

**The Design and Implementation of a
Three Dimensional Computerized
Treatment Planning System**

by

Roch Comeau

A Thesis Submitted to the
Faculty of Graduate Studies and Research
in Partial Fulfillment of the requirements for the Degree of

Master of Science

Medical Physics Unit
McGill University, Montreal
January, 1993

©Roch Comeau, 1992, 1993



November 16, 1992

Mr. Roch Comeau
Medical Physics Unit
McGill University
1650 Cedar Avenue
Montreal, Quebec
Canada H3G1A4

Dear Mr. Comeau:

I am responding to your letter of October 20. Please feel free to reproduce figure 9a & b from my publication "Arbitrary Oblique Image Sections for 3-D Radiation Treatment Planning".

Thank you for your interest in my paper.

Sincerely,

Radhe Mohan, Ph.D.

UNIVERSITY OF WASHINGTON

SEATTLE WASHINGTON 98195

*School of Medicine and University Hospital
Department of Radiation Oncology RC-08*

Divisions

CLINICAL RADIATION ONCOLOGY
MEDICAL RADIATION PHYSICS
EXPERIMENTAL BIOLOGY

November 13, 1992

Roch Comeau
Medical Physics Unit
McGill University
1650 Central Ave.
Montreal, Quebec
Canada
H3G 1A4

Dear Mr. Comeau:

This is in response to your request to reproduce in your thesis Figure 2 from my review article, "3-D Radiation Treatment Planning: Overview and Assessment." The article originally appeared in *American Journal of Clinical Oncology (Cancer Clinical Trials)* volume 13, number 4, pages 331 - 343 1990.

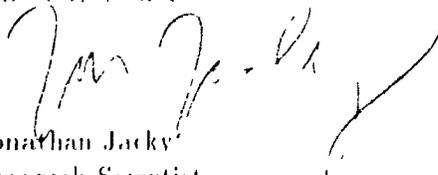
It is fine with me if you use the figure. However, I believe I am not the person you need to ask for permission. I believe you must get permission from the publisher of the journal. I signed an assignment of copyright to Raven Press, and my reprint of the article is clearly marked "Copyright 1990 Raven Press Ltd., New York". My correspondence regarding the copyright was with

Mr. Jonathan Geffner
RAVEN PRESS
1185 Avenue of the Americas
37th floor
New York, NY 10036

You should contact Mr. Geffner, or someone else at Raven Press.

I am curious about what use you made of the figure. Would you like to send me an abstract of your thesis?

Sincerely yours,

A handwritten signature in black ink, appearing to read 'Jon Jacky', written in a cursive style.

Jonathan Jacky
Research Scientist
Radiation Oncology RC-08
University of Washington
Seattle, WA 98195

(206) 548 4117

jon@radonc.washington.edu

Page 1 of 1

Attn: Roch Comeau
Medical Physics Unit
McGill University
1650 Cedar Ave.
Montreal, Quebec
H3G 1A4

From: Soyla Daccarett
Permissions Coordinator
Raven Press
1185 Avenue of the Americas
37th floor
New York, NY

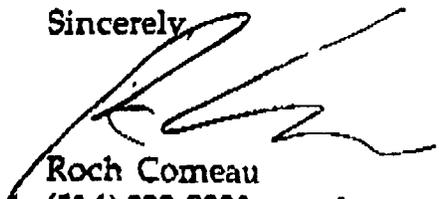
January 12th, 1993

Ms. Daccarett

I am writing this to request permission to reproduce a figure from one of your academic journals. The figure will be used in my M.Sc. thesis in a section reviewing previous investigators. The figure is figure 2 from a review article titled "3-D Treatment Planning: Overview and Assessment" by Jonathan Jacky. It was published in *American Journal of Clinical Oncology (Cancer Clinical Trials)*, volume 13, number 4, pages 331 - 343, 1990. I have already received permission from the author, but he also informed me that I require permission from you. I would therefore appreciate it if you could FAX me a letter (FAX # 514-397-9345) stating such as soon as possible, as I require only this in order to submit my thesis.

Please feel free to contact me if you have any questions regarding this request, and thank you for your cooperation.

Sincerely,



Roch Comeau
(514) 398-8330 - work
(514) 489-9840 - home

Permission is hereby granted for one-time reproduction of the material described herein, provided permission is also obtained from the author and full credit is given to the publication in which the material was originally published.

If a journal, you must cite " . . . Journal title, article title, author name(s), volume number, year, and page numbers.

If a book, you must cite book's author(s) or editor(s), book title, chapter author(s), chapter title, year of publication, and page numbers.

By Soyla Daccarett Date: 1/12/93
Soyla Daccarett, Permissions
RAVEN PRESS, LTD./NEW YORK

Roch Comeau
Medical Physics Unit
McGill University
1650 Cedar Ave.
Montreal, Quebec,
Canada
H3G 1A4

Dr. Michael Goitein
Dept of Radiation Medicine
Massachusetts General Hospital
Boston, MA
02114

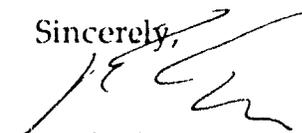
October 20th, 1992

Dr. Goitein,

I am writing this letter to request permission to reproduce a figure from one of your publications in my M.Sc. thesis. I wish to use Figure 3e and 3f from your publication "Multi-Dimensional Treatment Planning: II. Beam's Eye View, Back Projection, and Projection Through CT Sections". I would appreciate it if you could write me a return letter with your decision.

Thank you for your attention

Sincerely,



Roch Comeau

I'm delighted. Please feel
free to use these figures.

Michael Goitein



UNIVERSITY OF
NORTH CAROLINA HOSPITALS

January 5, 1993

Mr. Roch Comeau
Medical Physics Unit, McGill University
1650 Cedar Ave., Montreal, Quebec,
Canada, H3G1A4

Dear Mr. Comeau

Please excuse the delay in answering your letter of October 20th. I have been out of town a great deal the last three months and your letter "fell to the bottom of the stack" and became forgotten. I am sorry about that.

You asked permission to reproduce a figure from the publication:

Rosenman J, Sherouse GW, Fuchs H, Pizer SM, Skinner AL, Mosher C, Novins K, Tepper JE: Three-dimensional display techniques in radiation therapy treatment planning. *Int J Radiat Oncol Biol Physics* 16:263-269, 1989.

That is certainly OK with me. However, I believe that you have to have permission from the editor of the journal. Their address is.

Ms. Jacqueline Meehan, Managing Editor
International Journal of Radiation Oncology, Biology, and Physics
University of Rochester Cancer Center
Division of Radiation Oncology
601 Elmwood Avenues
Rochester, New York 14642

Incidentally, I am curious as to the subject of your Masters thesis. What problem are you researching?

Sincerely yours,

A handwritten signature in cursive script that reads "Julian Rosenman".

Julian Rosenman, M.D.
Associate Professor
Department of Radiation Oncology
University of North Carolina Hospitals
Chapel Hill, NC 27514



PERGAMON PRESS LTD

Registered Office: Headington Hill Hall, Oxford OX3 0BW, UK
Telephone (+44) 865 291141 Fax (+44) 865 60285

M. Roch Comeau
Medical Physics Unit
McGill University
1650 Cedar Avenue
Montreal, Quebec
CANADA

ACJ/J/Free0892
15 January, 1993

FAX ONLY 0101 514 934 8229

Dear Mr Comeau

Re: Int. Jour. Radiation Oncology Biol. Physics 16, pp. 263-269, 1989

Further to your fax, which I have just received I hereby grant you permission to reprint the aforementioned material at no charge provided that:

1. The material to be used has appeared in our publication without credit or acknowledgement to another source
2. Suitable acknowledgment to the source is given as follows

FOR JOURNALS: "Reprinted from Journal title, Volume number, Author(s), Title of article, Pages No., Copyright (Year), with permission from Pergamon Press Ltd, Headington Hill Hall, Oxford OX3 0BW UK".

3. Reproduction of this material is confined to the purpose for which permission is hereby given.
4. This permission is granted for non-exclusive world **English** rights only. For other languages please reapply separately for each one required.

Should your thesis be published commercially, please reapply for permission.

Yours sincerely

Anne-Cécile Junger (Ms)
Subsidiary Rights Manager

NB. Good luck for your thesis.

McGILL UNIVERSITY

FACULTY OF GRADUATE STUDIES AND RESEARCH

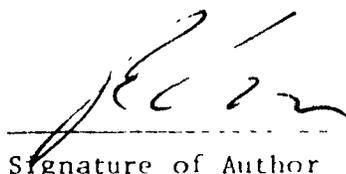
Date .. Oct 20, 1992

AUTHOR'S NAME Roch Comeau

DEPARTMENT Medical Physics DEGREE SOUGHT: M.Sc.

TITLE OF THESIS The Design and Implementation of a
Three Dimensional Treatment Planning System

1. Authorization is hereby given to McGill University to make this thesis available to readers in a McGill University Library or other Library, either in its present form or in reproduction. The author reserves other publication rights, and neither the thesis nor extensive extracts from it may be printed or otherwise be reproduced without the author's written permission.
2. The authorization is to have effect on the date given above unless the Executive Committee of Council shall have voted to defer the date on which it is to have effect. If so, the defered date is: _____


Signature of Author

Permanent Address:
5 des Violettes,
Ste-Therese, QC
J7E 2H4

Signature of Dean required if date is inserted in paragraph 2.

The Design and Implementation of a 3-D RTP System

Computerized

Abstract

An efficient and productive radiation treatment planning (RTP) system must make use of both appropriate visualization techniques and good user interface design. The suitability of several visualization techniques have been examined in the context of 3-D radiation treatment planning. These techniques include wire frame, surface rendering, volume rendering and a subset of volume rendering: reformatting of data. A rudimentary computerized RTP system was written using the most appropriate visualization techniques examined earlier. These techniques were used to display the anatomical data acquired from computed tomography (CT) scanners, the beam position within the anatomy, and finally, the dose distributions resulting from the entered plan. The program was written in ANSI C and runs on a Silicon Graphics Personal Iris UNIX workstation. The system makes use of effective user interface tools and efficient code which results in an efficient and interactive system. The accuracy of the system is verified by comparing dose profiles obtained with film dosimetry and from the computer calculations.

Résumé

Un système de planification de traitement par radiation doit se servir de techniques de visualisation appropriées et d'une interface à l'utilisateur efficace. Plusieurs techniques de visualisation graphiques ont été examinées dans le contexte d'un système de planification de traitement par radiation. Ces techniques incluent les modèles "fils de fer", le rendement de surface, le rendement de volume, et la restructuration des données. Un système rudimentaire de planification de traitement par radiation a été conçu en se servant des techniques que nous avons trouvées les plus appropriées. Ces techniques sont utilisées pour visualiser l'anatomie dérivée d'un Tomogramme Axial, la position du champ de traitement dans l'anatomie, et la distribution de dose qui résulte d'un traitement. Le logiciel a été écrit en "ANSI C" sous le système d'exploitation UNIX et fonctionne sur un ordinateur "Personal Iris" de Silicon Graphics. Ce système se sert aussi d'une interface à l'utilisateur qui est efficace et intuitive. La précision du système est vérifiée en faisant la comparaison de profils de dose dérivés de films et de profils obtenus du logiciel.

Original Contribution

Various visualization techniques for the display of anatomical data in the context of interactive 3-D radiation treatment planning are evaluated. Reformatting of data was determined to be the best suited for this task due to its low processor requirements, low preparation work load, and its ability to work effectively in tandem with other visualization techniques.

A complete 3-D treatment planning system was written in C operating on a Silicon Graphics Personal Iris workstation under the UNIX operating system utilizing a novel graphical user interface (GUI). The software uses a derivative of the Milan-Bentley method of calculating dose from a photon beam. An algorithm was developed to provide the ability to automatically detect and account for surface curvature in the dosimetry without the need for manual contouring. The software uses reformatting of data for the anatomical display, wire frame outlines for the beam representation and outline contours for dose distribution visualization.

The system stresses interactive real time response and included an assortment of tools to aid in the evaluation of the plan. These tools include calculation of dose to one axial slice or the entire volume, a cursor which gives dose as well as geometrical information and a slice or volume hot spot indicator.

Acknowledgments

I would like to thank Dr. B.G. Fallone for his continued and constructive support, patience and understanding throughout the project's duration.

I thank my colleagues, Mr. Alex Markovic, Mr. Ian Crooks and Mr. Sridar Narayanan for their support, both technical and moral, for their friendship and their encouragement. I also thank the Medical Physics students at the Montreal General Hospital for their thought provoking dialogue and assistance. I would like to thank the professors and clinical personnel at the Montreal General Hospital for their constructive criticism and for allowing me access to the treatment planning system and the treatment machines.

I thank Alias Research Inc., for providing me with a software and training grant which gave me extra insight into many current visualization techniques, and the MGH Research Institute for a hardware grant that provided the SGI computer essential to this project. I am grateful to Programmed Communications Ltd. for their flexibility and support in my part time work, which provided me with the personal funding required to undertake this project.

Finally, I thank my parents for their support and encouragement which was present throughout my academic career.

Table of Contents

Abstract	i
Résumé	ii
Original Contribution	iii
Acknowledgments	iv
Table of Contents	v
List of Figures	ix
List of Tables	xvi

Chapter 1 Introduction

1.1 General Introduction	1
1.2 Thesis Organization	2

Chapter 2 Evolution of Computerized Treatment Planning

2.1 Introduction	4
2.2 Evolution of computerized radiation treatment planning .	5
2.2.1 Early mainframe computers	5
2.2.2 Dedicated mini-computers	6
2.2.3 Three dimensional radiation treatment planning .	8
2.3 Three dimensional visualization techniques	9
2.3.1 Wire frame techniques	10
2.3.2 Surface rendering	12
2.3.3 Volume rendering	14

2.3.4	Reformatting of data	17
2.3.5	Beam's eye view (BEV)	18
2.4	Verification of 3-D RTP accuracy	19

Chapter 3 Materials and Methods

3.1	Introduction	25
3.2	Equipment layout	26
3.2.1	386 tower PC	26
3.2.2	Silicon Graphics Personal Iris workstation	26
3.3	Anatomical data	28
3.3.1	Acquisition	28
3.3.2	Surface rendering data preparation	28
3.3.3	Reformatting of data	30
3.4	Simulating uniform density phantoms	32
3.5	Software development platform	32
3.6	Dosimetry data calculation	34
3.7	Coordinate systems	36
3.8	Verification	39
3.8.1	Film dosimetry considerations	39
3.8.2	Film parallel to the beam using ^{60}Co radiation	39

Chapter 4 Visualization of Anatomical Data: Technique Evaluation

4.1	Introduction	48
4.2	Contouring and surface rendering using high order equations	49

4.2.1	Contouring the anatomical data	49
4.2.2	Generating a surface using the "patch" technique ..	52
4.2.3	Generating a surface using the "skin" technique ..	56
4.3	Automatic image generation	60
4.3.1	Automatic segmentation of surface boundary information	60
4.3.2	Creation of a polygon set from contour points	61
4.4	Surface rendering using polygonal data	61
4.4.1	Surface rendering using SolidView	61
4.5	Volume rendering	65
4.6	Summary	68

Chapter 5 3-D Treatment Planning System

5.1	Introduction	69
5.2	Starting the software	71
5.3	Anatomical volume	73
5.3.1	Entering an image set	73
5.3.2	Visualizing the volume	73
5.4	Beams Manipulation	77
5.4.1	Beam visualization	77
5.4.2	Creating and manipulating beams	77
5.5	Dosimetry	80
5.5.1	Dose distribution visualization	81
5.5.2	Evaluation tools	83
5.6	Summary	87

Chapter 6 Technical Specifications

6.1	Introduction	88
6.2	Image volume display	88
6.2.1	Raster allocation	89
6.2.2	Redraw organization	90
6.2.3	Look-up table (LUT) methods	96
6.3	Beam display algorithms	97
6.3.1	Beam entry/exit points	98
6.3.2	Beam outline drawing	100
6.4	Dose calculation	102
6.4.1	3-D beam coordinates and simplifying assumptions.	103
6.4.2	Surface correction grid calculation	103
6.4.3	Calculation volume optimization	106
6.5	Summary	107

Chapter 7 System Verification

7.1	Introduction	109
7.2	Film response curve	109
7.3	Profile comparison	111
7.3	Summary	113

Chapter 8 Conclusion

8.1	Summary	115
8.2	Future work	116

List of Figures

Figure #		Page
Figure 2.1	Example wire frame image by Jacky et. al.	12
Figure 2.2	Example surface rendered image by Goitein et. al.	16
Figure 2.3	Example volume rendered image by Rosseman et. al	16
Figure 2.4	Examples of the Beam's Eye View using wire frame and reformatting of data.	20
Figure 3.1	Overall layout of the imaging system.	27
Figure 3.2	Creating a series of polygons from a set of contiguous slices.	31
Figure 3.3	Creating uniform volume phantoms by combining several 2-D slice images.	33
Figure 3.4	General set-up for calculating dose within a volume.	35

Figure 3.5	Diagrams of the coordinate systems	38
Figure 3.6	General set-up of the T780 ^{60}Co unit and phantom for film calibration.	41
Figure 3.7	Film positions within the phantom for all measurements.	42
Figure 3.8	General set-up for film scanning and description of the principle of operation of the film scanner.	43
Figure 3.9	Procedure of cropping scanned images to obtain geometrical alignment for the profiles.	45
Figure 4.1	Alias Studio software with a CT image during the contouring process.	50
Figure 4.2	Perspective view of 13 contours of a head obtained using CT data as source material.	51
Figure 4.3a	Wire frame view of the head using the “patch” technique for surface generation.	53
Figure 4.3b	Alias “Quick Render” of the “patch” set.	54

Figure 4.4	Illustration of how the “patch” technique is not suitable for this type of object generation.	55
Figure 4.5a	Wire frame of the head using Alias “skinning” technique.	57
Figure 4.5b	Ray trace image of the “skinned” head.	58
Figure 4.6	Illustration showing the problems encountered using the “skinning” technique.	59
Figure 4.7a	Wire frame view of the head using our automatic segmentation software.	62
Figure 4.7b	Split view of wire frame view and surface rendering of the polygonal set using SolidView software.	63
Figure 4.7c	Split view of an example data set provided with the SolidView software.	64
Figure 4.8	Example of how a 3-D volume data set can be segmented to display 3 orthogonal views.	67
Figure 5.1	General flow diagram of the operations performed for a typical treatment plan.	70

Figure 5.2	Viewer software main screen.	72
Figure 5.3	Viewer software image list screen.	74
Figure 5.4	Viewer orthogonal images display and control windows.	76
Figure 5.5	Orthogonal view windows displaying the anatomical and beam information.	78
Figure 5.6	General beam parameter window	80
Figure 5.7	Viewer software dosimetry control window.	82
Figure 5.8	Orthogonal view windows displaying anatomical, beam and dosimetry information.	84
Figure 5.9	Axial view showing anatomical, beam, dosimetry and point cursor.	85
Figure 5.10	Viewer software information window.	86
Figure 5.11	Gray scale output of the dose distribution.	87
Figure 6.1	Diagram showing how the voxels of one slice are broken down into a series of rasters.	90

Figure 6.2	Example of how the rasters are used to create the axial and coronal views.	91
Figure 6.3	Pointer arrays and their relationships with rasters that are used to draw the axial and coronal views.	92
Figure 6.4	Pseudocode of a generic redraw sequence required to update the screens in the treatment planning system.	93
Figure 6.5	Pseudocode of the modified redraw sequence.	94-95
Figure 6.6	Modifying the LUT ramp table to control the image brightness.	97
Figure 6.7	Simple 2-D example of the beam entry and exit points are found.	99
Figure 6.8	All the combinations of intersections of the beam volume and the anatomical slice cutting plane.	101
Figure 6.9	Illustration of how the software determines intersections of the beam edges and the slice plane.	102

Figure 6.10	Comparison of the grids used to hold the off axis ratios in the standard Milan-Bentley method and in our software.	104
Figure 6.11	Comparison of the surface correction matrices for the standard Milan-Bentley method and our software.	104
Figure 6.12	Illustration of the effectiveness of the calculation volume optimization for varying θ .	107
Figure 7.1	Film H&D curve for ^{60}Co radiation and Kodak X-Omat V film.	110
Figure 7.2	Film response curve for ^{60}Co radiation and Kodak X-Omat V film.	111
Figure 7.3	Profiles taken across the beam central axis from the film data and the computer calculations.	112
Figure 7.4	Depths of the film measurements and positions of the profiles extracted from each film with respect to the beam central axis.	112

Figure 7.5	Profile comparisons for film and computer treatment planning system for profiles taken off the beam central axis.	113
Figure 8.1	Example of a hybrid system using a combination of reformatting of data and surface rendering.	118

List of Tables

Table		Page
3.1	Film label numbers and exposure times for film response curve for ^{60}Co radiation.	42
3.2	Film label numbers and depths of exposure for film placed in the BEV plane.	42

Chapter 1

Introduction

1.1 General Introduction

External beam radiotherapy has been an indispensable tool for the treatment of cancer for many years. During this time, the techniques have become more sophisticated and generally more successful in delivering the desired dose to the target tissue while minimizing the dose to the surrounding healthy tissue. Often, these new techniques are more complicated to evaluate and to implement. In order to evaluate a treatment before it is actually performed, dose calculations have to be made that account for the treatment technique and the patient anatomy. These calculations help determine the expected effectiveness of a treatment, and to help the radiation oncologist correct the technique for the particular patient's anatomy. One of the major tools that has helped the radiation oncologist and physicist is the computer. The computer is useful both in rapidly calculating dose distributions and in presenting the results in a manner that accelerates the evaluation and thus the implementation process. This work examines the capabilities of current computer 3-D visualization techniques as applied to interactive treatment planning, and uses the suitable ones by creating an interactive computerized radiation treatment planning (RTP) system.

1.2 Thesis Organization

Chapter 2 introduces treatment planning by briefly tracing the history of computers in this field. This will start with early 2-D mainframe applications to current implementations by other investigators using modern computer workstations. Emphasis will be placed on significant advances made in the visualization techniques used and in the improvement of plan evaluation. The issue of treatment planning system verification will also be discussed.

Chapter 3 presents the equipment and the experimental procedures used in our studies. This will include descriptions of all the computer hardware and software (both commercial and custom developed), the techniques used to acquire and simulate anatomical data sets, and how the data is transferred to the computer. The experimental techniques used to verify the final system's accuracy will also be presented.

Chapter 4 describes and evaluates the various visualization techniques available on typically affordable computer workstations. These techniques include surface and volume rendering and reformatting of data. Particular emphasis will be placed on the suitability of these techniques as applied to 3-D treatment planning by accounting for the computer processing requirements and in the work required in data preparation by the user.

Chapter 5 gives a functional description of the final treatment planning system that we developed after the evaluation process. This is accomplished by creating an example treatment plan. This begins with entering an

anatomical data set of a spherical phantom, creating and positioning one treatment beam, and calculating and evaluating the resulting dose distribution using the available tools.

Chapter 6 provides a technical description of our software package. Particular emphasis is given to how the memory structures and algorithms differ from typical implementations in order to improve the system response.

Chapter 7 presents the results of a film technique used to verify system accuracy. The film is used to obtain a 3-D distribution of dosimetry data in order to compare it to the computer predictions for the same treatment parameters.

Chapter 8 summarizes the conclusions of the visualization technique evaluations, the software implementation and the system verification techniques. The chapter also proposes future enhancements to the software in order to increase its functionality both as a treatment planning system, and as a tool to implement and evaluate new visualization techniques.

Chapter 2

Evolution of computerized treatment planning

2.1 Introduction

External beam therapy machines have provided medical personnel the flexibility to use radiation for treatment of a variety of conditions. By placing single, or multiple beams in different arrangements, complex radiation distributions can be obtained. These are used to achieve two goals: First to deliver a specified dose within $\pm 5\%$ to a specific target volume (e.g., a tumor), and second to minimize the dose to surrounding healthy tissue. In order to use the available equipment efficiently and effectively, one must be able to accurately calculate and evaluate the dose distribution within the volume of interest resulting from a proposed treatment.

In early radiation therapy, planning the treatment was achieved by either manually performing the calculations and plotting the results for evaluation, or by fitting the patient case to a standard case where the calculations were previously performed. This meant that the planner had to evaluate the quality of a plan with very little calculated information, or that only plans that fit "standard" models could be prescribed. In the late 1950's the development of the computer had advanced significantly, enabling it to be applied in treatment planning. The two most obvious contributions computers can make in radiation treatment planning (RTP) are in dose calculation and in image display as a visual aid during the planning

process. As computers have become more powerful, more elaborate and higher resolution calculation schemes have been adopted. In addition, the techniques to visualize both the patient anatomy and planning results have become equally elaborate.

As treatment modalities become more complex, the role of computers in treatment planning has become increasingly important[1]. This chapter will introduce computerized treatment planning by tracing the evolution of computer use in this field from early mainframe applications to some current examples of 3-D graphics workstations. 3-D visualization techniques including wire-frame outlines, surface and volume rendering, and beam's eye view (BEV) will be presented. Finally, methods of verification of the accuracy of a 3-D RTP system will be examined.

2.2 Evolution of computerized radiation treatment planning

2.2.1 Early mainframe computers

As early as 1958, Tsien used computers to help in dose calculations for rotational beam therapy[2]. The benefits of general computerized planning were still questioned, given the time and costs they then involved. In the 1960's, Bentley[3] developed a system running on a mainframe computer, where the plan and relevant data were entered using punch cards for batch processing. The results would arrive hours later and would have to be repeated if the plan was not successful, or an data input error

occurred. This turnaround time made it difficult or impossible to use these systems in an interactive manner, and the output device was a line printer which was not well suited for the task of printing graphics.

2.2.2 Dedicated mini-computers

In order to decrease the turnaround time of plan calculation, mini computers were employed. This began with a program functioning on a Programmed Console (PC) by Cunningham and Milan in 1969 at Washington University in St. Louis, U.S.A. The system was improved by Bentley and Milan[4] in 1971 with a two dimensional system operating on a Digital Equipment PDP-8I and various peripherals. Their system was based on the earlier PC but made use of increased computing power made available on the PDP-8I. As with many computerized planning systems to come, their system improved on past ones in the following areas: calculation complexity and speed, display techniques, and in user interface.

The computer system would typically perform the dose calculation at 1000 points in a uniform grid superimposed on the patient outline within a minute or two. Once the dose array was calculated, the user could enter isodose values that required display. A contour would then be calculated and displayed on an oscilloscope screen (later to be referred to as a vectored display) along with the patient contour, and any critical organs previously outlined by the user. After examination of the results, the user could modify the plan by changing any of the entered treatment parameters, including beam width, length, energy, wedge weighting and gantry angle. The new results were calculated and again examined on the screen. This ability to

modify any parameter(s) and view the results within a few minutes provided the user with an *interactive* treatment planning system. The final results were recalculated at a resolution of about 3000 points and printed on a plotter, which was better suited for graphical printing than the line printer.

With the improvement of diagnostic imaging tools, notably the computerized tomography (CT) scanner[5], and improvements in raster graphic displays, an improvement in the display of the patient data and dose distributions was achieved. The *Target* computerized treatment planning system introduced in the early 1980's (General Electric Corp.) uses a raster graphics screen, which can display gray scale images obtained from a CT scanner, as well as text information. The system combines the graphics screen and improved calculation capability to increase its functionality.

The system uses the CT data to facilitate the contouring process, and displays a CT slice on the screen with the patient contour, as well as the dose distributions obtained after calculation. This enables the user to view more completely, the relationship between the patient structures and the dose distribution within those structures. The faster computer is also capable of either obtaining the dose values faster or conversely, obtain dose points on a finer grid, increasing accuracy. The system also provides for better interactivity by using an improved user interface, more interactive tools and faster response.

Up to very recently, computerized treatment planning systems have still only replaced manual calculation. That is, the final result is a great improvement in time efficiency, but comparatively little improvement in the quality and quantity of information available have been achieved. When manual dose planning was performed, the planners were almost always restricted to a two dimensional set-up. Often the plane of calculation was chosen to cut the treatment volume along the beam central axis and the off axis dosimetry was either ignored, approximated or calculated at a few points for verification. The introduction of the computer replaced the manual calculations, but the results were still restricted to 2-D. The manufacturers and users chose the advantage of quick system response in 2-D instead of performing 3-D calculations. Furthermore, the hardware and software required to display 3-D results in an adequate manner was either not available, or was prohibitively expensive. Some inherently 2-D systems do provide for the calculation of the dose distribution for a few parallel slices, but the display tools are still based on displaying one slice at a time on a single plane. These are sometimes referred to as 2.5-D systems.

2.2.3 Three dimensional treatment planning

The recent availability of inexpensive and powerful computers, namely mini computers, computer graphics workstations (e.g., SUN & Silicon Graphics), and the improvement of the already inexpensive micro computers (e.g. IBM & Apple Macintosh) have changed the present situation. These computers provide at least an order of magnitude higher performance in CPU power, and superior 3-D graphics capability, when compared to mini computers of a few years ago. With these computers

readily available, 3-D visualization in treatment planning is becoming more common. Presently, one can purchase one of many potential 3-D treatment planning systems, produced both within the research and teaching sector, as well as from computer or medical equipment manufacturers[6]. Many of these systems run on different hardware platforms and use a wide variety of display and user interface techniques, each attempting to attract the attention of their target market. When examining a particular system, one should pay particular attention to how appropriate the visualization techniques are to display the anatomical, beam and dose distributions, the calculation techniques, the anatomical data pre-processing required and finally the interactive response and hardware/software design of the system as a whole.

2.3 Three dimensional visualization techniques

The ultimate goal in computer visualization is to communicate a message to the user[7]. In treatment planning, that message is whether the proposed plan will or will not provide the desired dose distribution within the organs or volume of interest. An imaging technique must be able to convey the message in a clear and rapid fashion. Since the variables involved are inherently 3-D, a 3-D technique is likely to be the most suitable. In order to choose the right technique or combination of techniques, a general understanding of the relative strengths and weaknesses of all the available techniques is required.

2.3.2 Wire frame techniques

The wire frame technique was one of the first 3-D techniques implemented in computer visualization. Wire frame requires comparatively little memory capacity and processor power to render on the screen. In addition, this simple display can clearly convey the spatial relationships between the objects on the screen. For these discussions, a technique will be referred to as wire frame when an object is displayed as a series of lines that represent the edges of that object in 3-D, or a projection of that 3-D object onto a 2-D plane. When the outline of a slice of the object is shown (e.g., outline of a single CT slice) this will be referred to as a contour and not a wire frame.

Wire frame displays are used in many current 3-D systems. One such system is described by Mohan et al. [8, 9] which uses a wire frame display to show the relative positions of the gantry, collimator and couch positions for a given treatment. Wire frame displays of the patient anatomy and beam trajectory are displayed when the system is using its beam's eye view (BEV) technique to be discussed later. The wire frame technique is also used to display patient anatomy and dose distributions, either by isodose surface or colour coded by dose to the surface of an organ.

One treatment planning system described by Jacky[10] relies heavily on wire frame display techniques to satisfy the visualization requirements (the description discussed other more elaborate visualization techniques, but no examples are given). Although wire frame images are often simple

to examine and manipulate, it can become cluttered. It can be difficult for the user to derive precise and useful information about the spatial relationships because the wire frame attempts to depict an inherently 3-D scene by projecting the outlines on a 2-D surface: the screen. Figure 2.1 shows an example of a cluttered image. The image shows patient anatomy, multiple beam trajectories, wedges and isodose distributions. With all the overlapping lines, it is difficult to visualize what is actually going on. This often is eased by rotating the view around so the user may mentally integrate the series of views and better understand what is being displayed.

Wire frame visualization is used in a variety of ways by many more investigators including Gotein et al.[11], Chin et al.[12] and McShan, Fraass and Lichter[13]. Each of these systems use this technique in concert with other techniques discussed later.

It is important to note that the information required to generate the wire frame images is often obtained after extensive work by the user. Most of the anatomical information used in 3-D treatment planning is obtained using CT or magnetic resonance imaging (MRI) scanners. In order to produce wire frame files from these scans, the user must contour every organ of interest and the treatment volume for every slice available. Most of this work is done manually using a variety of input devices including computer mice, digitizing tablets and light pens. Some software packages try to automate the process as much as possible[8] by providing edge detection tools for the most obvious thresholds, but the task is still extremely time consuming and often requires the aid of a physician for delineation of complicated or subtle structures. Although much work is being done to

accelerate this process, no purely automatic segmentation and contouring system is available at this time.

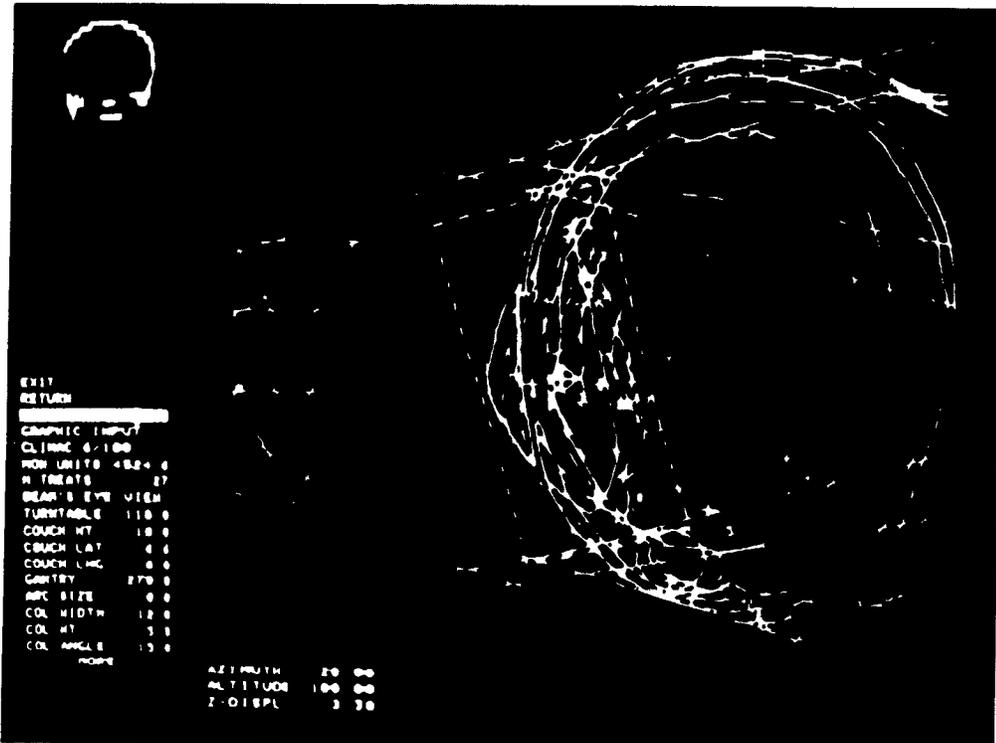


Figure 2.1: Example image shown by Jacky et. al.[10]. Although there is a lot of information in this image, it is very cluttered, making it difficult to visualize the data.

2.3.3 Surface rendering

Many of the early computer aided design (CAD) systems used wire frame displays for visualization. As in medical imaging, wire frames were used primarily because the technology for a superior visualization

technique was not yet available. As the technology improved, CAD system designers improved on their existing systems by filling in the gaps between the wire frames, and displaying their objects by rendering the actual surfaces.

An excellent example of the step from wire frame to surface rendering is given by Goitein[11]. Figure 2.2a is an example which shows the advantages and disadvantages of this technique in one image. The image easily shows the general spatial relationships between the structures, but some of the information is not visible because the larger structure in front obscures the rear objects. Furthermore, their system takes approximately 8 seconds to generate each view so real or near real time image manipulation was not possible at that time. Anticipated improvements included faster hardware for faster image generation, and the use of transparency in the surfaces to allow the user to see through the objects that obscure others. Until transparency is implemented, Fig 2.2b shows an interim solution. The obscuring object is displayed in wire frame without any of the surface patches. This enables the user to see through the structure to the other objects behind it.

Another group examining the uses of surface rendering in medical imaging is Rosenman et al.[14]. They have devised a series of tools that use edge and threshold detection to help automate the contouring process for outer contours and internal organs that have a high contrast gradient with the surrounding tissue. Their techniques use Gouraud[15] and Phong[16] shading to enhance the quality of the surface characteristics.

In general, surface rendering can yield informative images but require the same extensive work at contouring the structures before being able to view the images as was the case in wire frame techniques. Furthermore, by having to segment the data before being able to plan, structures that may require attention may not have been contoured because their significance may not have been obvious at the start of the process. This can result in important organs being exposed to unnecessary doses because they were not present on the screen and thus not in the mind of the dosimetrist during the planning process. Surface rendering requires a powerful processor in order to allow real or near real time processing which may still may not be generally affordable at this time.

2.3.4 Volume rendering

Although surface rendering and volume rendering may look similar to each other, the two are distinctively different. While surface rendering uses information about the surface of an object (i.e., the normal or orientation of a surface polygon with respect to the light source and viewer) volume rendering derives the image from the average value of a set of voxels along a projection line or ray, and displays that value on the screen. Another volume technique is to characterize all the voxels in a volume into a predetermined set of categories, e.g., bone or tissue. The result is a display that uses the characteristics of the entire volume, not just the surface boundary information. One of the obvious benefits of this approach is that the user no longer needs to outline the objects of interest before viewing the results. This eliminates the possibility of accidentally omitting any important organs from being monitored during the plan.

Although no real or near real time volume rendering is yet available, their availability is subject to the improvement of current computer hardware, which is inevitable, either in graphics co-processor design, or in parallel processing techniques. In anticipation of this event, Rosenman et al.[14] have examined the usefulness of a volume rendering based system. To generate the anatomical volume display, their system examines the CT number of each voxel within the volume and assigns them a probability of being a certain tissue type (e.g., bone, muscle, or air). The user then assigns a particular colour and transparency to each tissue class. The dose distribution is displayed as a separate series of colours and transparency values displayed within the anatomical volume. Figure 2.3 is an sample of what this display technique yields when displaying both the anatomy and the dosimetry simultaneously.

Although the quality of this display technique is obvious, particularly when compared to surface rendering, it is currently difficult or expensive to implement it in a real time interactive environment. Furthermore, the author states that this technique is intended as a complimentary display to help give the planner an “understanding at a glance”. This technique is intended to function in tandem with a more conventional display like reformatting of data.

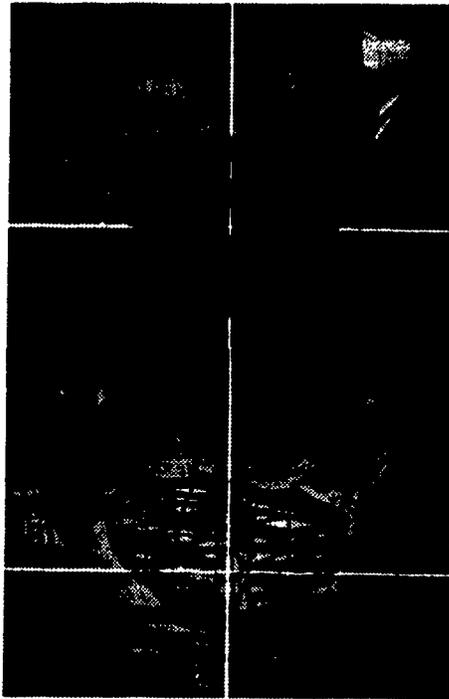


Figure 2.2: A) Surface rendered image presented by Goitein et. al.[11], which shows one disadvantage of surface rendering for visualization B) This image shows a simple solution to the problem presented in A



Figure 2.3: Volume rendering display presented by Rosseman et. al.[14]. Notice that the same visualization technique is used for both the anatomical display and the dose distributions

2.3.5 Reformatting of data

We have seen that both surface and volume rendering can yield images that are far superior to that of wire frame techniques. However, surface rendering requires that contouring of all the structures be performed, and volume rendering is still far from being implemented in an interactive environment. If an examination is performed on the shortcomings of conventional 2-D treatment planning systems, the most common complaint would be the inability to work or analyze the data in 3-D. The most logical step would be to create a display that is capable of displaying any one of a series of slices from a CT or MR scanner. If these contiguous slices were grouped together, a volumetric data set would be obtained. By interactively reformatting and displaying the data as 2-D subsets, a 3 dimensional feel can be conveyed to the user. In addition to the standard axial slice view, sagittal and coronal slices can easily be formatted and displayed. This technique is known as reformatting of data.

Many of the imaging systems that use surface or volume rendering use them as a complimentary display. The display that is most often the "default" or "fallback" technique is reformatting of data. There are significant advantages of this technique: The data need not be segmented by contouring which saves time, accidental omission of organs in the contouring process can not occur; and reasonably fast interactive response can be achieved. One example of this is the treatment planning system by Mohan et al.[8], described earlier. Wire frame is used in their beam's eye

view display (BEV), but axial, sagittal and coronal slices are also available as reformatted data.

Another advantage of the reformatting of data is its inherent ability to be combined with other visualization techniques. Many systems overlay the beam outline on the reformatted data slices to aid in beam positioning. By using two different techniques to simultaneously show two related yet different data sets (often referred to as a hybrid technique), their relationships can be seen without risk of getting them confused. If we examine Figure 2.3 again, it can be seen that the same imaging techniques are used to show the dose distributions and the anatomical data. If a hybrid technique were used here, it would be easier to see the difference between the anatomy and the dosimetry, while having an excellent view of their spatial relationship. By observing three orthogonal views simultaneously, and by cycling through several slices in each view, a 3-D analysis of the entire volume may be accomplished.

Reformatting of data may also be used in a perspective view. That is, the slices need not be orthogonal to each other, but rather a perspective view along any arbitrary direction, say along the beam central axis may be generated in real or near real time. This method may be used in a BEV display, aiding in beam positioning and in custom beam block design.

2.3.6 Beam's Eye View

Although the BEV display is not a fundamental visualization technique but rather a specific application of other techniques (e.g., wire

frame and reformatting of data), it merits mention here. Most techniques offer the ability to set up a perspective view. This view is produced by setting of the perspective view position within the volume to coincide with the view of the beam. The resulting image would be exactly what the beam would see within the volume it encompasses, enabling the planner to automatically see exactly what is within the beam and what is not.

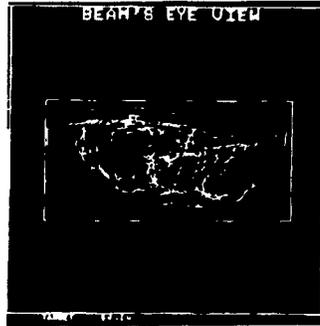
The BEV technique is used by many systems including those by Mohan et al.[8], Jacky[10], and Goitein and Abrams[11]. A mathematical treatment on how to generate BEV images from the volumetric data is presented by Mohan et al.[9]. Figure 2.4a is an example of the BEV using wire frame, and Figure 2.4b is an example of BEV using reformatting of data.

2.4 Verification of 3-D treatment planning system accuracy

Although the computer can make millions of calculations a second, the results are only as good as the algorithm that generated them in the first place. There are two types of errors that can be encountered when using a computer: A program code error and a program design error.

Code errors occur when a particular piece of program code has an unintentional flaw or “bug” in it such as a typing error. These errors can sometimes get through the de-debugging stages and cause either system crashes or erroneous results. Obviously system crashes are easier to detect,

A)



B)



Figure 2.4. Examples of beam's eye view (BEV) visualization presented by Mohan et. al.[9]. A) BEV using wire frame data. B) BEV image using reformatting of data.

while erroneous results can go undetected, particularly if the errors are small or within the range or appropriate values the user might have expected.

Design flaws can occur when the software is being used in a situation that was not considered by the designer(s). Well written software should inform the user whenever it is requested to function outside its accepted parameters. When this does not occur, it may again generate a system crash, or more likely provide erroneous results. These types of errors are difficult to detect, as they may occur only in some obscure situations.

For these reasons, it is important to perform acceptance testing when introducing a new treatment planning system, or even introducing a software upgrade. McCullough and Krueger[17] have presented a verification protocol for treatment planning systems for external photon beam. The goal of their protocol is to ensure that the data used by the program is consistent and dependable, and it covers the accuracy of the system for single and multiple beams for a variety of typical treatments. It specifies what characteristics of the beam merit close scrutiny. The protocol also attempts to familiarize the user with the limitations of the system, reducing the probability of encountering a design error. This protocol does not propose any special techniques for 3-D treatment planning systems.

In order to perform acceptance testing on a 3-D treatment planning system, Stern et al.[18] have devised a method of generating a true 3-D data

set. Instead of attempting to use an ionization chamber to measure all the points in a volume, film is placed in a water equivalent phantom at various depths in the BEV plane. The density of the film is converted to dose by obtaining a film calibration curve, and by normalizing the dose from each film to the dose on the beam central axis as measured by an ion chamber. A 3-D matrix is created by interpolating the dose between slices. This matrix of values can be compared to the matrix obtained by performing the equivalent experiment on the treatment planner.

This process can be repeated for several different set-ups. Variations in beam energy, wedge values, and position parameters can generate several sets of data in order to find out if the system is accurate over a variety of experimental conditions. The authors present a method of using differential dose volume histograms as an aid in determining system accuracy.

References

- 1 Doppke K.P., Goitein M., 1988, "A survey or the information gained from planning treatment with a computer", *Med. Phys.*, **15**(2), pp. 258-262
- 2 Tsien K.C., 1955, *Brit. J. Radiol.*, **28**, 432, 1958, *ibid* , 31, 32
- 3 Bentley R.E., 1964. "Digital computers in radiation treatment planning " *Brit. J. Radiol.*, **37**, pp 748-755
- 4 Bentley R E , Milan J , 1971 "An interactive computer system for radiotherapy treatment planning". *Brit. J. Radiol.*, **44**, No 527, pp. 826-833
- 5 Hounsfield G.N., 1973, "Computerized transverse axial scanning (tomography)", *Brit. J. Radiol.*, **46**, p. 1016
- 6 Various public advertisements, 1992, *Med. Phys.*, 18, p xxiv, xxviii, xxxi, xxxv, xli, xlvi, lvi, lxi, lxiii, lxx, lxxxvii
- 7 Newman W.M., Sproull R F., *Principles of interactive computer graphics* 2nd ed. (McGraw-Hill 1979)
- 8 Mohan R , Barest G., Brewster L J., Chui C.S , Kutcher G J , Laughlin J.S.,and Fuks Z., 1988, "A comprehensive three-dimensional radiation treatment planning system", *Int. J. Radiation Oncology Biol. Phys* , **15**, pp 481-495
- 9 Mohan R , Brewster L J., Barest G , Ding I Y , Chui C S , Shank B , and Vikram B , 1987, "Arbitrary oblique image sections for 3-D radiation treatment planning" ,*Int J Radiation Oncology Biol. Phys.*, **15**, pp.1247-1254
- 10 Jacky J., 1990, "3-D Radiation therapy treatment planning Overview and assessment", *Am J Clin Oncol* **13**(4), pp. 331-343
- 11 Goitein M., Abrams M., Rowell D , Pollari H , and Wiles J , 1983, "Multi-dimensional treatment planning II: Beam's Eye-view, back projection, and projection through CT sections", *Int J Radiation Oncology Biol. Phys* , **9**, pp. 789-797
- 12 Chin L.M., Siddon R.L., Svensson G K , and Rose C., 19-85, "Progress in 3-D treatment planning for photon beam therapy", *Int J Radiation Oncology Biol Phys.* **11**, pp. 2011-2020
- 13 McShan D.L., Fraass B.A., and Lichter A.S., 1990, "Computerized treatment planning", *Int J Radiation Oncology Biol Phys* , **18**(6), pp 1485-1494
- 14 Rosenman J., Sherouse G.W., Fuchs H., Pizer S M , Skinner A L , Mosher C , Novins B.S. and Tepper J.E., 1989, "Three-dimensional display techniques in radiation therapy treatment planning", *Int. J. Radiation Oncology Biol Phys* , **16**, pp 263-269

(figure reproduced with permission from Pergamon Press Ltd., Headington Hall, Oxford OX3 OBW, UK.)

- 15 Gouraud H., "Continuous shading of curved surfaces", 1971, *IEEE Trans. Comp.* **20**, pp 623-628
- 16 Phong B.T , 1973, "Illumination for computer generated images", Ph.D. Dissertation, University of Utah
- 17 McCullough E C., and Krueger A.M., 1980, "Performance evaluation of computerized treatment planning systems for radiotherapy: External photon beams", *Int. J. Radiation Oncology Biol. Phys* , **6**, pp 1599-1605
- 18 Stern R L , Fraass B A , Gerhardtsson A., McShan D.L., and Lam K.L., 1992, "Generation and use of measurement based 3-D dose distributions for 3-D dose calculation verification", *Med. Phys.* **19**(1), pp 165-173

Chapter 3:

Materials and Methods

3.1 Introduction

The implementation of a 3D interactive treatment planning system involved many steps. First, various visualization techniques were examined and evaluated. This involved acquiring anatomical data, processing it to conform to the requirements of each technique and evaluating the results. The processing often required writing custom software to obtain information from the raw data and arrange it to conform to a format compatible with the imaging software. Once the techniques were evaluated, the actual treatment planning system was written and tested. This chapter will describe the techniques, hardware and software used to accomplish these goals. First, a general description of the hardware used for all the studies will be presented followed by a description of anatomical data acquisition and pre-processing. Data processing and data simulation for each individual technique will be explained, including descriptions of the function of all custom software developed for these studies, as well as of various commercial packages. Finally, a description of the development environment for the actual treatment planning system, dosimetry techniques, coordinate systems, and the experimental set-ups and procedures for the verification of this system will be presented. A detailed description of the functionality of the final system will be presented in

chapter 5, and a description of the algorithms used in the system will be presented in chapter 6.

3.2 Equipment layout

Figure 3.1 shows the general layout of the various equipment used during these studies and their role in them. Real anatomical data comes from a Toshiba 900 S/F computed tomography (CT) scanner whose files are converted to a disk format readable by PC computers. The images are then brought to the 386 tower for further processing before arriving via Ethernet network to the Silicon Graphics Personal Iris (SGI) UNIX workstation for analysis.

3.2.1 386 Tower PC

The 386 tower contained a Matrox IM-1280 image processing board (Matrox Corp., Montreal) which was capable of performing a series of image processing functions. The board had 8MB of RAM enabling it to process and display multiple images at once. A software product, Visilog software (Noesis Vision Inc., France) functioned as an interface between the user and the card.

3.2.2 Silicon Graphics Personal Iris (SGI) workstation

The SGI workstation was the computer platform used for all the imaging evaluations, and ultimately the system used for the actual

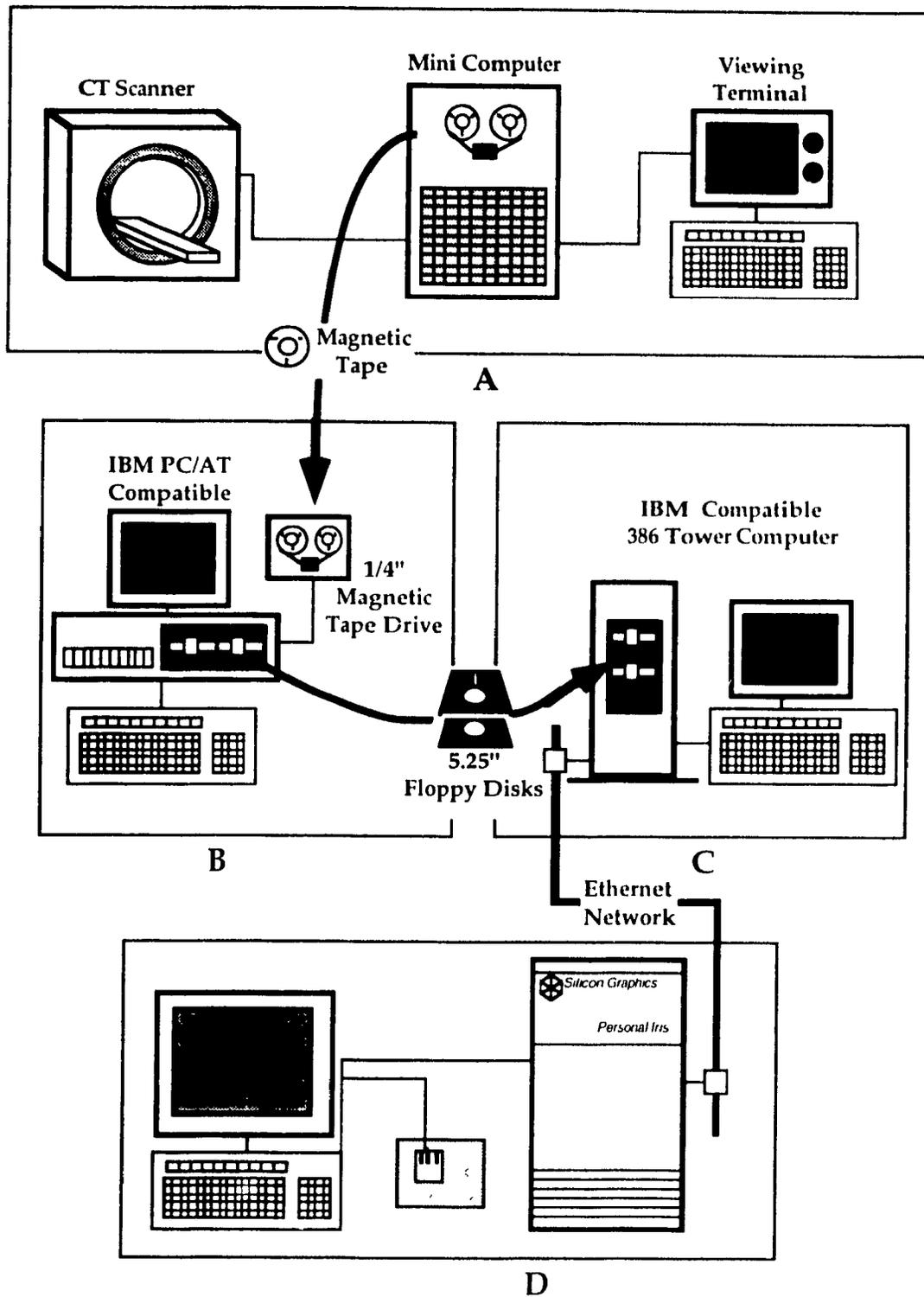


Figure 3.1: Overall layout of the imaging system (a) CT scanner (b) IBM PC/AT with magnetic tape reader and floppy drives to convert the images to binary format. (c) 386 tower PC with the IM-1280 imaging board, and (d) SGI UNIX workstation.

treatment planning system. The computer was an SGI Personal Iris 4D/20G with a 24 bit Z buffer display, 16MB of RAM and a 300MB hard disk. Silicon Graphics computers use a custom hardware/software graphics library (GL) which enables them to manipulate and display 3-D graphics quickly. The computer is equipped with an Ethernet network port and a 1/4" tape cartridge drive for file I/O.

3.3 Anatomical data

3.3.1 Acquisition

Two data sets were obtained from the CT scanner, one a series of slices of the head (13 slices), the other slices of the abdomen (9 slices). Both data sets were 512x512 pixels resolution with 12 bits in depth. These 12 bits contained both the image information, and separate text information which represented the patient and examination information overlaid on the image. The images were transferred to the IBM PC/AT where the software converted them to binary 8 bit images. These files were moved to the 386 tower and resampled to 256x256.

3.3.2 Surface rendering data preparation

Two software packages were used to examine surface rendering. The first was Alias Studio V3.0 (Alias Research Inc., Toronto, Canada) in which we examined the process of contouring and surface generation, as well as the visualization of the results. Alias Studio is used by designers in

manufacturing to create and visualize a variety of objects during initial design stages, and is also used by several universities for medical illustration. It was hoped that the techniques used to create objects for design visualization could be easily adapted for our purposes. The other software package used was SolidView (Silicon Graphics Inc , CA) which is primarily a finite element analysis program. This software required that polygon files be provided by some other source (e.g., by some other program or custom software) in a SolidView compatible file format.

In order to create the polygons to be stored SolidView formats, software had to be found or created to first generate the contours required from each slice, and then the polygons that accurately represent the surface boundary. The outer contours were obtained automatically using a modified version of software developed to segment ultrasound images[1]. The modification was done to allow for variable resolution, and a simplified output format.

There are many techniques available to convert a set of contours to a set of polygons[2, 3]. At this stage of evaluations, it is not necessary to generate objects more complex than the outer surface of a patient, so a simple algorithm was adopted. The algorithm assumed: 1) The number of points may vary from one contour to the next, and 2) That the object is simple enough that there will be only one contour per object per slice. The second assumption is a simplifying assumption that would not hold for complex objects that branch off such as fingers attached to a hand, but will hold for outer surfaces of the abdomen, and for the outer surface of the head

with a reasonable degree of accuracy. Figure 3.2 shows how polygons are generated from two contiguous sets of contour points.

The algorithm makes two passes around the contour pair. The first pass generates a series of triangles with the base of each triangle being a pair of points on the upper contour (Figure 3.2b). The third point is chosen from the neighboring contour such that it is closest to the middle of the line between the first two points. The second pass patches the holes left by generating triangles with the bases being on the lower contour (Figure 3.2c). In order to account for an unequal number of points between the two contours, the algorithm compares side 3 of each new triangle with side 1 of the neighboring triangle created in the first pass (Figure 3.2d). If they do not match, then an additional triangle is created with the base on the next set of points in contour two. This ensures that there will be no overlap between polygons, and no gaps left in the surface. Once the polygons are generated, it is a simple matter to store them in a format compatible with SolidView for analysis.

3.3.3 Reformatting of data

The data structure used in reformatting of data is the image file format used by the Visilog software residing on the 386 tower. In order to speed up response and minimize disk space requirements, the original 512 X 512 images were reduced to 256 X 256, which corresponds to at least double the resolution of the actual dose calculations. These images were then transferred to the SGI for use by the custom software which can accommodate the Visilog file format.

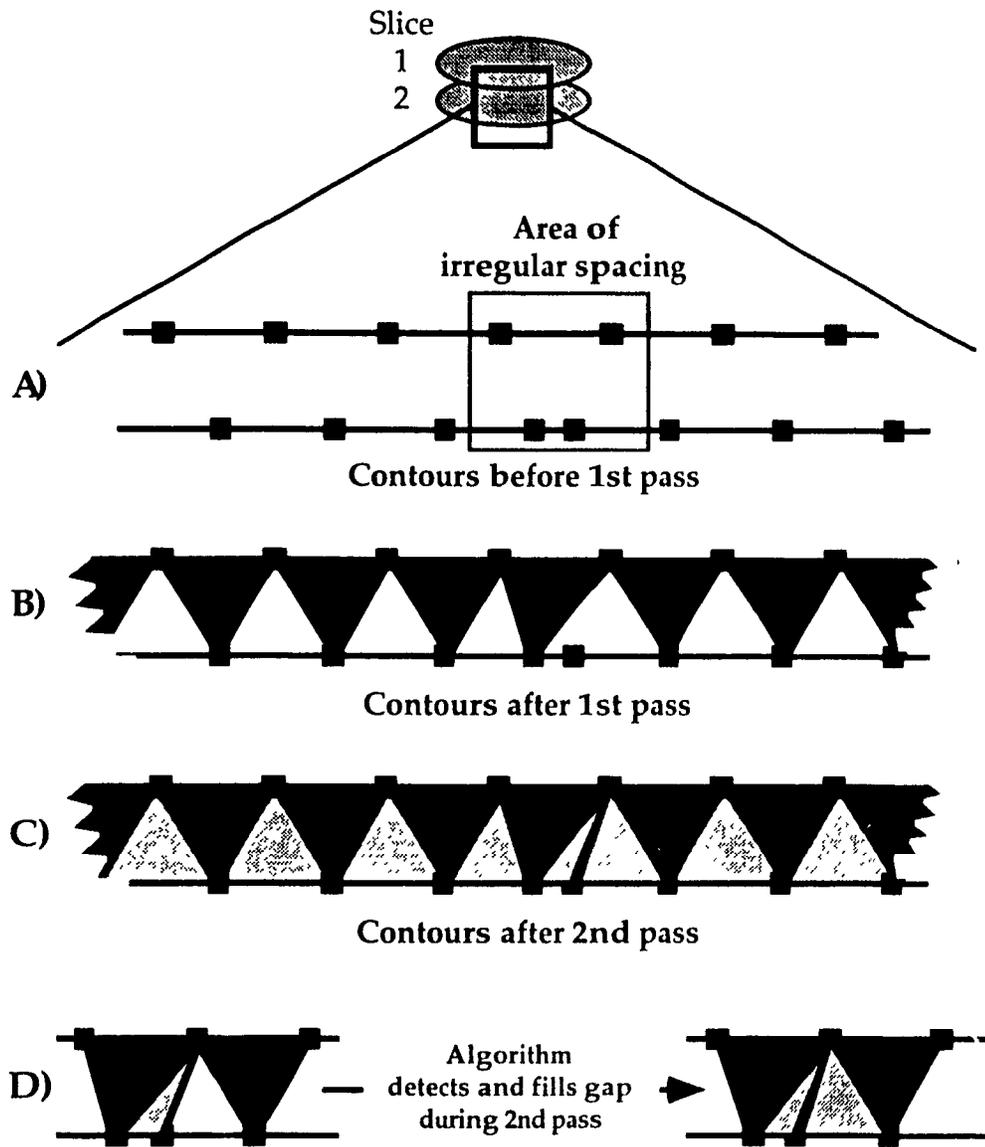


Figure 3.2: Creating a series of polygons from a set of contiguous contour slices

3.4 Simulating uniform density phantoms

Some of the later studies involved using water equivalent phantoms. In order to easily represent these on the computer, the drawing tools of Visilog were used to create a uniform gray box (128x128 pixels), and a blank image. By replicating these images, a volumetric data set of a cube was created.

Figure 3.3a shows how a series of blank and gray square images can be combined to create a cube within a volume. In order to create a spherical phantom, a series of circles of varying size with the radius calculated using the following equation:

$$\sqrt{(x \times t)_{(pixels)}^2 + r_{(pixels)}^2} = R_{(pixels)} \quad (3.1)$$

Where x is the slice number, t is the slice thickness in pixels, R is the radius of the sphere, and r is the radius of the circle of slice x . Figure 3.3b shows the combination of a series of circular slices to form a sphere.

3.5 Software development platform

In order to create custom software to perform a task, a programming development environment had to be selected. The SGI includes a proprietary graphics library (GL) designed to rapidly process and display graphics. These functions are available using either C or C++ languages, which are

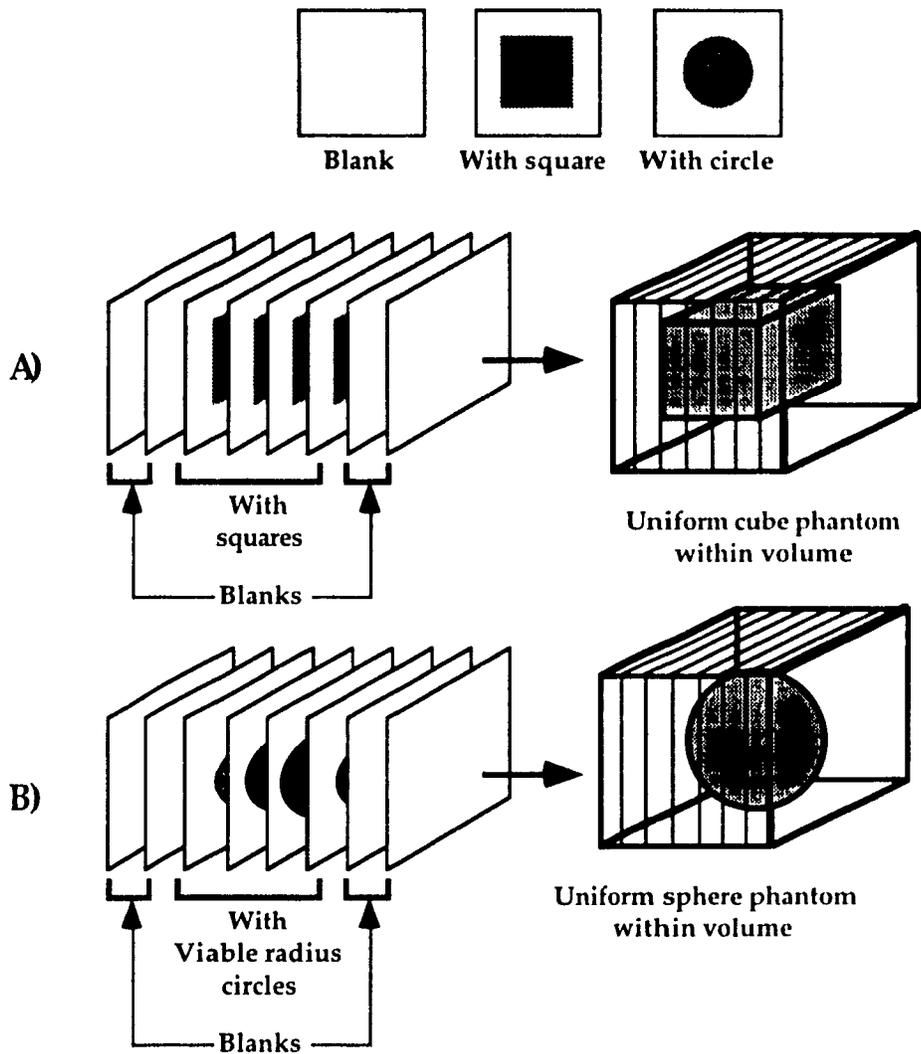


Figure 3.3: Creating a uniform phantoms. A uniform cube volume is created by replicating an image of a square repeatedly, and buffering the ends of the volume with blank images (surrounding air) . A spherical phantom was created by replicating blank and variable circle images.

also available on the SGI. Since all the software to be designed are calculation intensive and not suitable for object oriented programming (C++), standard C was used.

When writing software that requires an interface with the user, a set of tools that create a standard set of interface tools is desirable. A software library called FORMS[4] was used for many of these interface requirements. This provided some of the tools required to easily create an interface that was both intuitive, and consistent. The library also accelerated the process of software design.

3.6 Dosimetry Data

During some of the initial tests as well as in the final software, dose distributions had to be calculated given the target shape and beam parameters. During the initial examinations, these calculations were performed by an external program that saved the results in a file that was compatible with the imaging software. These routines were later modified and incorporated into the integrated treatment planning system. Although many calculation schemes exist (e.g., Milan-Bentley, pencil beam, Monte-Carlo etc...), a simple measurement based technique was adopted for evaluation purposes. Figure 3 4 shows the relationship between the variables in a typical Milan-Bentley[5] set-up for a curved contour.

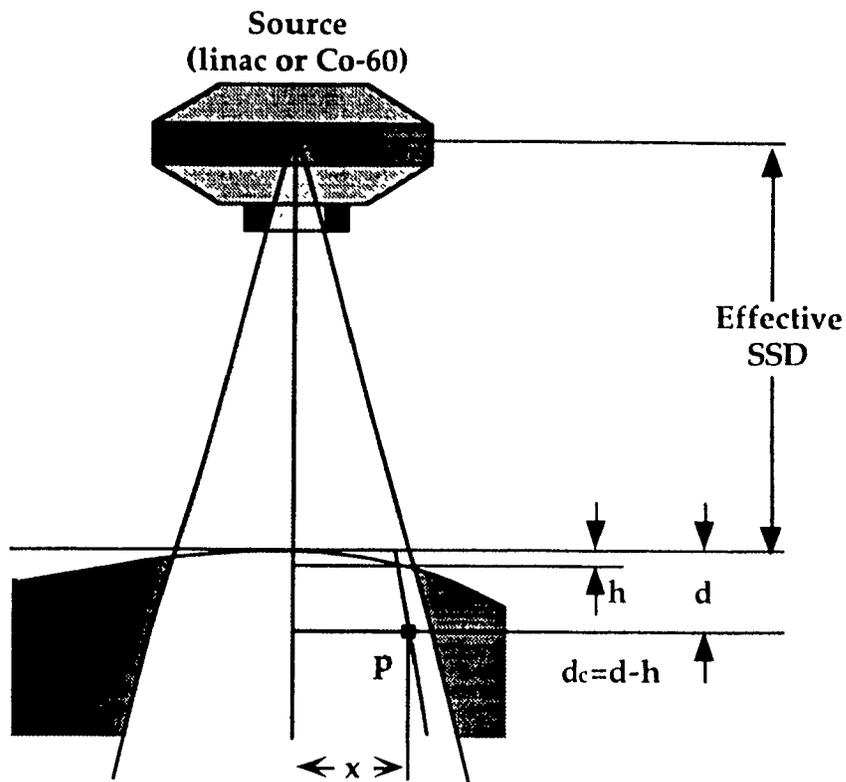


Figure 3.4: General set-up used to illustrate the relationships between the variables for calculation of dose within a volume.

The dose at any point within a volume can be represented by:

$$D(d, x) = D(d_m) * PDD(SSD, E, A, d_c, x) * OAR(SSD, E, A, d_c, x) \frac{(SSD + d)^2}{(SSD + d + h)^2} \quad (3.2)$$

where SSD is the source to surface distance (taken along the beam central axis), d is the effective depth of the calculation point, d_c is the true depth of point p correcting for the surface curvature, d_m is the depth of maximum dose along the beam central axis within the volume, E is the beam energy, h is the true surface to effective surface distance (d-d_c), and A is the beam area at the surface. PDD(SSD, E, A, d_c) is the percent depth dose for the given parameters, and OAR(SSD, E, A, d_c, x) is the ratio of the dose at point

p to the central axis dose at depth d_c . This technique can be extended into 3-D by measuring the off axis ratio for the complete area of the beam for any given depth, or by assuming square symmetry for that beam area. If square symmetry is assumed, the OAR term in equation 3.2 is replaced by:

$$\text{OAR}(\text{SSD}, \text{E}, \text{A}, d_c, x) * \text{OAR}(\text{SSD}, \text{E}, \text{A}, d_c, y) \quad (3.3)$$

where x and y are the coordinates of point p in the beam plane.

3.7 Coordinate Systems

In the final system, there are four distinct coordinate frames. The image or couch frame, the isocenter frame, the beam coordinate frame and the dose coordinate frame. Figure 3.5 shows the relationship of the first three systems. The dose coordinate frame shares the same origin and orientation as the image frame, with the only difference being that the grid spacing or resolution of the dose coordinate frame is different.

The coordinates (x, y, z) in the dose frame can be found given the coordinates in the image frame (x_i, y_i, z_i) by:

$$(x \ y \ z) = (x_i \ y_i \ z_i) \begin{bmatrix} \frac{1}{G} & 0 & 0 \\ 0 & \frac{1}{G} & 0 \\ 0 & 0 & \frac{1}{G} \end{bmatrix} \quad (3.4)$$

Where G is the scaling factor between the image and dose grids. A scaling factor of 2 would indicate that the dose grid resolution is half the resolution of the image grid.

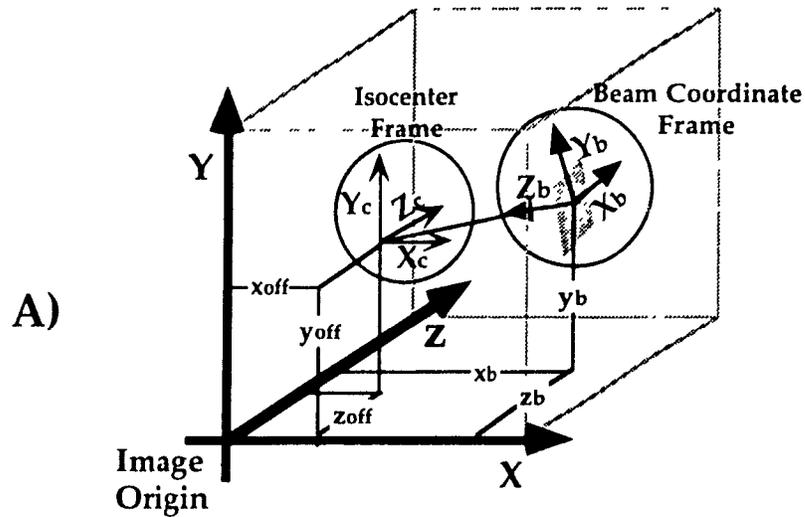
The final software requires the ability to map a point in the isocenter frame (x_c, y_c, z_c) to a point in the image frame. This can be achieved by multiplying the isocenter coordinates with the conversion matrix:

$$(x \ y \ z) = (x_c \ y_c \ z_c) \begin{bmatrix} \cos(\phi) & 0 & \sin(\phi) \\ 0 & 0 & 0 \\ -\sin(\phi) & 0 & \cos(\phi) \end{bmatrix} + (x_{off} \ y_{off} \ z_{off}) \quad (3.5)$$

Where the variables are those described in Figure 3.5. The most common transformation to be used is from a point in the dose frame (x_d, y_d, z_d) to the beam frame, which can be obtained using:

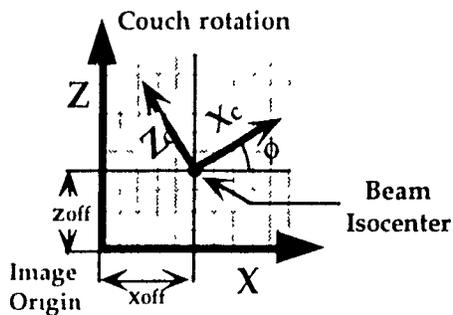
$$(x \ y \ z) = (x_d \ y_d \ z_d) \begin{bmatrix} \cos(\theta)\cos(\phi) & \sin(\phi) & \sin(\theta)\cos(\phi) \\ \sin(\theta) & 0 & -\cos(\theta) \\ -\cos(\theta)\sin(\phi) & \cos\phi & -\sin(\theta)\sin(\phi) \end{bmatrix} + (x_b \ y_b \ z_b) \begin{bmatrix} \frac{1}{G} & 0 & 0 \\ 0 & \frac{1}{G} & 0 \\ 0 & 0 & \frac{1}{G} \end{bmatrix} \quad (3.6)$$

Where (x_b, y_b, z_b) are the coordinates of the beam entry point in the image frame.

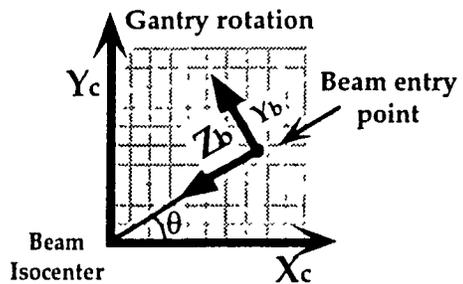


In Image Frame:

- **Isocenter coordinates:** $(X_{off}, Y_{off}, Z_{off})$
- **Beam entry coordinated:** (X_b, Y_b, Z_b)



B)



C)

Figure 3.5. Illustrations of the relationships of the three distinct coordinate frames. (A) shows the overall relationships between the image, isocenter and the beam coordinate frames (B) shows the role of the angle ϕ which is the couch rotation angle. (C) shows the gantry rotation angle θ between the isocenter frame and the beam frame.

3.8 Verification

In order to ensure that the results provided by the computer are accurate, experimental verification of the system is required. Because of the amount of data involved in a 3-D dose distribution, film dosimetry was used. The procedure followed the method outlined by Stern et. al[6] which involved using both film placed in the beam's eye view (BEV) plane and normalizing each film with the central axis dose.

3.8.1 Film dosimetry considerations

When using film as a dosimeter, it is required that an H&D curve be obtained to relate exposure or dose to optical density. The curve is unique for a given type of film (or even lot #), the processor used and the film measurement technique. If any of these parameters change during the experiment or between sessions, a new curve should be obtained. It is advisable to always obtain a new curve between sessions as the film processor conditions may vary slightly over time, due to temperature variations and changes in the chemistry concentrations through use.

3.8.2 Film parallel to the beam using ^{60}Co radiation

Figure 3.6 shows the general set-up and phantom configuration used to obtain the film calibration curve. The radiation source was a T780 ^{60}Co unit (Theratronics International Ltd., Ontario Canada) The phantom was a series of water equivalent polystyrene sheets, which enabled easy

variation of the depth of exposure by varying the number of sheets on top of the film. Kodak X-Omat V film was used due to its slow speed and the convenience of having them in a light tight "Ready Pak" envelope. The film processor was a Kodak X-ray processor used routinely in the Radiology department at the Montreal General Hospital.

A sheet of film was placed in the phantom at d_{\max} (0.5 cm). The field size was set to 10x10 cm at SSD which was itself set to 80 cm. The film was exposed for 0.07 minutes (un-corrected for shutter error). After exposure, the film was removed and replaced by a fresh one. It was also exposed, but for 0.12 minutes. This process continued for all the times listed in table 3.1. The films were processed and kept for later examination with a film scanner.

In order to ensure that the difference in the attenuation characteristics of the film and phantom do not affect the measurements, the film was placed perpendicular to the beam (BEV), as performed by Stern et al.[6]. These films can be scanned profiles extracted for comparison, or interpolated into a 3-D volume for comparison with the 3-D volume derived from the computer software.

With the SSD and field size unchanged from above, a sheet of film was placed at a depth of 0.5 cm in the center of the beam. Alignment with the beam was achieved by using the light field in the collimator head of the unit. The film was then exposed for 0.32 minutes and promptly removed. Other pieces of film were similarly exposed in different positions within the phantom as illustrated in Figures 3.7. The exposures were done separately

to avoid one film being exposed to a beam that may be perturbed by the presence of another film in its vicinity. Table 3.2 shows the film label numbers and their corresponding depths of exposure.

Once all the films were exposed, they were developed in the automated processor. The processed films were then ready to be scanned. The scanner was an AGFA ARCUS film scanner with the transmission light cover option, connected to a Macintosh IIfx computer via a small computer systems interface (SCSI) cable. The scanner could be controlled using a "desk accessory" utility software or driven directly from Adobe Photoshop software by use of a Photoshop "plug-in" software driver provided with the scanner. Figure 3.8 shows the set-up for film scanning, and the general principle behind the transmission film scanner.

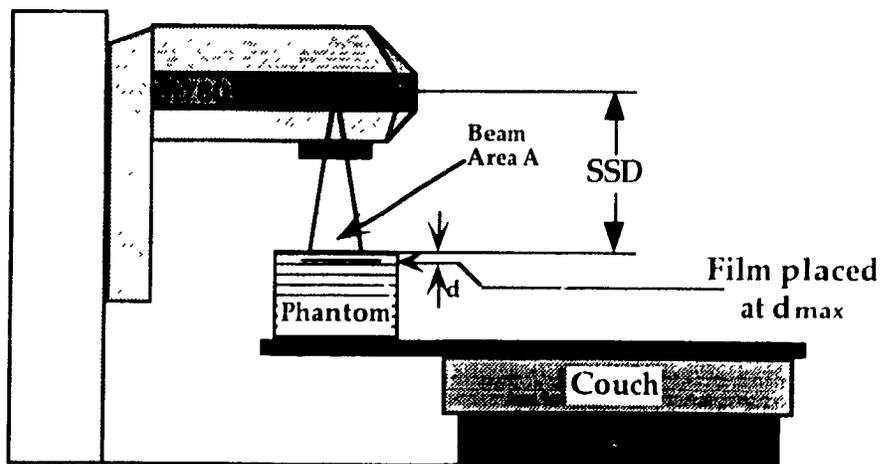


Figure 3-6: General set-up of the T780 Cobalt-60 unit and phantom for film calibration curve measurements.

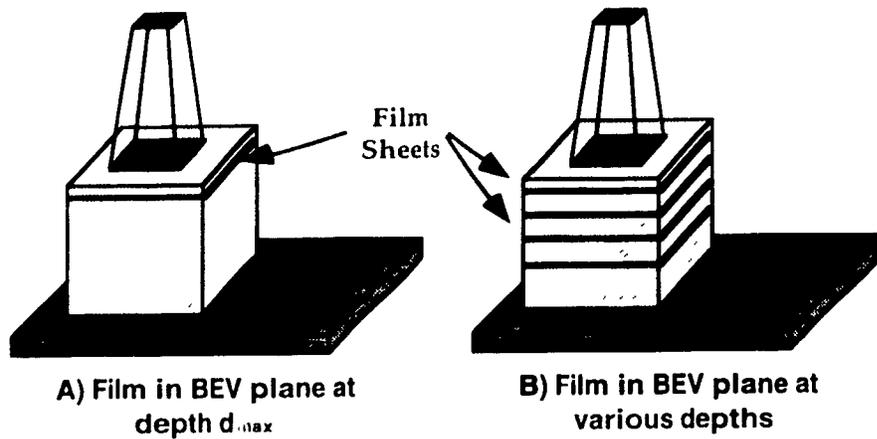


Figure 37 Film positions for both the film response curve, and for the BEV plane measurements

Table 3-1 Film label numbers and exposure times for film response curve for ^{60}Co radiation

Film #	Exposure (Min)
5-1	0.08
5-2	0.13
5-3	0.18
5-4	0.23
5-5	0.27
5-6	0.32

Table 32 Film label number and depths of exposure for films placed in the BEV plane. All films were exposed for time of 0.28 min (un-corrected for shutter error) in a 10cm^2 beam, with an SSD of 80cm

Film #	Depth (Cm)
6-1	0.5
6-2	5.0
6-3	10.0
6-4	15.0

Various gamma values, shadow, and highlight settings were tried with the films in order to obtain settings that allowed the scanner to detect the film fog while maintaining a good dynamic range throughout the range of possible film densities. The settings that gave the best results were Gamma=1, Highlights=30, Shadows=100. All the films were scanned at a resolution of 100 dots per inch, and with the above exposure settings.

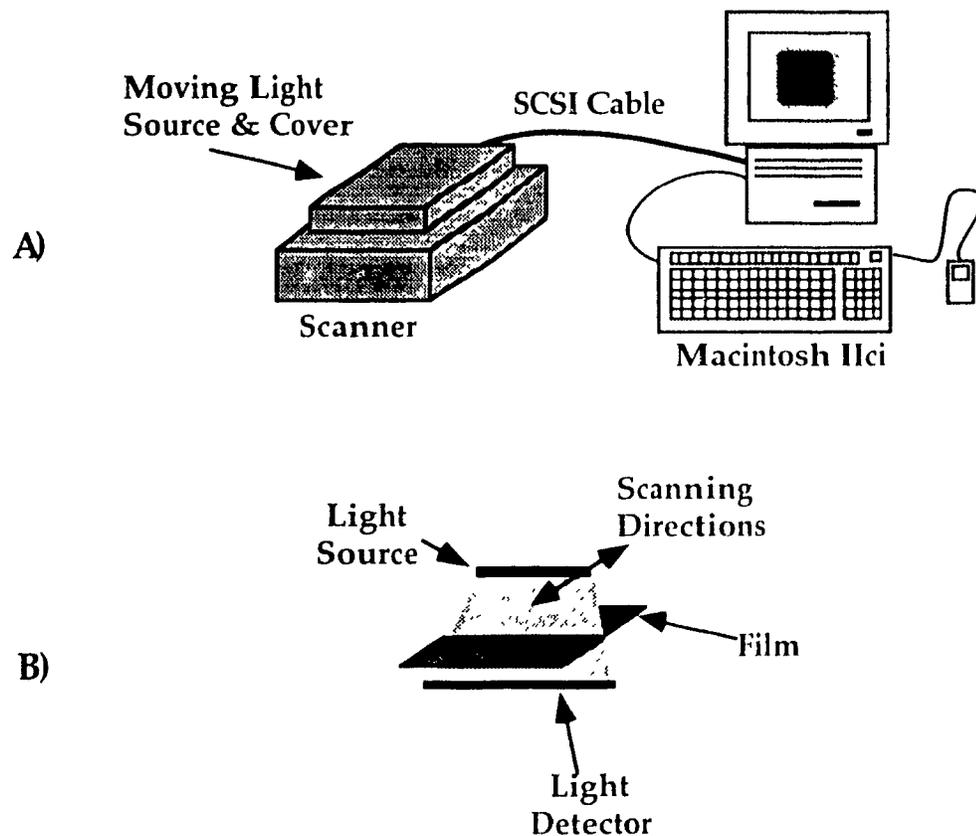


Figure 3.8: A) General set-up of the equipment used for film scanning. The computer used was a Macintosh IIfx and an AGFA ARGUS flatbed scanner with a transmission source option. B) The general principle of operation of a transmission scanner

In order to obtain the data required to generate a film response (H&D) curve, the pixel value at the center of each film was obtained by averaging the pixel values in the central region (1 cm²) of each response curve film

(films 5-1 to 5-6). In addition, the exposures films 5-1 and 5-2 were repeated in order to verify reproducibility in both the exposure and in the film processing. The results of these, and the resulting response curve are given in chapter 7.

When examining the films, it is important to preserve the relative geometries and scales of the films. In order to achieve this, each film was cropped to extract the exposed area from the rest of the film. Figure 3.9 demonstrates this procedure. These cropped films were resampled to a common size of 256x256 pixels by 8 bits. These files were saved in a binary (sometimes referred to as RAW) format for easy manipulation. A utility program was written in C on the Macintosh to extract profiles from the 2-D images, and store them in an ASCII column format to facilitate importing them into an analysis program. The program can extract any horizontal profile from files of varying resolutions.

The film exposure profiles can be converted to dose profiles in a few steps. The first step is to convert the pixel values of each profile to relative exposure using the film response curve. In order to increase the accuracy, each profile was then normalized with respect to the exposure value at the beam center for that particular film. This normalized profile can be multiplied by the percent depth dose at the depth of that film. The resulting profile is the relative dose profile for that depth. In chapter 7, we will present these profiles and compare them to those calculated by our system.

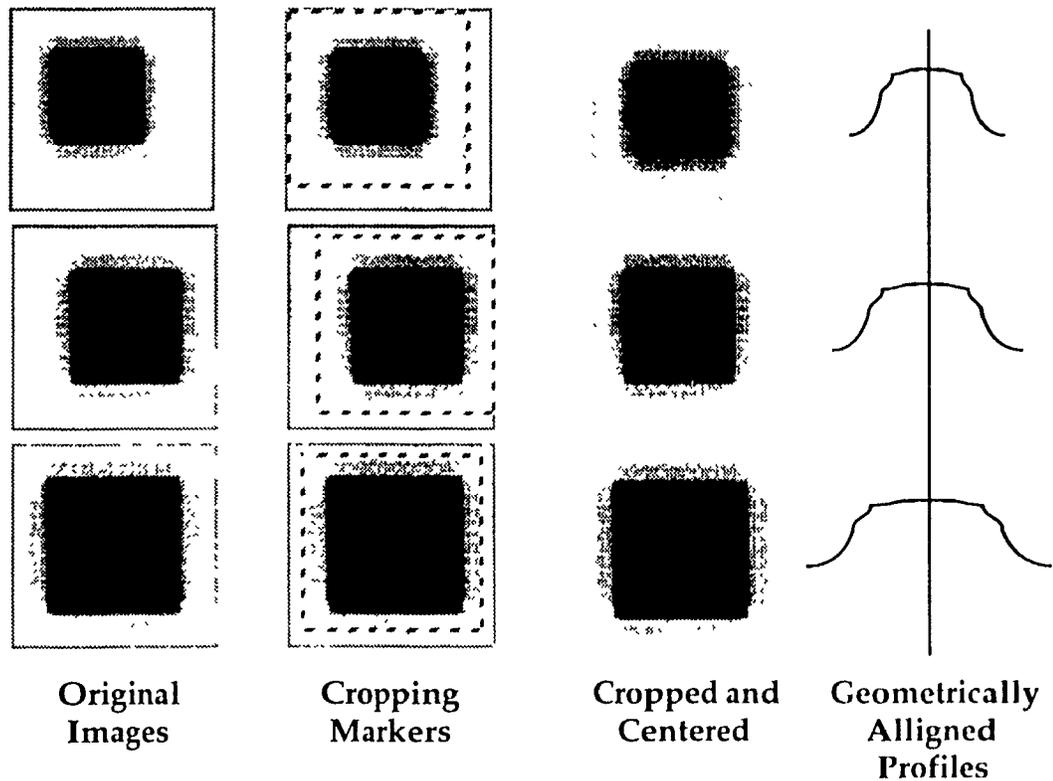


Figure 3.9: Procedure of cropping scanned images to obtain geometrical alignment for the profiles.

Our computer RTP program has an export feature that can export 2-D slices of either the image (anatomical, beam and dose distributions as seen on the computer screen) or a gray scale image of the dose distribution analogous to the exposed films discussed earlier. The dose images have a gray pixel value equal to the relative dose calculated at that pixel. This means that a pixel value of 100 corresponds to a relative dose of 100%. These

images were similarly cropped and converted to the binary format as performed on the film scans. Profiles were extracted using the same program as used for the binary film images, and plotted. These results are shown again in chapter 7.

References

- 1 Ian Crooks, McGill University, Montreal, Canada "Personal communication "
- 2 Boissonnat J-D., 1988 "Shape Reconstruction from Planar Cross Sections",
Computer Vision, Graphics, and Image Processing, **44**, pp 1-29
- 3 Sinclair B , Hannam A.G , Lowe A A , and Wood W W , 1989, "Complex Contour
Organization for Surface Reconstruction", *Computers and Graphics*, **13**, pp 344-349
- 4 Overmars M.H., *FORMS A C-library for dialogues*, Department of Computer
Science, Utrecht University, P.O.Box 80 089, 3508 TB Utrecht, the Netherlands
- 5 Bentley R E., Milan J , 1971 "An interactive computer system for radiotherapy
treatment planning". *Brit J. Radiol* , **44**, No 527, pp 826-833
- 6 Stern R.L , Fraass B A , Gerhardsson A , McShan D L , and Lam K L , 1992,
"Generation and use of measurement based 3-D dose distributions for 3-D dose
calculation verification", *Med Phys* **19**(1), pp 165-173

Chapter 4

Visualization of Anatomical Data: Technique Evaluation

4.1 Introduction

During the initial design phase of the treatment planning system, various visualization techniques to view the anatomical data were considered. In order to be able to make informed decisions as to the final techniques to be implemented, some preliminary work with these techniques was in order. This chapter will show some of the advantages and disadvantages of surface and volume rendering by first going through the steps involved in creating a medical image with surface rendering, then by examining the computer requirements to implement real or near real time volume rendering. The surface rendering study will begin by extracting the surface boundary information from the source anatomical data using a commercial high quality rendering package and viewing the results. During this process, relevant points regarding the implementation of this technique to treatment planning will be discussed. Another technique which involved writing custom software to automatically extract surface contours and create a polygon set compatible for real time display will also be described. After examining volume rendering, a subset technique of volume rendering, reformatting of data will be presented.

4.2 Contouring and Surface Rendering Using High Order Equations

4.2.1 Contouring the anatomical data

In order to represent the shape of an object using surface rendering, the proper information must be supplied to the rendering software. Alias Studio can create or accept non uniform rational B-spline (NURBS) files to mathematically describe the surface of an object. In order to create a three dimensional object from a series of CT images, each image must be contoured, and these must be interpolated into a surface. Figure 4 1 shows a screen on the SGI running Alias Studio with a typical CT image during contouring.

A program was written to convert the Visilog format CT image file to a "pix" format used by Alias. With the image displayed in the background, standard drawing tools were used to draw the contour of an outer surface. Since some of the surface generating algorithms require that each contour have the same number of points, the most complex outline was selected first. To speed up the process, this contour was duplicated and adjusted to fit the next outline. This process of duplicating and adjusting contours was performed for each slice of the study. In order to space these contours in the third dimension, the slice thickness of the study was noted and converted to thickness in pixels. The contours were spaced according to the thickness before attempting to generate a surface. Figure 4 2 shows a perspective view of all the contours before creating a surface.

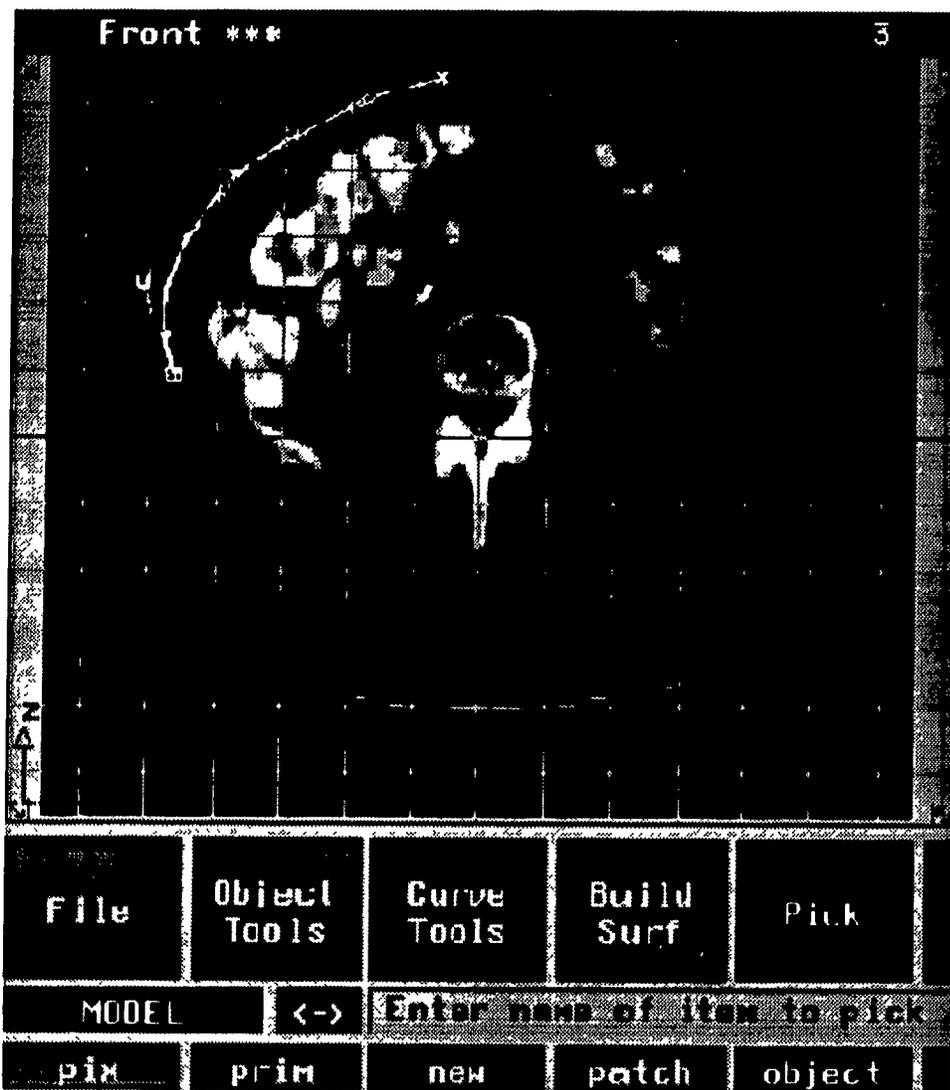


Fig 41 Example of Alias Studio software with a background image to be used as a template for contouring

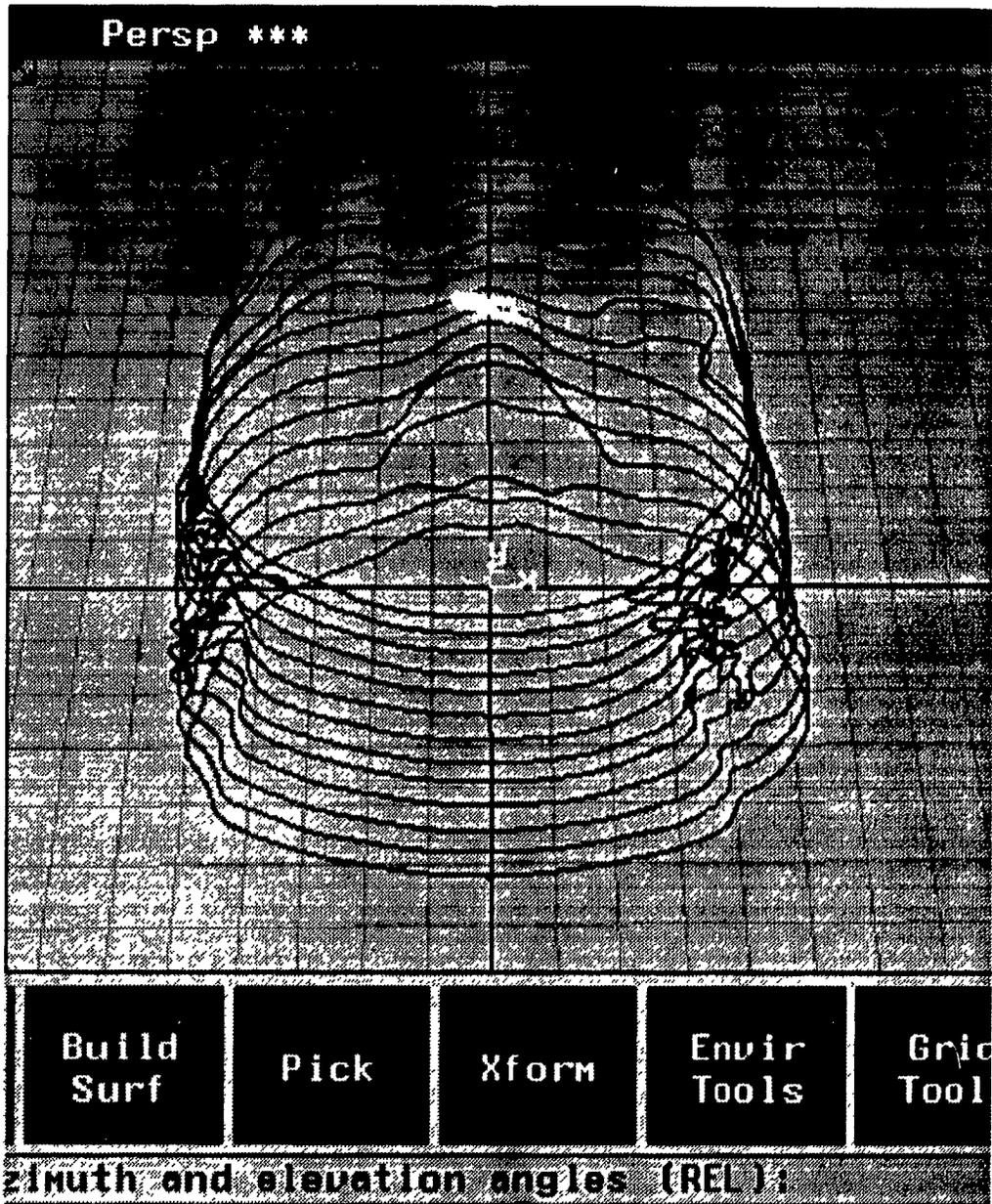


Figure 4.2: Perspective view of 13 contours of a head obtained using CT data as source material.

4.2.2 Generating a surface using the “patch” technique

Alias Studio can create surfaces using a variety of techniques. The technique to be used is dependent on the nature of the lines used to describe the surface and the overall shape of the object to be represented. In the case of an outer contour surface, two techniques may be suitable: the patch and the skin techniques. The patch technique is designed to create a surface by creating a series of bridge lines between two guide lines. Figure 4.3a shows the wire frames of the surfaces created when each pair of contour lines were patched. Figure 4.3b shows a quick-render of the wire frame.

The resulting image using the patch technique is obviously not suitable for the types of data encountered. The technique fails in two areas: First, there is a lack of continuity of the surface from one slice pair to another, which renders it almost indistinguishable. The other problem lies in the way Alias builds the patch surfaces from the points along the contour. Figure 4.4 shows how the algorithm fails. Alias takes one point on each curve and joins them. Then it follows along both bounding curves and joins each pair of points with a line. As more point pairs are joined, their positions with respect to each other on the curves may shift due to a different number of points or a change in complexity in one of the curves. This shift tends to show up as a wrinkle or wave distortion on the surface of the object.

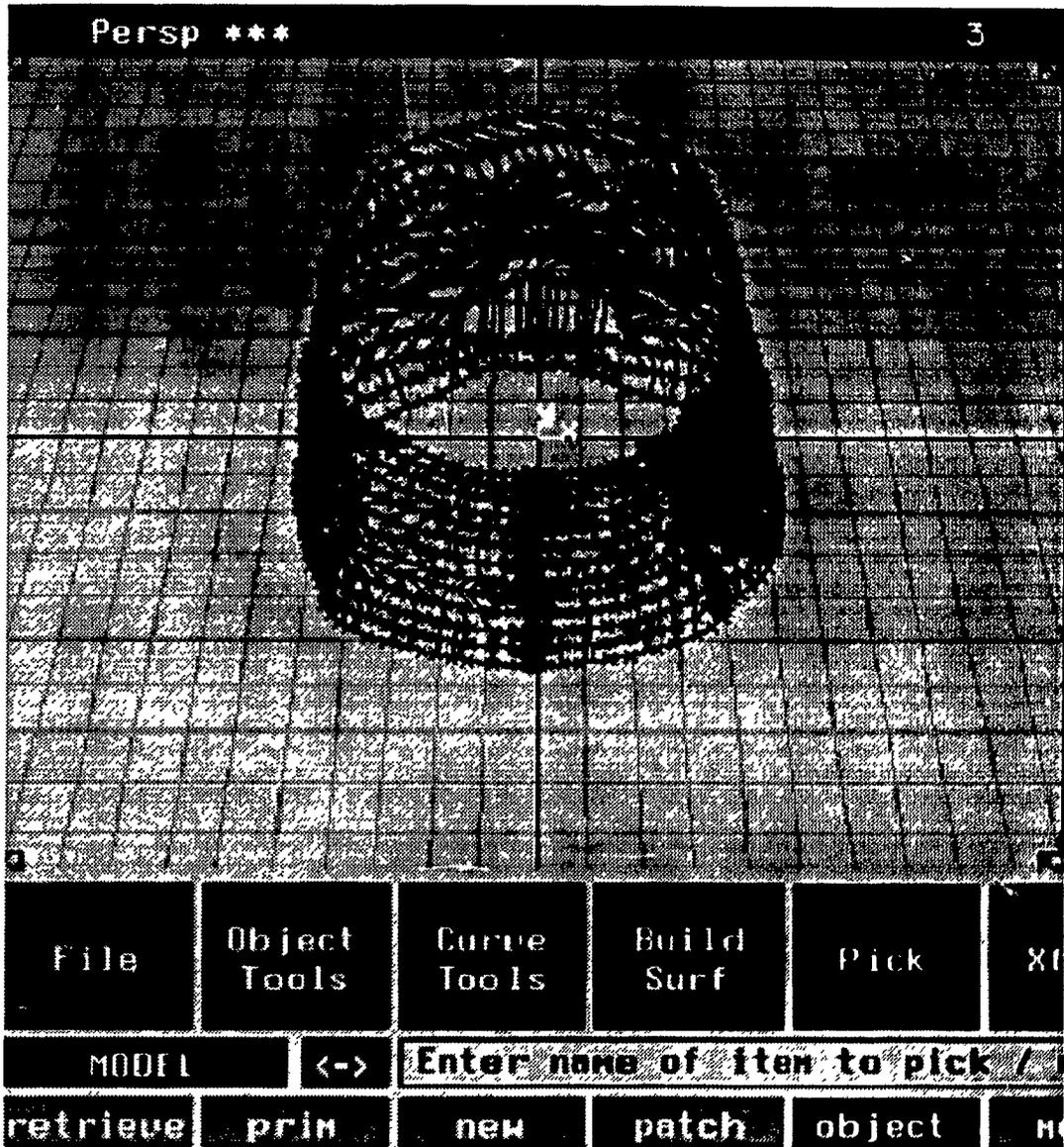


Figure 4 3a Perspective view of a wire frame representation of the head The wire frame was generated by using the "patch" technique successively on each pair of contours

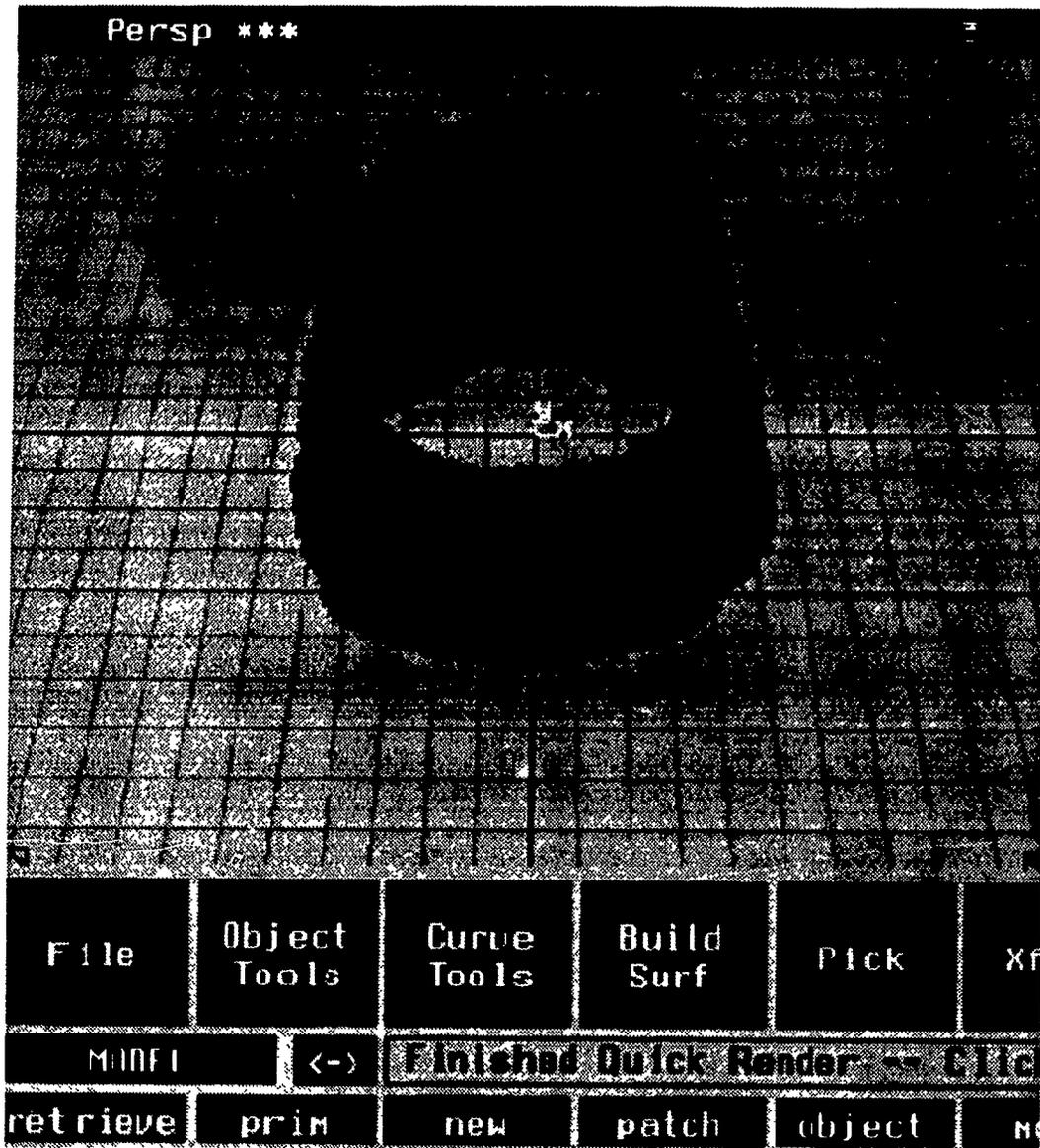
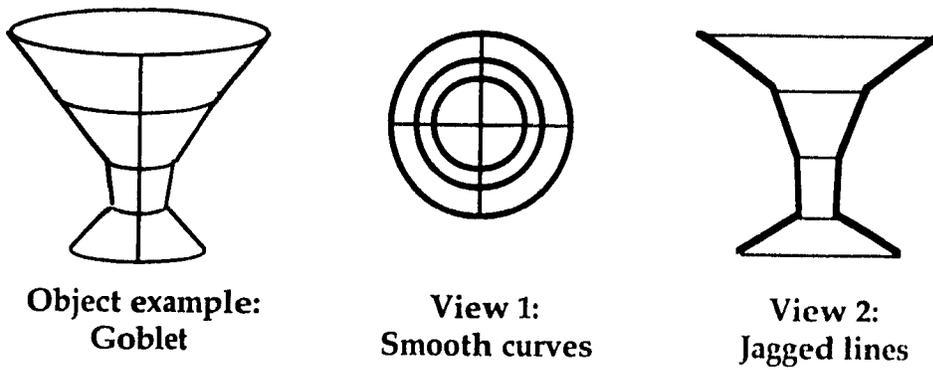
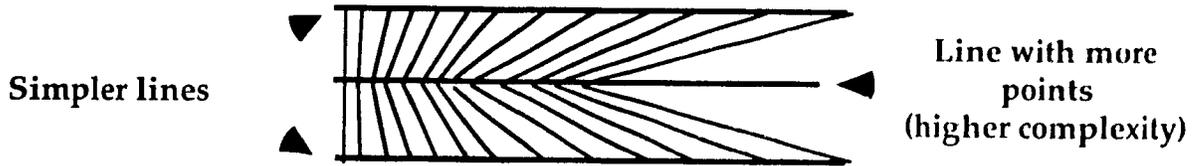


Figure 4 3b Alias "Quick Render" of the patch set. This technique generates the least number of points because it only joins two contours at a time, but the resulting image lacks continuity between the surface patches



A)



B)

Figure 4.4: Illustration of how the patch technique is not suitable for this type of object generation. The two reasons are A) The lack of continuity between patches. B) The algorithm's inability to adequately handle curves of different complexity (i.e., the number of points to describe any section of it). The latter can cause surface wrinkle artifacts in the image.

4.2.3 Generating a surface using the “skin technique”

In order to obtain an image that has smooth transitions from slice to slice, the skin technique was examined. This technique uses information from all the contours when creating the surface boundary lines, resulting in lines with a high order of accuracy. Figure 4.5 shows the resulting surface from the same set of contours as used in the patch example. Note in Figure 4.5a how many lines were created using this technique. Although the final rendered image is very realistic (Figure 4.5b), the inefficiency of the algorithm results in extremely poor response time. The inefficiency in the algorithm can be explained in the following manner. Figure 4.6a shows 3 lines (with their vertex points) that are about to be “skinned”. Figure 4.6b shows the connecting lines that are created in the skinning process. Note that although each line has the same number of points, and that the points lie at approximately on the same place on each line, the algorithm projects each point onto the other lines and creates new points at each projection point, regardless of how close that point comes to an already existing point. The results are that the complexity of each line, that is the number of points that define any given line, is unnecessarily increased depending on how many other lines are participating in the skinning process. If the 12 slice set shown here is a typical set, this algorithm used 12 lines to define a section of surface where 1 would have been sufficient.

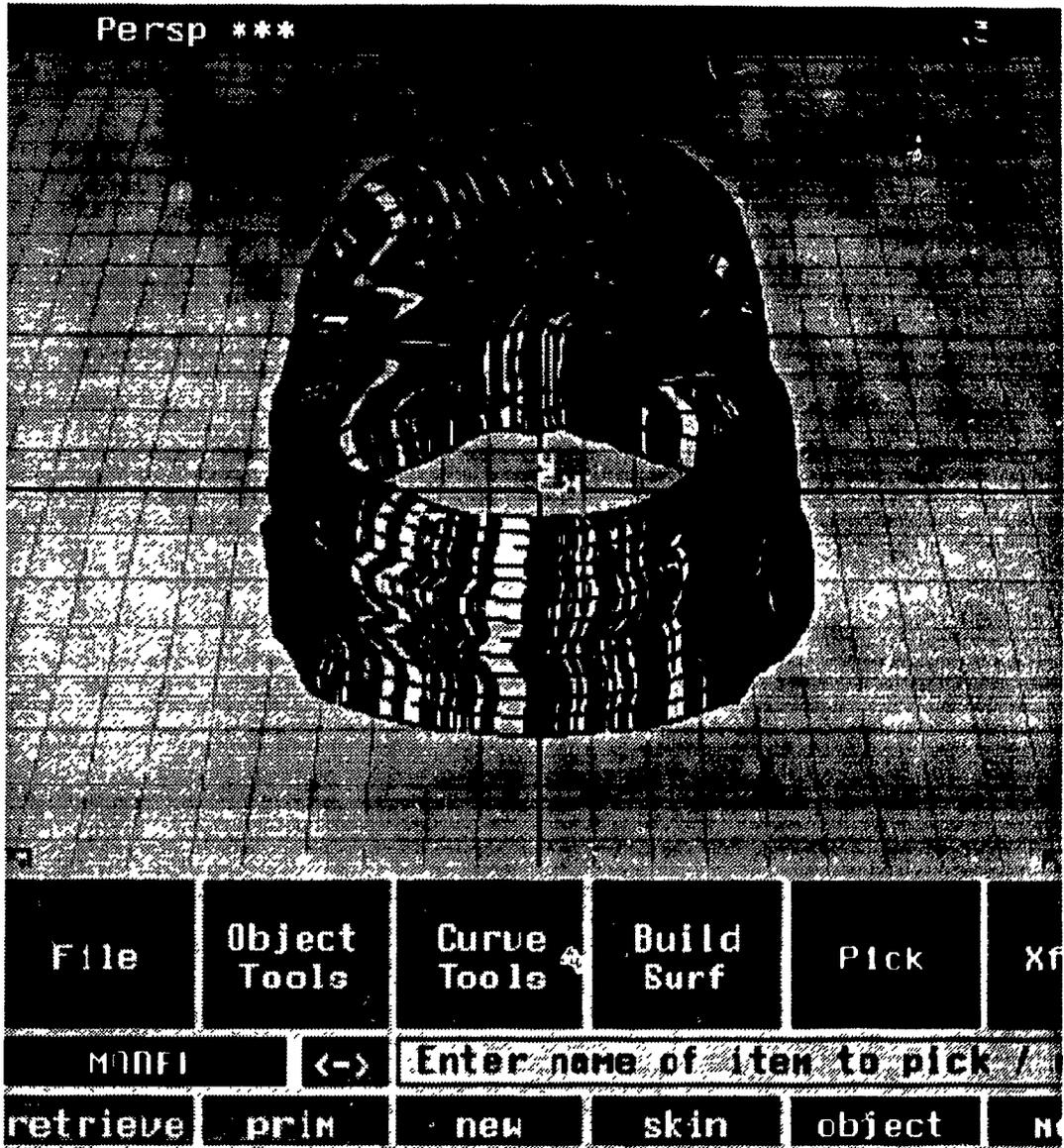


Figure 4.5a The wire frame image of the head using the skinning technique. Although the resulting image quality is good, the amount of lines created to represent the surface shows a lot of redundancy.



Figure 4.5b Ray trace image of the "skin" surface shown in Figure 4.5a. The accuracy is good but there are too many redundant lines.

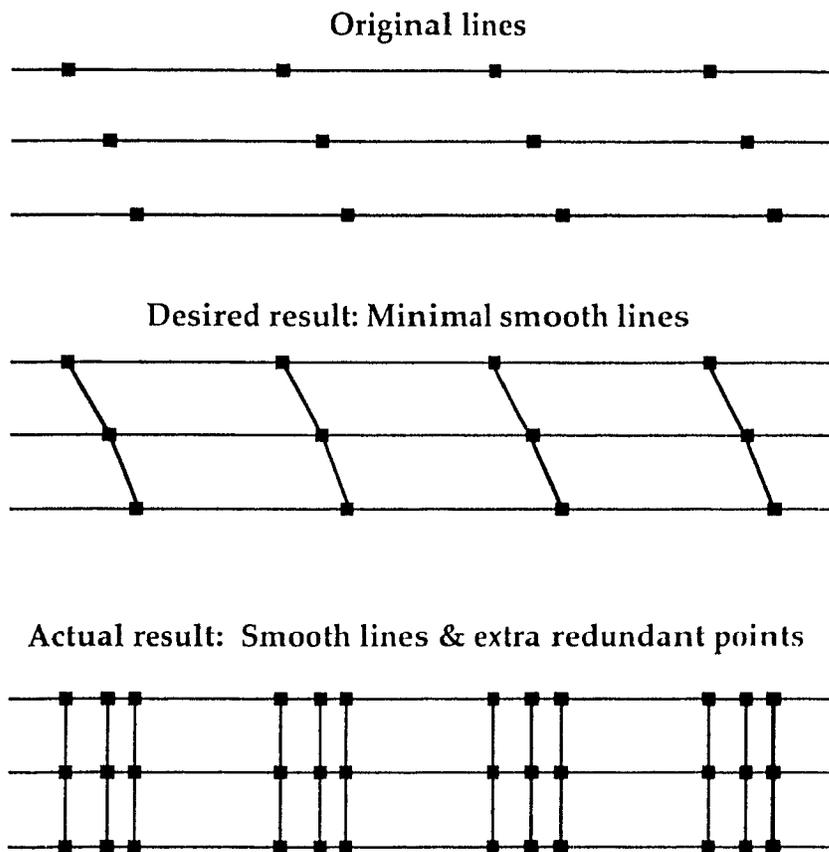


Figure 4.6: Illustration showing the problems encountered using the "skinning" technique

After examining these two techniques, it can be stated that neither of them is suitable for our needs given the type of data available. The patch technique minimizes the number of points and lines used to define the surface but lacks the continuity between slices required to properly represent the data. Skinning provides very accurate images but the algorithm clearly was not designed to create surfaces from the type of data set provided used here. The resulting wire mesh contains an order of magnitude more lines than is required to represent the data.

4.3 Automatic Image generation

Given the results of the previous section, a different technique is needed to generate and represent the surface boundary information. This technique must be more suitable to the type of data given and capable of representing it accurately. Another commercial (public domain) surface rendering software package available to us was SolidView. Instead of using high order polynomials or NURBS to smoothly represent data, this package uses a set of polygons. In order to view images in this manner, that polygon set must be created.

4.3.1 Automatic segmentation of surface boundary information

Since software had to be written to segment the image, attempts were made to provide some automatic segmentation of data. Although it is not yet possible to automatically segment anatomical data within the volume of interest, particularly low contrast objects such as a tumor, automatic

segmentation of the outer surface was attempted. All the contouring was performed on the 386 tower using software that was a combination of command scripts running under Visilog and the IM-1280 board, and a custom C program to format the output.

4.3.2 Creation of a polygon set from contour points

Once a set of points were obtained for each contour and stored in a single image file, software was written to create a set of polygons stored in a file format that would be compatible with SolidView. The amount of polygons created depended on the resolution of the contours provided. Figure 4.7a shows a SolidView wire frame view obtained by using the polygon creation program.

4.4 Surface Rendering using Polygonal Data

4.4.1 Surface Rendering Using SolidView

Once the polygonal data was obtained, surface rendered results could be viewed. The image could be manipulated in real time with only a slight delay. This delay is dependent on the number of polygons used to represent the surface of the object. The quality of the image also depended on the quantity of polygons used. In order to provide a reasonable looking image and preserve the real time manipulations, a compromise resolution was selected, which was around 5000 polygons per image. Figure 4.7b shows the image displayed using a skin user material to dictate the surface

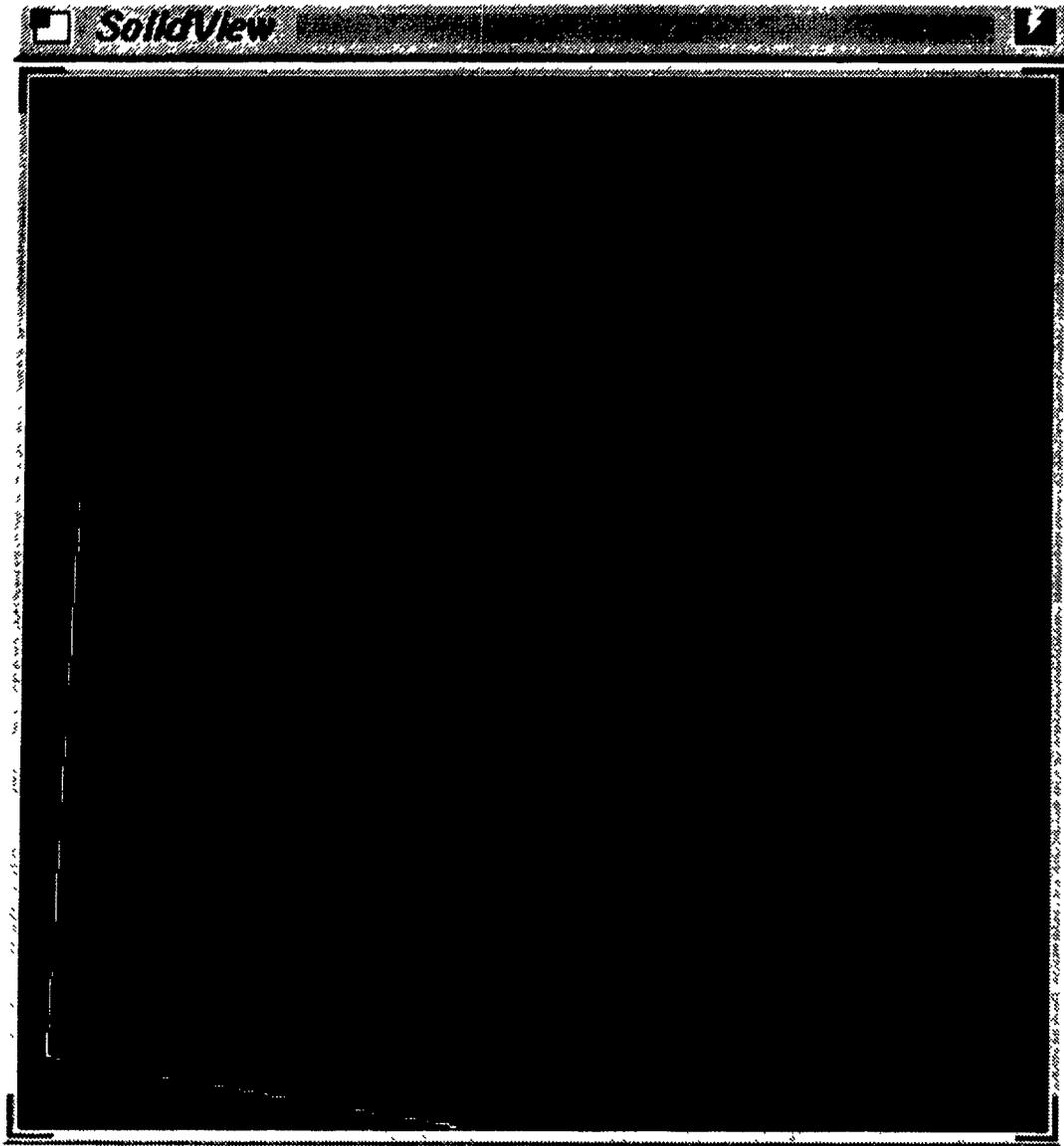


Figure 17a Wire frame view of the head generated using our segmentation software, and our polygon generation software. Although the accuracy is good, one can often notice a vertical seam where the contour is closed.

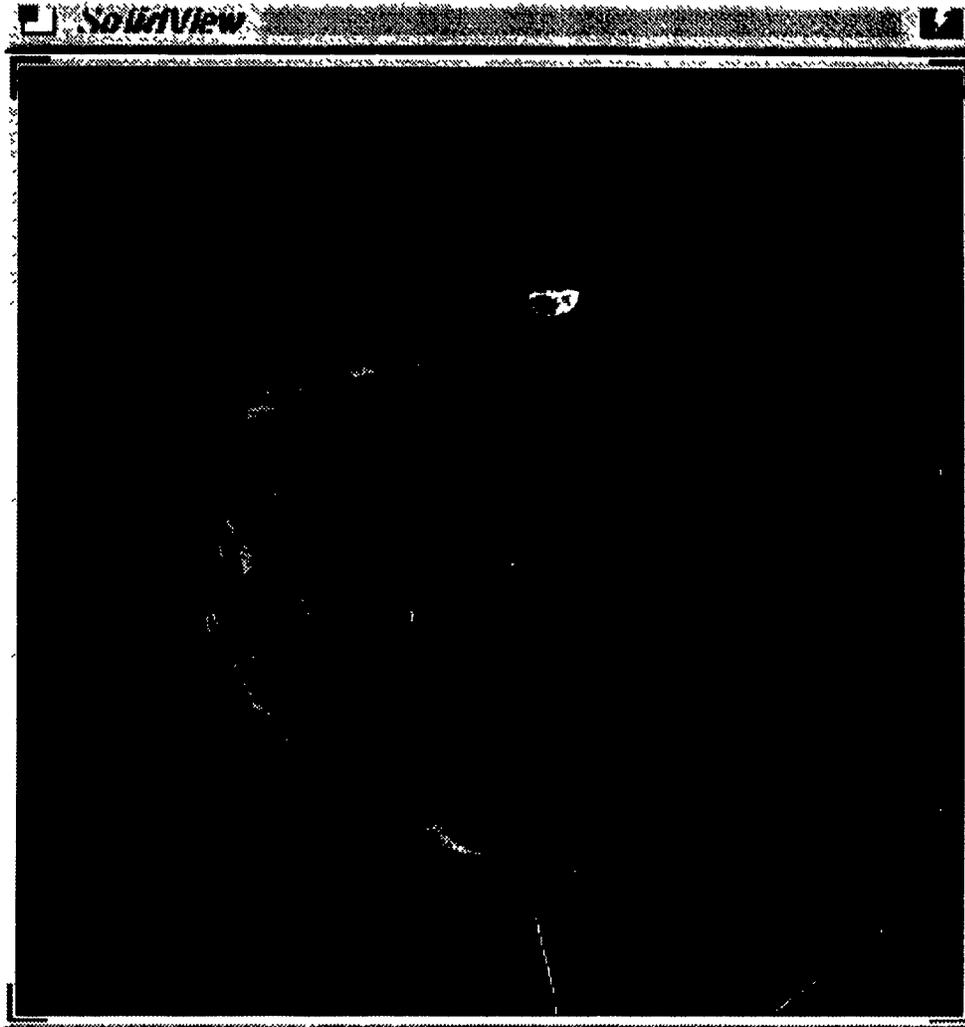


Figure 4 7b Split view of the surface rendered image and the wire frame used to generate it. The response of the system during manipulation shows that it would be suitable for real time analysis The roughness of the image suggests that more slices are needed for an optimum image

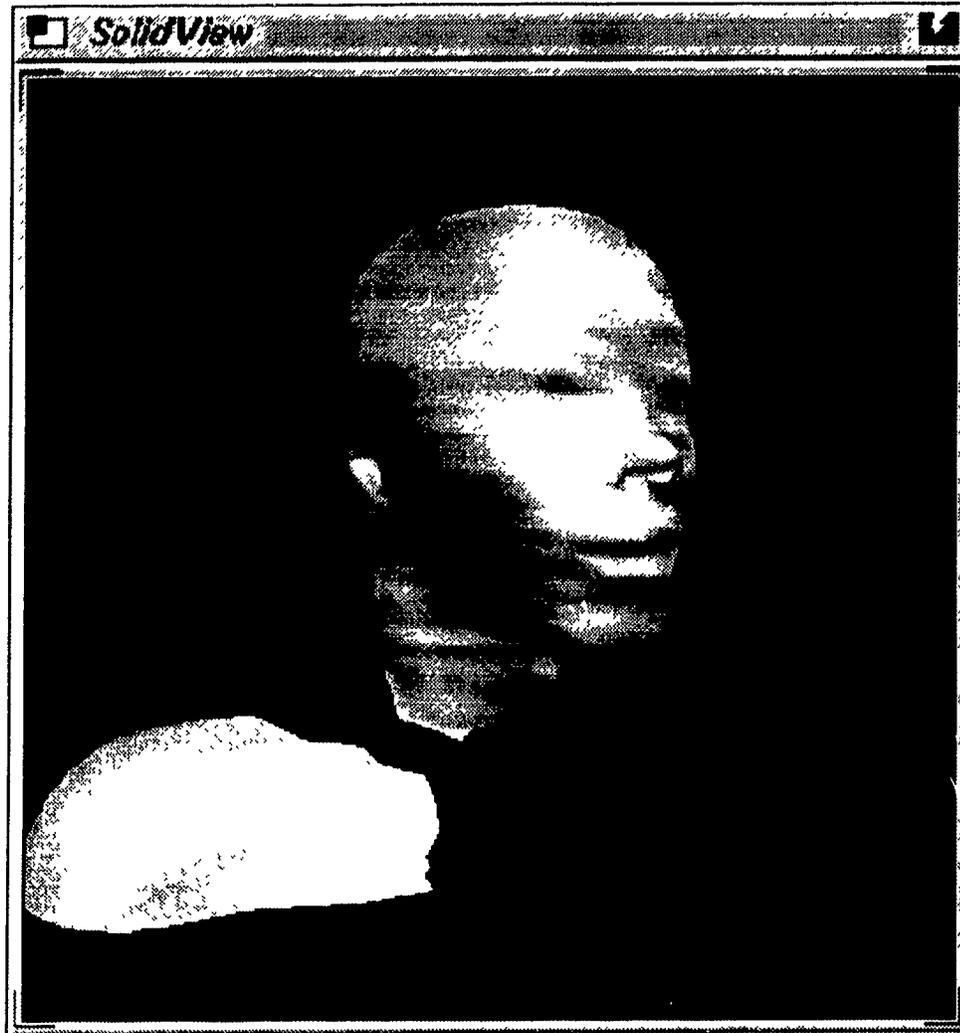


Figure 4 7c An example data set provided with the SolidView software package shows the potential of surface rendering for visualization. No fully automatic segmentation technique is presently available to perform this task, which makes the value of this type of system given the work involved, subjective.

characteristics, and Figure 4.7c shows an example image of an upper body. This data set was provided with the SolidView software to demonstrate its surface rendering abilities.

After performing these tests, it was determined that this technique could be suitable for certain applications in treatment planning given its information quality and its capability for real time manipulation only if automated or rapid manual segmentation of the pertinent structures were possible.

4.5 Volume rendering

Volume rendering promises to provide the most informative and most flexible 3-D display, due to the availability of all the data acquired for a given volume at all times. This contrast greatly with surface rendering which undergoes a contouring stage which discards most of the information within the volume. Despite this advantage, it has serious drawbacks in terms of memory and processor requirements when manipulating any of the display parameters.

Although surface rendering can provide a simple and clear display of the spatial relationships between the structures that have been contoured, some of the detail may be lost by contouring. Volume rendering provides a display that consults every data point within the volume when generating an image. The result is an image that truly depicts the volume of interest with all the structures within it. To generate a clear image,

manipulation of the parameters which control the visibility of each pixel is required in order to highlight desired structures and subdue others.

This display capability has an inherent cost. In a typical volume data set generated by a CT scanner, 512x512 pixels by 30 slices may be generated. This results in 7.86 million points to be stored and manipulated at all times. With an 8 bit pixel (256 colour or gray values), this results in a data storage requirement of over 7.8 MB per volume set. In a graphics workstation, storage of this volume is not atypical, but in order to manipulate the display, each one of these points must be consulted. If the data set is reduced to 256x256 pixels, the volume set would still contain 1.97 million points. Even with the reduced points, an extremely fast computer, or parallel processor(s) would be required to be able to manipulate and display it in a real time fashion. At this time, no reasonably affordable system is available to perform this task with acceptable speed.

In order to obtain a display that maintains the entire volume for viewing, reformatting of data was examined. This technique preserves all the volume data in memory for display, but only 2-D subsets of this data are displayed at any given time. By simultaneously generating several different views that are all subsets of the volume, a 3-D feel to the volume may be obtained, without the processing requirements described earlier.

Figure 4.8 explains how the volume data is reformatted to create the 3 orthogonal views. The axial view (a) simply cycles through the slices originally provided by the user. Each axial slice image displays the full

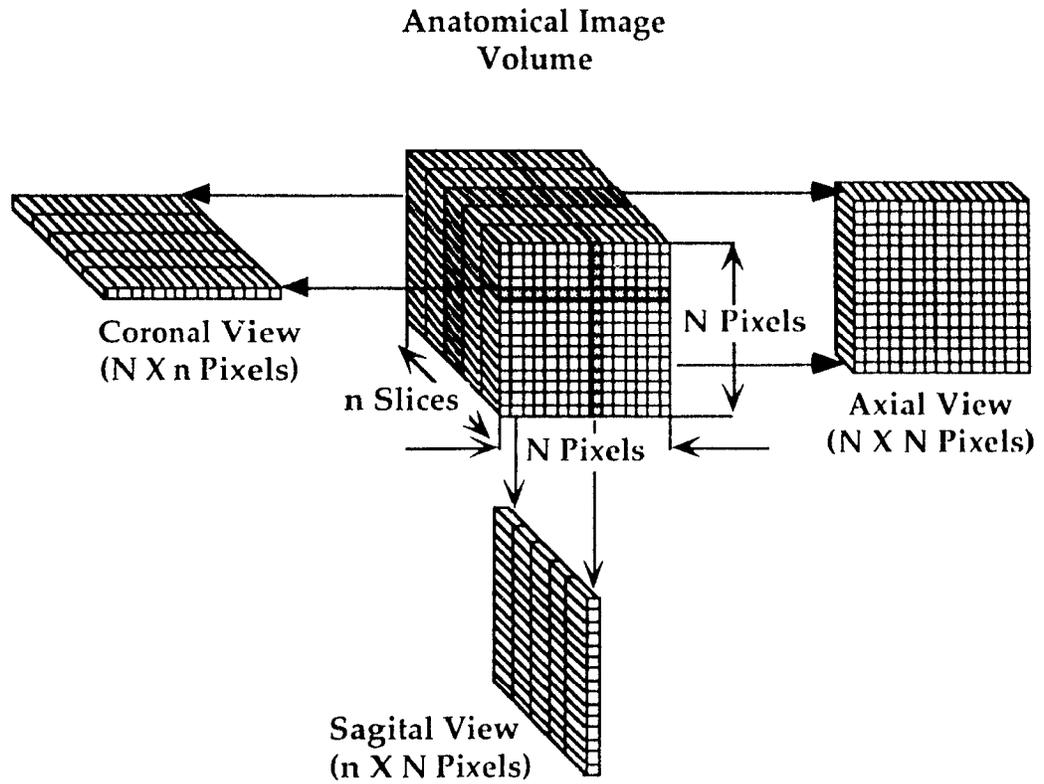


Figure 4.8: Example of how a 3-D volume data set can be segmented to display 3 orthogonal views. This technique is commonly referred to as reformatting of data.

256x256 resolution. Any given coronal slice (b) will only have 256x#slices pixels to display, so each slice must be given a thickness in pixels according to the CT slice thickness, resulting in an a display with the proper scale. The sagittal view (c) is handled in the same way as the coronal view, except the image is stretched in the vertical plane according to the slice thickness.

4.6 Summary

We have examined and attempted 3 different implementations of surface rendering, and have found that among these, polygonal data representation is best suited for viewing of anatomical data. Although this technique works well, we have also found that any surface rendering technique requires extensive work by the user in extracting the boundary information for each structure, even when much of the work is automated. We have also found that real or near real time volume rendering is not practical at this time given the amount of data involved, and the capabilities of typically affordable computers. Although full volume rendering is not practical, we have found that reformatting of data provides much of the advantages of volume rendering, namely the constant availability of all the volume data for examination, while still enabling real time manipulation of the image.

Chapter 5

3-D Treatment Planning System

5.1 Introduction

Chapter 4 described the work performed in evaluating the various techniques available for viewing anatomical data. After an understanding of the viability of each technique has been obtained, the actual treatment planning system was constructed. In order for the system to function effectively, the visualization techniques used for each data set had to satisfy two criteria: To be able to represent the data set clearly and rapidly, and be able to do so without interfering with other techniques that may be used simultaneously. In order to accomplish this, some compromises may be required in order to achieve the best balance for the system as a whole

This chapter will describe the functions implemented in our interactive RTP system. This will be accomplished by presenting an example anatomical data set (spherical phantom) and going through all the steps involved in using this data for a treatment plan. Particular emphasis will be placed on describing how the system behaves in an interactive environment. This chapter deals mainly with the functionality of the software. A detailed description of the data structures and the algorithms used will be reserved for chapter 6. Figure 5.1 shows the basic steps involved in using the treatment planning system

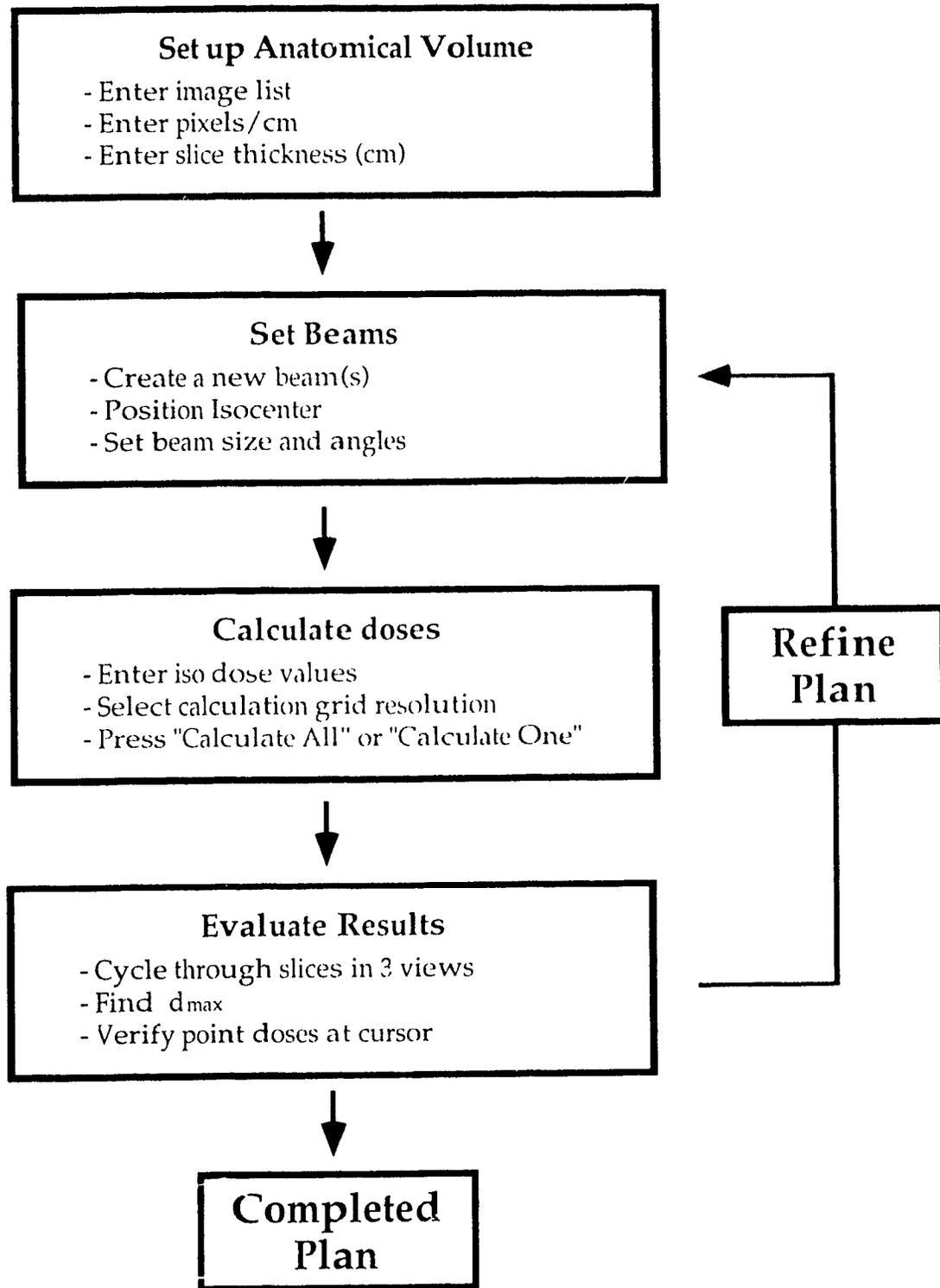


Figure 5.1: general flow diagram of the operations performed for a typical treatment plan

In the following example, a project file will be created and images that comprise the anatomical volume (uniform sphere phantom) entered. Once the volume is set up, a beam will be placed to simulate a treatment. This will provide the opportunity to demonstrate the beam placement tools, and how the visualization techniques work together. Once the beam is in an initial position, isodose values will be entered and a dose calculation will be performed. All the options available for dose calculation and display will then be presented. The isodose values will be examined and the plan will be refined by modifying beam parameters. Another calculation at a higher resolution will be performed, and these results will be evaluated using a variety of utilities.

5.2 Starting the software

Figure 5.2 shows the Viewer main window. It contains general controls and menus that are in use during the entire operation of the program. The software was designed to have multiple windows that open only when required, which reduces clutter on the screen and helps make the software more intuitive and easier to use. The top text line contains the name of the current project in memory. The software can be started up through a standard UNIX shell by typing

```
viewer <file name>[Enter]
```

If a file name is specified, the software will start up, and immediately load a previously saved project of name *filename* assuming that the UNIX

shell is at the directory where the file is kept. If no file name is specified, the program will initialize the required data structures to create a new project.

Other general controls include a series of buttons that govern the function of the mouse. The mouse can be used to perform any of a series of functions over the three orthogonal views. The user selects from these by pressing on any of these buttons. Note that the program will not allow the user to select any beam related function (e.g. Iso-Center) when the beam window is not open.

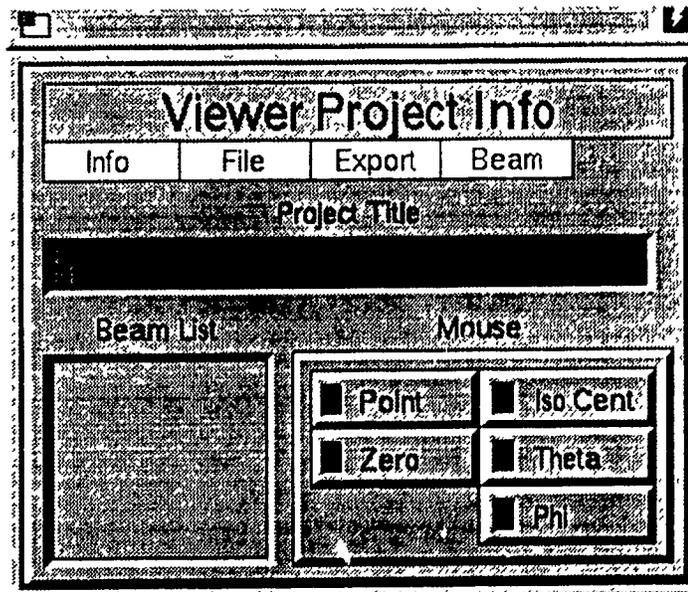


Figure 5.2: Viewer software main screen containing general menus and mouse function buttons.

5.3 Anatomical volume

5.3.1 Entering an image set

For this example, a spherical phantom data set was created as explained in chapter 3. Figure 5.3 shows the file list window where the file names and paths are entered into the system. The user types in the path and a file name. The *Add* button is pressed and the file name is entered into the list. By typing each file name and pressing *Add* one at a time, the image list is built. The file name is appended at the end of the list unless the user highlights one of the files already on the list. If a name is highlighted, any new files are inserted into the list before the highlighted file. The order of the files in the list determine the order in which they will be combined into a volume set. The user may also highlight and remove files from the list by pressing the *Remove* button. In addition to the names and path to the image files, the user must enter the number of pixels/cm for the image group and the slice thickness in cm. This enables the system to relate the volume of pixels to a volume in cm, and determine the slice thickness in pixels. Once the entries are complete, pressing *OK* begins the process of image importing.

5.3.2 Visualizing the volume

Once the anatomical data has been entered, it can be visualized. Reformatting of data was used as the primary technique for the anatomical display, which has the advantage of not requiring contouring of the

structures within the volume as is required in surface rendering. Figure 5.4 shows the three orthogonal slices all cutting into the center of the sphere. Note that in the coronal and sagittal windows, the thickness of each slice is observable as steps along the outer perimeter of the sphere.

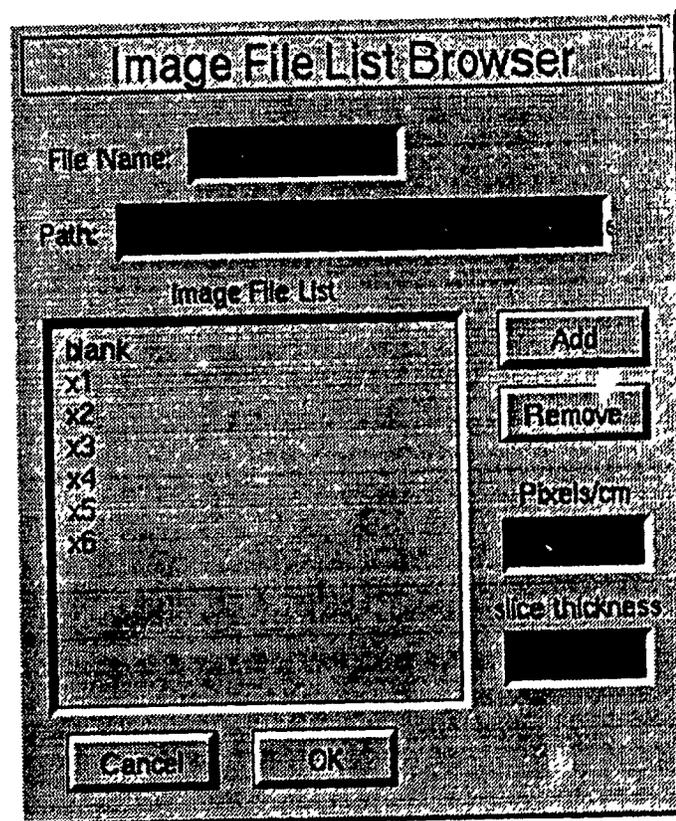


Figure 5.3: Viewer lister screen to complete the image list, and enter scaling factors

In addition to the image windows, there is a window showing a perspective outline of a cube. The cube outline represents the volume, while the three blue planes cutting into the cube represent the relative positions of the slice planes of the three orthogonal views. The 3-D axis is displayed at the bottom as a reference to the 2-D axes displayed in each of the image windows. The relative window level (gray level brightness) of the image may be altered by manipulating a slider bar located at the bottom of the cube reference window. This enables the user to optimize the brightness to help bring out any obscure structures within the image. It should be noted that the window width (relative contrast) may not be altered here because the Toshiba file transfer software reduces the contrast to 7 bits during the process, which can be adequately represented with the SGI screen without requiring window width adjustments.

The scroll bar at the bottom of the window is used to select the slice to be displayed. The user may drag the slider to cycle through the volume or use the arrows at either end to move precisely slice by slice. The image, perspective cube and slice indicator bars are updated automatically and in real time. A pop-up menu is available that can adjust the zoom of the image. By selecting a factor of two, the image doubles in size, namely to 512x512 pixels.

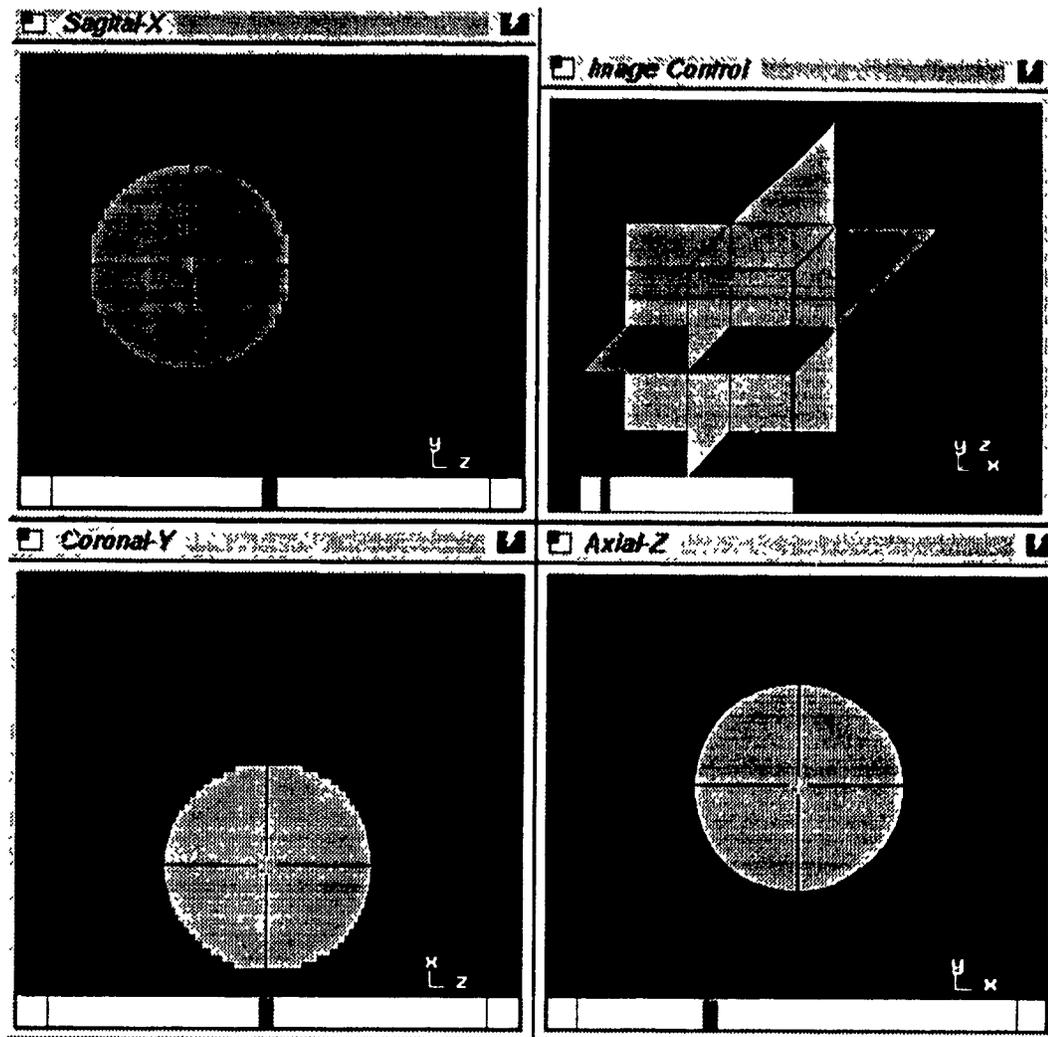


Figure 5.4: Viewer image windows. These include 3 orthogonal slice views and a perspective cube view to facilitate orientation. The slider bar at the bottom of every slice window control the current slice, which are displayed as blue lines on the other 2 orthogonal views. The slider on the cube view controls the gray level brightness.

5.4 Beam Manipulation

5.4.1 Beam visualization

In order to best represent the position of the beam within the volume, an outline of the intersection of that beam with each slice is displayed as a line contour overlaid on the anatomical display. This technique effectively shows what part of the anatomical volume is encompassed by the beam without obscuring the anatomy. This technique is also extremely fast to perform, minimizing response time. Figure 5.5 shows the three orthogonal views displaying the anatomical data as well as the beam outline. Note that in addition to the beam outline (shown in blue), the isocenter position is shown (in green) when a slice displayed coincides with it. In addition, the projection of the beam central axis (shown in red) on each slice is also displayed to aid in beam positioning.

5.4.2 Creating and Manipulating beams

In order to place a beam, the user must create a new one. This is done by selecting “New Beam” from the “Beam” menu in the main window. Figure 5.6 shows the beam information window. The user must select a beam energy, SSD, and size that coincides with previously entered beam data files. These files contain PDD and OAR data for varying SSD, beam energy and area. The position information can be entered manually by typing in the values, or interactively by manipulating it on the screen.

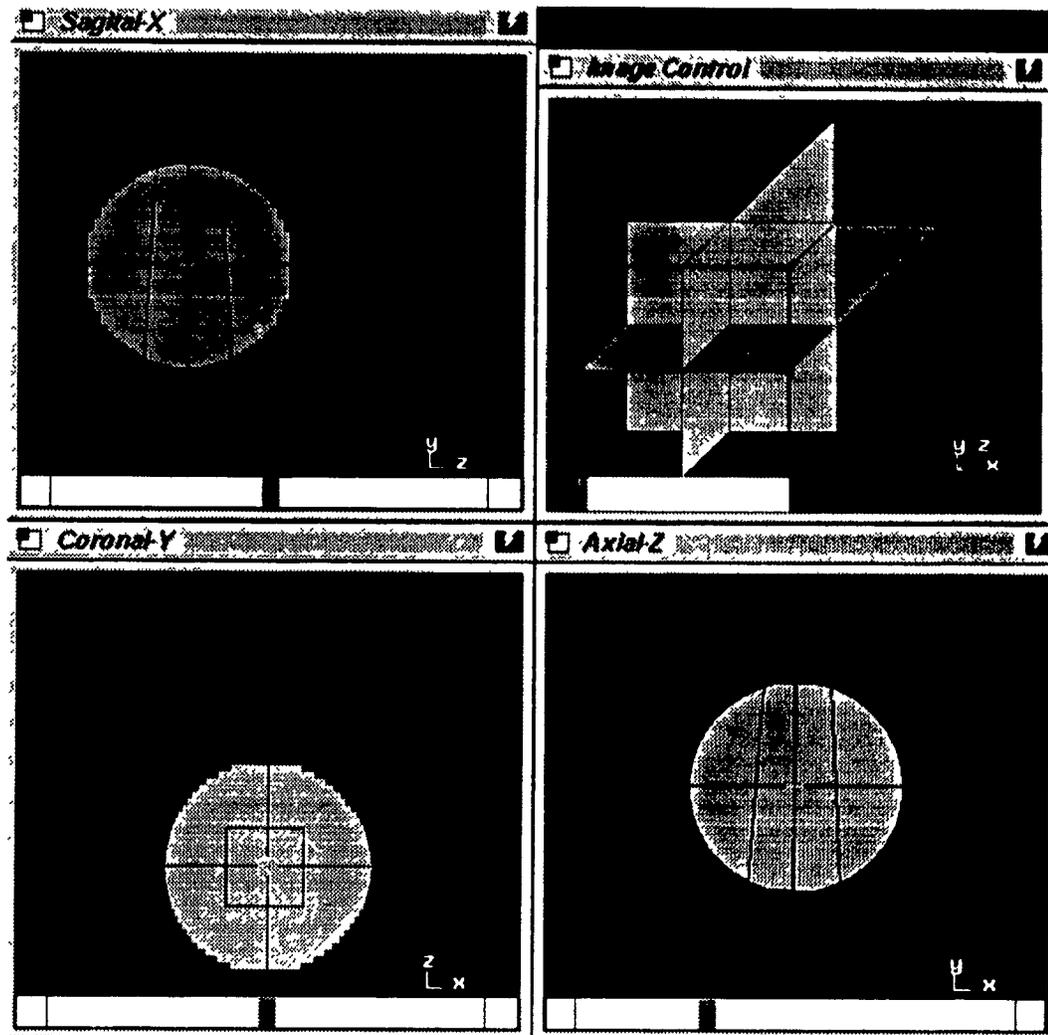


Figure 5.5 Orthogonal view windows displaying the anatomical information (sphere phantom) as well as the beam outline. The green cross denotes the beam isocenter which may be placed interactively.

The beam isocenter may be positioned by selecting “Isocenter” from the mouse function buttons on the main window. When the user places the mouse over an image and presses and holds the left mouse button, the isocenter indicator will track the mouse cursor. The user may use any of the three windows to place the isocenter. During positioning, the other two windows automatically modify their slice display to track the isocenter position, as the mouse is moved. While the display is being modified, the program also calculates the beam entry and exit points within the anatomical volume, and displays the outline of the beam intersection with each slice in real time. When the mouse is released, the last position becomes the isocenter.

By selecting the “Theta” mouse function button (θ in Figure 3.5), the user may use the mouse to set the beam entry angle. Again, this is done by moving the mouse over an image (usually the axial display) and pressing and holding the left mouse button. The beam outline will again be generated by determining the entry and exit points within the volume, and the intersection of that beam with each slice. This allows for interactive beam placement.

In order to adjust the couch rotation (ϕ in Figure 3.5), the “Phi” button is selected from the mouse function selection. In order to interactively set it, the mouse must be positioned over the coronal or sagittal windows. The angle is set by the relative horizontal position of the mouse to the window while the left button is being held down. The leftmost position is taken as 0 degrees, while the rightmost position is taken to be 360 degrees. As before,

the three views are continuously being updated in real time, as the user drags the mouse back and forth over the image.

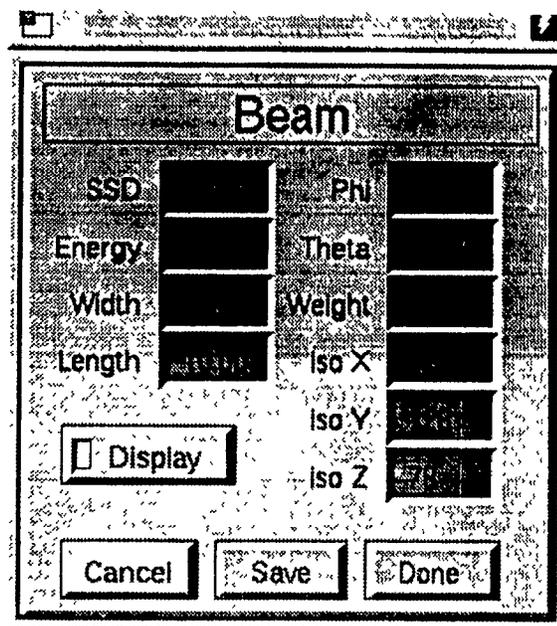


Figure 5.6: General beam parameter window The user may enter all the beam parameters manually, or manipulate some of them by manipulating the beam on the screen with the mouse. During manipulation, the values in this window are constantly updated

5.5 Dosimetry

Once the beam(s) are in place, the dosimetry control window can be opened by selecting “Dosimetry...” from the “Beams” menu in the main window. Figure 5.7 shows the dosimetry control window This window allows the user to set many dosimetry related parameters, and to start the calculation process. In a typical example, the user selects whether the system should calculate the dose distribution for the selected beam(s) (highlighted from the beam list in the beam window) or for all the beams entered. The user then selects the dose grid resolution from the resolution menu. The user can select from various axial resolutions from 32x32 up to

256x256. The third dimension is always the number of slices present in the system. The user then enters the list of isodose values to be displayed. This is entered in advance for the pre-contouring of the axial isodose display. The system automatically assigns a color to the value and displays the legend in the information window. The user can choose to have the relative doses with respect to the dose at d_{\max} or to have the dose matrix normalized to the maximum dose encountered. The calculation process is started by pressing "Calculate All" for the entire volume, or for the current axial slice by pressing "Calculate One".

5.5.1 Dose distribution visualization

One can employ many different techniques to display the dose distributions. In order to effectively show the distribution within the anatomy, isodose contours were used. As in the beam representation, this technique complements anatomical reformatting of data well. Some typical finite element analysis software (e.g. SolidView) will show data distributions as filled contours with either discrete or continuous color distribution. In either case, the filled contours tend to obscure the other data sets being displayed, which is the anatomical and beam data in our case. In the case of continuous filled contours, the user may obtain a good general feel of the distribution, but will encounter difficulty in trying to obtain exact dose values at discrete points or exact dose threshold lines because it is difficult to judge the exact colour of a particular pixel and match it with the legend. In order for the software to know which contours to display, the user enters the values prior to calculation. Figure 5.8 shows the familiar 3 orthogonal views with anatomical information, the beam outline, and the

relative dose distribution calculated using the system. The fourth window shows where the slices were taken within the volume.

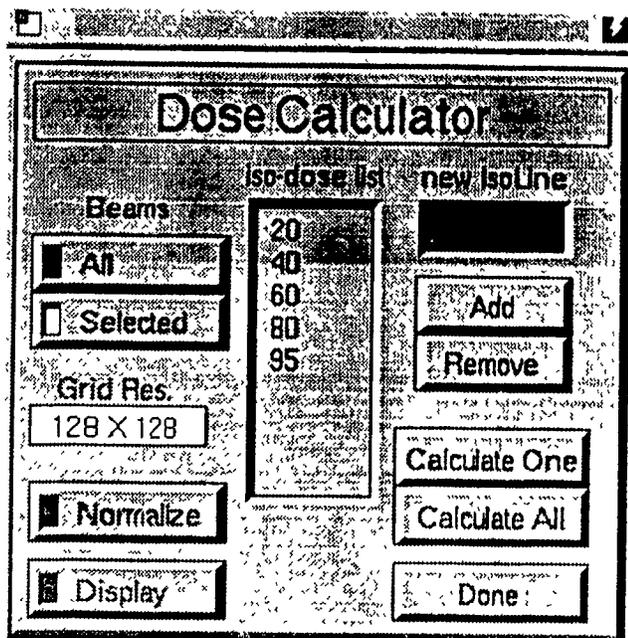


Figure 5.7: Viewer dosimetry control window. The user uses this window to enter all pertinent parameters for dose calculation, and to execute the dose calculations.

In order to minimize the response time of the dosimetry display, the contour information for the axial view is calculated and stored immediately after the dose calculation process and before updating the visual display. Since the resolution of the sagittal and coronal displays are usually much lower than the axial display, the SGI can contour and display them in real time, thus the contouring is not done in advance as in the axial display. Once the calculation and contouring are done, the results are displayed. The user may interact with the display tools to examine the dose distribution throughout the volume.

5.5.2 Evaluation Tools

In addition to the ability to view the dose distribution within the volume for all slices and for all views, some tools are provided to enable the user to better evaluate the quality of the plan. By selecting the “Point” mouse function button in the main window, the user may drag a point dose cursor over any of the windows. The software correlates the cursor position with the dose matrix and displays the dose to the voxel in the information window. The window also displays position of the cursor relative to the origin, which is also relocatable. One function for this would be to relocate the origin to the entry point of a particular beam, and to use the point dose cursor to examine the central axis dose by dragging it down that central axis. The origin is set by selecting the “Origin” mouse function button in the main window, and dragging the mouse (with the left button down) to the point where the origin is required. As the user drags either the origin or the cursor, the relative position of the cursor and the origin are continuously updated in pixels (image scale) and in cm. Figure 5.9 shows the axial view of the sphere with a dose distribution, and both the origin point (a small white cursor), and the point dose cursor (larger red cursor).

It is often convenient to know where the maximum dose or “hot spot” is. In order to find it quickly, a “Hot Spot” pop-up menu is available for each view. The user may choose between the hot spot for that particular slice, or for the entire volume. If the particular slice displayed does not contain the volume hot spot, no point is shown. If a hot spot is found, the hot spot voxels are displayed in pink. In addition, the hot spot value is displayed in the

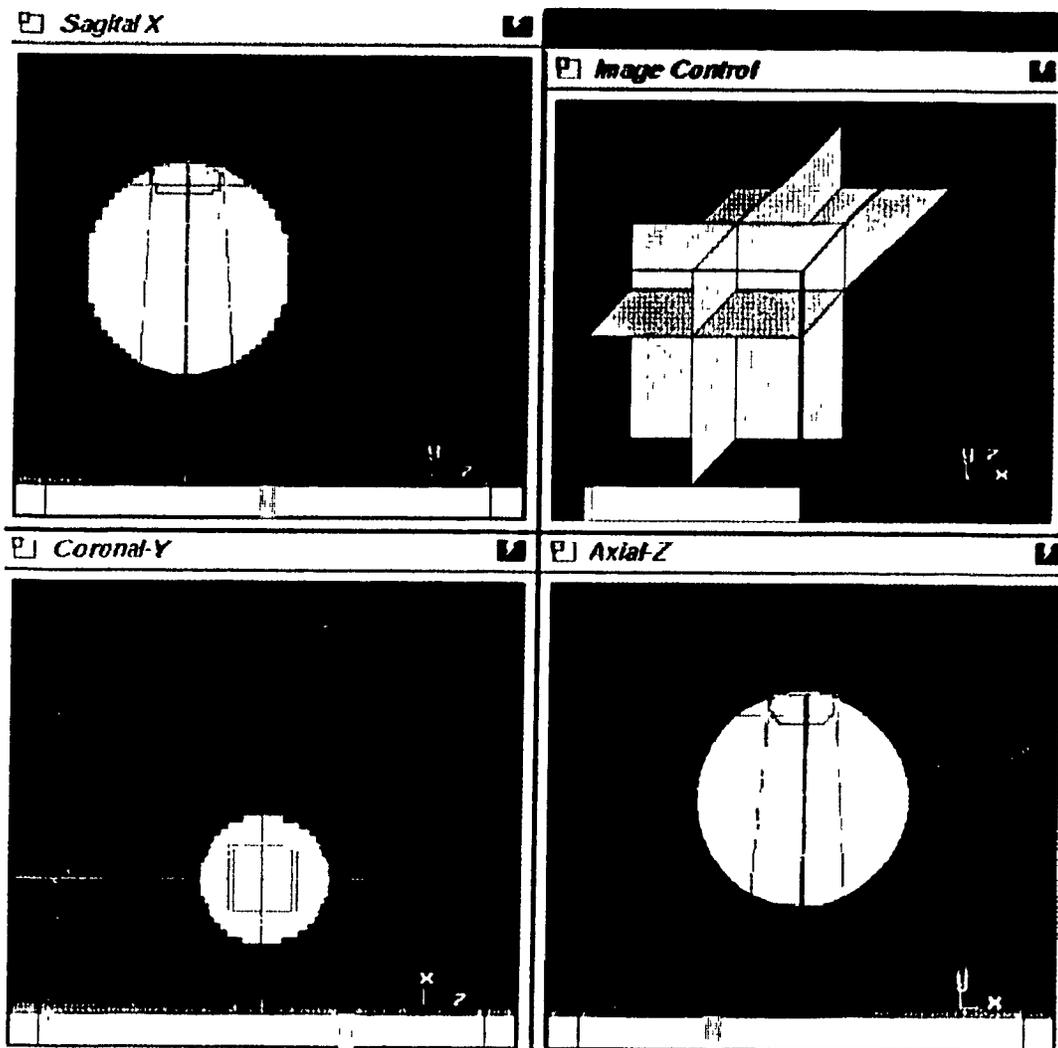


Figure 5 8 Viewer orthogonal windows displaying the anatomical data, the beam outline, and the resulting dose distributions after a calculation run. The calculation (128x128 points) and display of the isodose distributions on the SGI take approximately 20 seconds.

information window. Instead of simply not displaying the hot spot if it were not contained within the current slice, one might think to have the system simply find the volume hot spot and alter the slice selection to display it. This can not easily be implemented as there are usually multiple points within the volume that all have the same maximum dose and each view may only display one slice at a time. Figure 5.10 shows information window displaying the cursor position relative to the origin both in pixels and in cm, as well as the current slice hot spot value.

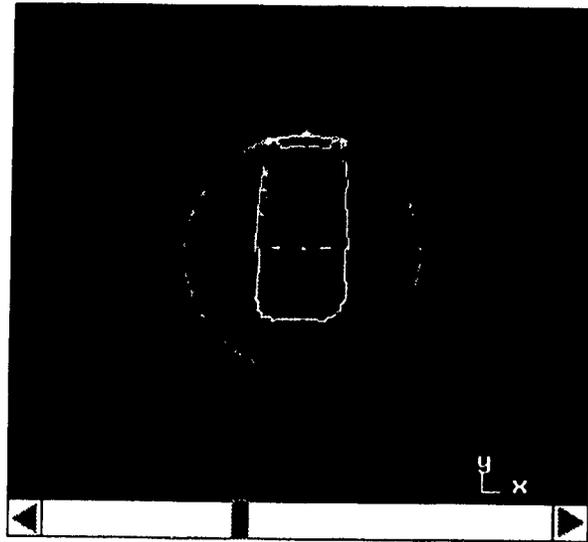


Figure 5.9: Axial view showing anatomy, beam outline and distribution. Note also the presence of the white cursor which is the relocatable origin, and the red cursor which serves as a point dose cursor.

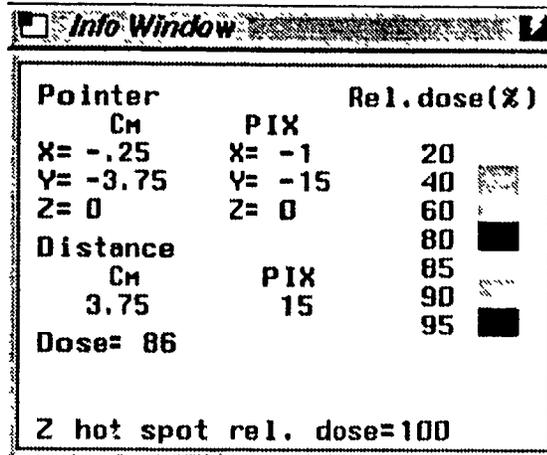


Figure 5.10: Information window displaying cursor related information, dose distribution color legend, and current Z slice hot spot Bottom line usually serves as a general status information line throughout the operation of the program

Once the dose distribution is displayed and evaluated, the user may wish to change any of the dose parameters in order to modify the plan. This can be done at any time and the dose may be recalculated simply by pressing the "Calculate All" button in the dosimetry window. If the user wishes to make numerous fine adjustments in the axial view, the best interactive placement can be achieved by selecting "Calculate One" instead of "Calculate All". This will ask the system to only calculate and display the dose distribution for the one axial slice. This feature dramatically shortens response time by restricting the system to a 2-D distribution. This may be used for some fine positioning and the full volume may be calculated at any time.

At any time during the planning process, the user may export images of either of the three views with dosimetry information. It is often desirable to have a high resolution dose distribution for this purpose. The resolution may be raised to 256X256 (equal to the image resolution) by using

the menu in the dose information window, and pressing "Calculate All" again. The user then exports the image by selecting "Image" from the "Export" menu in the main window. The user is prompted for a file name and for the view that is to be exported. The image is saved in the Alias "pix" format which can later be converted to a variety of formats including the TIFF format. If a dose volume has been calculated, an image of the distribution that resembles a film may also be exported. The image will have gray values equal to the dose to each pixel of the slice exported. Figure 5.11 shows the gray level dose distribution image of the slice shown in Figure 5.9.

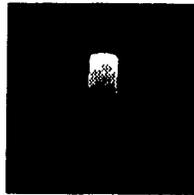


Figure 5.11: Gray scale output of the dose distribution shown in earlier diagrams. This image can be compared to a linearity corrected image of a film used to verify the system.

5.4 Summary

This chapter provided a detailed description of the functionality of the final software program. It was shown that the visualization techniques chosen were effective in conveying the anatomy, beam position and finally the dose distribution while using information from all three dimensions. In addition, the effectiveness of these techniques in providing an interactive environment and the addition of specialized tools for the planner have shown how such a system can be made to function in an effective and intuitive manner.

Chapter 6

Technical Specifications

6.1 Introduction

Chapter 5 described the functionality of the treatment planning system. This chapter will describe in detail, the memory structures and algorithms that comprise the final software. This will begin with a description of the image formats, memory management and algorithms used to generate the anatomical display. Technical details explaining how the beam trajectory information is obtained and how it is used to draw the beam outline will then be presented. Finally, the memory structures and important algorithms that pertain to the dosimetry will be presented. Throughout the chapter, emphasis will be placed on how these techniques may differ from the most straightforward techniques in order to improve the response of the program, including some changes in how we apply the Milan-Bentley method to calculate the dose distribution in a three dimensional grid.

6.2 Image Volume Display

The image information in a simple system, which originates as a series of slices, is read in and stored in a three dimensional array. In order to generate the orthogonal views, selected pixels are copied from the 3-D volume onto 2-D screen buffer arrays. Recalling Figure 4.8, subsets of the

volume data are segmented and used to create three orthogonal views. In order to maximize the redraw speed, some optimization techniques can be used in the manner the data is stored and accessed, and in the organization of the redraw techniques

6.2.1 Raster allocation

A technique to accelerate the generation of the axial and coronal views was implemented by using a novel memory pointer scheme. Instead of storing the points as a standard 3-D grid of pixels, the volume was grouped as a series of rasters. Figure 6.1 shows how one $N \times N$ pixel slice is broken into a series of N rasters of N pixels each. These rasters are the basic building blocks for the axial and coronal display. Figure 6.2 illustrates how the axial and coronal views can be generated by moving a series of rasters. As the images are read in, raster memory blocks are allocated and a complex set of pointer arrays are generated for each view. Figure 6.3 schematically demonstrates the relationships and operation of the pointer arrays. These arrays are designed to minimize the steps involved in sifting and moving the data from the volume array to the 2-D view arrays. The improvement in speed can be largely attributed to the increased efficiency in using pointer arithmetic and pointer dereferencing in determining the addresses of the pixels to be moved, as opposed to calculating the pixel addresses using an array base address and x y z offsets.

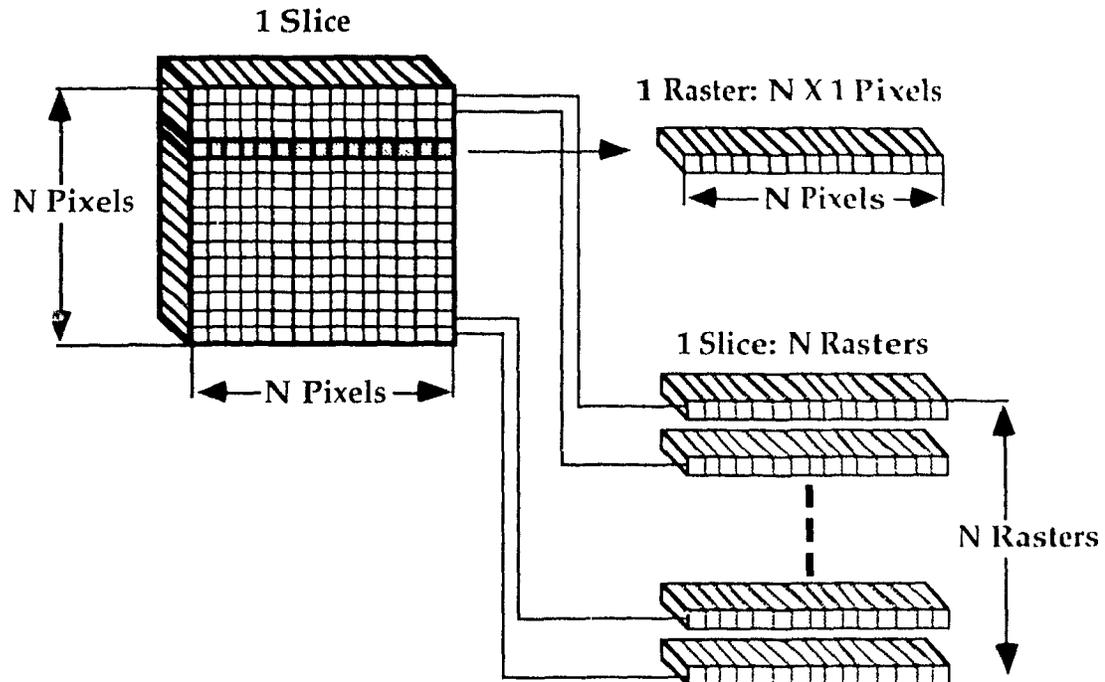


Figure 6.1· Diagram showing how the volume pixels (voxels) are consulted to create the three orthogonal views.

6.2.2 Redraw organization

Using the pointer arrays generated during image loading, it is a simple matter to quickly move the data about. It is important to consider all the information and how it is to be viewed before organizing the redraw routines. When the user interacts with anything that changes the display (e.g. changing a slice view or moving a beam on the screen), the three orthogonal views must be updated to reflect the new scene. Figure 6 4 shows a simple redraw sequence for the three orthogonal views. This series of operations is performed every time the user interacts with the program.

Despite the optimization of the anatomical data display, it still is one of the longest tasks to perform. One way to improve system response is to

eliminate any redundant processes in the redraw sequence. If one considers all the actions that can generate a redraw event, one can see that few interactions actually involve the anatomical display. Figure 6.5 shows a modified redraw sequence.

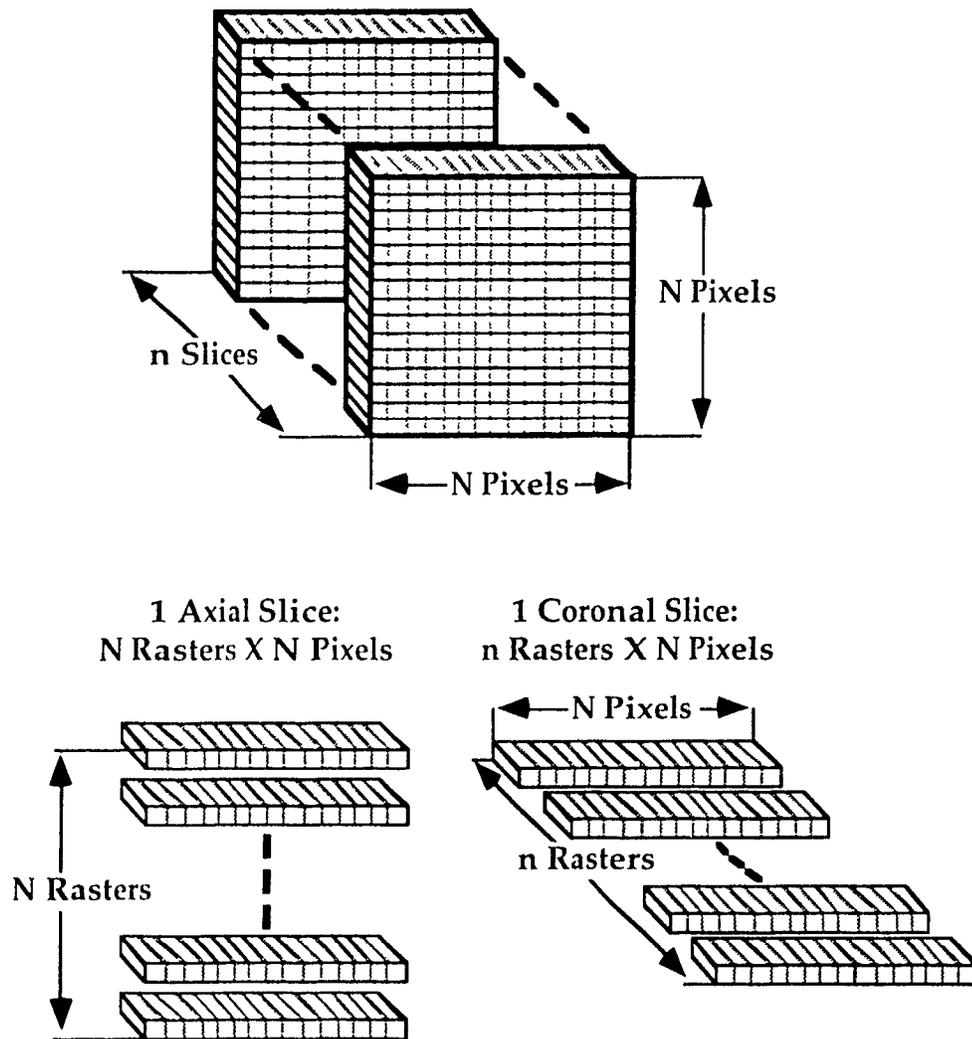


Figure 6 2. Example of how the rasters are used to create the axial and sagittal views.

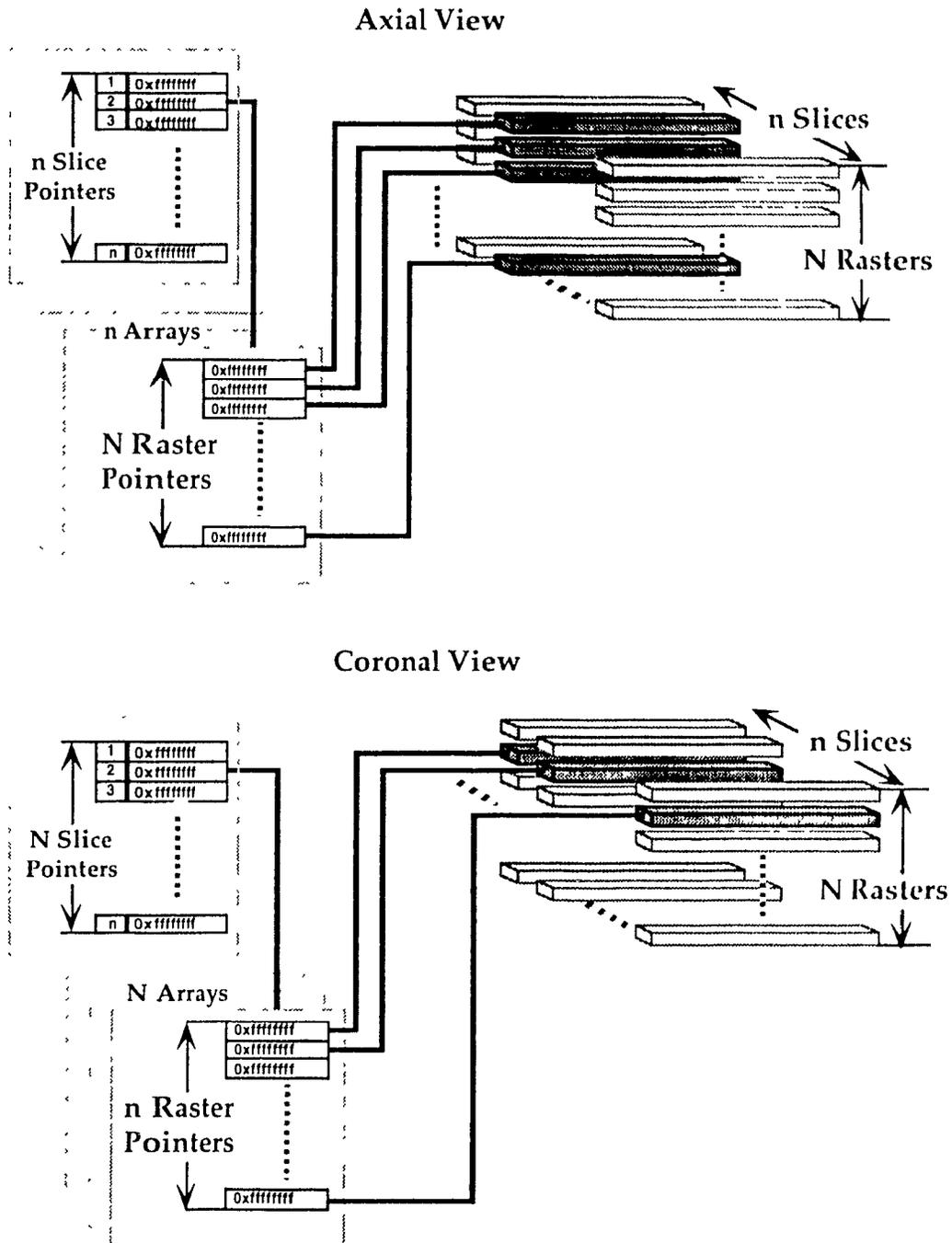


Figure 6.3: Pointer arrays and their relationships with rasters that are used to draw axial and coronal views

```

on redraw all {
  copy pixels to axial screen buffer
  copy pixels to coronal screen buffer
  copy pixels to sagittal screen buffer
  draw beam indicators

  if (beamDisplay=on) then
    calculate beam entry & exit coordinates
    calculate beam corner coordinates
    calculate & draw axial beam outline
    calculate & draw coronal beam outline
    calculate & draw sagittal beam outline
  end if

  if (doseDisplay=on) then
    contour & draw axial isodoses
    contour & draw coronal isodoses
    contour & draw sagittal isodoses
  end if

  if (cursorDisplay=on) then
    draw cursor in current window
    update info window
  end if

  update cube display

```

Figure 6 4: Pseudocode of a generic redraw sequence required to update the screens in the treatment planning system.

```

on slice selection change
  if (theWindow=axial) then
    copy rasters from volume to axial memory buffer
  end if
  if (theWindow=coronal) then
    copy pixels from volume to coronal memory buffer
  end if
  if (theWindow=sagittal) then
    copy rasters from volume to sagittal memory buffer
  end if
  redraw all
end slice selection change

```

```

on beam interaction
  calculate beam entry & exit coordinates
  calculate beam corner coordinates
  redraw all
end beam interaction

```

```

on dose calculation
  calculate doses
  contour axial view-> store contours in pointer list
  redraw all
end dose calculation

```

```

on cursor movement
  calculate new cursor coordinates
  redraw all
end cursor movement

```

(continued...)

Figure 6.5: Decentralized event oriented redraw scheme. Despite the increase in code quantity, this technique is much faster due to its increased intelligence. The longest operations are the pixel or raster moves when consulting the volume to update the anatomical display, and determining the beam entry and exit points. Both these operations are removed from the redraw loop and are performed only when necessary.

```
on redraw all
  block copy from axial memory buffer to axial screen buffer
  block copy from coronal memory buffer to coronal screen buffer
  block copy from sagittal memory buffer to sagittal screen buffer
  draw slice indicators

  if (beamDisplay=on) then
    calculate & draw axial beam outline
    calculate & draw coronal beam outline
    calculate & draw sagittal beam intersection
  end if

  if (doseDisplay=on) then
    draw axial isodoses
    contour & draw coronal isodoses
    contour & draw sagittal isodoses
  end if

  if (cursorDisplay=on) then
    draw cursor in current window
    update info window
  end if

  update cube display
end redraw all
```

Figure 6.5: (Continued)

The main change is that the overall approach is decentralized. For example, anatomical views are first created in an intermediate buffer before being copied into the actual screen memory. This means that the anatomical data updating and the screen redraws are independent. When the program changes anything to do with the anatomical display, it updates the display buffer and then asks for a complete redraw event. The redraw event simply copies the buffer directly into the screen RAM without consulting the volume. The advantage is that any other interaction that requires a screen redraw that does not involve the anatomical information (e.g. changing the beam angle or generating isodose values) need not consult the volume information. Another advantage to this is that the GL library has a rapid block copy from RAM to the screen buffer, and that this copy operation can incorporate integer scaling. This eases the work involved in offering an image zoom feature. Similarly, the beam routines were separated into independent updating and drawing routines along the same decentralization scheme. Even though there is considerably more program code in the new scheme, the actual execution is considerably faster due to better organization.

6.2.3 Look-up table (LUT) methods

As discussed in chapter 5, the user may change the gray brightness level of the image. There are two ways of implementing this operation. One would be to actually change the gray values in memory, either in the volume array itself, or by adding the offset to each pixel during the copy process. The other way would be to change the look-up table. The latter

method is substantially faster, as it only needs to modify 256 values once, and preserves the original data. Figure 6.6 shows how the LUT ramp can be moved up or down.

Since the display uses 256 gray levels and the images are 8 bit (gray scale), raising the brightness level may cause some of the brighter areas to saturate, as they are moved up the LUT ramp. The user must decide which structures are more important. This phenomenon does not affect the operation of the surface detection algorithms discussed later, as they always examine the actual volume data, not the brightness corrected image.

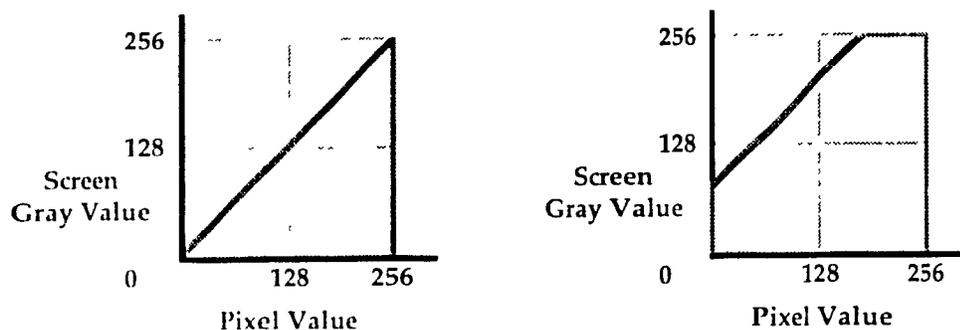


Figure 6.6 Modifying the LUT ramp table to increase brightness. Note that the brighter areas of an image may be saturated in the process.

6.3 Beam Display Algorithms

In the current version of the software, the beam geometry was limited to square or rectangular beams. In order to properly represent these beams in the orthogonal views, many variables had to be considered. These include the beam isocenter, the SSD, the entry and exit points, and finally

the beam angles (gantry and couch angles). In order to be able to update the beam outline in real time, and allow the user to interactively modify any of the beam parameters, a fast method of determining the beam entry and exit points was required. In addition, the routines that calculate and display the intersection of the beam and each slice had to be developed.

6.3.1 Beam entry/exit points

Figure 6.7 shows a simplified 2-D example of how the beam entry point is found. The process is started by calculating the pixel coordinates along the beam central axis for increasing r (the ray from the isocenter to the calculation point) toward the beam source. If a pixel value of zero was found at that point, it is assumed that a tissue-air transition was encountered. That point is then stored and r is increased again. This continues until the ray from the isocenter reaches the edge of the image. The last tissue-air threshold is taken to be the surface of the object. By continuing to the edge of the image, the system ensures that it does not mistakenly take an air bubble within the volume as the surface of the object. Similarly, a ray is projected along the beam central axis away from the source, to determine the beam exit point.

In reality, these calculations are performed in the three dimensional volume. The x , y , z coordinates of the pixel in image space, given the isocenter coordinates x_{off} , y_{off} , z_{off} , the gantry angle θ , and the couch angle ϕ (refer to Figure 3 5), can be determined using the equations:

$$\begin{aligned}
 x &= -r \cdot \sin(\theta) \cos(\phi) + x_{\text{off}} \\
 y &= r \cdot \cos(\theta) + y_{\text{off}} \\
 z &= r \cdot \sin(\theta) \sin(\phi) + z_{\text{off}}
 \end{aligned}
 \tag{6.1}$$

The calculation of the entry-exit points are performed continuously when the user is either moving the beam isocenter, or changing the angles θ or ϕ . This permits the user to see how the variation of any parameter will affect beam placement in real time.

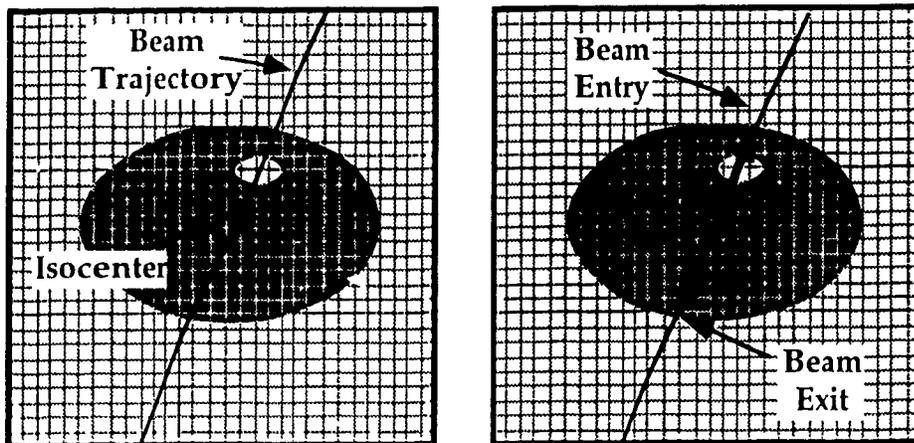


Figure 6.7 Simple 2-D example of how the beam entry and exit points are found. Note that the algorithm searches until it has reached the edge of the image to avoid interpreting a tissue-gas threshold as a surface air threshold.

6.3.2 Beam outline drawing

Once the beam entry and exit coordinates have been established, the beam outline may be drawn. The shape of the beam volume can be considered to be a pyramid. The 8 corners are determined by the entry and exit coordinates, the beam width and length at the entry point, and by the SSD. The SSD determines the divergence of the beam. With the ability to modify both θ and ϕ , the beam can intersect with any slice within the anatomical volume in many different ways.

Figure 6.8 illustrates how a pyramid shaped beam can intersect with a slice. The program uses lines to represent the intersection of the beam volume with the slice. In order to represent this, the points that make up the edges or that outline must be found. The intersection can have as little as three, and as many as six points of intersection, depending on how the slice cuts into the beam volume.

The program routine that determines the edge lines of the intersection can be broken down into two sections. The first section determines if there is an edge line along the top and/or bottom of the beam pyramid (Figure 6.9a). It does so by determining if any perimeter edges of the upper or lower squares or the beam pyramid go through the slice. If it does, the intersection point is stored. The result is a pair of points whose coordinates are the end points of the intersection line.

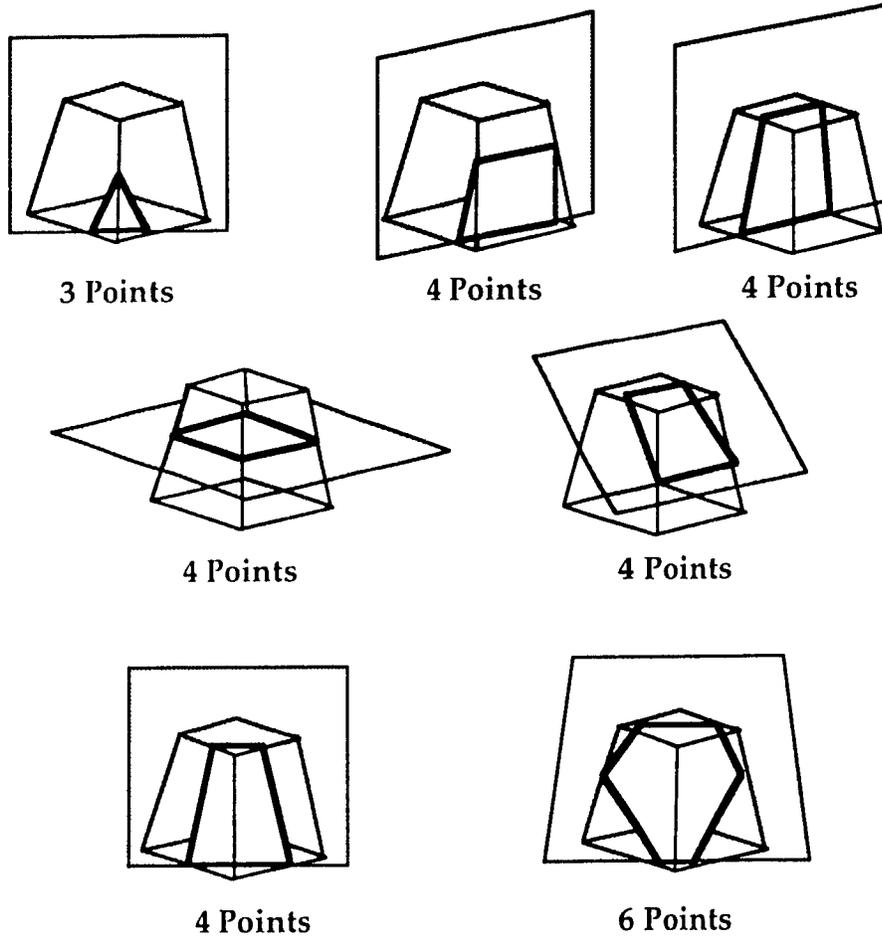


Figure 6 8: All the combinations of intersections of the beam volume and the anatomical slice cutting plane.

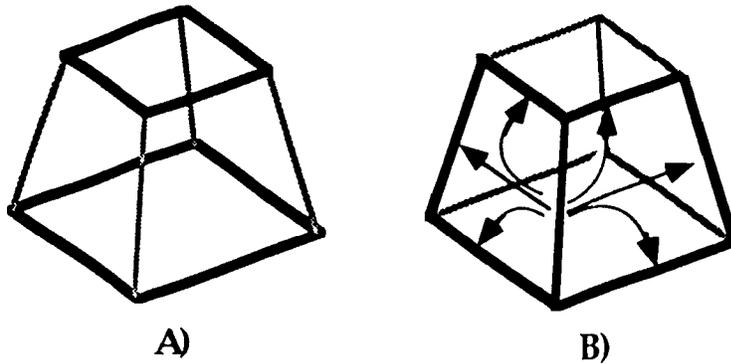


Figure 6.9: A) shows how the top and bottom sections of the beam is examined for intersections. B) shows how the center edges and their neighbors are examined for intersections with the slice plane.

The second section is more complex. It determines if there is an intersection point along any of the beam's vertical edges (Figure 6.9b). If it finds one, it searches all the connected edges to find the other intersection point that will complete the slice intersection line. In the case where there are 3 points of intersection (Figure 6.8), there are two intersection lines that use the same vertical intersection point. The algorithm takes this into account. When the algorithm is complete, a series of lines that describe the intersection of the beam volume with the slice plane are obtained.

6.4 Dose Calculation

Once the beam information is entered, the dose calculation may be performed. As explained in chapter 3, the model implemented is a modified version of the Milan-Bentley technique. The modifications were to enable

the calculation of dose using an inherently 3-D coordinate system, to accommodate patient surface curvature, and to simplify the model by assuming a homogeneous volume.

6.4.1 3-D beam coordinates and simplifying assumptions

In the original Milan-Bentley technique[1], the off-axis ratio (OAR) points of measurement were taken along fan lines that followed the divergence of the beam. In effect, the horizontal spacing of the points decreased as the depth of measurement increased (Figure 6.10a). In order to calculate the dose to a volume and to permit the summing of doses from multiple beams, a uniform OAR grid was adopted (Figure 6.10b). In order to minimize the measurements and database sizes for the OARs, they are assumed to be square symmetric. When one requires the OAR for a point in the dose grid, the coordinate transformation described in chapter 3 (Equation 3.6) is used to obtain the off axis distances (x & y) and the effective depth (z). This depth must then be corrected for surface curvature.

6.4.2 Surface curvature correction grid calculation

The Milan-Bentley technique uses a user-defined contour to determine the surface curvature correction values that are the distance of the actual surface to the effective SSD. These values are used to correct the effective depth to a true depth value for looking up the percent depth dose (PDD), OAR values, and in an inverse square law correction (see Equation 3.2 of chapter 3). Their technique only required a 1-D array of corrections for

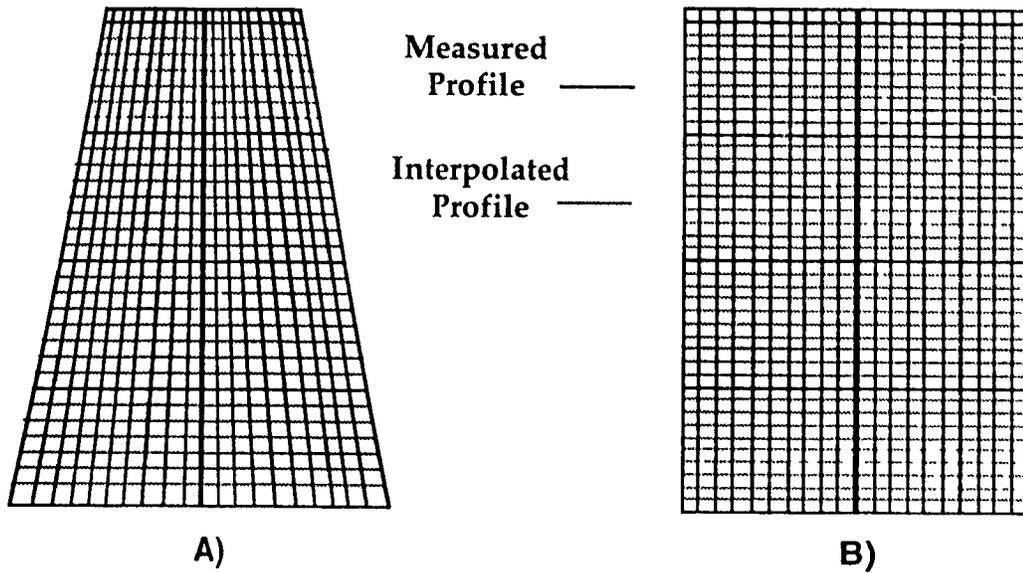


Figure 6.10: Comparison of the grids used to hold the off-axis ratios (OARs) for A) The standard Milan-Bentley method, and B) the modified technique for calculations in a 3-D grid.

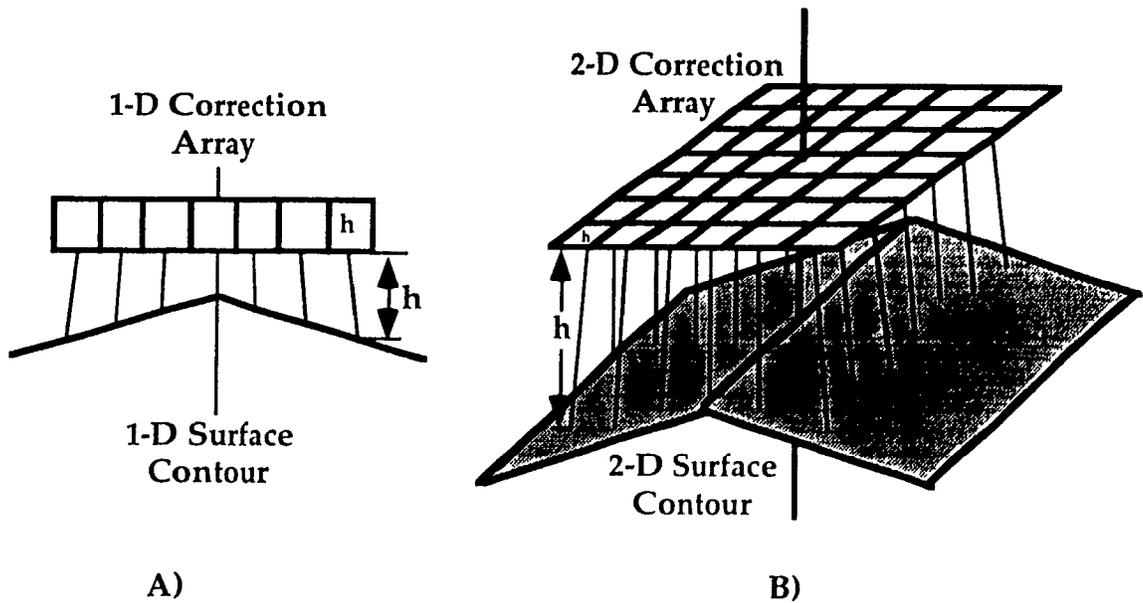


Figure 6.11: Comparison of the surface correction matrices for A) The standard 2-D Milan-Bentley technique, and B) Our modified 3-D technique.

a 2-D dose array (Figure 6.11a). In order to account for surface curvature in a 3-D dose volume, a 2-D array of surface correction values is required (Figure 6.11b) and an algorithm was developed to obtain it.

When the user requests a dose distribution calculation, the system calculates the correction grid for each beam. According to the coordinate systems described in chapter 3, a 2-D grid may be set up in beam space at the entry point of the beam into the volume ($z=0$), with the size equal to the size of the beam area at this point. The resolution of the grid is set to equal the resolution of the dose grid. For each point on this grid, a ray is traced away from the starting point along or against the beam direction. Each point along the ray is converted to a point in patient space and the corresponding pixel in the anatomical volume is examined. The coordinates (x, y, z) of a point in image space can be determined given a point in beam space ($\mathbf{x}, \mathbf{y}, \mathbf{z}$) using the equations:

$$\begin{aligned}
 x &= (\mathbf{x} * \cos(\theta) \cos(\phi) + \mathbf{y} \sin(\phi) + \mathbf{z} \sin(\theta) \cos(\phi)) * G + x_b \\
 y &= (\mathbf{x} * \sin(\theta) - \mathbf{z} \cos(\theta)) * G + y_b \\
 z &= (-\mathbf{x} * \cos(\theta) \sin(\phi) + \mathbf{y} \cos(\phi) - \mathbf{z} \sin(\theta) \sin(\phi)) * G + z_b
 \end{aligned}
 \tag{6.2}$$

where the variables are as described in Figure 3.5.

If that point is 0 (air) then the length of the ray is increased by 1 pixel, If it is not 0, the starting point is assumed to be within the volume, and the length is increased in the negative direction by 1 pixel. This operation continues until the tip of the ray encounters threshold change that signifies

if it has encountered the surface from either direction. These distances are stored in the array for use in the dose calculation.

6.4.3 Calculation volume optimization

In a generalized dose volume, not all voxels within that volume may fall within the exposed volume for every beam. In order to reduce the number of needless calculations, the coordinates of the beam are used to create a smaller volume for calculation for each beam. The most sophisticated system might use interpolation in all three dimensions to precisely define the beam volume, however the extra overhead involved might outweigh the benefits. A simple method is to quickly create bounds by comparing the 8 points that define the corners of the beam. By taking the minimum and maximum points in all three dimensions, an upright cubical volume that contains the beam volume is created. The efficiency of this technique depends on the orientation of the beam. Figure 6.12 illustrates the concept in a two dimensional example.

When the gantry angle is a multiple of 90° , the number of needless calculations is minimized, as there are few points within the area of the square that are not within the beam area. The simplification is least efficient when the beam is at a multiple of 45° , as the upright square contains many points that are not contained within the volume.

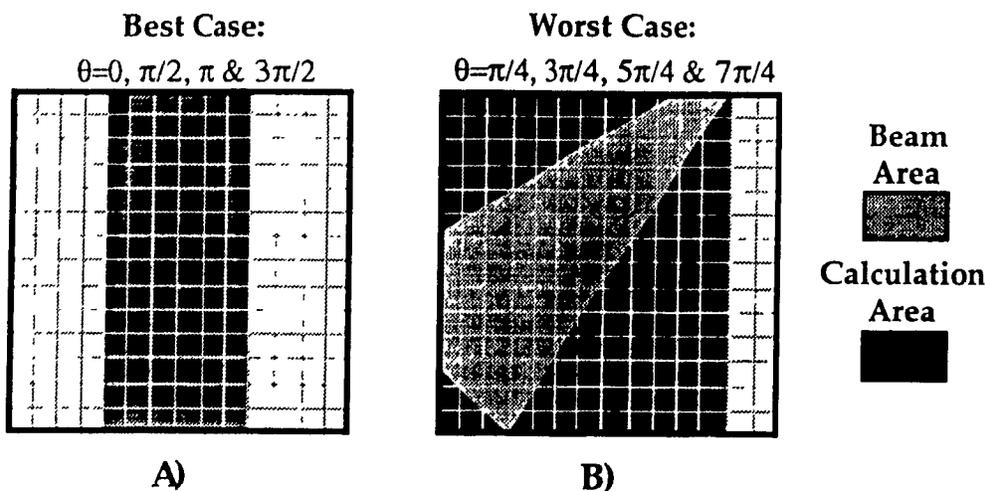


Figure 6.12: Optimization of the dosimetry calculation by decreasing the number of unnecessary calculations. this simple technique requires little calculation work but is often effective.

6.5 Summary

In order to be able to provide as much real time information as possible, a variety of optimizations are required both in the organization of the information, and in the actual algorithms that perform the tasks. This chapter has shown the more notable examples used in this project. The complete package was written with real time display and interactivity in mind, which resulted in a fast and efficient system.

References

-
- 1 Milan J., Bentley R.E., 1973. "The Storage and Manipulation of Radiation Dose Data in a Small Digital Computer." *Brit. J. of Radiol.*, **47**, pp. 115-121.

Chapter 7

System Verification

7.1 Introduction

When a new tool is introduced that performs calculations and provides results, care must be taken to ensure that the results given are accurate. Computerized treatment planning is no exception. In order to ensure that the system is performing properly, computer calculated data must agree with experimental data. This chapter presents results obtained from an experiment whose technique is outlined by Stern et. al., [1]. This experiment uses film dosimetry in a water equivalent phantom to verify the computer results by placing the film in the beam's eye view (BEV), which is perpendicular to the beam. These films can then be used to obtain profiles at various depths directly, or by interpolating between two film measurements. This procedure uses a percent depth dose curve along the beam central axis to help limit errors when using film dosimetry. The film response curve will be shown first, followed by the profiles from both the computer calculations and film dosimetry for comparison.

7.2 Film Response curve

Figure 7.1 shows the film calibration curve obtained for the Theratron ^{60}Co unit, Kodak X-Omat V film and the film processor.

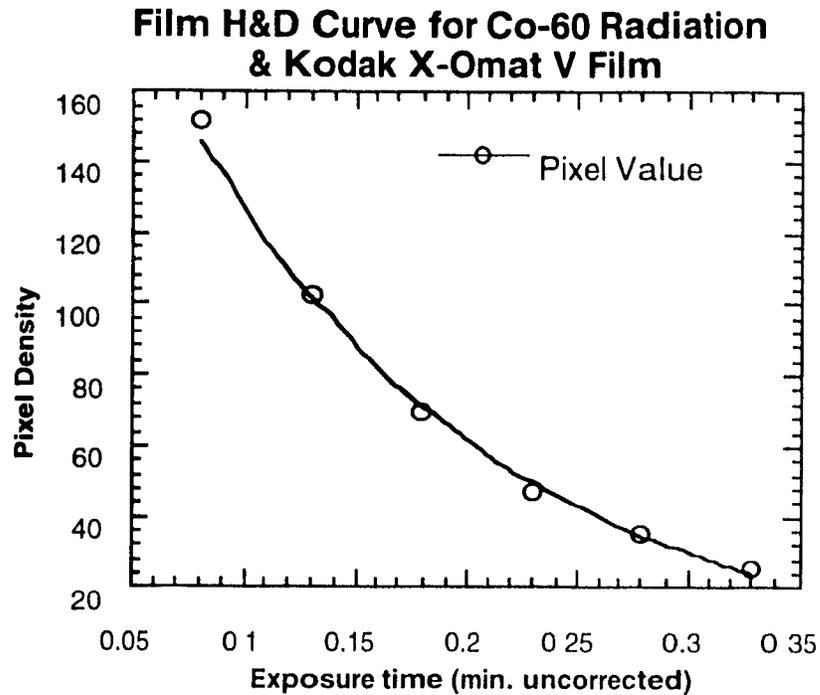


Figure 7.1: Film H&D curve for ^{60}Co radiation, SSD=80cm, 10Cm² field for Kodak X-Omat V film and processor, and the transmission film scanner combination

Although this curve is the usual method of presenting film response, a curve with the axes transposed is required to relate the pixel value to the corresponding exposure. Figure 7.2 shows this data. A logarithmic transformation was used to fit the data.. The logarithmic fit (Equation 7.1) was used to convert the profile data into relative exposure.

$$\text{Exposure} = M0 + M1 \cdot \log(\text{Pixel Value}) \quad 7.1$$

where:

M0 -81.729894057

M1 -209.1631365

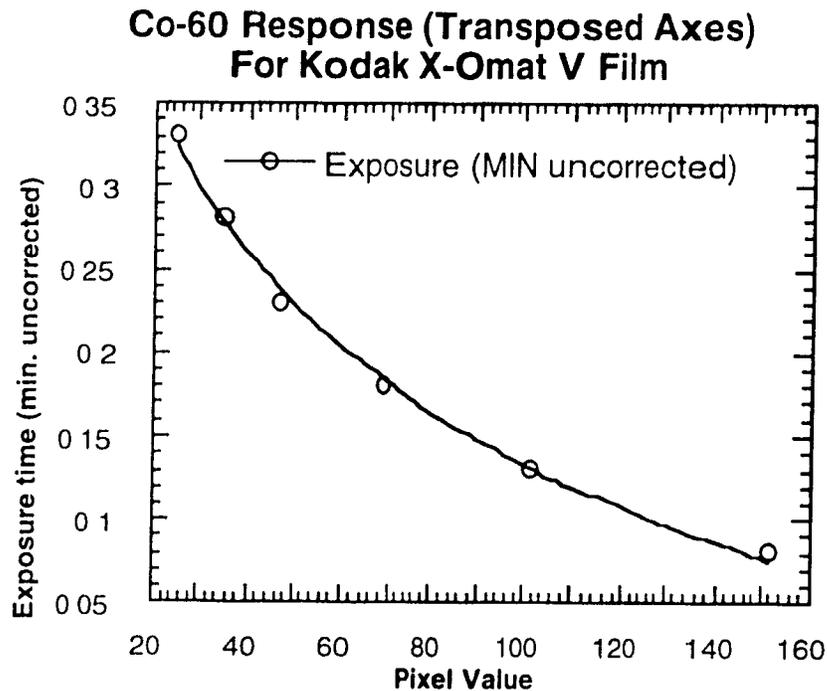


Figure 7.2 Transposed H&D curve to allow for the log curve fit to yield an exposure value given a pixel value. The fit is used to convert the profiles to relative exposure.

7.3 Profile comparison

Figure 7.3 compares profiles from the film dosimetry results and the computer calculated results at four different depths. These profiles were taken from the center of each beam. That is, each profile crosses the beam central axis. Another set of profiles was taken off the axis. Figure 7.4 shows the locations from which the profiles for the first and second set were taken from. Figure 7.5 shows the second set of profiles.

Beam Profiles at Various Depths Through Beam Central Axis

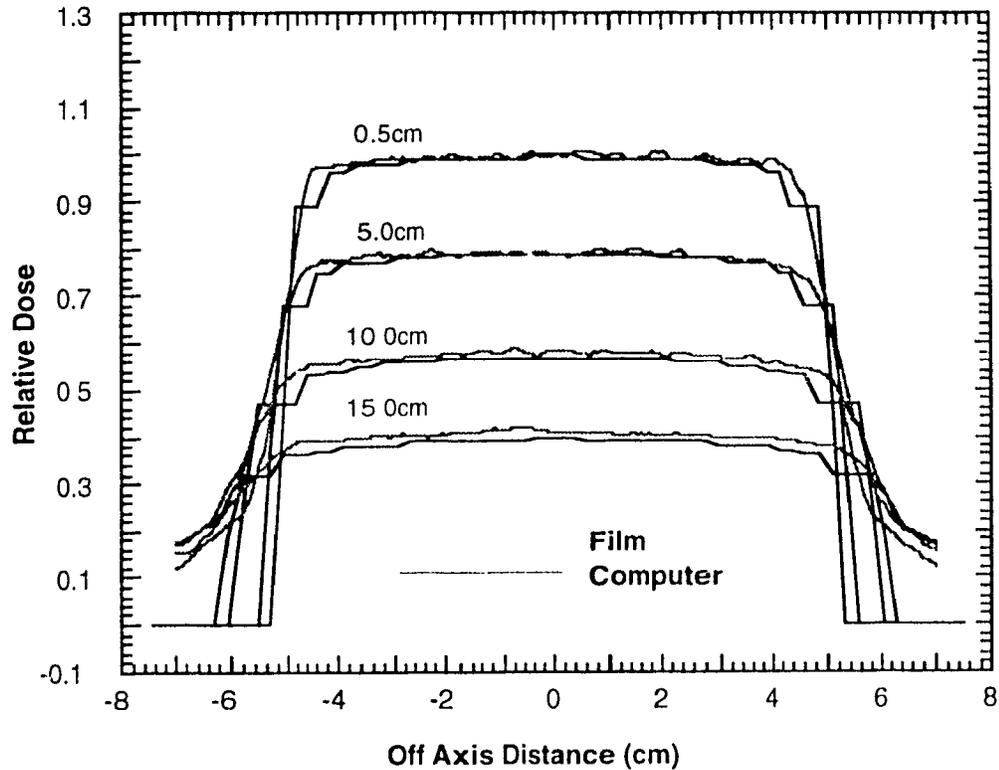


Figure 7.3: Profiles taken across the beam central axis from the film data and the computer calculations from the treatment planning system

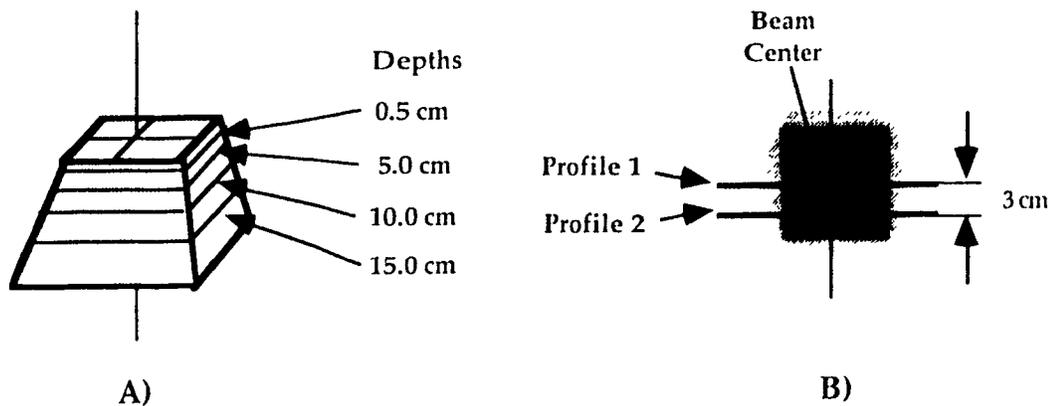


Figure 7.4. Depths of the film measurements and the positions of the profiles extracted from each film with respect to the beam central axis.

Beam Profiles at Various Depths 3cm off Beam Central Axis

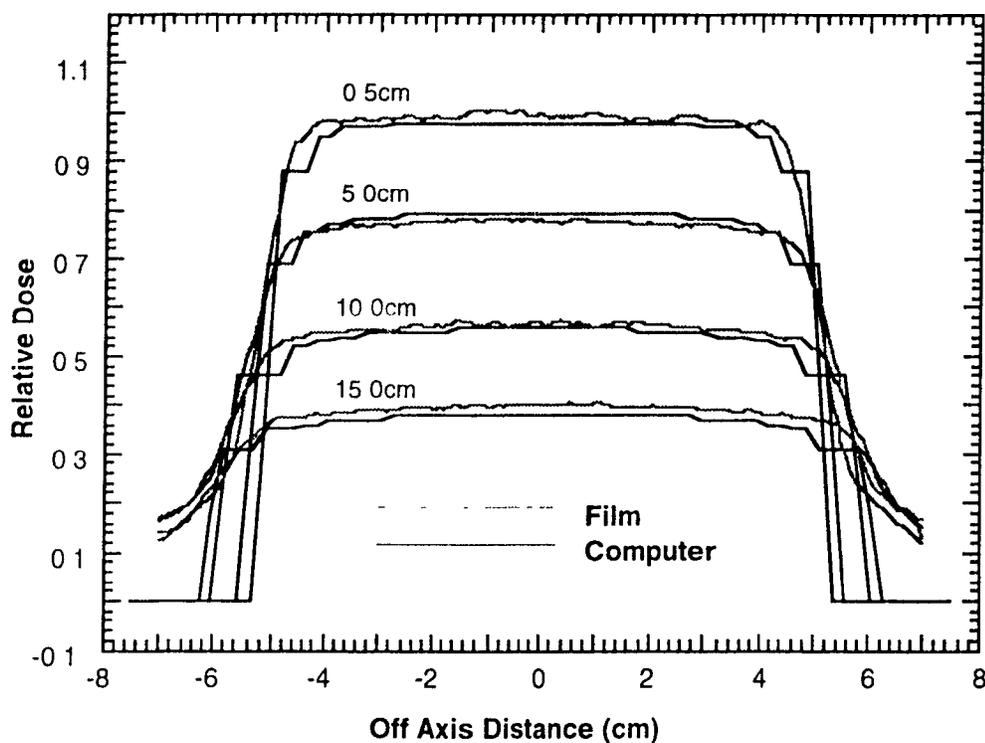


Figure 7.5. Profile comparisons for film and computer treatment planning system for profiles taken off the beam central axis (see Figure 7.4b).

7.3 Summary

This experiment has shown that the interactive RTP system we implemented can successfully predict the dose distributions within a homogeneous volume within 5% for the geometries presented. It should be noted that if support for more beams and more complicated set-ups is to be added (e.g., beam blocks, wedges and rectangular beams), this verification technique should be repeated for each variation from the simple model demonstrated here.

References

-
- 1 Stern R.L., Fraass B.A , Gerhardsson A., McShan D.L., and Lam K.L , 1992, "Generation and use of measurement based 3-D dose distributions for 3-D dose calculation verification", *Med Phys.* **19**(1), pp 165-173.

Chapter 8

Conclusion

11 Summary

Many visualization techniques are commonly used in computer imaging, and computerized treatment planning is no exception. Many techniques can do an adequate job at relating the significant variables but some do so at a higher cost than others. This project examined techniques such as surface and volume rendering, reformatting of data, wire frame outlines and contouring. The examinations initially focused of the visualization of the anatomical volume, where the information is most relevant and potentially involves the most work in data preparation. The other variables to be represented are the beam position and the dose distributions. The techniques used for these were selected to work best with the modality chosen for the anatomical data, which was reformatting of data. The beam position was thus displayed as a wire frame of the intersection of the beam volume and the anatomical slice, and the dose distributions were displayed as isodose contours.

In addition to the visualization techniques, a novel graphical user interface (GUI) was developed to promote interactive experimental use. The system was designed to provide as much real time updating as possible given the computer available for a reasonable amount of money. The result is a system that successfully helps the user create and evaluate treatment

plans with the potential for more experimentation which can lead to better plans, and thus a better treatment.

A 3-D film verification technique was examined and attempted in order to show the accuracy of the treatment planning system. The technique showed that it was capable of determining if a system is performing within acceptable accuracy (5%) while minimizing the work involved in taking measurements.

1.2 Future work

This system has the potential to expand in many areas. The two main areas are in calculation technique and in visualization techniques. As available systems become faster and more affordable, improvements can be made in both areas without sacrificing real time response.

There currently exists many other techniques to calculate dose distributions other than the Milan-Bentley technique used here. Many of these take the surface inhomogeneity into account, but they also take into account the inhomogeneity within the volume. These vary from the equivalent path length technique, to various other techniques that improve on the estimate of scatter from the inhomogenous volume surrounding the point of calculation. These new techniques can be added to the software by adding a new menu in the dosimetry control window that would select the preferred method given the situation, and the amount of time permitted for calculation. This would allow the user to either select the technique that is

best suited for a particular case, or even use several methods for a given case to compare each different prediction.

As with the system presented by Rosseman et. al. [1], Other visualization techniques may be used in tandem with one another to help the user visualize the variables. Figure 8.1 is an example of this technique. Note that the usual reformatting of data display is present with all of its associated tools, and a new window with a contoured surface rendered image of the outline contour is also present. If automatic segmentation of the inner structures were available, it could be included with little extra work required by the user.

The system developed for this project has provided much insight into current visualization techniques, dosimetry calculation methods and in verification. Its best use is in its potential for becoming a suitable test platform for future imaging techniques such as hybrid images using multiple modalities (i.e., CT anatomy and MRI spectroscopy), or testing new calculation techniques for both photons and electrons. The system may be taken into a more clinical direction by exploring treatment plan evaluation tools such as dose volume histograms and other mathematical evaluation tools.

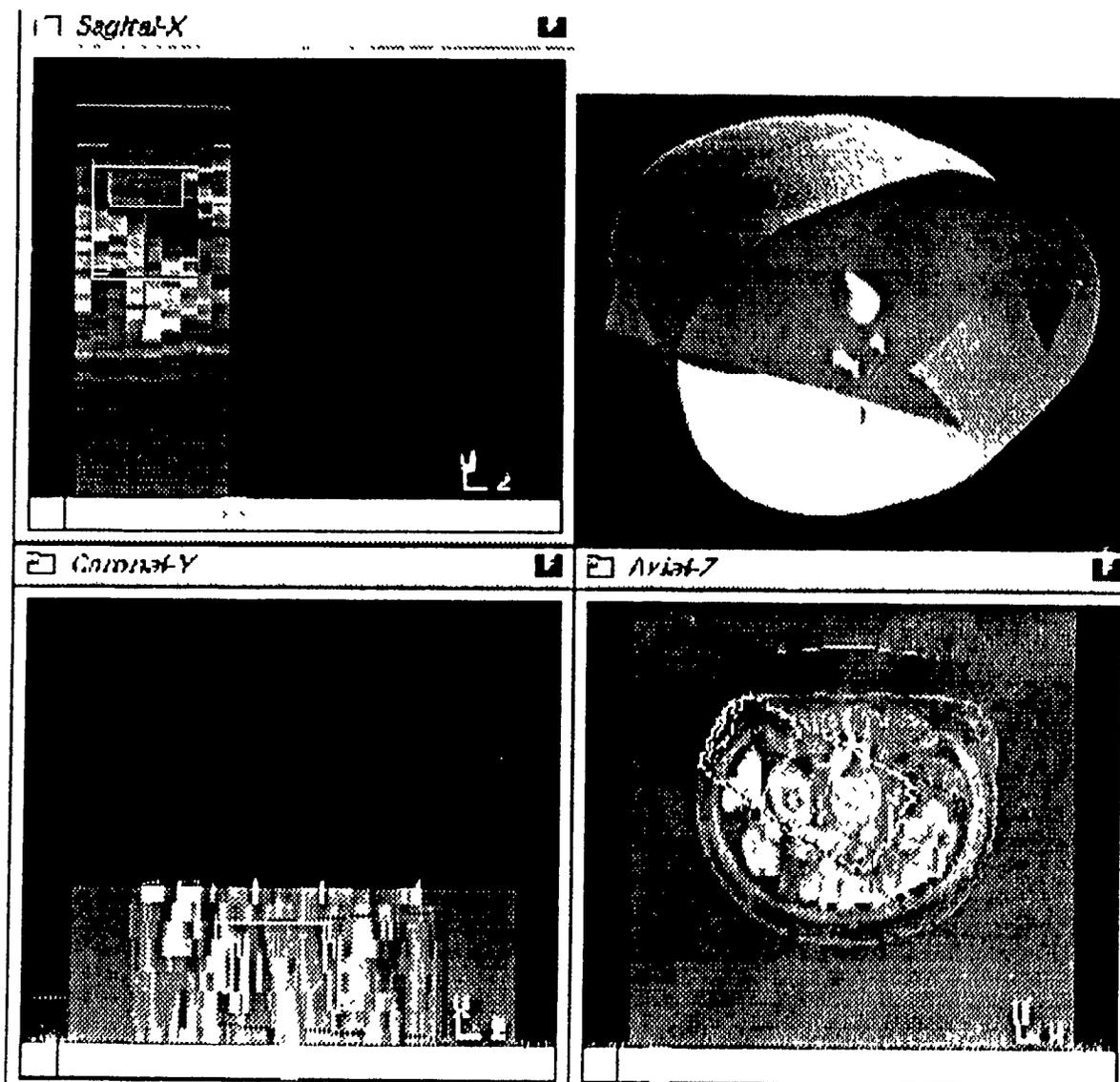


Figure 8.1 Example of a hybrid system using a combination of reformatting of data and surface rendering. The reformatting of data provides a detailed view of the volume while the surface rendering helps provide the "understanding at a glance"

Bibliography

Bentley R.E., 1964. "Digital computers in radiation treatment planning" *Brit. J. Radiol.*, **37**, pp. 748-755. [5]

Bentley R.E., Milan J., 1971. "An interactive computer system for radiotherapy treatment planning". *Brit. J. Radiol.*, **44**, No 527, pp. 826-833 [6, 34]

Boissonnat J-D , 1988. "Shape Reconstruction from Planar Cross Sections", *Computer Vision, Graphics, and Image Processing*, **44**, pp 1-29.[29]

Chin L.M., Siddon R.L., Svensson G.K., and Rose C., 19-85, "Progress in 3-D treatment planning for photon beam therapy", *Int J Radiation Oncology Biol. Phys* **11**, pp. 2011-2020 [11]

Doppke K.P., Goitein M., 1988, "A survey or the information gained from planning treatment with a computer", *Med. Phys.*, **15**(2), pp. 258-262 [5]

Goitein M., Abrams M., Rowell D., Pollari H., and Wiles J., 1983, "Multi-dimensional treatment planning II: Beam's Eye-view, back projection, and projection through CT sections", *Int J Radiation Oncology Biol. Phys.*, **9**, pp. 789-797 [11, 13, 16, 19]

Gouraud H., "Continuous shading of curved surfaces", 1971, *IEEE Trans. Comp.* **20**, pp 623-628 [13]

Hounsfield G.N., 1973, "Computerized transverse axial scanning (tomography)", *Brit. J. Radiol.*, **46**, p. 1016 [7]

Jacky J., 1990, "3-D Radiation therapy treatment planning: Overview and assessment", *Am J Clin Oncol* **13**(4), pp. 331-343 [10, 12, 19]

McCullough E.C., and Krueger A.M., 1980, "Performance evaluation of computerized treatment planning systems for radiotherapy: External photon beams", *Int. J. Radiation Oncology Biol. Phys.*, **6**, pp 1599-1605 [21]

McShan D.L., Fraass B.A., and Lichter A.S., 1990, "Computerized treatment planning", *Int J Radiation Oncology Biol. Phys* , **18**(6), pp. 1485-1494 [11]

Milan J., Bentley R.E., 1973. "The Storage and Manipulation of Radiation Dose Data in a Small Digital Computer." *Brit. J. of Radiol.*, **47**, pp. 115-121 [103]

Mohan R., Barest G., Brewster L.J., Chui C.S., Kutcher G.J., Laughlin J.S., and Fuks Z., 1988, "A comprehensive three-dimensional radiation treatment planning system", *Int. J. Radiation Oncology Biol. Phys.*, **15**, pp. 481-495 [10, 11, 17, 19]

Mohan R., Brewster L.J., Barest G., Ding I.Y., Chui C.S., Shank B., and Vikram B., 1987, "Arbitrary oblique image sections for 3-D radiation treatment planning" ,*Int. J. Radiation Oncology Biol. Phys.*, **15**, pp.1247-1254 [10, 19]

Newman W.M., Sproull R.F., *Principles of interactive computer graphics* 2nd ed. (McGraw-Hill 1979) [9]

Overmars M.H., *FORMS A C-library for dialogues*, Department of Computer Science, Utrecht University, P.O.Box 80.089, 3508 TB Utrecht, the Netherlands. [34]

Phong B.T., 1973, "Illumination for computer generated images", Ph.D. Dissertation, University of Utah [13]

Rosenman J., Sherouse G.W., Fuchs H., Pizer S.M., Skinner A.L., Mosher C., Novins B.S. and Tepper J.E., 1989, "Three-dimensional display techniques in radiation therapy treatment planning", *Int. J. Radiation Oncology Biol. Phys.*, **16**, pp. 263-269 [13, 15]

Sinclair B., Hannam A.G., Lowe A.A., and Wood W.W., 1989, "Complex Contour Organization for Surface Reconstruction", *Computers and Graphics*, **13**, pp 344-349. [29]

Stern R.L., Fraass B.A., Gerhardsson A., McShan D.L., and Lam K.L., 1992, "Generation and use of measurement based 3-D dose distributions for 3-D dose calculation verification", *Med. Phys.* **19**(1), pp 165-173 [21, 39, 40, 109]

Tsien K.C., 1955, *Brit. J. Radiol.*, **28**, 432; 1958, *ibid.*, 31, 32. [5]