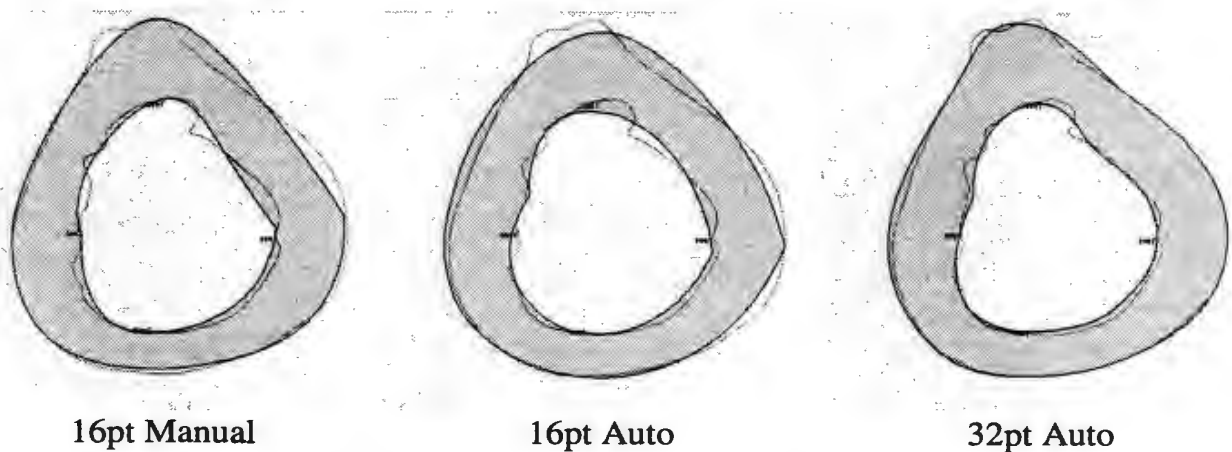
Figure 4.5   Model slices (shaded) and true edges for Slice 005

16pt Manual   16pt Auto   32pt Auto
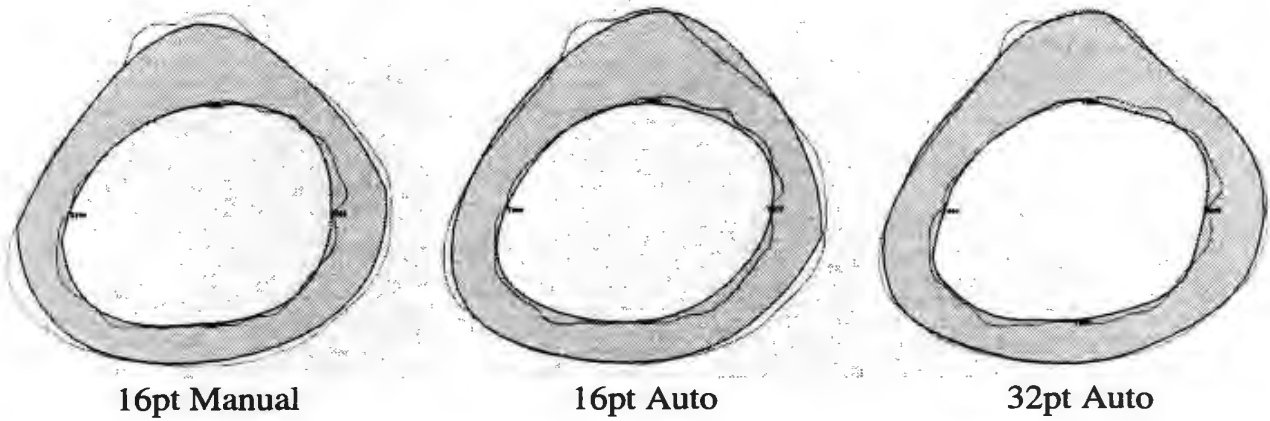
Figure 4.6   Model slices (shaded) and true edges for Slice 065

Again, five observers, not previously involved in this project, were asked to independently judge the fit of the computed edges versus the true edge. The ANSYS plots with the overlaid true edges were shown and the observers were asked to subjectively rank the three models in order of best fit, 1 being closest to the true edge, 3 being the farthest. The results are given in Table 4.2.

Table 4.2 Observer rankings of best edge fit			
	16pt Manual	16pt Auto	32pt Auto
Observer A	2	3	1
Observer B	2	3	1
Observer C	3	2	1
Observer D	2	3	1
Observer E	3	2	1

67

Clearly, the 32pt Auto model was selected as having the best visual fit, with no decisive difference between the 16pt models.

4.5.2 Direct Slice Measurement Comparison

As the visual comparison was subjective, more objective methods were developed involving direct measurement comparisons between the actual edges of the bone and the generated models. Using digital calipers, measurements were taken from each of the models along 16 equiangular lines radiating from the geometric center of the bone slice (as taken from the original CT image), plus four additional lines spanning the full height and width of the inner and outer edges (lines 17, 18, 19 and 20).
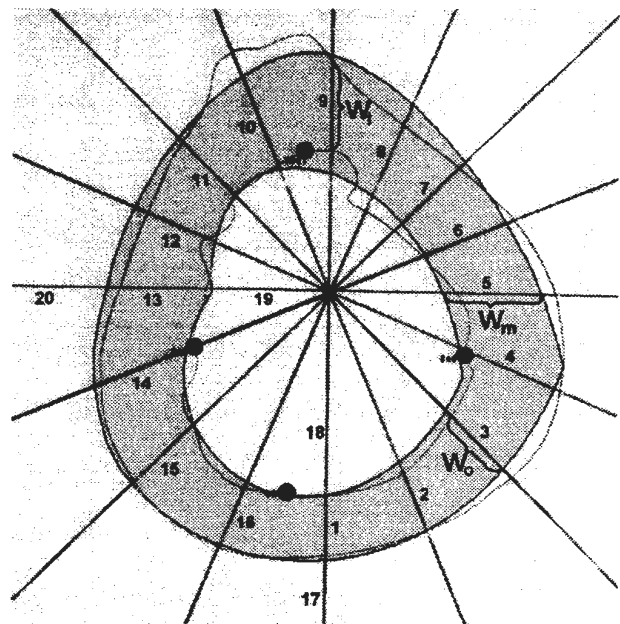
Figure 4.7
Method of direct measurement

Three measurements were taken for each line, as shown in Figure 4.7.

W_t = the width between the true edges

W_m = the width between the modelled edge

W_o = the width of the overlap between the true and modelled edges

68

To analyze the fit between the true CT edges and each of the models, the following calculations were performed:

Percentage Correct = Percentage of model width which coincided with the true width

$$\% \, Correct \;=\; \frac{W_o}{W_t} * 100 \tag{4.1}$$

Percentage Wrong = Percentage of the total width of edge underestimates and overestimates to the true width, where an edge underestimate was a model edge which underestimated the true edge, and an overestimate was a model edge which overestimated the true edge

$$\% \, Wrong \;=\; \frac{(W_m - W_o) + (W_t - W_o)}{W_t} * 100 \tag{4.2}$$

Mr. John Tucker, Research and Project Engineer in the Biomedical Engineering Centre, suggested that the following error analysis equations might also provide useful results:

Absolute Error, AE = Absolute difference between the true edge width and the overlap width

$$AE = |W_t - W_o| \qquad\qquad (4.3)$$

Sum of Absolute Error, SAE = Sum of absolute differences between the true edge width and the overlap width, for a given number of linear measurements, n

$$SAE = \sum_{i=1}^{n} |W_t - W_o| \qquad\qquad (4.4)$$

To express the error as a percentage of the true width, variations of the AE and SAE, the Absolute Percentage Error (APE) and Sum of Absolute Percentage Error (SAPE), were used.

$$APE = \frac{|W_t - W_o|}{W_t} * 100 \qquad\qquad (4.5)$$

$$SAPE = \sum_{i=1}^{n} \frac{|W_t - W_o|}{W_t} * 100 \qquad\qquad (4.6)$$

The measurements and error calculations are presented in Tables 4.3 through 4.8., with graphical results shown in the radial plots of Figures 4.8 through 4.15. The radial plots of Figures 4.8, 4.9, 4.12 and 4.13 show width versus line number, with the radial coordinate along any given line being the width measurement in millimeters. The closer a plot follows that of the true edge the better. The radial plots of Figures 4.10, 4.11, 4.14 and 4.15 show the AE and APE value versus line number, and the radial coordinate along any given line is either the AE (mm) or APE (%). In these plots showing a measure of error, the closer a plot is to zero (the center) the better. A small footprint indicates less error. Measurements for lines 17 through 20 were excluded from the graphs to limit the data range and control the scale of the graphs.

Table 4.3 Direct measurement results for Slice 005 16pt Manual model

Line	W_t (mm)	W_m (mm)	W_o (mm)	% Correct	% Wrong	AE (mm)	APE (%)
1	14.5	13.0	13.0	89.7	10.3	1.5	10.34
2	15.0	17.0	15.0	100.0	13.3	0.0	0.00
3	19.5	22.9	19.5	100.0	17.4	0.0	0.00
4	24.5	24.0	24.0	98.0	2.0	0.5	2.04
5	24.0	26.0	20.0	83.3	41.7	4.0	16.67
6	23.0	22.0	19.0	82.6	30.4	4.0	17.39
7	22.0	21.0	21.0	95.5	4.5	1.0	4.55
8	23.0	22.5	18.5	80.4	37.0	4.5	19.57
9	21.0	27.5	21.0	100.0	31.0	0.0	0.00
10	29.5	25.5	24.5	83.1	20.3	5.0	16.95
11	17.5	20.0	17.5	100.0	14.3	0.0	0.00
12	14.5	18.0	14.5	100.0	24.1	0.0	0.00
13	21.0	19.0	18.0	85.7	19.0	3.0	14.29
14	25.0	23.5	23.5	94.0	6.0	1.5	6.00
15	20.5	23.5	20.5	100.0	14.6	0.0	0.00
16	15.0	14.5	14.5	96.7	3.3	0.5	3.33
17	120.1	123.0	117.5	97.8	6.7	2.6	2.16
18	105.0	102.9	101.1	96.3	5.4	3.9	3.71
19	83.0	82.1	82.1	98.9	1.1	0.9	1.08
20	59.0	57.0	53.5	90.7	15.3	5.5	9.32
			Average	93.6	15.9		
			Sum			38.4	127.41

Table 4.4 Direct measurement results for Slice 065 16pt Manual model

Line	W_t (mm)	W_m (mm)	W_o (mm)	% Correct	% Wrong	AE (mm)	APE (%)
1	14.1	13.5	13.5	95.7	4.3	0.6	4.26
2	13.2	13.2	13.2	100.0	0.0	0.0	0.00
3	15.0	12.8	12.8	85.3	14.7	2.2	14.67
4	17.1	14.1	12.7	74.3	33.9	4.4	25.73
5	14.8	15.3	14.8	100.0	3.4	0.0	0.00
6	14.7	14.6	14.6	99.3	0.7	0.1	0.68
7	17.8	18.2	16.8	94.4	13.5	1.0	5.62
8	30.9	24.8	24.8	80.3	19.7	6.1	19.74
9	31.9	28.6	28.1	88.1	13.5	3.8	11.91
10	19.6	19.8	18.1	92.3	16.3	1.5	7.65
11	13.1	12.4	12.4	94.7	5.3	0.7	5.34
12	11.8	12.4	10.6	89.8	25.4	1.2	10.17
13	14.8	17.9	13.6	91.9	37.2	1.2	8.11
14	17.8	17.1	15.8	88.8	18.5	2.0	11.24
15	15.5	13.9	13.9	89.7	10.3	1.6	10.32
16	12.6	12.6	11.0	87.3	25.4	1.6	12.70
17	127.4	123.1	123.1	96.6	3.4	4.3	3.38
18	119.0	117.9	117.9	99.1	0.9	1.1	0.92
19	80.1	79.9	79.9	99.8	0.2	0.2	0.25
20	88.4	83.6	83.6	94.6	5.4	4.8	5.43
			Average	92.1	12.6		
			Sum			38.4	158.12

Table 4.5 Direct measurement results for Slice 005 16pt Auto model

Line	W_t (mm)	W_m (mm)	W_o (mm)	% Correct	% Wrong	AE (mm)	APE (%)
1	14.5	15.5	14.5	100.0	6.9	0.0	0.00
2	15.0	17.9	15.0	100.0	19.3	0.0	0.00
3	19.5	19.5	17.5	89.7	20.5	2.0	10.26
4	24.5	25.0	23.0	93.9	14.3	1.5	6.12
5	24.0	23.5	21.0	87.5	22.9	3.0	12.50
6	23.0	21.0	21.0	91.3	8.7	2.0	8.70
7	22.0	21.0	19.0	86.4	22.7	3.0	13.64
8	23.0	24.5	19.0	82.6	41.3	4.0	17.39
9	21.0	29.0	21.0	100.0	38.1	0.0	0.00
10	29.5	24.0	23.0	78.0	25.4	6.5	22.03
11	17.5	19.5	17.5	100.0	11.4	0.0	0.00
12	14.5	21.0	14.5	100.0	44.8	0.0	0.00
13	21.0	25.0	21.0	100.0	19.0	0.0	0.00
14	25.0	29.1	22.5	90.0	36.4	2.5	10.00
15	20.5	20.5	20.5	100.0	0.0	0.0	0.00
16	15.0	16.5	14.5	96.7	16.7	0.5	3.33
17	120.1	123.0	120.1	100.0	2.4	0.0	0.00
18	105.0	104.0	100.5	95.7	7.6	4.5	4.29
19	83.0	78.0	78.0	94.0	6.0	5.0	6.02
20	59.0	55.0	55.0	93.2	6.8	4.0	6.78
			Average	93.9	18.6		
			Sum			38.5	121.06

Table 4.6 Direct measurement results for Slice 065 16pt Auto model

Line	W_t (mm)	W_m (mm)	W_o (mm)	% Correct	% Wrong	AE (mm)	APE (%)
1	14.1	14.9	13.6	96.5	12.8	0.5	3.55
2	13.2	14.6	13.2	100.0	10.6	0.0	0.00
3	15.0	16.3	15.0	100.0	8.7	0.0	0.00
4	17.1	15.6	14.9	87.1	17.0	2.2	12.87
5	14.8	16.5	14.8	100.0	11.5	0.0	0.00
6	14.7	16.4	14.7	100.0	11.6	0.0	0.00
7	17.8	19.7	17.4	97.8	15.2	0.4	2.25
8	30.9	25.8	25.8	83.5	16.5	5.1	16.50
9	31.9	33.1	31.4	98.4	6.9	0.5	1.57
10	19.6	28.8	19.6	100.0	46.9	0.0	0.00
11	13.1	19.3	13.1	100.0	47.3	0.0	0.00
12	11.8	13.1	11.7	99.2	12.7	0.1	0.85
13	14.8	14.0	11.5	77.7	39.2	3.3	22.30
14	17.8	17.1	15.8	88.8	18.5	2.0	11.24
15	15.5	15.8	14.0	90.3	21.3	1.5	9.68
16	12.6	15.7	12.4	98.4	27.8	0.2	1.59
17	127.4	127.2	126.2	99.1	1.7	1.2	0.94
18	119.0	116.4	116.0	97.5	2.9	3.0	2.52
19	80.1	78.7	78.7	98.3	1.7	1.4	1.75
20	88.4	84.9	84.9	96.0	4.0	3.5	3.96
			Average	95.4	16.7		
			Sum			24.9	91.57

Table 4.7 Direct measurement results for Slice 005 32pt Auto model

Line	W_t (mm)	W_m (mm)	W_o (mm)	% Correct	% Wrong	AE (mm)	APE (%)
1	14.5	16.0	14.5	100.0	10.3	0.0	0.00
2	15.0	17.0	15.0	100.0	13.3	0.0	0.00
3	19.5	21.5	19.5	100.0	10.3	0.0	0.00
4	24.5	24.5	24.5	100.0	0.0	0.0	0.00
5	24.0	25.0	24.0	100.0	4.2	0.0	0.00
6	23.0	23.0	22.5	97.8	4.3	0.5	2.17
7	22.0	22.9	22.0	100.0	4.1	0.0	0.00
8	23.0	22.0	21.1	91.7	12.2	1.9	8.26
9	21.0	26.0	21.0	100.0	23.8	0.0	0.00
10	29.5	27.5	27.5	93.2	6.8	2.0	6.78
11	17.5	19.0	17.5	100.0	8.6	0.0	0.00
12	14.5	19.5	14.5	100.0	34.5	0.0	0.00
13	21.0	24.9	21.0	100.0	18.6	0.0	0.00
14	25.0	26.0	25.0	100.0	4.0	0.0	0.00
15	20.5	20.5	19.9	97.1	5.9	0.6	2.93
16	15.0	17.0	15.0	100.0	13.3	0.0	0.00
17	120.1	122.0	119.0	99.1	3.4	1.1	0.92
18	105.0	107.0	104.0	99.0	3.8	1.0	0.95
19	83.0	79.0	79.0	95.2	4.8	4.0	4.82
20	59.0	57.0	57.0	96.6	3.4	2.0	3.39
			Average	98.5	9.5		
			Sum			13.1	30.22

Table 4.8 Direct measurement results for Slice 065 32pt Auto model

| Line	W_t (mm)	W_m (mm)	W_o (mm)	% Correct	% Wrong	AE (mm)	APE (%)
1	14.1	15.6	14.1	100.0	10.6	0.0	0.00
2	13.2	14.5	13.0	98.5	12.9	0.2	1.52
3	15.0	15.1	15.1	100.7	-0.7	0.1	0.67
4	17.1	17.4	16.3	95.3	11.1	0.8	4.68
5	14.8	16.3	14.8	100.0	10.1	0.0	0.00
6	14.7	16.6	14.7	100.0	12.9	0.0	0.00
7	17.8	19.1	17.8	100.0	7.3	0.0	0.00
8	30.9	26.7	26.9	87.1	12.3	4.0	12.94
9	31.9	32.3	31.9	100.0	1.3	0.0	0.00
10	19.6	23.1	19.6	100.0	17.9	0.0	0.00
11	13.1	14.4	12.3	93.9	22.1	0.8	6.11
12	11.8	13.8	11.8	100.0	16.9	0.0	0.00
13	14.8	17.9	14.1	95.3	30.4	0.7	4.73
14	17.8	19.6	16.9	94.9	20.2	0.9	5.06
15	15.5	14.9	14.9	96.1	3.9	0.6	3.87
16	12.6	14.1	11.5	91.3	29.4	1.1	8.73
17	127.4	126.2	127.4	100.0	0.6	0.0	0.00
18	119.0	119.3	118.5	99.6	1.1	0.5	0.42
19	80.1	79.0	79.0	98.6	1.4	1.1	1.37
20	88.4	84.3	84.3	95.4	4.6	4.1	4.64
			Average	97.3	11.3		
			Sum			14.9	54.74
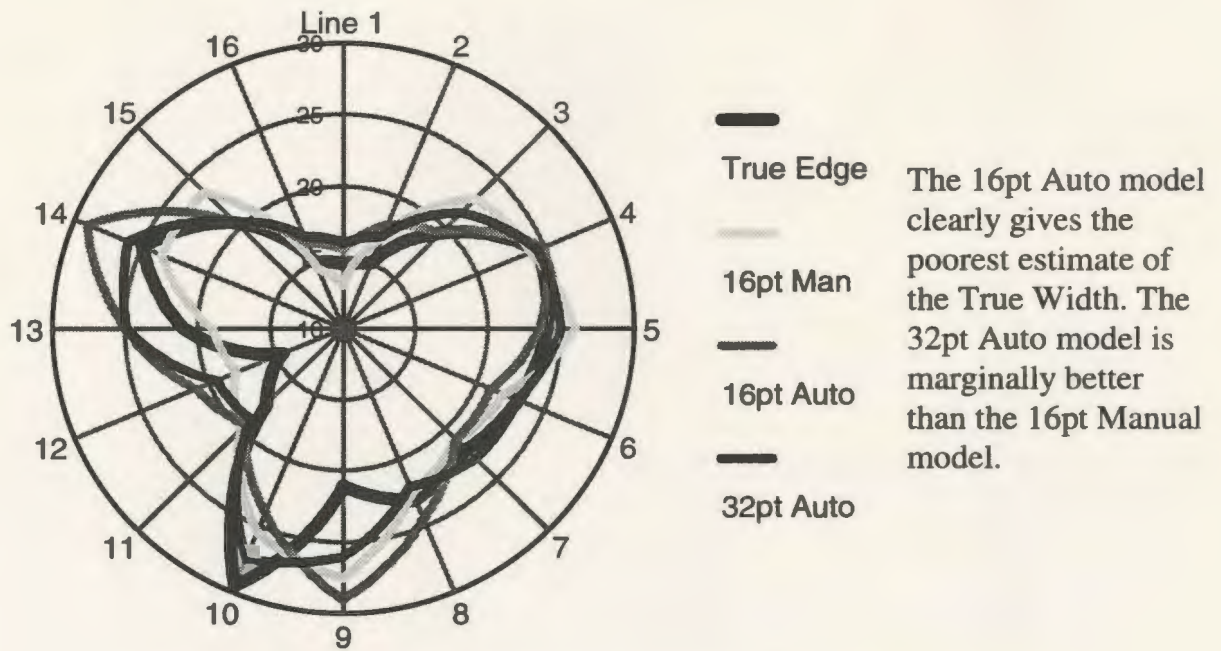
Figure 4.8   Slice 005 Model Width compared with True Width (mm)

The 16pt Auto model clearly gives the poorest estimate of the True Width. The 32pt Auto model is marginally better than the 16pt Manual model.
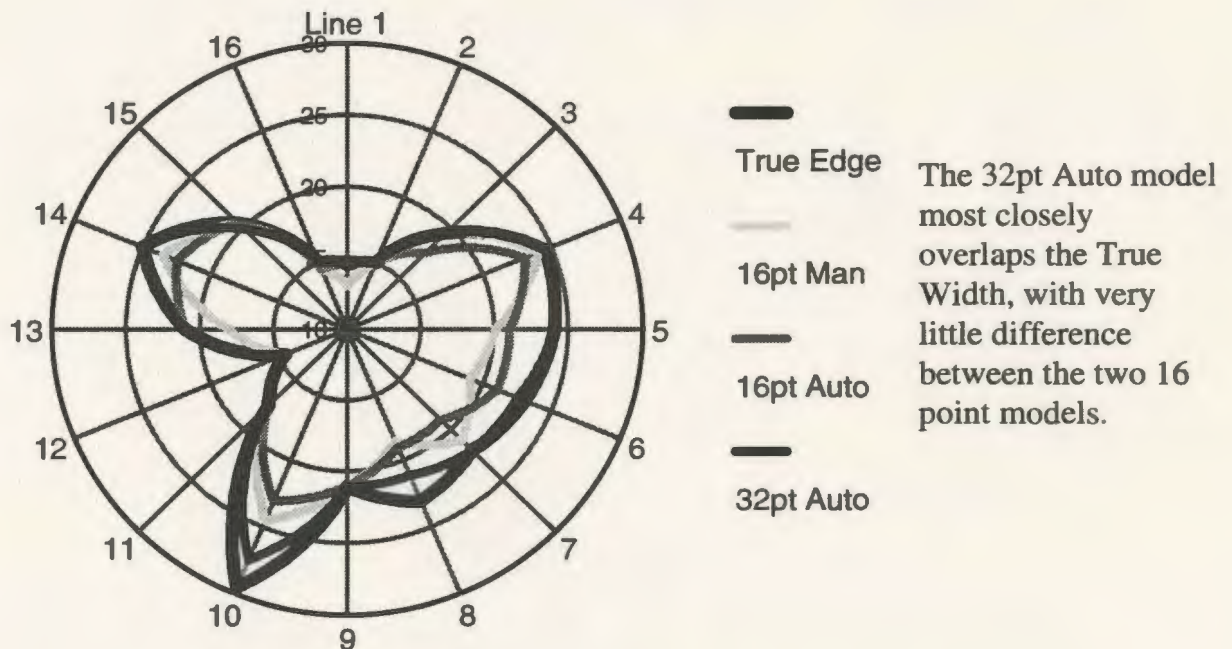
Figure 4.9   Slice 005 Model Overlap compared with True Width (mm)

The 32pt Auto model most closely overlaps the True Width, with very little difference between the two 16 point models.

75

Figure 4.10   Slice 005 Absolute Error (AE) (mm)
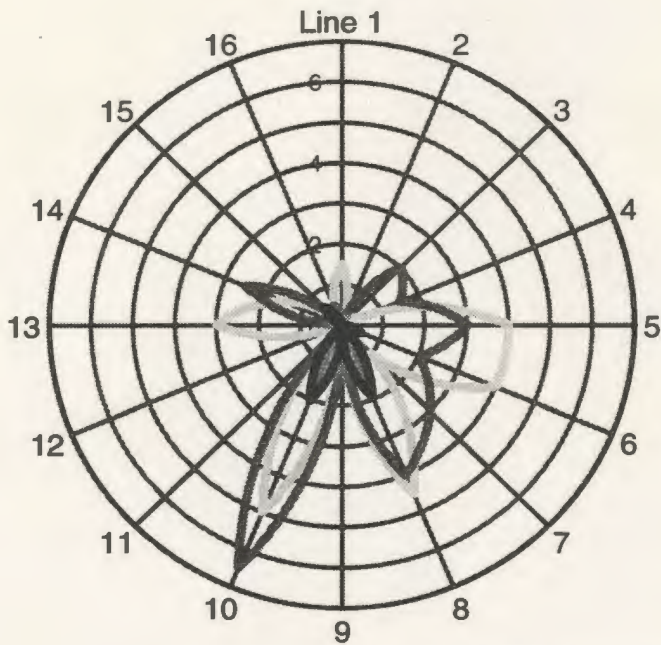
With the smallest AE footprint, the 32pt Auto model is the best of the three. The 16pt Manual model is marginally better than the 16pt Auto model.

16pt Man
16pt Auto
32pt Auto

Figure 4.11   Slice 005 Absolute Percentage Error (APE) (%)

The small APE footprint indicates the 32pt Auto model is the better of the three. The 16pt Manual model is the worst fit.

16pt Man
16pt Auto
32pt Auto

The 32pt Auto Model Widths most closely match the True Widths. The 16pt Auto model tends to overestimate the widths, but is still marginally better than the 16pt Manual model, which tends to underestimate the widths.
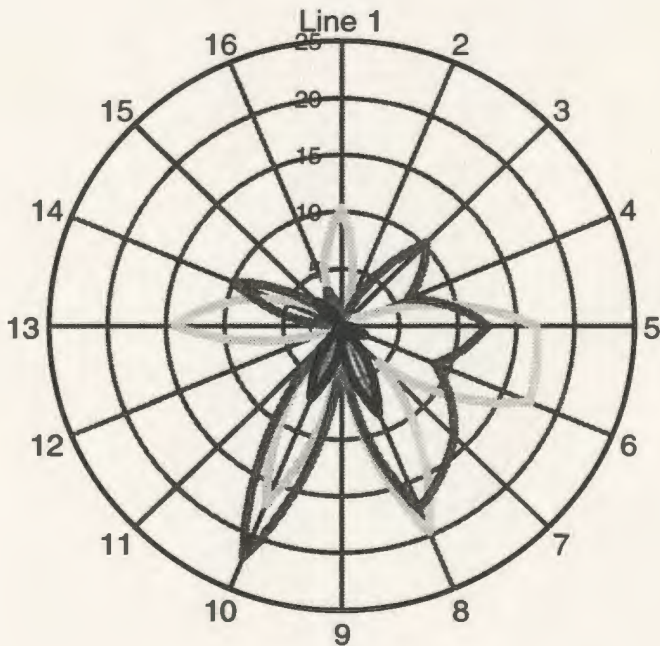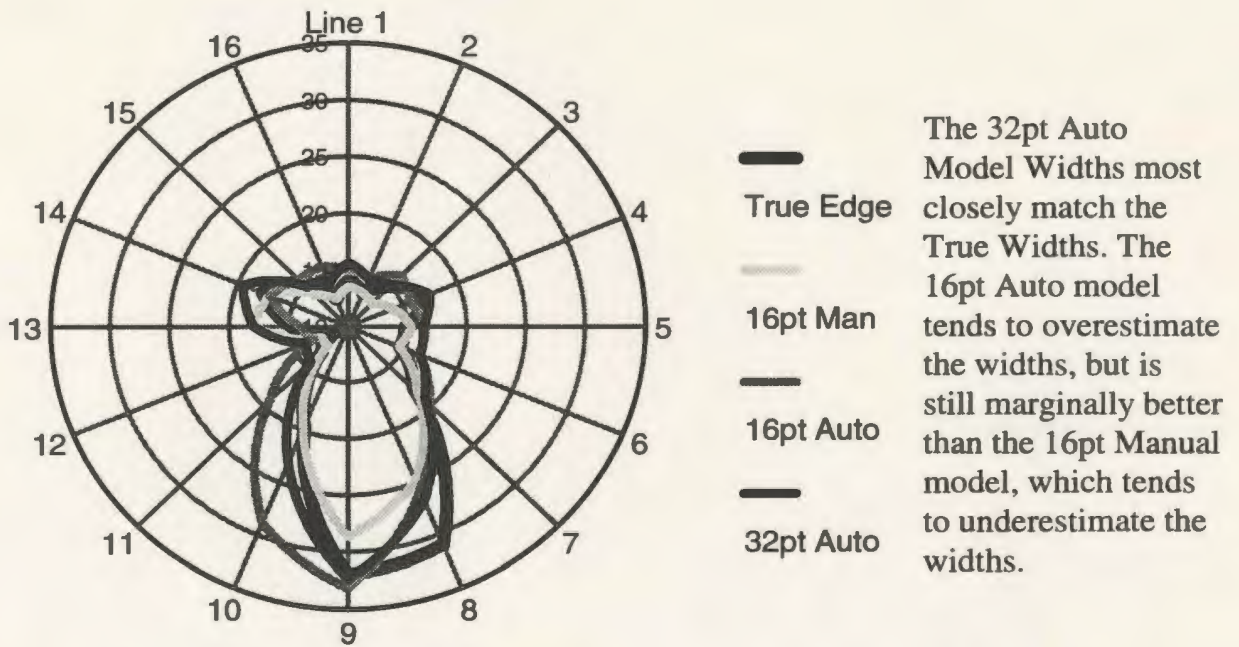
Figure 4.12   Slice 065 Model Width compared with True Width (mm)

The 16pt Manual model does not perform as well as the others. The 32pt Auto model is marginally better than the 16pt Auto model.

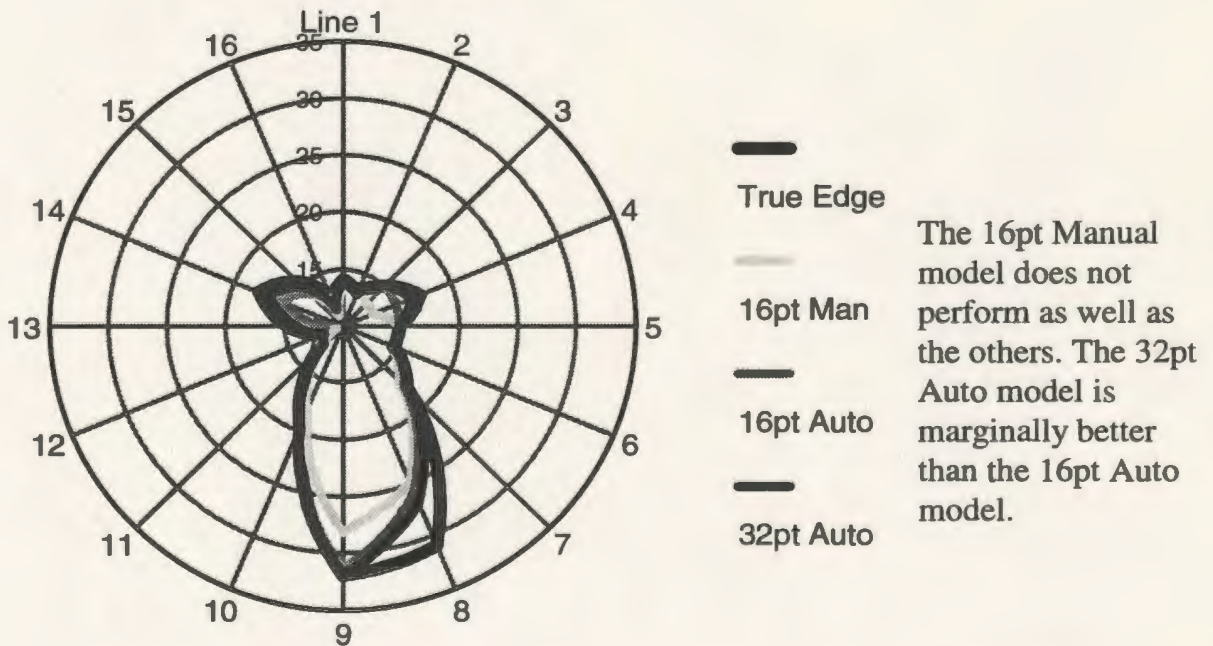Figure 4.13   Slice 065 Model Overlap compared with True Width (mm)
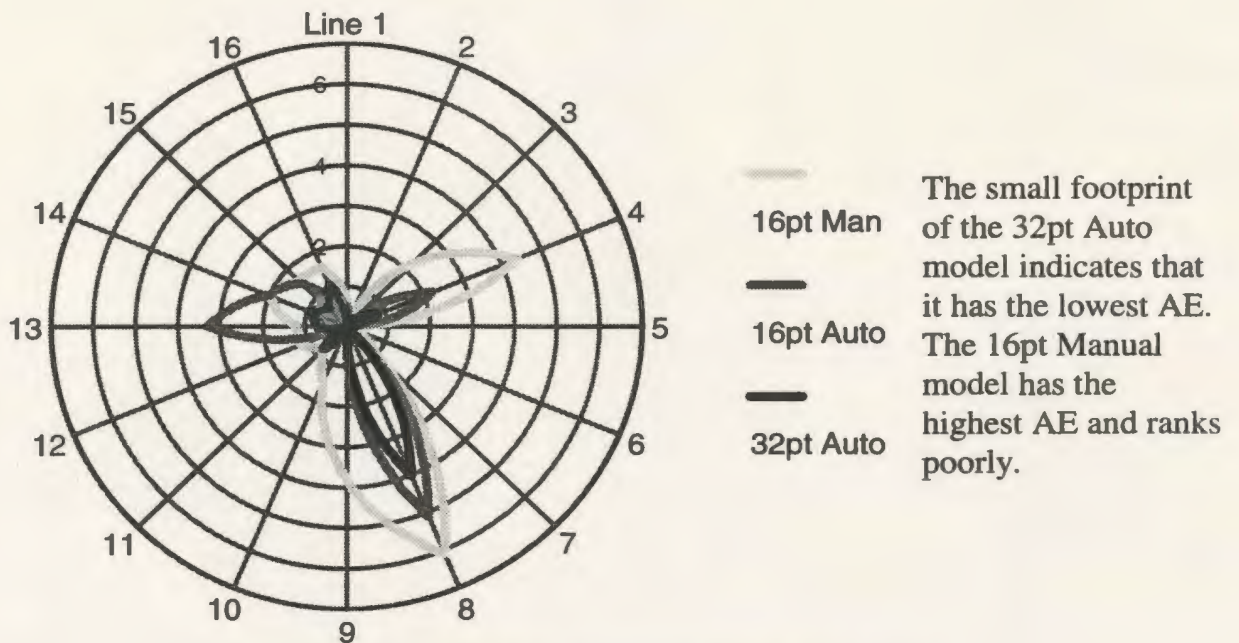
Figure 4.14   Slice 065 Absolute Error (AE) (mm)

The small footprint of the 32pt Auto model indicates that it has the lowest AE. The 16pt Manual model has the highest AE and ranks poorly.
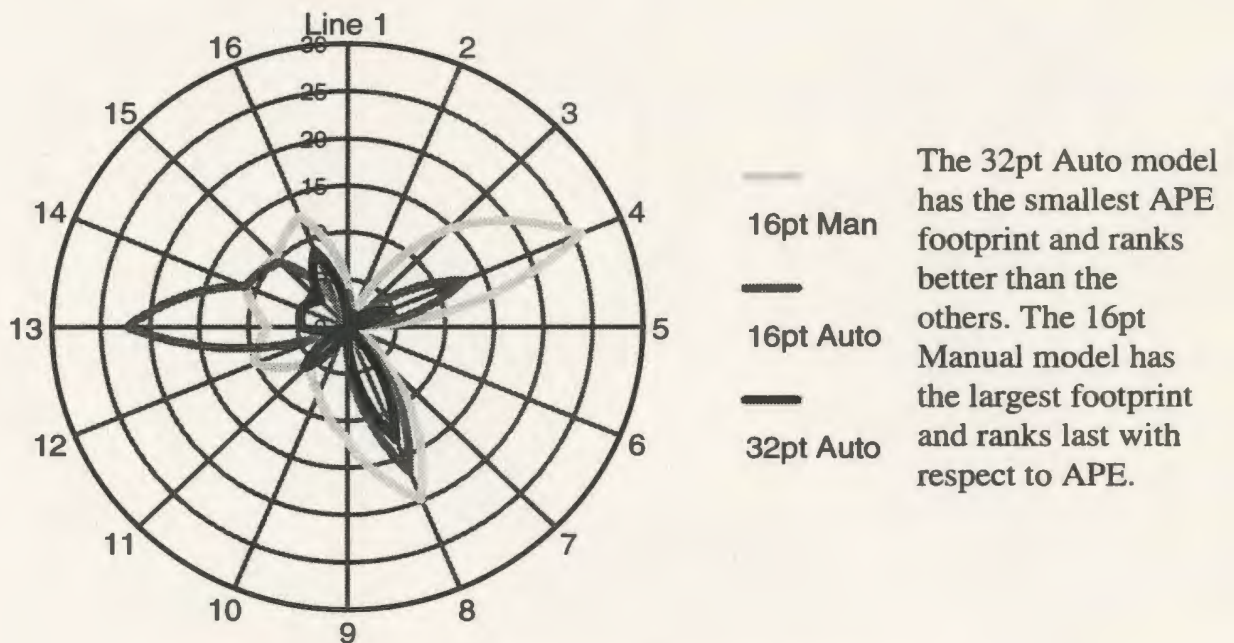
Figure 4.15   Slice 065 Absolute Percentage Error (APE) (%)

The 32pt Auto model has the smallest APE footprint and ranks better than the others. The 16pt Manual model has the largest footprint and ranks last with respect to APE.

Although subject to measurement error, these results support the visual comparisons, with the %Correct and %Wrong showing the 32pt Auto model to be the best fit, with the greatest Percentage Correct and the least Percentage Wrong. The SAE and SAPE values indicate the two 16 point models are very similar for Slice 005, but, for Slice 065, the 16pt Auto model performs better. The graphs show that the 32pt Auto model most closely follows the true edge, and its small footprint in the AE and APE plots verifies this. The ridge running along the posterior surface of the femur (measured by lines 9 and 10) is poorly modelled with the 16 point models, but the added points of the 32 point model help in its definition. As expected, the 32pt Auto model most closely represents the true femur, and the 16pt Auto model marginally outperforms its manual counterpart.

Although the measurements were taken along a series of equiangular lines, with no bias towards any one area of the slice, it was still possible that these discrete measurements did not reflect the overall discrepancy between the edges. Chance may have placed the measurement lines in non-representative areas. Rather than decrease the angles between measurement lines, again risking non-representative measurements, it was decided that a comparison of the model areas was needed.

4.5.3 Comparison of Slice Areas

The complex shapes of the femur slices made area determination difficult. As the BEC had no appropriate image processing software available at that time, the simple solution adopted by Adams et al. [1] was used, whereby plots of these slice areas were cut out and the paper weighed. By weighing pieces of the same paper of known area, a relationship between area and weight was determined. Using this technique, various areas of interest can be plotted, cut out, weighed, and converted into areas.

Again, the true edges were manually traced onto ANSYS plots of the three models, and using a method similar to that described by Farman-Ara et al. [14], key areas were identified, cut out and weighed. Due to the small areas involved, a precision, digital Mettler AE 200 scale (reading to 1/10,000th of a gram) was used to weigh the paper samples involved. Using this technique, the following areas were determined:
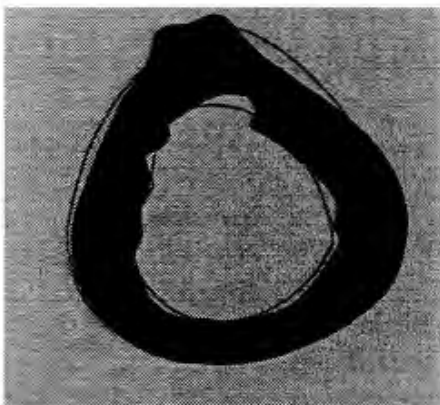

True Area

- The area enclosed by the tracing of the CT image's true edges. This represents the true area of the cortical bone.
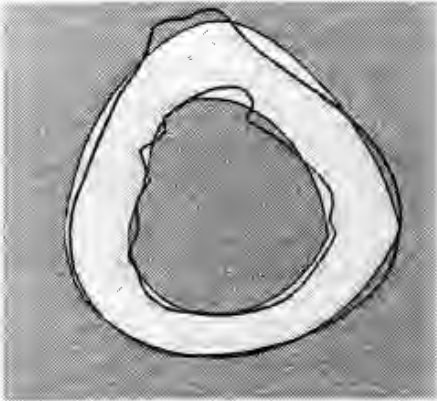
Figure 4.16
True area

80

Figure 4.17
Computed area

Computed Area

- The area enclosed by the ANSYS model's edges. This area represents the computed model area of cortical bone.
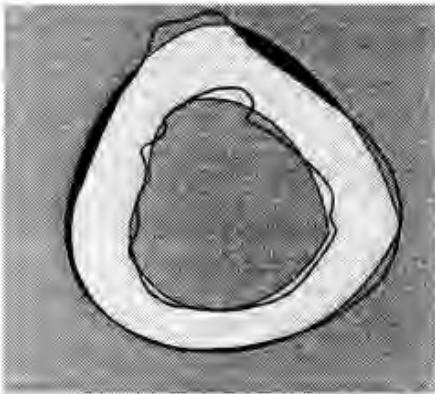

Figure 4.18
Outer false positive

Outer False Positive

- Areas of the model which extend beyond the true outer edge. They are falsely identified as areas of cortical bone.
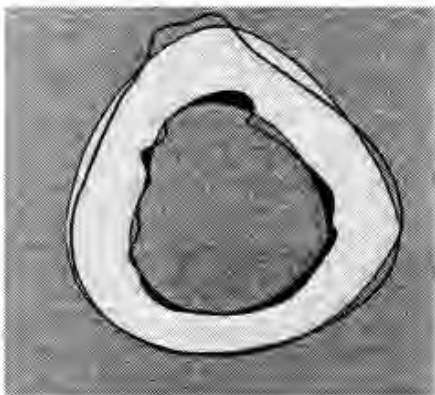
Figure 4.19
Inner false positive

Inner False Positive

- Areas of the model which extend beyond the true inner edge. They are falsely identified as areas of cortical bone.
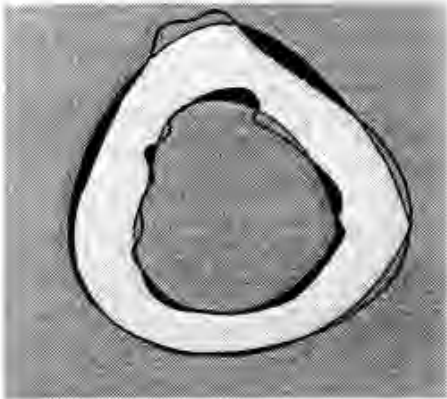
Figure 4.20
Total false positive

Total False Positive

- Total area falsely identified as cortical bone.
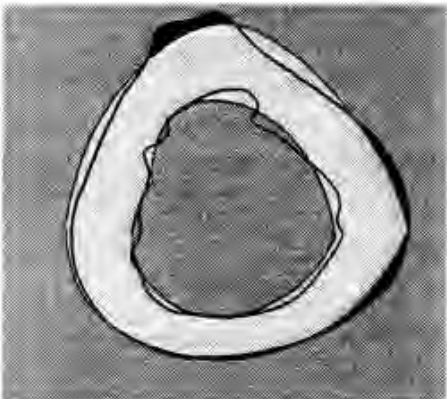
Figure 4.21
Outer false negative

Outer False Negative

- Areas of the true cortical bone which extend

beyond the model's outer edge. These areas

are falsely identified as air.
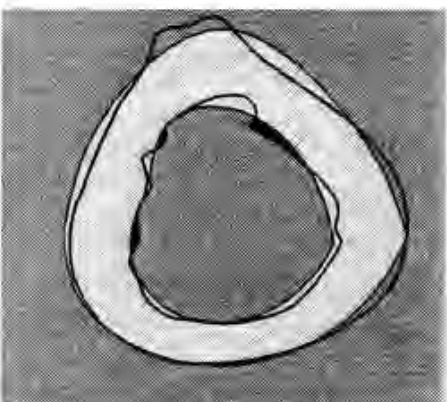
Figure 4.22
Inner false negative

Inner False Negative

- Areas of the true cortical bone which extend

beyond the model's inner edge. These areas

are falsely identified as air.
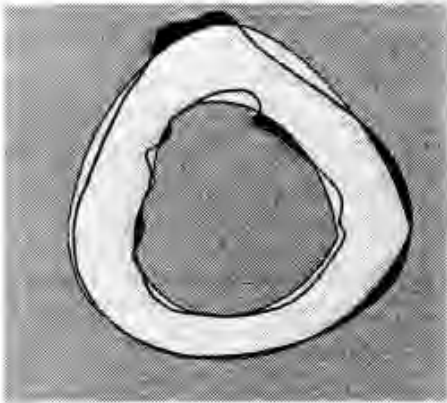
Figure 4.23
Total false negative

Total False Negative

- Total area falsely identified as air.
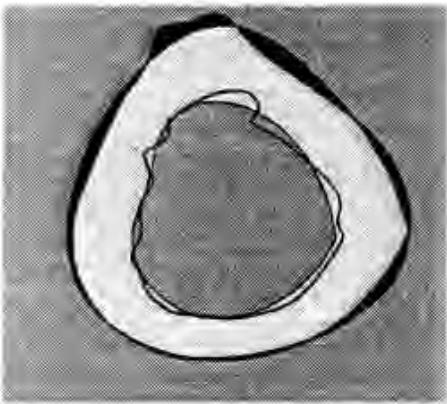
Figure 4.24
False outer edge

False Outer Edge

- Areas along the outer edge falsely identified
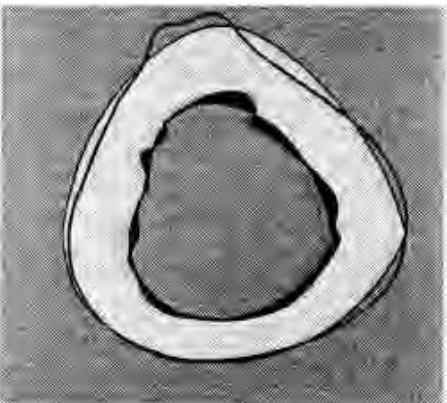
as being air or bone.

Figure 4.25
False inner edge

False Inner Edge

- Areas along the inner edge falsely identified
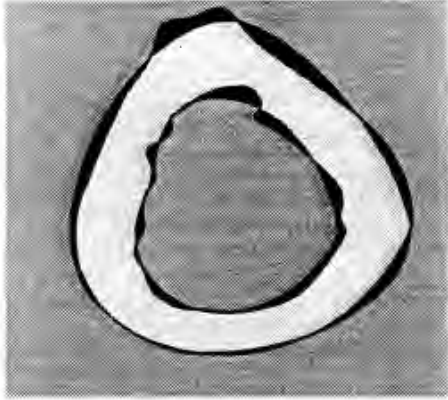
as being air or bone.

Figure 4.26
Total false edge

Total False Edge

- Total area falsely identified as being air or

bone.

These areas gave an indication of the fit of each of the models and represented width and position displacement errors. The resulting measurements are given in Tables 4.9 and 4.10, and the basic measurements are presented graphically in Figures 4.27 and 4.28. Further detailed graphs are shown in Appendix C.

| Table 4.9 Weights of model areas as measured and calculated (grams)											
	True Area	Computed Area	True/ Computed Overlap	Inner False Positive	Inner False Negative	Outer False Positive	Outer False Negative	Total False Positive	Total False Negative	False Inner Edge	False Outer Edge	Total False Edge
Slice 005												
16pt man	1.025	1.057	0.948	0.060	0.020	0.049	0.058	0.109	0.078	0.080	0.107	0.187
16pt auto	1.033	1.091	0.957	0.061	0.019	0.073	0.057	0.134	0.076	0.080	0.130	0.210
32pt auto	1.045	1.119	1.028	0.041	0.007	0.051	0.010	0.092	0.017	0.048	0.061	0.109
Slice 065												
16pt man	0.980	0.927	0.880	0.040	0.014	0.008	0.086	0.048	0.100	0.054	0.094	0.148
16pt auto	0.979	1.041	0.917	0.058	0.007	0.067	0.056	0.124	0.062	0.064	0.122	0.186
32pt auto	0.967	1.026	0.939	0.052	0.008	0.036	0.021	0.088	0.029	0.059	0.057	0.116

| Table 4.10 Model areas expressed as a percentage											
	% of True Area		% of Computed Area									
Slice 005												
16pt man	100.0%	103.1%	89.7%	5.7%	1.9%	4.6%	5.5%	10.3%	7.3%	7.6%	10.1%	17.7%
16pt auto	100.0%	105.6%	87.7%	5.6%	1.8%	6.7%	5.2%	12.3%	7.0%	7.3%	11.9%	19.2%
32pt auto	100.0%	107.1%	91.8%	3.7%	0.6%	4.5%	0.9%	8.2%	1.6%	4.3%	5.4%	9.8%
Slice 065												
16pt man	100.0%	94.7%	94.8%	4.3%	1.5%	0.9%	9.3%	5.2%	10.8%	5.8%	10.2%	16.0%
16pt auto	100.0%	106.4%	88.1%	5.6%	0.6%	6.4%	5.3%	12.0%	6.0%	6.2%	11.7%	17.9%
32pt auto	100.0%	106.1%	91.5%	5.0%	0.7%	3.5%	2.1%	8.5%	2.8%	5.8%	5.6%	11.4%
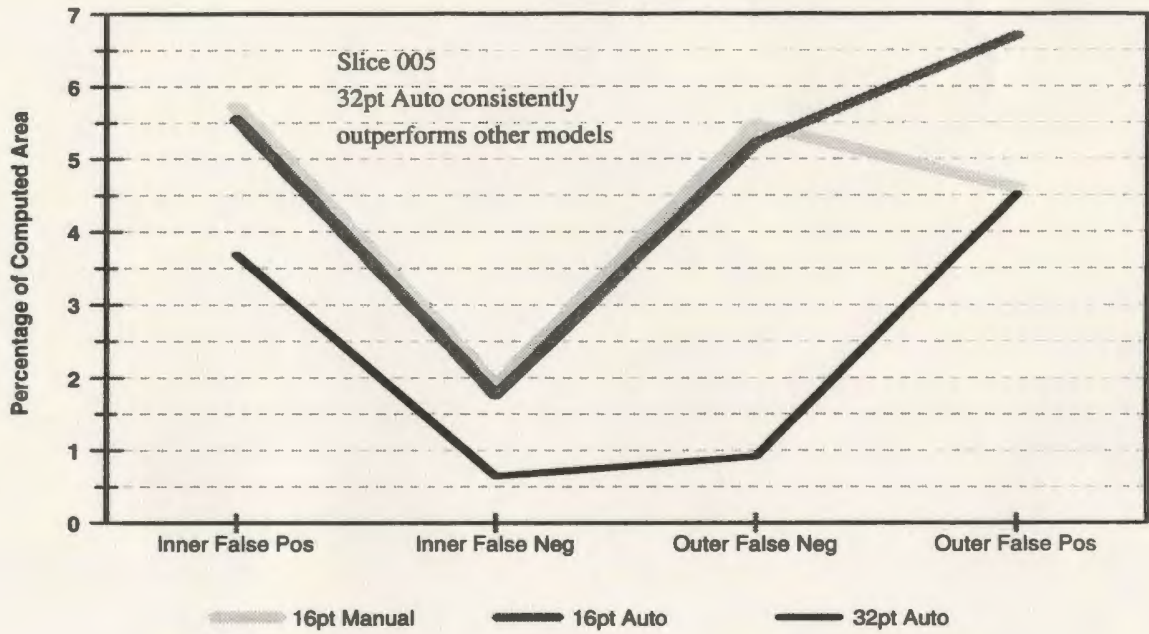
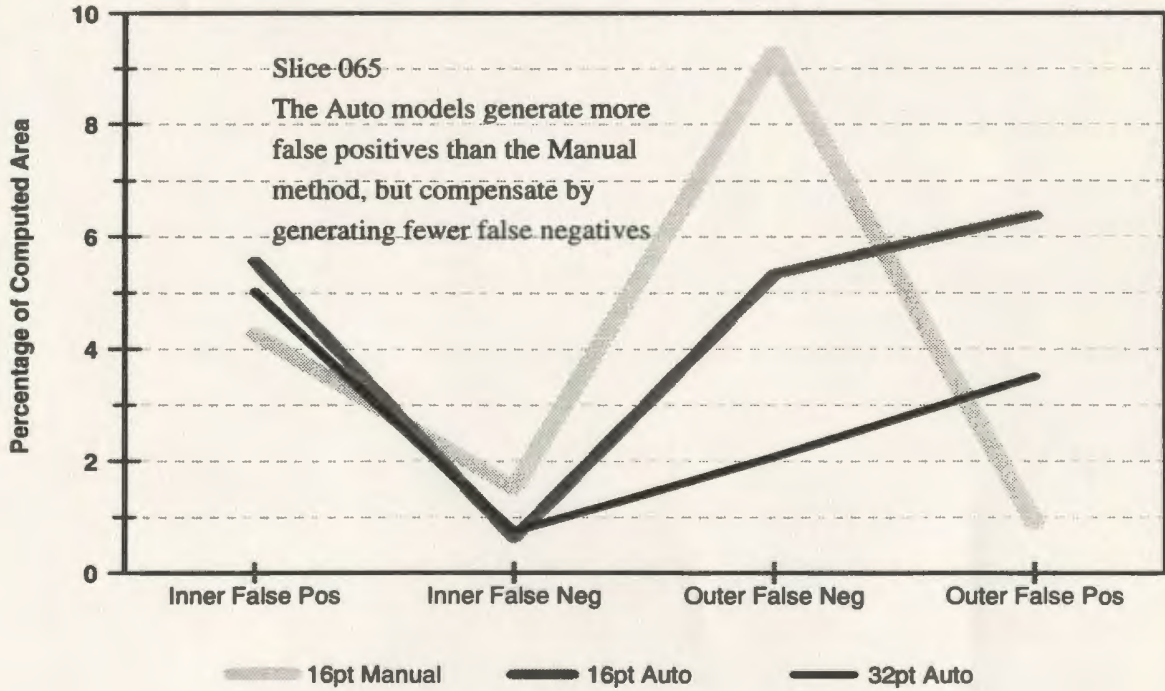Figure 4.27 Slice 005 False Edge Areas for each model (%)

Figure 4.28 Slice 065 False Edge Areas for each model (%)

There were several sources of error in these measurements, including accuracy in judging the actual edge, manually tracing it to each of the plots, cutting and weighing each of the areas, and the accuracy of the scale. As such, the quantitative accuracy of the results may be questioned, but overall, they should be seen as a general indicator of best fit.

The poor performance of the 16pt Auto model was somewhat surprising as it had been anticipated that the results between the two 16pt models would have been closer. This may be partially explained as

1) An aberration in the test data resulting from too limited a sample size - Comparisons among a greater number of slices may have produced more accurate, and different, results. Other slice comparisons may have shown the 16pt Auto model to be a better fit.

2) Testing bias - The same user who manually placed the points to build the 16pt Manual model, also determined and traced the true edges onto each of the plots. Therefore, the same biases that were used to determine point placement were also used in determining the true edges.

3) Use of an inaccurate method for determining areas - Weighing pieces of paper is not an ideal method. A better, computer-based method of area

87

determination is needed for future tests.

4)   No attempt was made to refine the initially selected thresholds - The first

      model generated was used for test purposes and the initial edges were not

      adjusted or fine-tuned. With some threshold adjustments, better edges might

      have resulted, altering the model rankings.

The difference in the results between the 16pt models is in direct conflict with the earlier

visual comparisons and linear measurements which indicated that they produced very similar

results. To see what the cause of this discrepancy might be, the original CT images were

plotted with the selected points overlaid, as shown in Figures 4.29 and 4.30.

16pt Manual*   16pt Auto   32pt Auto
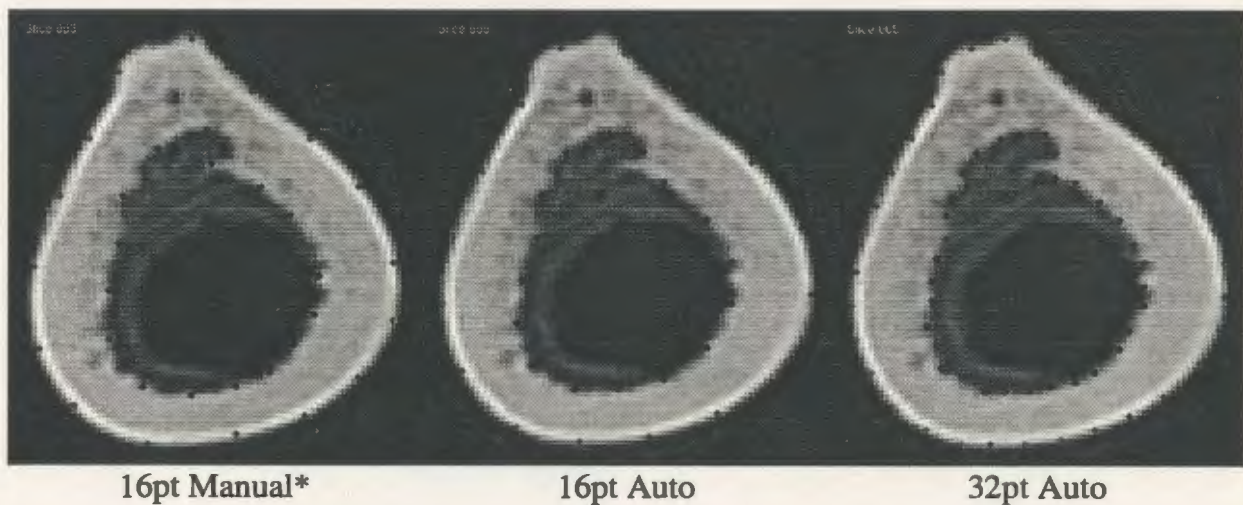
Figure 4.29   Slice 005 models with selected edge points

* Note the inconsistency in the spacing between the manually placed points.

88

16pt Manual   16pt Auto   32pt Auto

Figure 4.30   Slice 065 models with selected edge points

Again, upon visual inspection, there appeared to be little difference between them, certainly nothing that should cause such a difference in their results. Comparing the placement of the selected edge points with the edges of the generated ANSYS models, however, indicates that factors other than point placement may play a major role in the model definition. As shown in Figure 4.31, not all of the selected points appear to influence the placement of the model

Figure 4.31   Possible effects of splining and skinning - Some points
don't appear to influence placement of model edges

89

edges. This indicates that the ANSYS splining and skinning operations are not modelling the surfaces of the femur as anticipated. Clearly, this is an area which requires further investigation, as the accuracy of the model is significantly affected.
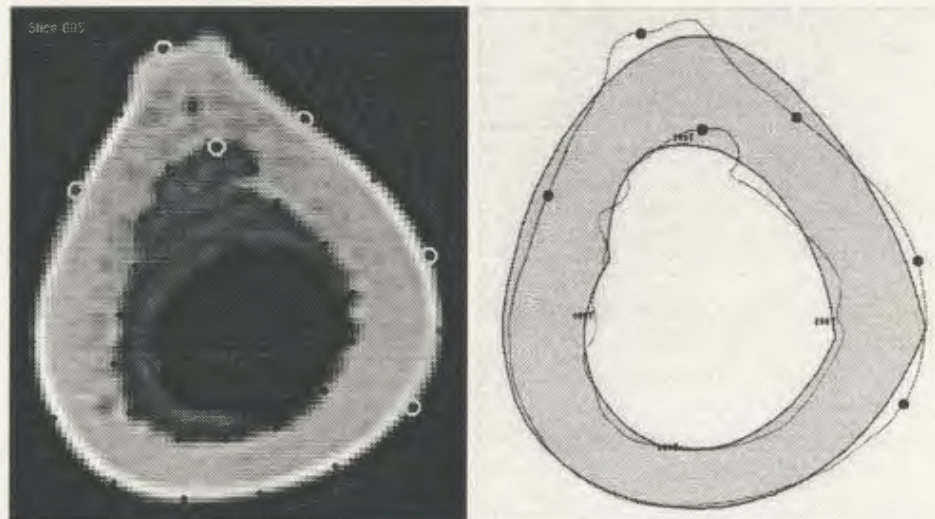
4.6   Limitations of the Modelling Software

The developed software modules are capable of defining the cortical edges using any specified number of points. Unfortunately, modelling capabilities were limited because the ANSYS finite element modelling software being used by Memorial University's Centre for Computer Aided Engineering is the version approved for educational use (ANSYS/ED), which imposes certain restrictions on its users. Using this version, models can have a maximum of 10,000 nodes, which limits the size of the model that can be solved. Testing, conducted by engineering graduate student Paul Smith, has shown that, with the present restrictions, the largest model of a femur that can be generated is one using 32 outer edge points and 16 inner edge points. As a result, no attempt has been made to generalize the software to model using more than 32 points per edge. If the full ANSYS package is used, the software should be modified to allow more detailed modelling.

4.7 General Comments

These results have been presented merely to illustrate the potential value of this software. The test objects for this study, human femurs, are not easily obtained, and are made available to the BEC only once a year. Local CT scanners are generally not used for research purposes, and tests must be conducted as clinic schedules permit. Each of these conditions tends to limit the amount of data available to researchers in the BEC. As such, the software has only been tested on one femur shaft, one complete femur, and one seal baculum, making the test base far too small to draw any valid conclusions. The results must be interpreted with caution, but overall, are very encouraging, and indicate that the automated models are equal, or superior, to those generated using the manual method. The accuracy of the 16 point models is comparable, but the 16pt Auto model is favoured in terms of time savings. The 32pt Auto model is superior in all respects.

There is little doubt that the accuracy of the results was affected by the scanning techniques used. The 5mm slice width may be sufficient to model the relatively uniform femur shaft, but is too wide to accurately reflect the complex geometry of the head of the femur. Unfortunately, technical restrictions limit the amount of data that can be transferred to researchers in the BEC. The local hospitals have limited data transfer capabilities, and 3½" floppy disks must be used to transfer the data in Toshiba format. With each slice requiring 500Mb of storage space, an average femur scan using 5mm slice intervals requires 50

diskettes, and hospital CT technicians must be willing to copy the data to these diskettes for transfer. While the technicians have obliged until now, it is felt that decreasing the slice interval to 3mm or even 1mm, would be too much of an imposition on their time. Until more efficient data transfer links are established, full femurs must be scanned at a minimum of 5mm slice intervals, reserving smaller intervals for smaller test objects.

CHAPTER 5

CONCLUSIONS

Practical application software has been developed for use in the Biomedical Engineering Centre which automates the generation of 3D computer models of the human femur. The software consists of a series of computer programs which assist the user in building 3D solid models of the human femur. The programs perform the following tasks:

A) Transfer and Rename Data - Raw CT data is copied from diskette to a processing computer, and each file is assigned a unique name as an identifier.

B) Translate and Crop Data - Raw data files are read, translated into a compatible format, and cropped to exclude all extraneous data other than bone.

C) Detect Inner and Outer Edges of Cortical Bone - Using preselected values, a thresholding technique is used to define the perimeter edges of the cortical bone.

D) Sort Edge Data To Indicate Connectivity Between Points - The stored perimeter data points are converted from raster format and sorted, linking adjacent edge points

93

within the data file.

E) Select User Specified Number of Points From The Edges - A user-specified number of
points is taken from the edges by selecting equidistant points along the perimeters.

F) Generate ANSYS Input File - With the keypoints selected, an ANSYS input file is
automatically generated to plot keypoints, draw splines, skin over areas and generate
a solid volume model of the femur.

Other supporting programs allow the user to edit and display edge points, and manipulate CT
image data.

The software was developed using computer resources available in the BEC and CCAE.
Much of the research being done elsewhere in this area has relied on the use of dedicated and
specialized hardware systems and advanced software packages. This research project differs
from most in that no specialized CT image processing systems have been used, and much of
the processing can be done on a personal computer.

Due to the small number of femur specimens available, software testing has been limited.
Further testing must be performed on more bone specimens to provide a larger test base from
which to draw stronger conclusions. The preliminary results are encouraging though, and
suggest that the new modelling software will generate models which are comparable to, if
not better than, those produced manually. The ability to easily place a greater number of

94

keypoints will produce smoother, more accurate models than those generated using the manual method. The thresholding and contour detection algorithms used have worked well on the existing test data, but further testing is needed to prove the software robust for all cases.

While user input is required, the automated model building process is much less labour intensive than the former manual method, and the learning curve required to construct models has been shortened. This software allows novice users to build 3D femur models from raw CT data with minimal training, in an expeditious manner. The new software will permit anyone to construct a preliminary ANSYS finite element model after reading the users guide which accompanies the software. Much of the tedium has been eliminated from the manual method, and reduced user-involvement has made the process less prone to human error. Automatic edge detection and keypoint selection make routine generation of finite element models feasible.

Although developed for modelling femurs, this same methodology and software can be used to help model other complex geometries, and has proved useful in modelling orthopaedic structures other than femurs.

CHAPTER 6

RECOMMENDATIONS

Further work is needed in three areas - improvement in the CT scanning techniques used in this type of research, software enhancements, and additional testing.

6.1   Improvement in CT Scanning Techniques

The BEC has not adopted a standard procedure to be used when scanning *in vitro* bone specimens. As only five scans have been conducted to date, this has not been an area for concern. Each specimen has been mounted between platens, or in a plexiglass mounting frame and scanned surrounded by air. No calibration tools have been used. Image quality tests were not performed as part of this research, but clearly, there are issues relating to image quality. What effect did scanning in air have on image quality? Do the materials used in the mounting frame affect the CT numbers? Several slices of the CT imagery were

adversely affected by outside sources. Where CT image quality is important, the literature in this area strongly recommends the use of water bath scanning, and calibration tools [8,23]. Image quality is also affected by other scanning parameters, such as slice thickness and slice interval [38]. Severe image artifacts were also introduced by the metallic platens used to hold each end of the femur. This prevented acquisition of accurate CT slice data in these regions. As a result, alternate non-metallic platen materials should be investigated to permit full femur scans.

The importance of correct CT data to the 3D modelling procedure cannot be underestimated, and as such, every effort should be made to ensure that the highest quality images are being obtained, not just to improve edge detection, but also to ensure that the CT numbers are not being adversely affected by improper scanning techniques.

6.2   Software Enhancements

The purpose of this project was to develop convenient software to take data from CT scans for the generation of 3D computer models. This has been accomplished, but there is always room for improvement and software enhancement. There are several areas that should be addressed in the future.

1)   All software components should be consolidated into a single package. At present,

97

several different programming languages and packages have been used, and a certain user familiarity is required to generate a model from start to finish. A graphical user interface (GUI) should be developed to allow even the novice user to easily use the software.

2) The CT data decoding routine was written in FORTRAN 77, and was developed quickly to test the methodology. Code optimization was not a primary concern, and it is expected that the routine would run more quickly if written in machine language. An even better solution to this problem would be to simplify the decoding procedure by using existing data conversion tools. Alternate or more efficient methods of converting the CT data from Toshiba format should be investigated. The ability of other image processing packages, such as OSIRIS, to display the data, or a close approximation, indicates that the byte-swapping format used by Toshiba is known. Unfortunately, the source code for these programs is unavailable, and none of the programs have tools providing direct access to the CT numbers. With further investigation, another off-the-shelf image processing package may be able to both read and offer the user access to the data, eliminating the need for this step in the preprocessing. It would be ideal if Toshiba would provide its own decoding routine.

3) Edge detection threshold selection should be automated. The threshold might be calculated based on histogram levels [7], or found by searching for maximum values which provide contour closure [34].

4) Edge detection software should be made more robust. The software has not had any

98

difficulties with the test data, but potential problems can be foreseen that might cause difficulties, for example, infinite looping while connecting the edges. Further testing is required.

5)   If the entire femur is to be modelled, the software must be modified to address the problem of model bifurcation at each end of the femur.

6)   Automatic area and volume calculation tools, and a method for aligning images produced from different modalities would be useful for validation purposes.

6.3   Additional Testing

As mentioned, the test base for this study was very small. More extensive testing of both the software and the methodology are required, and this can come only as more specimens are obtained. As more user testing is done, the software can be modified as experience may indicate. The effects of varying different parameters, such as threshold levels, slice intervals, and number of points, need further study to ensure that the generated models are as accurate as possible. In addition, the results of this study have raised questions about the effects of splining through keypoints and then skinning over these lines on model accuracy. The splining and skinning options within ANSYS should be investigated.

CHAPTER 7

REFERENCES

[1]   J.E. Adams et al., "Measurement of Trabecular Bone Mineral by Dual Energy Computer Tomography," *Journal of Computer Assisted Tomography*, Vol. 6, no. 3, June 1982, pp. 601-607.

[2]   Ehud Artzy, Gideon Frieder, and Gabor T. Herman, "The Theory, Design, Implementation and Evaluation of a Three-Dimensional Surface Detection Algorithm," *Computer Graphics and Image Processing*, Vol. 15, 1981, pp.1-24.

[3]   Kyongtae T. Bae et al., "Automatic Segmentation of Liver Structure in CT Images," *Medical Physics*, Vol. 20, No. 1, January/February 1993, pp. 71-78.

[4]   F.A. Bandak et al., "An Imaging-Based Computational and Experimental Study of Skull Fracture: Finite Element Model Development," *Journal of Neurotrama*, Vol. 12, No. 4, August 1995, pp. 679-688.

[5]   D.L. Bartel, D.L. Taylor, and M.A. McCarthy, "Structural Analysis of Bone-Implant Systems: The Effects of Modeling" in *Computational Methods in Bioengineering*, ed. R.I. Spiker and B.R. Simon, June 1988, pp. 199-210.

[6]   P.K. Basu et al., "3-D Femoral Stress Analysis Using CT Scans and P-Version FEM," *Biomaterials, Medical Devices and Artificial Organs*, Vol.13, Nos. 3 & 4, 1985-86, pp. 163-186.

[7]   K.L. Bell et al., "Cortical and Cancellous Bone in the Human Femoral Neck: Evaluation of an Interactive Image Analysis System," *Bone*, Vol. 19, No. 5, November 1996, pp. 541-548.

[8]   G.A. Binder, V.M. Haughton, and K.C. Ho, "Computer Tomography of Anatomic Specimens," *Journal of Computer Assisted Tomography*, Vol. 2, September 1978, pp. 506-507.

[9]   Commean, et al., "Finite Element Modeling and Experimental Verification of Lower Extremity Shape Change Under Load," *Journal of Biomechanics*, Vol. 30, No. 5, May 1997, pp. 531-536.

[10]   B. Cortet et al., "Methods for Quantitative Analysis of Trabecular Bone Structure," *Revue Du Rhumatisme (English Edition)*, Vol. 62. No. 11. December 1995, pp. 781-793.

[11]   E. P. Durand and P. Rüegsegger, "Cancellous Bone Structure: Analysis of High Resolution CT Images with the Run-Length Method," *Journal of Computer Assisted Tomography*, Vol. 15, No. 1, January/February 1991, pp. 133-139.

[12]   Samuel J. Dwyer III, et al., "Medical Image Processing in Diagnostic Radiology," *IEEE Transactions on Nuclear Science*, Vol. NS-27, No. 3, June 1980, pp. 1047-1055.

[13]   K.H. Englmeier et al., "Methods and Applications of Three-Dimensional Imaging in Orthopedics," *Archives of Orthopaedic and Trauma Surgery*, Vol. 109, No. 4, 1990 pp. 186-190.

[14]   B. Farman-Ara et al., "Edge Detection in Brain SPET Perfusion Imaging: A Comparison of Several Methods," *Nuclear Medicine Communications*, Vol. 18, No. 12. December 1997, pp. 1122-1129.

[15]   H. Freeman, "Computer Processing of Line-Drawing Images," *ACM Computer Surveys*, Vol. 6, No. 1, March 1974, pp. 57-97.

[16]   H. Fuchs, Z.M. Kedem, and S.P. Uselton, "Optimal Surface Reconstruction from Planar Contours," *Communications of the ACM*, Vol. 20, No. 10, October 1977, pp. 693-702.

[17]   J.X. Gao and L. Waite, "Patellofemoral Joint Study via Image Processing," *Proceedings of the International ISA Biomedical Sciences Instrumentation Symposium*, No. 32, 1996, pp. 151-160.

[18]   L.M. Gutkowski, D.D. Raftopoulos, and G. Williams, "Computer Techniques for In Vivo Determination of Geometric Properties of Human Femur and Tibia," *Medical and Biological Engineering and Computing*, Vol. 33, May 1995, pp. 341-347.

[19]   M.A. Hadiekker et al., "Evaluation of the Cortical Structure in High Resolution CT Images of Lumbar Vertebrae by Analysing Low Bone Mineral Density Clusters and Cortical Profiles," *The British Journal of Radiology*, Vol. 70, December 1997, pp. 1222-1228.

[20]   G.T. Herman, and H.K. Liu, "Dynamic Boundary Surface Detection," *Computer Graphics and Image Processing*, Vol. 7, 1978, pp. 130-138.

[21]   W.E. Higgins and R.D. Swift, "Distributed System for Processing 3D Medical Images," *Computers in Biology and Medicine*, Vol. 27, No. 2, March 1997, pp. 97-115.

[22]   "Hip Fracture Solution Guide," Internet page www.howmedica.com /products/catalog.htm. Howmedica Inc., Pfizer Medical Technology Group, ©1997.

[23]   G.N. Hounsfield, "Computerized Transverse Axial Scanning (Tomography): Part I. Description of System," *British Journal of Radiology*, Vol. 46, No. 552, December 1973, pp. 1016-1022.

[24]   J.L. Humm, et al., "Image Analysis for the Study of Radionuclide Distribution in Tissue Sections," *The Journal of Nuclear Medicine*, Vol. 35, No. 7, July 1994, pp. 1217-1225.

[25]   Y.K. Kang et al., "Three Dimensional Shape Reconstruction and Finite Element Analysis of Femur Before and After the Cementless type of Total Hip Replacement," *Journal of Biomedical Engineering*, Vol. 15, No. 6, November 1993, pp. 497-504.

[26]   J.H. Keyak et al., "Automated Three-Dimensional Finite Element Modelling of Bone: A New Method," *Journal of Biomedical Engineering*, Vol. 12, No. 5, September 1990, pp. 389-397.

[27]   J.L. Kuhn et al., "Evaluation of a Microcomputed Tomography System to Study Trabecular Bone Structure," *Journal of Orthopaedic Research*, Vol. 8, No. 6, November 1990, pp. 833-842.

[28]   T.F. Lang et al., "Volumetric Quantitative Computed Tomography of the Proximal Femur: Precision and Relation to Bone Strength," *Bone*, Vol. 21, No. 1, July 1997, pp. 101-108.

[29]   Jeffrey C. Lotz, Tobin N. Gerhard, and Wilson C. Hayes, "Mechanical Properties of Trabecular Bone from the Proximal Femur: A Quantitative CT Study," *Journal of Computer Assisted Tomography*, Vol. 14, No. 1, January/February 1990, pp. 107-114.

[30]   Olivia Louis et al., "Size of Cortical Bone and Relationship to Bone Mineral Density Assessed by Quantitative Computed Tomography Image Segmentation," *Investigative Radiology*, Vol. 28, No. 9, September 1993, pp. 802-805.

[31]   Shlomo A. Marom and Martin J. Linden, "Computer Aided Stress Analysis of Long Bones Utilizing Computed Tomography," *Journal of Biomechanics*, Vol. 23, No. 5, May 1990, pp. 399-404.

[32]   Jeffrey L. Marsh, and Michael W. Vannier, "Surface Imaging From Computerized Tomographic Scans," *Surgery*, Vol. 94, No. 2, August 1983, pp. 159-165.

[33]   Jeffrey L. Marsh, and Michael W. Vannier, "The 'Third' Dimension in Craniofacial Surgery," *Plastic and Reconstructive Surgery*, Vol. 71, No. 6, June 1983, pp. 759-767.

[34]   Michael F. McNitt-Gray et al., "Development and Testing of Image-Processing Methods for the Quantitative Assessment of Airway Hyperresponsiveness from High-Resolution CT Images." *Journal of Computer Assisted Tomography*, Vol. 21, No. 6, November 1997, pp. 939-947.

[35]   Bhavin V. Mehta, Sailesh Rajani, and Guatam Sinha, "Comparison of Image Processing Techniques (Magnetic Resonance Imaging, Computed Tomography Scan and Ultrasound) for 3D Modeling and Analysis of the Human Bones," *Journal of Digital Imaging*, Vol. 10, No. 3, Suppl.1, August 1997, pp. 203-206.

[36]   R. Müller et al., "Morphometric Analysis of Noninvasively Assessed Bone Biopsies: Comparison of High-Resolution Computed Tomography and Histologic Sections," *Bone*, Vol. 18, No. 3, March 1996, pp. 215-220.

[37]   R. Müller and P. Rüegsegger, "Three-Dimensional Finite Element Modelling of Non-Invasively Assessed Trabecular Bone Structures," *Medical Engineering and Physics*, Vol. 17, No. 2, March 1995, pp. 126-133.

[38]   Derek R. Ney et al., "Three-Dimensional Volumetric Display of CT Data: Effect of Scan Parameters upon Image Quality," *Journal of Computer Assisted Tomography*, Vol. 15, No. 5, September 1991, pp. 875-885.

[39]   S.H. Ong, et al., "Image Analysis of Tissue Sections," *Computers in Biology and Medicine*, Vol. 26, No. 3, May 1996, pp. 269-279.

[40]   A. Pepino et al., "Preliminary Notes on the Conversion of Three-Dimensional Images of Bones into CAD Structures for the Pre-Operative Planning of Intertrochanteric Osteotomies," *Medical and Biological Engineering and Computing*, Vol. 31, No. 5, September 1993, pp. 529-534.

[41]   F. Peyrin et al., "3D Display of High Resolution Vertebral Structure Images," *Computerized Medical Imaging and Graphics*, Vol. 17, Nos. 4/5, July 1993, pp. 251-256.

[42]   P.J. Rubin et al., "The Morphology of the Proximal Femur: A Three-Dimensional Radiographic Analysis," *The Journal of Bone and Joint Surgery [British]*, Vol. 74-B, January 1992, pp. 28-32.

[43]   Tamas Sandor et al., "Compact and Trabecular Components of the Spine Using Quantitative Computed Tomography," *Calcified Tissue International*, Vol. 50, No. 6, June 1992, pp. 502-506.

[44]   David J. Sartoris et al., "Trabecular Bone Density in the Proximal Femur: Quantitative CT Assessment," *Radiology*, Vol. 160, September 1986, pp. 707-712.

[45]   Daniel S. Schlusselberg et al., "Use of Computed Tomography for a Three-Dimensional Treatment Planning System," *Computerized Medical Imaging and Graphics*, Vol. 12, No. 1, January 1988, pp. 25-32.

[46]   J. Schmitt et al., "Die Anwendung voxelorientierter Femurmodelle zur Spannungsanalyse [The Use of Voxel-Oriented Femur Models in Stress Analysis]," *Biomedizinische Technik*, Vol. 40, Iss. 6, June 1995, pp. 175-181, (Translation).

[47]   J.D. Schwardt et al., "Structural and Mechanical Properties of Cancellous Bone from 3D Magnetic Resonance Imaging and Finite Element Models," *Journal of Bone and Mineral Research*, Vol. 11, No. S1, 1996, p.T702, (Abstract).

[48] David Siu et al., "Femoral Articular Shape and Geometry: A Three-dimensional Computerized Analysis of the Knee," *The Journal of Arthroplasty*, Vol. 11, No. 2, February 1996, pp. 166–173.

[49] Howard W. Smith, Arthur A. De Smet, and Errol Levine, "Measurement of Cortical Thickness in a Human Cadaver Femur: Conventional Roentgenography versus Computed Tomography," *Clinical Orthopaedics and Related Research*, No. 169, September 1982, pp. 269-274.

[50] Sargur N. Srihari, "Representation of Three-Dimensional Digital Images," *Computing Surveys*, Vol. 13, No. 4, December 1981, pp. 399-424.

[51] Dale R. Sumner et al., "Computed Tomography and Automated Image Analysis of Prehistoric Femora," *American Journal of Physical Anthropology*, Vol. 68, October 1985, pp. 225-232.

[52] D. R. Sumner et al., "Computed Tomographic Measurement of Cortical Bone Geometry," *Journal of Biomechanics*, Vol. 22, No. 6/7, June/July 1989, pp. 649-653.

[53] Charles R. Sutherland, Stephen J. Bresina, and Donald E. Gayou, "Use of General Purpose Mechanical Computer Assisted Engineering Software in Orthopaedic Surgical Planning: Advantages and Limitations," *Computerized Medical Imaging and Graphics*, Vol. 18, No. 6, November 1994, pp. 435-442.

[54] Jayaram K. Udupa, "Display of 3D Information in Discrete 3D Scenes Produced by Computerized Tomography," *Proceedings of the IEEE*, Vol. 71, No. 3, March 1983, pp. 420-431.

[55] J. Van Cleynenbreugel et al., "A Semiautomatic Three-Dimensional Segmentation Method for Disarticulation of Bone Structures on Spiral Computed Tomography Images," *Journal of Digital Imaging*, Vol. 8, No. 4, November 1995, pp. 156-161.

[56] Michael W. Vannier, Jeffrey L. Marsh, and James O. Warren, "Three Dimensional CT Reconstruction Images for Craniofacial Surgical Planning and Evaluation," *Radiology*, Vol. 150, January 1984, pp. 179-184.

[57] Michael W. Vannier, and Jeffrey L. Marsh, "Three-Dimensional Imaging, Surgical Planning, and Image-Guided Therapy," *Radiologic Clinics of North America*, Vol. 34, No. 3, May 1996, pp. 545-563.

[58]   Santosh G. Zachariah, Joan E. Sanders, and George M. Turkiyyah, "Automated Hexahedral Mesh Generation from Biomedical Image Data: Applications in Limb Prosthetics," *IEEE Transactions on Rehabilitation Engineering*, Vol. 4, No. 2, June 1996, pp. 91-102.

APPENDICES

APPENDIX A

Transfer of CT data from diskette to BEC computers

The hospital's CT computer system copies the data files to diskette using the same name for each file it copies, so the first step in processing the data was to copy the data from diskette to a network PC, assigning unique names to each of the image slices. The naming conventions were as follows:

Table A.1 File name conventions	
Toshiba Output Filename	**Renamed As**
P001i001.img	ct005005.img
P001i002.img	ct005010.img
P001i001.inf	ct005005.inf
P001i002.inf	ct005010.inf
dirdef.tbl	ct005005010.tbl
nicsfdf.ctl	ct005005010.ctl
nicsfdf.dir	ct005005010.dir

Image files were named 'ct005###.img', where ### was the slice position in mm. The 005 was the identifying, assigned scan number.

Once all data was transferred and renamed, an archive copy was retained on more easily

managed media, such as a 100MB Zip™ disk, and the 3½" diskettes were reformatted for future use. As the data was to be processed and fed into the ANSYS finite element modelling software package, the CT data files were copied to the CCAE's UNIX system, where the ANSYS software modules and other modelling and image processing packages are available.

APPENDIX B

Format of control file

```
001     SCAN NUMBER (Slices range from 005 to 145)
005     START SLICE
065     END SLICE
5       INCREMENT BETWEEN SLICE NUMBERS
511     TOP OF BONE MOUNT (Row #)
0450    B/M THRESHOLD
0000    A/B THRESHOLD
5       PAD
32      NUMBER OF INNER EDGE POINTS TO SELECT
32      NUMBER OF OUTER EDGE POINTS TO SELECT
/users/biomedical/ctscan/          ROOT DIRECTORY
.img            RAW SCAN DATA
.hdr            REGIONS OF INTEREST HEADER FILE
.roi            REGIONS OF INTEREST
.ied            BINARY IMAGE OF INNER EDGE
.oed            BINARY IMAGE OF OUTER EDGE
.ico            LIST OF CONNECTED INNER EDGE POINTS
.oco            LIST OF CONNECTED OUTER EDGE POINTS
.pts            SELECTED INNER AND OUTER EDGE POINTS
.ans            GENERATED ANSYS INPUT FILE
```

APPENDIX C

Detailed Graphical Results

Figure C.1   Slice 005 False Positive Areas for each model
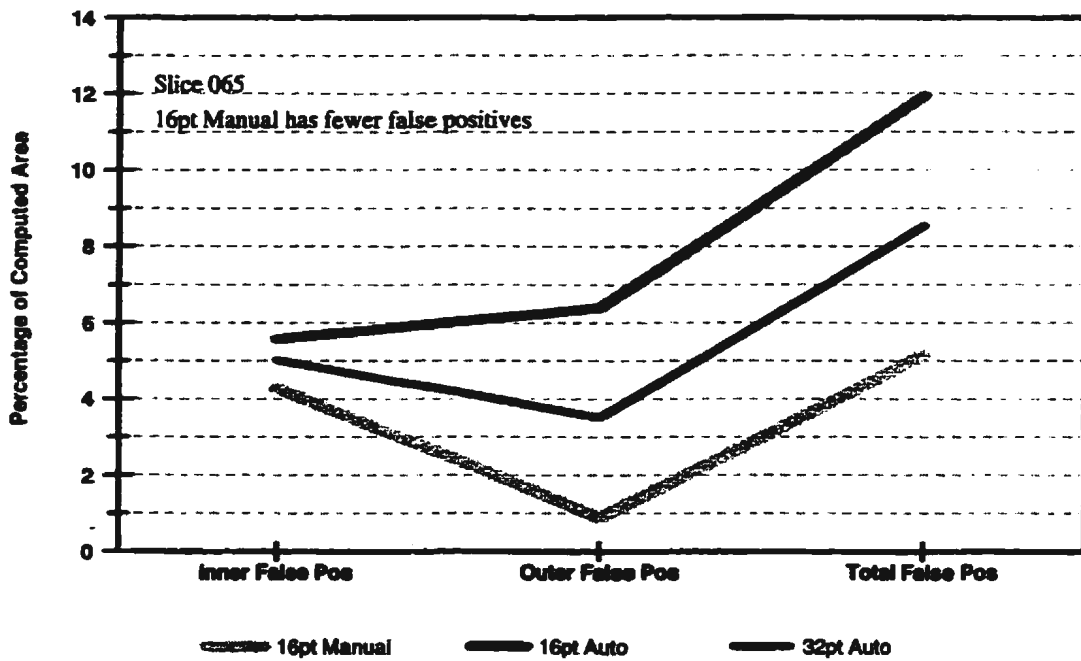
Figure C.2   Slice 065 False Positive Areas for each model
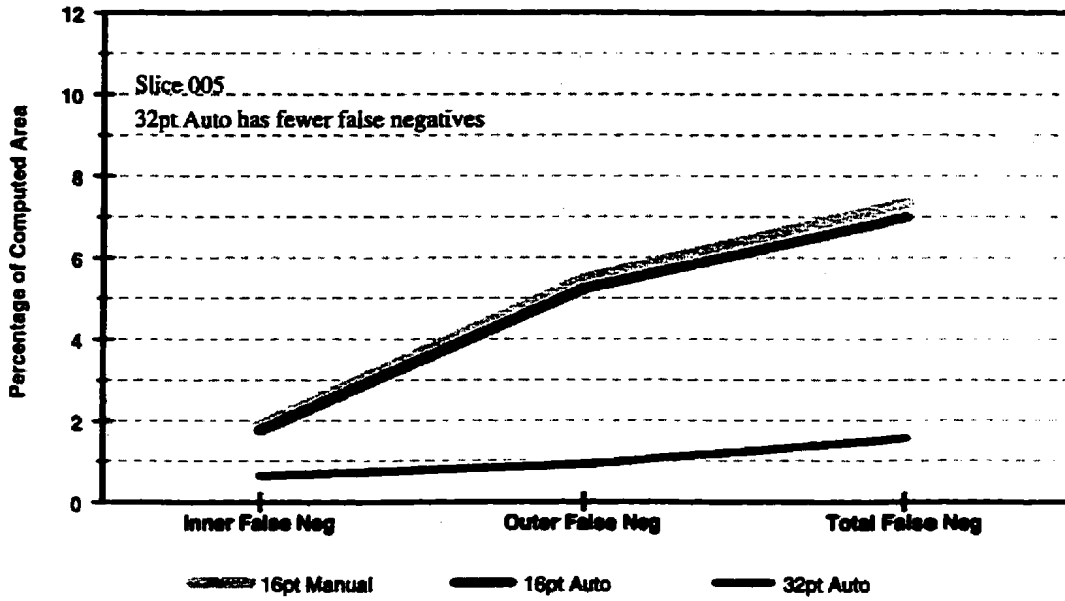
112

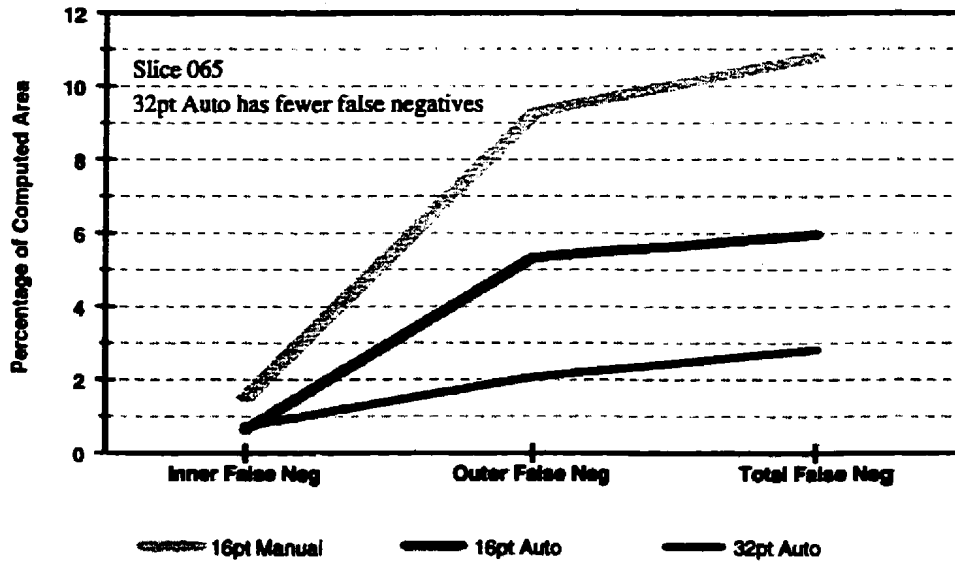**Figure C.3   Slice 005 False Negative Areas for each model**
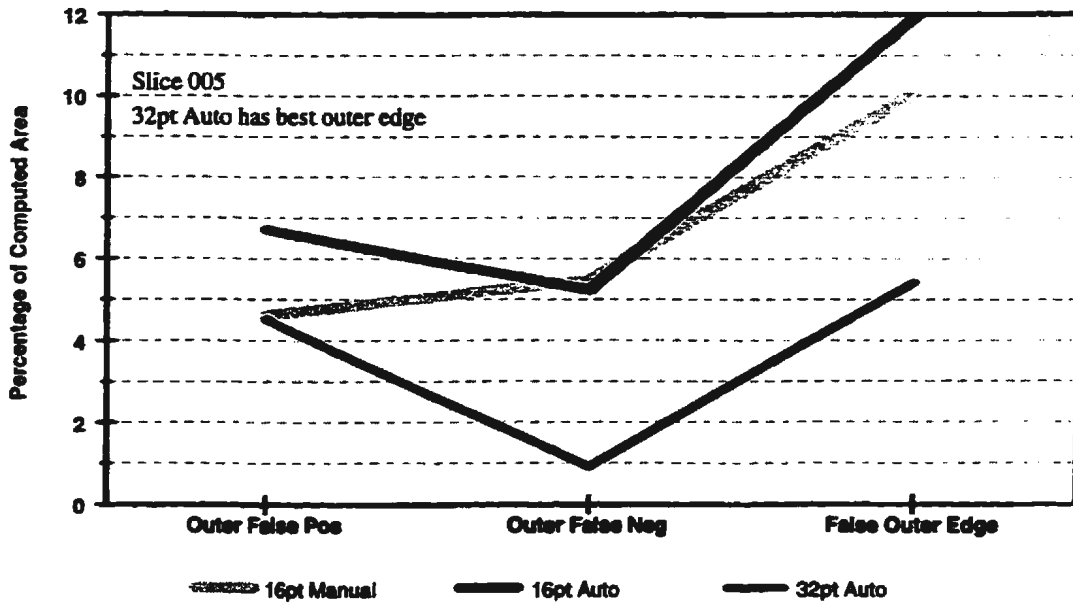

Figure C.4   Slice 065 False Negative Areas for each model

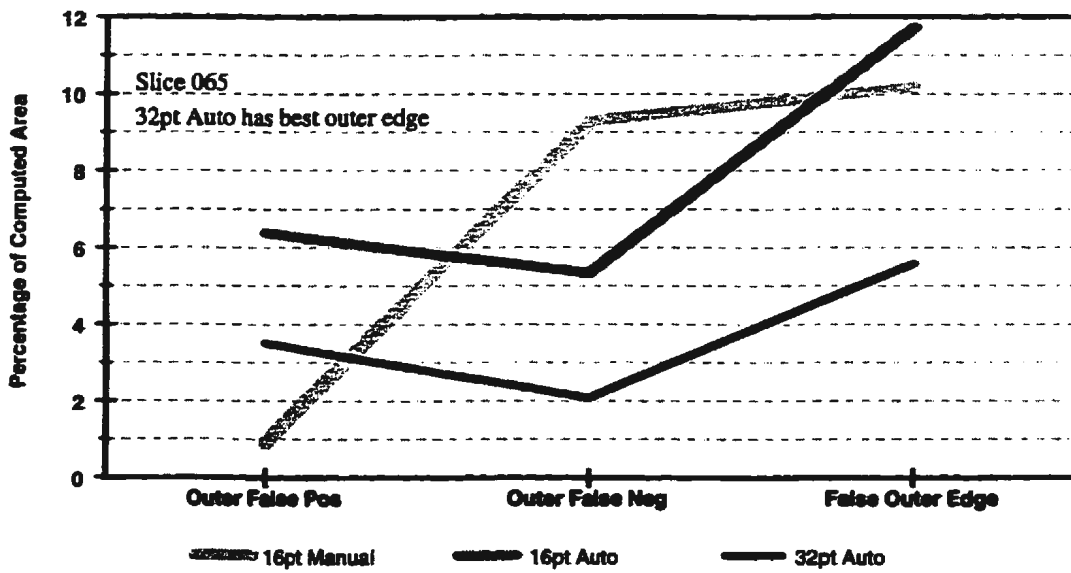Figure C.5   Slice 005 False Outer Edge Areas for each model

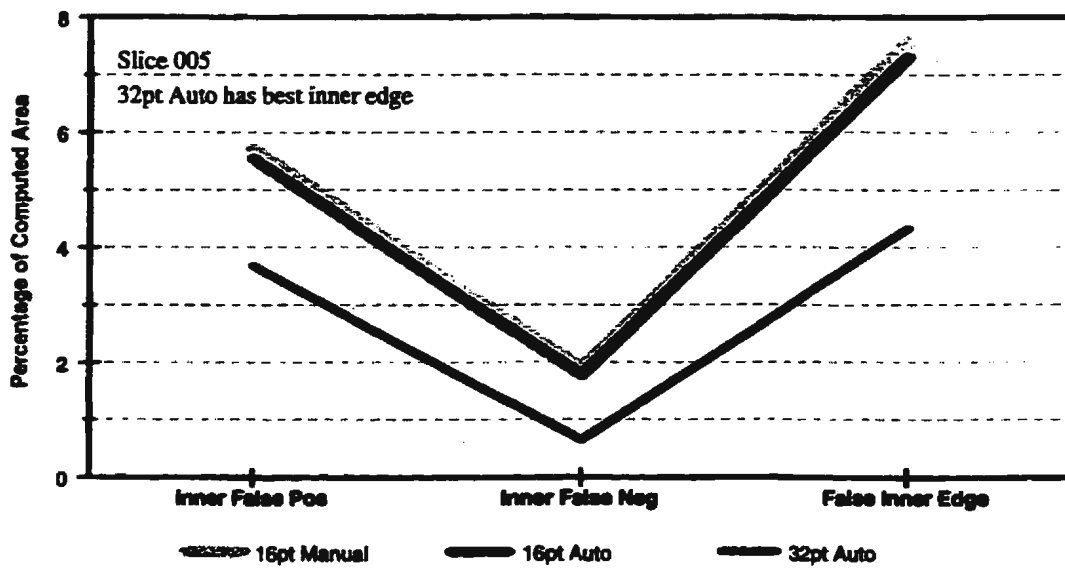Figure C.6   Slice 065 False Outer Edge Areas for each model

114

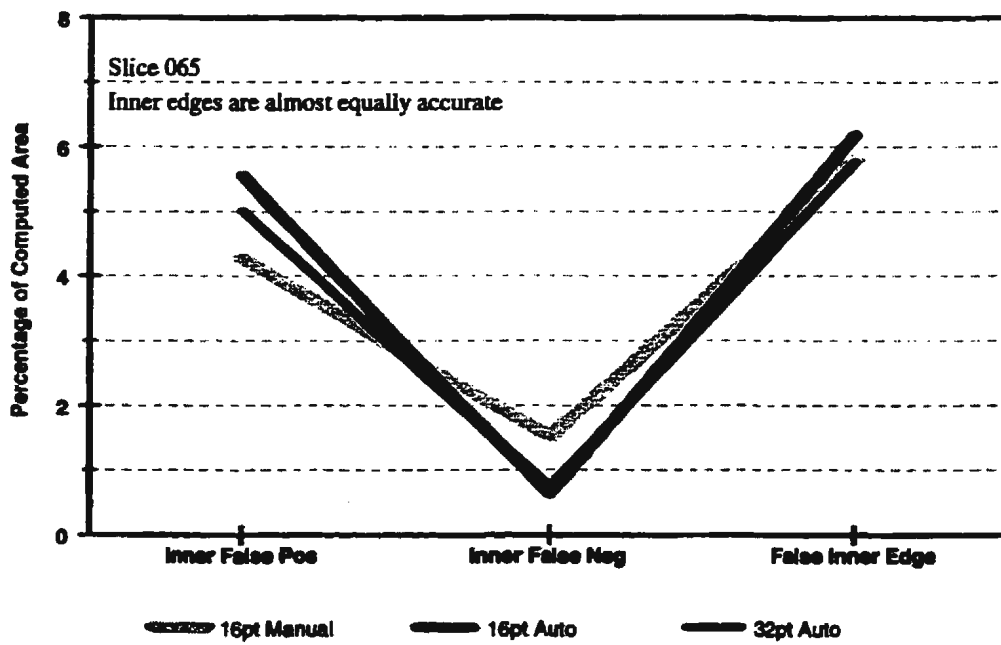Figure C.7   Slice 005 False Inner Edge Areas for each model

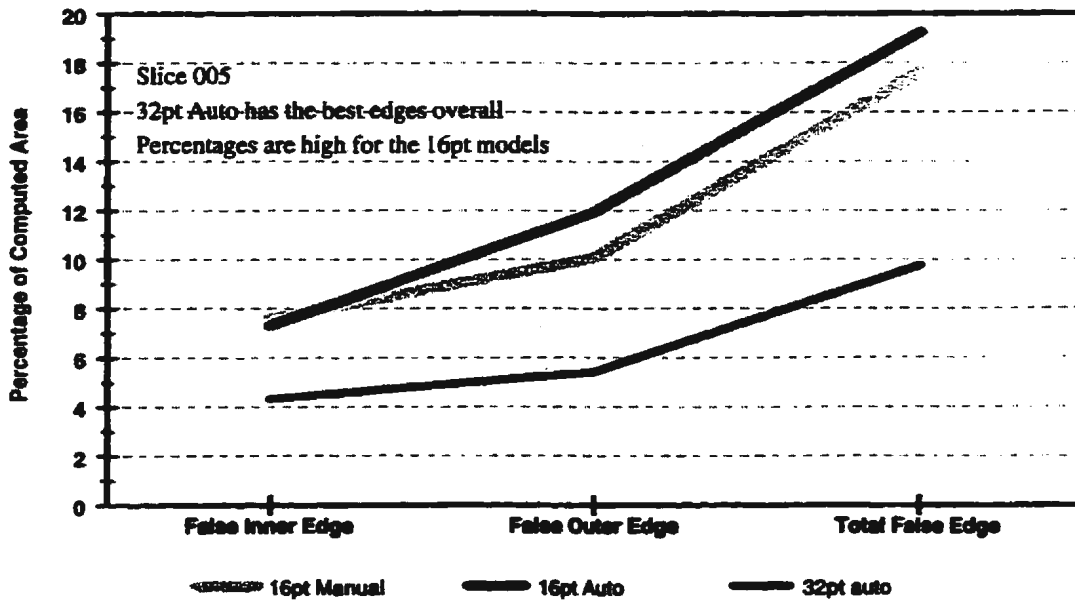Figure C.8   Slice 065 False Inner Edge Areas for each model
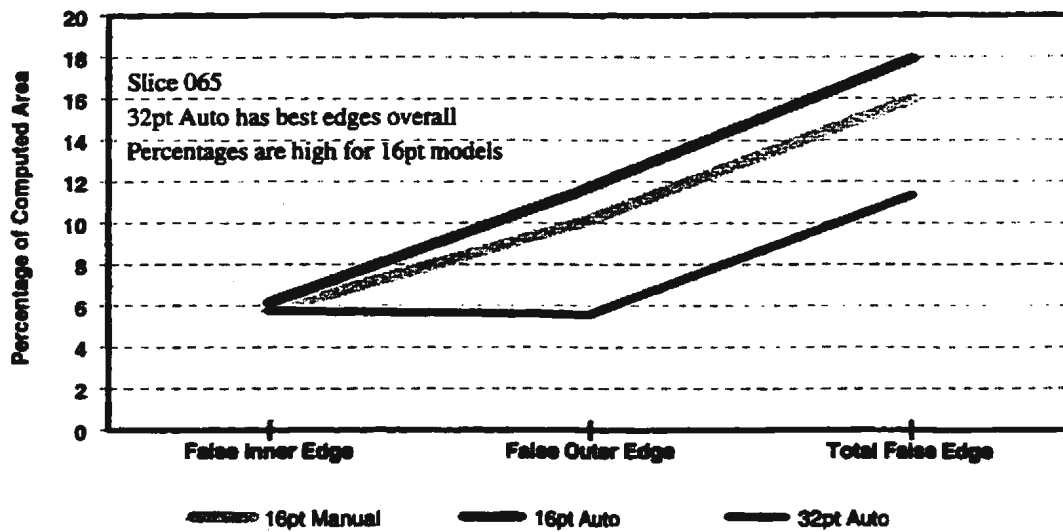
Figure C.9   Slice 005 False Edge Areas for each model

Figure C.10   Slice 065 False Edge Areas for each model

116

BIBLIOGRAPHY

Akiba, H., et al., "Three Dimensional CT Reconstruction Image Using the Personal Computer," *Rinsho Hoshasen. Japanese Journal of Clinical Radiology*, Vol. 35, No. 4, April 1990, pp. 475-481.

Berland, L.L., *Practical CT Technology and Techniques*, Raven Press, New York, 1987.

Cann, C.E., "Quantitative CT for Determination of Bone Mineral Density: A Review," *Radiology*, Vol. 166, No. 2, February 1988, pp. 509-522.

Case, J., L. Chilver, and C.T.F. Ross, *Strength of Materials and Structures with an Introduction to Finite Element Methods*, 3rd ed., Edward Arnold, London, 1993.

Cristofolini, L., et al., "A Minimal Parametric Model of the Femur to Describe Axial Elastic Strain in Response to Loads," *Medical Engineering Physics*, Vol. 18, No. 6, September 1996, pp. 502-514.

Elke, R.P.E., et al., "Three-Dimensional Anatomy of the Cancellous Structures within the Proximal Femur from Computed Tomography Data," *Journal of Orthopaedic Research*, Vol. 13, No. 4, July 1995, pp. 513-523.

Field, R.E., et al., "Bone Density Distribution Within the Femoral Head and Neck: An Examination by High-Resolution Computer Tomography," *Skeletal Radiology*, Vol. 19, No. 5, 1990, pp. 319-325.

Gonzalez, R.C., and R.E. Woods, *Digital Image Processing*, Addison-Wesley Publishing Company, New York, 1992.

Gross, A.E., "The Arthritic Hip," *Hip Hip Hooray Journal*, The Canadian Orthopaedic Foundation, Toronto, Spring 1998, pp. 16-21.

Hemmy, D.C., et al., "A Decade of Clinical Three-Dimensional Imaging: A Review. Part I. Historical Development," *Investigative Radiology*, Vol. 29, No. 4, April 1994, pp. 489-496.

Keyak, J.H., et al., "Validation of An Automated Method of Three-Dimensional Finite Element Modelling of Bone," *Journal of Biomedical Engineering*, Vol. 15, No. 6, November 1993, pp. 505-509.

Keyak, J.H., and H.B. Skinner, "Three-Dimensional Finite Element Modelling of Bone: Effects of Element Size," *Journal of Biomedical Engineering*, Vol. 14, No. 6, November 1992, pp. 483-489.

Lee, R.L., J.E. Dacre, and M.F. James, "Image Processing Assessment of Femoral Osteopenia," *Journal of Digital Imaging*, Vol. 10, No. 3,, Suppl. 1, August 1997, pp. 218-221.

Levinthal, C. and R. Ware, "Three Dimensional Reconstruction from Serial Sections," *Nature*, Vol. 236, March 31, 1972, pp. 207-210.

Rho, J.Y., M.C. Hobatho, and R.B. Ashman, "Relations of Mechanical Properties to Density and CT Numbers in Human Bone," *Medical Engineering and Physics*, Vol. 17, No. 5, July 1995, pp. 347-355.

Thompson, C.M., and L. Shure, *Image Processing Toolbox For Use with MATLAB*, The MathWorks, Inc., Natick, Mass., 1993.

Vander Sloten, J., et al., "The Development of a Physiological Hip Prosthesis: Evaluation of the Strains After Implantation of a Prototype of Hip Implant: Experiment in a Dry Femur," *Bio-Medical Materials and Engineering*, Vol. 3, No. 1, Spring 1993, pp. 1-13.

Weinstein, M., and K.R. Castleman, "Reconstructing 3-D Specimens From 2-D Section Images," *Proceedings of the Society of Photo-Optical Instrumentation Engineers*, Vol. 26, May 1971, pp. 131-138.

Woolson, S.T., et al., "Three-dimensional Imaging of Bone from Computerized Tomography," *Clinical Orthopaedics and Related Research*, No. 202, January 1986, pp. 239-248.