

**FLEXIBLE OCCLUSION RENDERING FOR
IMPROVED VIEWS OF THREE-DIMENSIONAL
MEDICAL IMAGES**

David Borland

A dissertation submitted to the faculty of the University of North Carolina at Chapel Hill in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Computer Science.

Chapel Hill
2007

Approved by:

Russell M. Taylor II

John P. Clarke

Anselmo Lastra

Guido Gerig

Gary Bishop

© 2007
David Borland
ALL RIGHTS RESERVED

ABSTRACT

DAVID BORLAND: Flexible Occlusion Rendering for Improved Views of
Three-Dimensional Medical Images
(Under the direction of Russell M. Taylor II)

The goal of this work is to enable more rapid and accurate diagnosis of pathology from three-dimensional (3D) medical images by augmenting standard volume rendering techniques to display otherwise-occluded features within the volume. When displaying such data sets with volume rendering, appropriate selection of the transfer function is critical for determining which features of the data will be displayed. In many cases, however, no transfer function is able to produce the most useful views for diagnosis of pathology.

Flexible Occlusion Rendering (FOR) is an addition to standard ray cast volume rendering that modulates accumulated color and opacity along each ray upon detecting features indicating the separation between objects of the same intensity range. For contrast-enhanced MRI and CT data, these separation features are intensity peaks. To detect these peaks, a dual-threshold method is used to reduce sensitivity to noise. To further reduce noise and enable control over the spatial scale of the features detected, a smoothed version of the original data set is used for feature detection, while rendering the original data at high resolution. Separating the occlusion feature detection from the volume rendering transfer function enables robust occlusion determination and seamless transition from occluded views to non-occluded views of surfaces during virtual flythroughs.

FOR has been applied to virtual arthroscopy of joints from MRI data. For example, survey views of entire shoulder socket surfaces have been rendered to enable rapid evaluation by automatically removing the occluding material of the humeral head. Such views are not possible with standard volume rendering. FOR has also been successfully applied to virtual ureteroscopy of the renal collecting system from CT data, and knee fracture visualization from CT data.

ACKNOWLEDGMENTS

I would like to thank Russ Taylor for being a wonderful advisor throughout my graduate school career. His intelligence, creativity, and enthusiasm have been invaluable to me, and he has always managed to somehow find funding for me to work on something interesting. I would also like to thank John Clarke, whose idea was the inspiration for this work, and who has been a great source of knowledge in medical imaging and medicine. I also thank the rest of my committee members—Anselmo Lastra, Guido Gerig, and Gary Bishop—for their help and feedback.

I would like to thank the CISMM group, as well as the UNC Computer Science and Radiology Departments, especially Tim Quigg and Joe Lee, for helping fund my dissertation research. I would also like to thank Julia Fielding for collaboration and funding on the virtual ureteroscopy study.

While in graduate school, I've had the good fortune to work on a number of interesting projects. I would like to thank the Keck TEM facility, especially Lu-Chang Qin, for supporting my work on TEM simulation. I would like to thank the NC State Department of Education, especially Gail Jones and James Minogue, for their help and support on the HapticCell project. I would like to thank NC A&T University, especially Ajit Kelkar, for allowing me to co-teach their Visualization course, and the students of the Visualization course at NC A&T and UNC for their support. I would also like to thank the Durham Museum of Life and Science, especially Troy Livingston and Brad Herring, for funding me during my hectic last semester in graduate school.

The members of the UNC OTD, especially James Deane, Jackie Quay, and Lisa Darmo, and the law firm of Jenkins, Wilson, Taylor & Hunt, especially Greg Hunt, have been extremely helpful in protecting the intellectual property contained in this dissertation.

I would like to thank NSRG, the research group I have been a member of throughout my

time in graduate school. I would especially like to thank Rich Superfine and Sean Washburn for their expertise, David Marshburn for all of his technical support throughout the years, and Cory Quammen for his work on implementing VDP on graphics hardware and our often amazingly insightful and trenchant conversations.

I would like to thank all the members of the UNC Computer Science Department. I truly feel that this is a very special department, and I am proud to have been a small part of it. I thank the faculty and staff, including Janet Jones, Sandra Neely, Tammy Pike, Kelli Gaskill, Missy Wood, David Harrison, John Thomas, David Musick, and Brian White, among others, for keeping the department running as remarkably smoothly as it does. I would also like to thank my fellow graduate students for their friendship throughout the years, especially James Branigan, without whom I probably would never have gotten through my first two years, at least not with my sanity relatively intact. I also thank Bill Baxter, Eric Bennett, Eli Broadhurst, Greg Coombe, Brian Eastwood, Casey Goodlett, Chad Hantak, Mike Henson, Jason Jerald, Luv Kohli, Ben Lok, Jason Stewart, Brad Moore, Kyle Moore, Andrew Nashel, Michael Noland, Stephen Olivier, Cory Quammen, Sharif Razzaque, Alvin Richardson, Brian Salomon, Adam Seeger, Rick Skarbez, Josh Stough, Jeff Terrel, Tim Thirion, Chris Vanderknyff, Kelly Ward, Ben Wilde, Chris Weigle, Michelle Weigle, Kirstin Williams, for a myriad of reasons, both technical and personal.

I also thank my wonderful girlfriend, Tabitha Peck, who has supported me in the final stages of my dissertation work, including during the minor interruption resulting from major knee surgery.

Most importantly, I would like to thank my family. Gillian, Patrick, Sam, Erin, and Hayley Young have always been a fun and loving distraction from my graduate school work. My wonderful parents, Tom and Jean Borland, have always supported me in every way imaginable, and this dissertation is dedicated to them.

TABLE OF CONTENTS

LIST OF TABLES	xiii
LIST OF FIGURES	xiv
LIST OF ABBREVIATIONS	xix
1 INTRODUCTION	1
1.1 Goal	1
1.2 Description	2
1.3 Thesis Statement	3
1.4 Results	3
1.5 Overview	4
2 MEDICAL IMAGING	6
2.1 Background	6
2.1.1 Ionizing Modalities	7
2.1.2 Non-Ionizing Modalities	7
2.2 Magnetic Resonance Imaging	8
2.2.1 History	8
2.2.2 Nuclear Magnetic Resonance	8
2.2.2.1 Microscopic magnetization	8
2.2.2.2 Precession	11
2.2.2.3 Net magnetization	12
2.2.2.4 Radio frequency excitation	14
2.2.2.5 Transverse and longitudinal magnetization	14

2.2.2.6	Transverse relaxation	17
2.2.2.7	Spin echoes	17
2.2.2.8	Longitudinal relaxation	21
2.2.2.9	Contrast mechanisms	21
2.2.3	Image Formation	23
2.2.3.1	Slice selection	24
2.2.3.2	Slice frequency encoding	24
2.2.3.3	Slice phase encoding	27
2.3	Computed Tomography	29
2.3.1	History	29
2.3.2	X-Ray Radiography	29
2.3.2.1	X-rays	29
2.3.2.2	X-ray source	30
2.3.2.3	Attenuation	31
2.3.3	Tomography	32
2.3.3.1	The Radon transform	33
2.3.3.2	The Fourier-slice theorem	34
2.3.3.3	Simple backprojection	35
2.3.3.4	Filtered backprojection	38
2.4	Medical Images	39
2.4.1	Anatomy	39
2.4.1.1	Directions	39
2.4.1.2	Reference planes	40
2.4.2	Medical Image Display	40
2.4.3	Comparison of Magnetic Resonance and Computed Tomography Images	42
2.4.3.1	Anatomical information	43
2.4.3.2	Image characteristics	43
2.4.3.3	Image artifacts	44

2.4.3.4	Scan time and cost	45
3	BACKGROUND AND RELATED WORK	47
3.1	Volume Graphics	47
3.1.1	Surface Rendering	48
3.1.2	Volume Rendering	48
3.1.2.1	Sampling	49
3.1.2.2	Transfer function evaluation	52
3.1.2.3	Shading	55
3.1.2.4	Compositing	55
3.1.2.5	Other volume rendering methods	55
3.1.3	Comparison of Surface Rendering and Volume Rendering	56
3.1.4	Determining the Transfer Function	57
3.2	Dealing with Occlusion in Volume Graphics	58
3.2.1	Illustrative Techniques	58
3.2.2	Focus-and-Context Techniques	59
3.3	Volume Rendering of Medical Images	60
4	FLEXIBLE OCCLUSION RENDERING	62
4.1	Motivating Example	62
4.1.1	The Desired System	63
4.1.2	Potential Solutions	64
4.1.2.1	Clipping plane	64
4.1.2.2	Segmentation	65
4.1.2.3	Isosurfaces	68
4.2	Volumetric Depth Peeling	69
4.2.1	Fixed-Threshold Method	72
4.2.1.1	Algorithm	72
4.2.1.2	Selecting the threshold	74
4.2.1.3	Extensions	78

4.2.1.4	Limitations	82
4.2.2	Intensity Peak Determination Methods	84
4.2.2.1	Global maximum	84
4.2.2.2	First local maximum	84
4.2.2.3	Dual-threshold first local maximum	85
4.3	Noise and Filtering	89
4.4	Flexible Occlusion Rendering	91
4.4.1	Fixed-threshold FOR	91
4.4.2	First local maximum FOR	91
4.4.3	Knee fracture visualization	92
4.4.3.1	Problems with FOR	92
4.4.3.2	Flood-filled mask	93
4.4.3.3	Flood-filled mask at scale	95
4.5	Recap	97
5	VIRTUAL ARTHROSCOPY	98
5.1	Joint Anatomy	98
5.1.1	The Shoulder	99
5.1.2	Common Derangements	100
5.1.2.1	Bankart lesion	100
5.1.2.2	Hill-Sachs lesion	100
5.1.2.3	SLAP tear	101
5.2	Standard Joint Derangement Diagnosis	101
5.2.1	Diagnostic Clinical Arthroscopy	102
5.2.2	Magnetic Resonance Imaging	102
5.2.2.1	MRI sequences used	103
5.2.2.2	Advantages	104
5.2.2.3	Disadvantages	104
5.2.3	Standard Virtual Arthroscopy	104
5.2.3.1	Advantages	104

5.2.3.2	Disadvantages	105
5.3	Flexible Occlusion Rendering for Virtual Arthroscopy	106
5.3.1	Shoulder virtual arthroscopy	106
5.3.1.1	Torn middle glenohumeral ligament	107
5.3.1.2	Bankart lesion	107
5.3.1.3	Hill-Sachs lesion	109
5.3.1.4	SLAP tear	110
5.3.1.5	Multiple shoulders	112
5.3.1.6	Selecting scale	112
5.3.1.7	Discussion	113
5.3.2	Other Joints	114
6	VIRTUAL URETEROSCOPY	116
6.1	The Renal Collecting System	116
6.1.1	The Kidneys and Ureters	116
6.1.2	Common Pathologies	117
6.1.2.1	Kidney stones	117
6.1.2.2	Tumors	117
6.1.2.3	Deformed calyces	117
6.2	Standard Ways of Diagnosing Pathology in the Renal Collecting System . . .	118
6.2.1	Ureteroscopy	118
6.2.2	CT Urogram Slice Viewing	118
6.3	Flexible Occlusion Rendering for Virtual Ureteroscopy	119
6.3.1	Glove-Phantom Study	119
6.3.2	Contrast-Enhanced CT Study	121
6.3.3	Conclusion	122
7	KNEE FRACTURE VISUALIZATION	124
7.1	The Knee	124
7.1.1	Tibial Plateau Fractures	125

7.2	Standard Ways of Viewing Tibial Plateau Fractures	125
7.2.1	X-Ray Imaging	125
7.2.2	CT Slice Viewing	126
7.3	Flexible Occlusion Rendering for Knee Fracture Visualization	127
7.3.1	Knee 1	127
7.3.1.1	Grazing effects	130
7.3.2	Knee 2	130
7.3.3	Limitations	133
7.4	Conclusion	133
8	CONCLUSION	134
8.1	Results	134
8.2	Limitations and Future Work	135
8.2.1	MRI Data	135
8.2.2	CT Data	136
8.3	Future Directions	136
A	INNERSPACE	138
A.1	Interface	139
A.1.1	Loading Data	139
A.1.2	Interacting with the Data	140
A.1.2.1	Slice views	140
A.1.2.2	3D view	141
A.1.2.3	FOR controls	143
A.1.2.4	Rendering controls	145
A.1.2.5	Miscellaneous	147
A.2	VTK Extensions	147
A.2.1	FOR	147
A.2.2	Widgets	147

A.2.3 Performance	148
-----------------------------	-----

BIBLIOGRAPHY

149

LIST OF TABLES

2.1	MRI Contrast Generation Timing Parameters	22
2.2	CT numbers for common materials	44
A.1	Innerspace Rendering Performance	148

LIST OF FIGURES

1.1 Comparison of shoulder socket images	2
2.1 The first medical image	7
2.2 MRI image of a knee	9
2.3 Spinning nucleus	10
2.4 Random nuclear spin axes	10
2.5 Aligned nuclear spin axes	11
2.6 Phase angle of a nuclear spin axis	12
2.7 Sum of individual nuclear magnetic moments	13
2.8 Net magnetic moment aligned with B_0	13
2.9 Effect of B_1 on an individual nucleus	14
2.10 Effect of B_1 on $M(t)$	15
2.11 Net magnetic moment precession	16
2.12 Nuclear spins projected onto transverse plane	16
2.13 T_2 and T_2^* decay	18
2.14 T_2 and T_2^* dephasing	19
2.15 Spin echo formation	20
2.16 T_1 recovery	21
2.17 MRI Sequences	23
2.18 Applying a z -gradient to B_0	24
2.19 Slice selection using frequency encoding	25
2.20 Summed signal amplitudes	25
2.21 Frequency encoded signal	26
2.22 Phase encoding	28
2.23 K -space	28
2.24 CT image of a knee	29
2.25 X-ray tube	30

2.26	X-ray attenuation	31
2.27	CT scanner	32
2.28	The Radon transform.	33
2.29	The Fourier-slice theorem	36
2.30	Simple backprojection	36
2.31	Filtered backprojection	38
2.32	Anatomical reference planes.	39
2.33	Window/level controls	41
2.34	Window/level example	41
2.35	Multi-planar reconstruction	42
3.1	Voxels	47
3.2	Isosurface rendering of a knee from CT data	48
3.3	Sampling a volume	49
3.4	Nearest neighbor interpolation artifacts	50
3.5	Trilinear interpolation	50
3.6	Sampling artifacts	51
3.7	Color transfer function	52
3.8	Scalar opacity transfer function	52
3.9	Transfer function effects	53
3.10	Gradient transfer function effects	54
3.11	Volume rendering data flow	56
3.12	Occlusion-handling techniques	58
3.13	Virtual colonoscopy	61
4.1	Shoulder socket illustration	62
4.2	Shoulder socket rendered using transparency	63
4.3	Shoulder diagram	64
4.4	Clipping plane diagram	65
4.5	Segmentation failure diagram	67

4.6	Self occlusion diagram	67
4.7	Isosurface backface culling diagram	69
4.8	Standard volume rendering of a shoulder socket from MRI data compared with VDP	70
4.9	Shoulder MRI	71
4.10	Fixed-threshold VDP diagram	72
4.11	Fixed-threshold VDP within a standard volume rendering implementation.	73
4.12	Self occlusion fixed-threshold VDP diagram	74
4.13	Shoulder MRI with black-body radiation color map	75
4.14	Idealized shoulder MRI intensity profile	75
4.15	Cartilage surface scalar opacity transfer function	76
4.16	Diagram of different thresholds	77
4.17	Effects of different thresholds	77
4.18	Accumulated color and opacity modulation	78
4.19	Display of rays that don't reach the threshold	79
4.20	Accumulated color and opacity modulation combined with display of rays that don't reach the threshold.	80
4.21	Skip-to-threshold VDP.	81
4.22	Fixed-threshold VDP limitations	83
4.23	Intensity profile along a ray where the threshold is not reached	83
4.24	First local maximum VDP.	86
4.25	First local maximum VDP thresholds	87
4.26	First local maximum VDP example	87
4.27	Using a gradient opacity transfer function	88
4.28	Highlighting areas where surfaces touch	89
4.29	Rendering smoothed data	90
4.30	Diagram of a knee CT	92
4.31	Diagram of a thresholded knee CT	93
4.32	Diagram of a thresholded and flood-filled knee CT	94

4.33 Overestimate of separation	95
4.34 Underestimate of separation	96
4.35 Mask smoothing	96
5.1 Shoulder joint illustration	99
5.2 Joint capsule illustration	100
5.3 Glenoid illustration	101
5.4 Arthroscopy example	102
5.5 Orthogonal shoulder MRI slices	103
5.6 Standard shoulder virtual arthroscopy	105
5.7 Shoulder socket cutaway illustration compared with FOR	106
5.8 FOR shoulder socket survey view	107
5.9 Torn ligament	108
5.10 Bankart lesion	108
5.11 Humeral head	109
5.12 Hill-Sachs lesion	110
5.13 FOR shoulder socket survey view of SLAP tear	111
5.14 SLAP tear	111
5.15 Multiple shoulder sockets	112
5.16 Occlusion feature scale	113
5.17 Other joints rendered using FOR	114
6.1 Kidney diagram	117
6.2 Orthogonal CT slices of the renal collecting system.	119
6.3 Standard volume rendering of glove phantom compared with FOR.	120
6.4 Glove phantom rendered using FOR	121
6.5 Standard volume rendering of the renal collecting system compared with FOR	122
6.6 Standard volume rendering of a deformed calyx compared with FOR	123
7.1 Knee diagram	124
7.2 Knee x-ray image	126

7.3	Coronal slice of a knee CT	127
7.4	Volume rendering of knee bones from CT data	128
7.5	Volume rendering of a tibial plateau fracture from CT data	129
7.6	FOR rendering of a tibial plateau fracture	129
7.7	Rendering of the femur from within the tibia.	130
7.8	Different amounts of smoothing applied to the occlusion data.	131
7.9	Survey view of a severe tibial plateau fracture.	131
7.10	Standard volume rendering of a severe tibial plateau fracture compared with FOR.	132
8.1	Engine block rendered using FOR	137
A.1	Innerspace main window	138
A.2	DICOM series selection dialog	139
A.3	Camera widget	140
A.4	Slice Views Controls window	140
A.5	Intensity profile	142
A.6	VDP Controls window	144
A.7	Occlusion Data Controls window	145
A.8	Rendering Controls window	146

LIST OF ABBREVIATIONS

1D	One-dimensional
2D	Two-dimensional
3D	Three-dimensional
CAT	Computer-aided tomography or computed axial tomography
CT	Computed tomography
DICOM	Digital Imaging and COmmunications in Medicine
FFT	Fast Fourier transform
FID	Free induction decay
FOR	Flexible occlusion rendering
FPS	Frames per second
GDCM	Grass roots DICOM
GUI	Graphical user interface
HU	Hounsfield unit
MPR	Multi-planar reconstruction
MRI	Magnetic resonance imaging or magnetic resonance image
NMR	Nuclear magnetic resonance
NMRI	Nuclear magnetic resonance imaging
PET	Positron emission tomography
RF	Radio frequency
ROI	Region of interest
SLAP	Superior labrum anterior-posterior
SPECT	Single photon emission computed tomography
T	Tesla
VDP	Volumetric depth peeling
VTK	Visualization ToolKit
WEDESS	Water excitation dual echo in the steady state

CHAPTER 1

INTRODUCTION

1.1 Goal

The goal of this work is to enable more rapid and accurate diagnosis of pathology from three-dimensional (3D) medical images by augmenting standard volume rendering techniques to display otherwise-occluded features within the volume. Viewing 3D reconstructions of objects from magnetic resonance imaging (MRI) and computed tomography¹ (CT) data is more intuitive than mentally reconstructing these objects from orthogonal slices of the data because the human visual system understands 3D shape through perception of surfaces (Gibson, 1979; Ware, 2004). Viewing slices of medical images becomes increasingly problematic as data set sizes increase due to ever-improving scanner resolutions.

When displaying such data sets with volume rendering, appropriate selection of the transfer function is critical for determining which features of the data will be displayed. The work in this dissertation is necessary because it is often the case that, due to occlusion, no transfer function is able to produce the most useful views for diagnosis of pathology.

Flexible occlusion rendering (FOR) was developed to make view-dependent display of occluded objects in a volume possible and easy to generate for physicians performing diagnosis. For example, when evaluating a shoulder joint, a view showing the entire shoulder socket is the most useful view for rapid evaluation. However, such views are impossible using standard volume rendering, as the material in the humeral head (the “ball” in this ball-and-socket joint) will occlude the shoulder socket from this vantage. Adjusting the transfer function to reduce the opacity of the humeral head will also reduce the opacity of the shoulder socket, combining

¹Computed tomography is also known as computer-aided tomography or computed axial tomography (CAT).

to form a confusing image. (Interrante et al., 1997) shows the difficulty of determining the shape of two overlapping semi-transparent surfaces. Using FOR, the occluding material of the humeral head is automatically removed, enabling display of the entire shoulder socket without transparency. Figure 1.1 shows a typical cutaway text-book illustration of a shoulder socket, with the ball of the shoulder removed, compared to similar views using standard volume rendering with transparency and FOR to render data of a real shoulder. This example

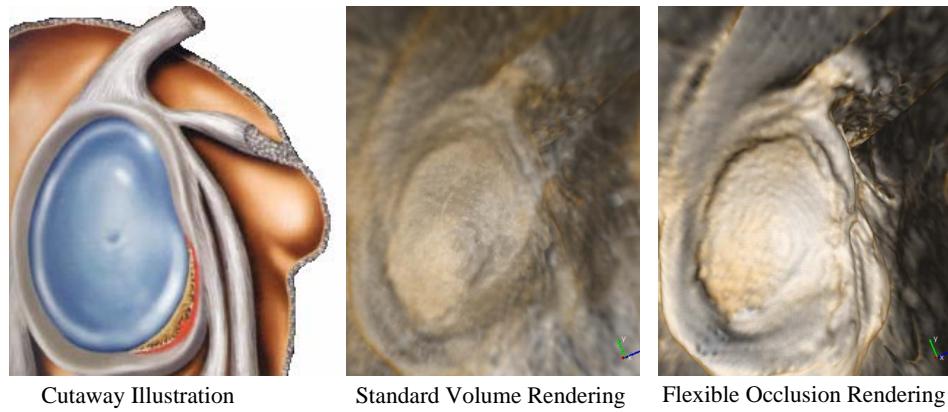


Figure 1.1: Cutaway illustration of a shoulder socket from a textbook (Stoller et al., 2004) compared with standard volume rendering using transparency and flexible occlusion rendering of magnetic resonance imaging data.

is indicative of the types of views that FOR can provide to aid physicians in performing diagnosis.

1.2 Description

FOR is an addition to standard ray cast volume rendering that modulates accumulated color and opacity along each ray when certain features are detected. Important for the applications described in this dissertation are features representing the separation between objects of the same intensity range. For contrast-enhanced MRI and CT data, these separation features are intensity peaks along the ray. To detect these peaks, a dual-threshold method is used to reduce sensitivity to noise. To further reduce noise and enable control over the spatial scale of the features detected, a smoothed version of the original data set is used

for feature detection, while rendering the original data at high resolution. For visualizing bone fractures from non-contrast-enhanced CT data, a smoothed binary mask of the original data is used for occlusion determination. Upon detecting occlusion features, the accumulated color and opacity along the ray is modulated or reset, exposing the far surface. Separating the occlusion feature detection from the volume rendering transfer function enables robust occlusion determination and seamless transition from occluded views to non-occluded views of surfaces during virtual flythroughs.

1.3 Thesis Statement

Decoupling occlusion determination from the volume rendering transfer function by finding *separation features* at *suitable scales*, using appropriate and *possibly different* data sets for rendering and feature detection, enables rapid view-dependent occlusion handling without requiring segmentation. This enables more effective and intuitive views for diagnosing pathology in virtual arthroscopy, virtual ureteroscopy, and knee fracture visualization.

1.4 Results

FOR has been successfully used for a virtual ureteroscopy study of contrast-enhanced CT data (Fielding et al., 2006). The study consisted of an in vitro portion using CT scans of three rubber glove phantoms and an in vivo portion using ten CT urograms (20 kidneys). The rubber glove phantoms were filled with contrast agent, along with high-density objects, such as stones and shells, and lower-density objects, such as bean and pieces of meat, representing high-density kidney stones and lower-density tumors. In all three cases, being able to see the interior of the glove from exterior positions enabled rapid identification of all objects with no false negatives. For the in vivo portion of the experiment, no tumors were detected in either the FOR renderings or the original CT images, so there were no tumor false positives. All abnormally-shaped kidney features being studied were identified, and the interiors of all ureters were visible from exterior positions.

Applying FOR to contrast-enhanced shoulder MRIs has enabled views of the entire shoul-

der socket from positions within the humeral head (Borland et al., 2005). This enabled rapid survey of the shoulder socket and diagnosis of derangements including torn ligaments and Bankart lesions (Borland et al., 2006) along with Superior Labrum Anterior-Posterior (SLAP) tears.

Shoulder virtual arthroscopy using FOR has also enabled views of the humeral head not possible using standard volume rendering. This enabled diagnosis of Hill-Sachs lesions (Borland et al., 2006). Upon viewing a movie of a Hill-Sachs lesion generated using FOR, a musculoskeletal surgeon not involved with the project remarked that, were it his patient, he would change his approach to surgery given his new appreciation of the injury versus that gained from MRI slice viewing.

FOR has also been applied to contrast-enhanced MRIs of elbows, wrists, hips, knees, and ankles (Borland et al., 2006). In all cases, FOR enabled views of anatomical structures from viewpoints not possible with standard volume rendering. A non-contrast MRI of the authors's own knee has also been rendered using FOR, enabling views of the femur from within the tibia. Such views show an osteochondral lesion on the medial femoral chondyle, along with holes left from a cartilage transplant from a non-weight-bearing portion of the femur.

A program, Innerspace, has been created for displaying medical images using FOR. Innerspace can load DICOM files, an industry standard format for storing medical images, and the user can use FOR along with slice viewing to investigate the data. Innerspace is able to render MRI and CT data at interactive rates.

International Patent Application No. PCT/US2006/009331, *Methods, systems, and computer programming products for processing three-dimensional image data to render an image from a viewpoint within or beyond an occluding region of the image data*, has been filed by the inventors, encapsulating portions of the FOR framework. A second provisional patent application has also been filed, covering the remainder of the full FOR description.

1.5 Overview

The rest of this dissertation is organized as follows:

Chapter 2 gives background material in medical imaging, focusing on MRI and CT.

Chapter 3 gives an introduction to volume rendering along with previous work on dealing with occlusion in volume rendering and using volume rendering to visualize medical images.

Chapter 4 describes the FOR framework, using virtual arthroscopy of a shoulder from MRI data as a motivating example.

Chapter 5 gives the results of using FOR for virtual arthroscopy from contrast-enhanced and non-contrast-enhanced MRI data of various joints.

Chapter 6 gives the results of using FOR for virtual ureteroscopy from contrast-enhanced CT data of the urinary system and rubber-glove phantoms.

Chapter 7 gives the results of using FOR for visualizing knee fractures from CT data.

Chapter 8 concludes and gives future work.

Appendix A describes the Innerspace program used to visualize volumetric data using FOR.

CHAPTER 2

MEDICAL IMAGING

The Background and Magnetic Resonance Imaging sections in this treatment summarize portions of (Prince and Links, 2006), a good reference for those wishing a more in-depth presentation of medical imaging. Many of the diagrams in these sections are modifications of images from this text. The Computed Tomography section also borrows heavily from (Lalush, 2006).

2.1 Background

Medical imaging enables physicians to inspect the interior of the body without invasive surgical procedures. Because medical imaging is less invasive, it is typically safer and cheaper than invasive procedures. Medical imaging can also present information that cannot be obtained through surgical procedures. It is often used for diagnosis of pathology, for planning and guiding intervention, for monitoring therapy, and for understanding the function of biological processes.

Wilhelm Conrad Röntgen, the discoverer of x-rays (also known as Röntgen radiation) and winner of the first Nobel Prize in Physics, published the first medical image, an x-ray radiograph of his wife's hand (Figure 2.1). The first clinical use of x-rays occurred only two months later, in February 1896.

Many different medical imaging methods, or modalities, have been developed since the work of Röntgen. All of these modalities involve detecting physical signals emanating from or passing through the body. These signals arise from processes that can be divided into two categories: ionizing and non-ionizing.



Figure 2.1: The first medical image, an x-ray image of the hand of Wilhelm Conrad Röntgen’s wife (RHL, 2007). Note that the bone is dark. Current x-ray imaging techniques typically display the negative of this, with higher-density materials such as bone appearing bright.

2.1.1 Ionizing Modalities

X-ray radiography, CT, and nuclear medicine all produce images using ionizing radiation. Ionizing radiation carries enough energy to ionize an atom or molecule by removing an electron from its orbit. In large enough doses, ionizing radiation can be harmful to the human body, so patient dose is carefully regulated for safety. X-ray radiography works by transmitting x-rays through the body, which are attenuated to different degrees based on tissue density, and imaging the resulting 2D projection. CT uses a series of x-ray projections from various angles to reconstruct a 3D volume of tissue density. The three nuclear medicine modalities—radionuclide imaging, or scintigraphy, single-photon emission computed tomography (SPECT), and positron emission tomography (PET)—image gamma rays emitted from a radiotracer introduced within the body¹.

2.1.2 Non-Ionizing Modalities

Ultrasound, MRI, and optical imaging all use non-ionizing signals that do not damage tissue in the body in standard practice. Ultrasound images high-frequency sound waves reflecting off tissue within the body. MRI detects the precession of atomic spin systems in large magnetic fields. Optical imaging, such as optical coherence tomography (OCT), collects visible light to image tissue in the body.

¹SPECT and PET are both emission tomographies, distinguished from CT, which is also known as transmission tomography. All three modalities use similar tomographics reconstruction techniques.

This dissertation will concentrate on 3D rendering of both MRI and CT data. The remainder of this chapter will therefore describe these modalities in more detail, along with contrast agents used with each modality to aid in diagnosis of pathology.

2.2 Magnetic Resonance Imaging

2.2.1 History

MRI uses the process of nuclear magnetic resonance (NMR) to create images. NMR was first described independently in 1946 by both Felix Bloch and Edward Mill Purcell, who were awarded the 1952 Nobel Prize in Physics for this work. Originally, only a single NMR signal was generated for an entire volume, but in the early 1970s, ideas were developed for spatially encoding the NMR signal to create images. Raymond V. Damadian patented the use of NMR for detecting cancer. Paul Lauterbur and Sir Peter Mansfield were the first to discover that magnetic field gradients could be used to create images of NMR signals, and they were awarded the 2003 Nobel Prize in Medicine for this work. MRI was originally called nuclear magnetic resonance imaging (NMRI), but “nuclear” was dropped due to the negative connotations of the word. The first clinical MRI scanners were developed in the 1970s, and there has been continuous technical development and clinical use since then. MRI has a wide range of applications, including imaging soft tissue such as the brain, muscles, ligaments, tendons, and cartilage. A slice of an MRI image of a knee is shown in Figure 2.2.

2.2.2 Nuclear Magnetic Resonance

To understand MRI, it is necessary to understand the physics of NMR. Therefore, generating a single NMR signal from a volume will be discussed first, before discussing how this NMR signal is spatially encoded to enable MRI.

2.2.2.1 Microscopic magnetization

As its name suggests, NMR deals with the interaction between atomic nuclei and magnetic fields. All nuclei consist of elementary particles called protons and neutrons, and all elemen-



Figure 2.2: A slice of an MRI image of a knee showing dark bone, grey soft tissue, and a bright contrast agent injected into the joint.

tary particles have an intrinsic angular momentum Φ . This intrinsic angular momentum can be explained by assigning each elementary particle a property called spin. This spin is conceptually similar to the Earth spinning on its axis². If two elementary particles of the same type are paired together, their spins will cancel. Nuclei consist of different energy levels for protons and neutrons. To achieve the lowest possible energy state, protons will pair with protons, and neutrons will pair with neutrons. Therefore, all nuclei having either an odd atomic number or an odd mass number will have a net spin, and thus a net angular momentum. Such nuclei can be thought of as a spinning ball of charge, which creates a microscopic magnetic field with magnetic moment³

$$\boldsymbol{\mu} = \gamma \boldsymbol{\Phi}, \quad (2.1)$$

where γ is the gyromagnetic ratio of the nucleus in radians per second per tesla (Figure 2.3). In conventional MRI, only hydrogen nuclei, which consist of a single proton, are imaged. Hydrogen is abundant in the human body, mainly due to its occurrence in fat and water.

Under normal circumstances the spin axes of the hydrogen nuclei in a given volume align at random orientations (Figure 2.4). However, when a large static external magnetic field

²(Griffiths, 2004) gives a more in-depth discussion of quantum spin and intrinsic angular momentum.

³The magnetic moment is a vector pointing along the axis of a magnet, from the south pole to the north pole.

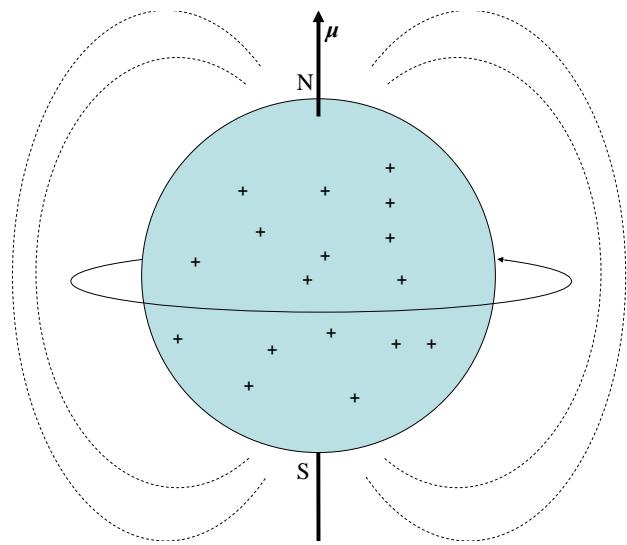


Figure 2.3: A spinning nucleus creates a magnetic moment μ . Image adapted from (Prince and Links, 2006).

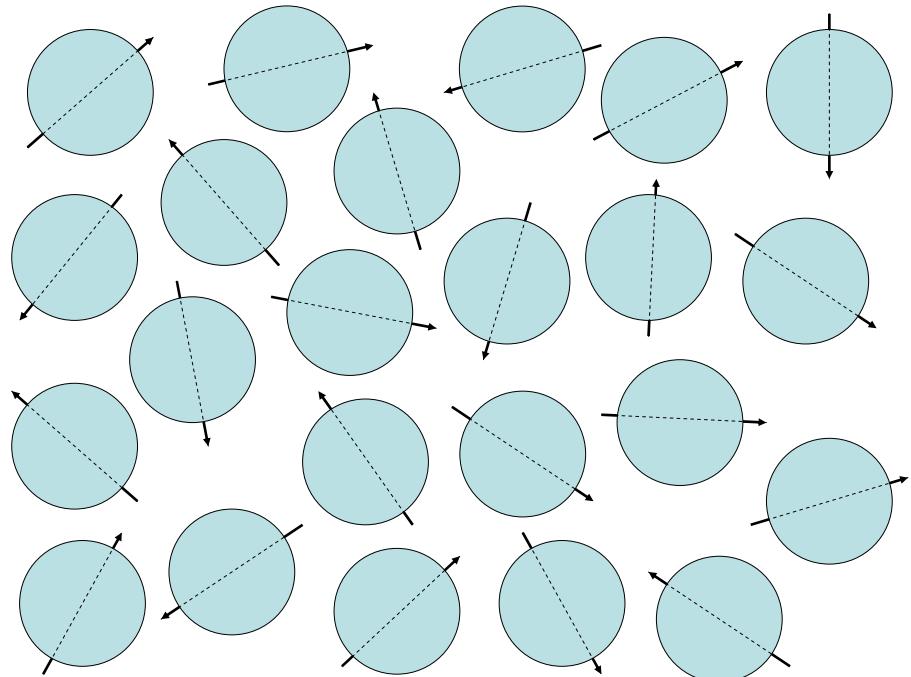


Figure 2.4: Nuclear spin axes with random orientations.

\mathbf{B}_0 is applied⁴, the axes align with respect to \mathbf{B}_0 . Due to quantum effects, this alignment is either at an up orientation angle of 54 degrees, or a down orientation angle of $180 - 54 = 126$ degrees (Figure 2.5). There exists a slight preference for the up orientation because it is the

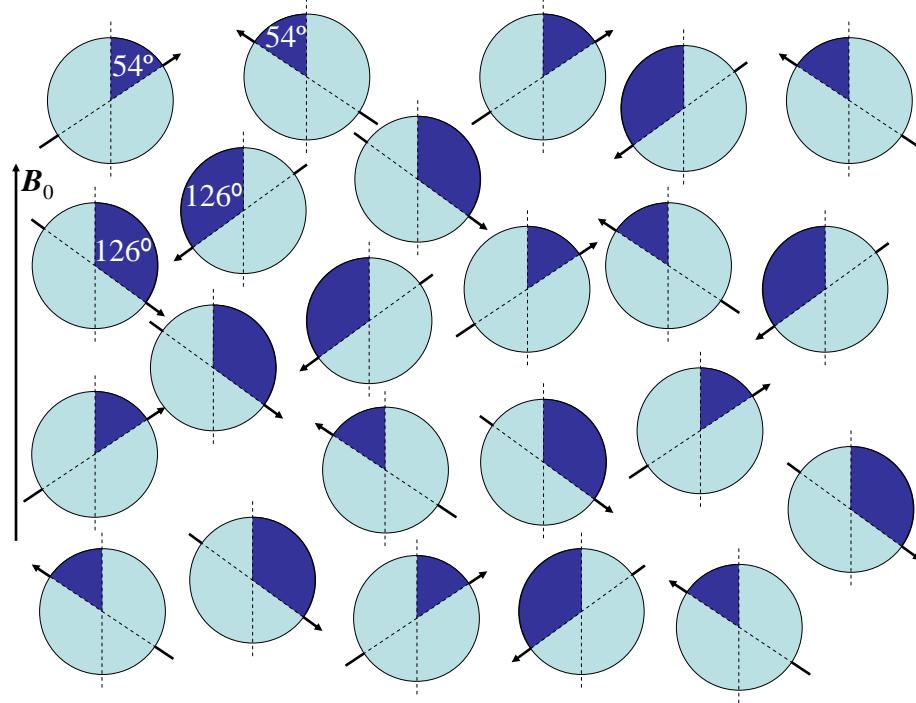


Figure 2.5: Nuclear spin axes align with external magnetic field \mathbf{B}_0 at the up orientation angle of 54 degrees or the down orientation angle of 126 degrees.

lower energy state. In addition to the orientation angle, each nuclear spin axis also has a randomly distributed phase angle around \mathbf{B}_0 (Figure 2.6).

2.2.2.2 Precession

Because μ for each nucleus is a magnetic moment, it experiences a torque when the external magnetic field \mathbf{B}_0 is applied, causing it to rotate, or precess, clockwise around \mathbf{B}_0 with frequency

$$\omega_0 = \gamma B_0, \quad (2.2)$$

⁴A typical strength of \mathbf{B}_0 for a clinical MRI scanner is 1.5 teslas (T), more than 100,000 times larger than the Earth's magnetic field on the surface of the Earth.

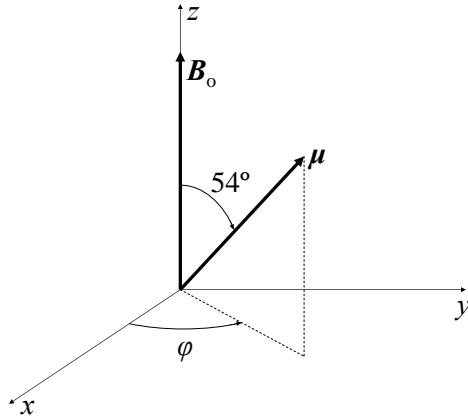


Figure 2.6: Each nuclear spin axis has a phase angle ϕ around \mathbf{B}_0 .

known as the Larmor frequency, where B_0 is the magnitude of \mathbf{B}_0 and γ is the gyromagnetic ratio of hydrogen. During this gyromagnetic precession there are two spins occurring: the intrinsic spin of the nucleus around its spin axis, and the rotation of the spin axis around \mathbf{B}_0 .

2.2.2.3 Net magnetization

The sum of the individual nuclear magnetic moments results in a net magnetic moment \mathbf{M} , which reaches an equilibrium value \mathbf{M}_0 as the individual nuclei align with respect to \mathbf{B}_0 . \mathbf{M}_0 has magnitude

$$M_0 = \frac{B_0 \gamma^2 \hbar^2}{4kT} P_D, \quad (2.3)$$

where \hbar is Planck's constant, k is Boltzmann's constant, T is the temperature in degrees Kelvin, and P_D is the proton density⁵. The random distribution of the phases of the spin axes as they precess around \mathbf{B}_0 results in a net magnetization of zero in the plane perpendicular to \mathbf{B}_0 . The slight preference for the up orientation results in a net magnetization aligned with \mathbf{B}_0 (Figure 2.7). The result is a net magnetic moment that is proportional to B_0 , aligned with \mathbf{B}_0 (Figure 2.8).

⁵In general, the P_D term is the number of targeted nuclei per unit volume. For conventional MRI, the targeted nuclei are hydrogen nuclei, each of which consists of a single proton. Thus the name proton density is used.

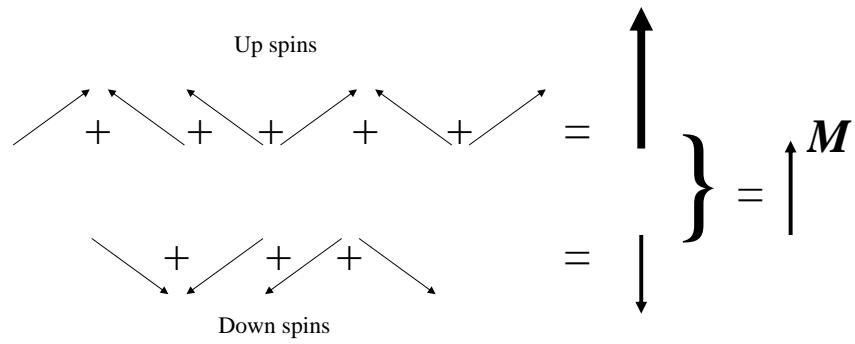


Figure 2.7: The net magnetic moment M is the sum of the individual nuclear magnetic moments.

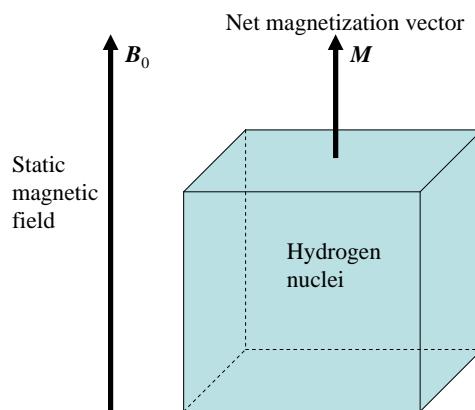


Figure 2.8: The net magnetic moment M is aligned with and proportional to B_0 . Figure recreated from (Prince and Links, 2006).

2.2.2.4 Radio frequency excitation

For a given nucleus, applying a small magnetic field \mathbf{B}_1 oriented along the x -axis will cause a small precession of $\boldsymbol{\mu}$ around the x -axis, resulting in a small motion of $\boldsymbol{\mu}$ towards the y -axis. By applying \mathbf{B}_1 as a radio frequency (RF) pulse at the Larmor frequency, $\boldsymbol{\mu}$ can be repeatedly tipped towards the y -axis through an incremental precession (Figure 2.9). By

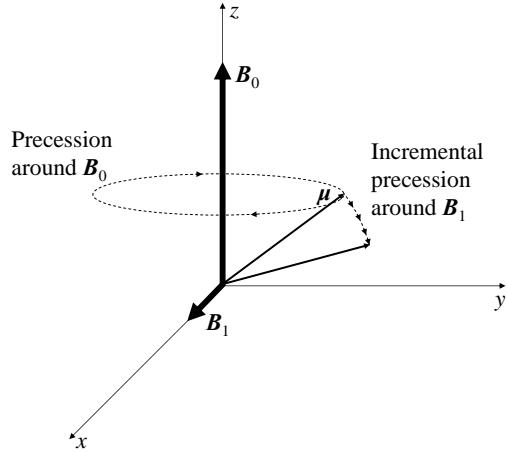


Figure 2.9: Applying \mathbf{B}_1 along the x -axis at the Larmor frequency causes $\boldsymbol{\mu}$ to incrementally rotate towards the y -axis.

applying \mathbf{B}_1 to tip over the individual nuclei, an angle α is introduced between \mathbf{M} and \mathbf{B}_0 (Figure 2.10). \mathbf{B}_1 is therefore referred to as an α -pulse. The individual nuclei will continue to precess around \mathbf{B}_0 at the Larmor frequency, causing \mathbf{M} to do the same.

2.2.2.5 Transverse and longitudinal magnetization

As it precesses, $\mathbf{M} = \mathbf{M}(t)$ is a function of time. $\mathbf{M}(t)$ can be conceptualized as having two components, the longitudinal component $M_z(t)$ oriented along the axis of \mathbf{B}_0 , and the transverse component $M_{xy}(t)$ lying in the plane orthogonal to the axis of \mathbf{B}_0 . The transverse magnetization is a complex quantity, written as

$$M_{xy}(t) = M_0 \sin \alpha e^{-i(\omega_0 t - \phi)}, \quad (2.4)$$

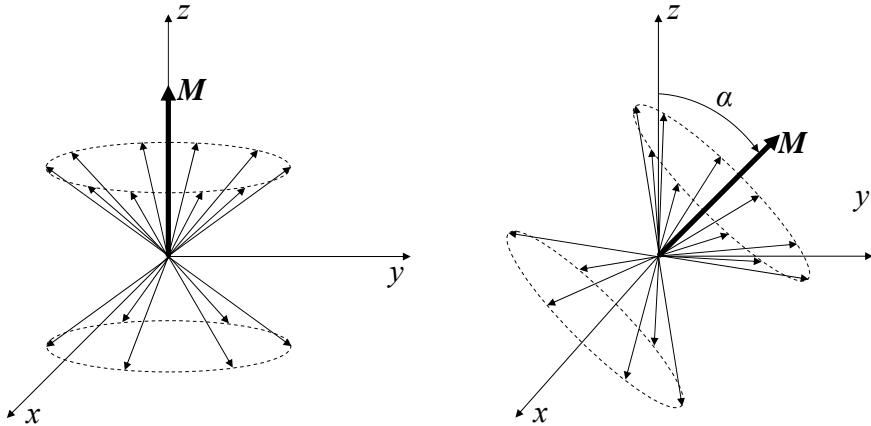


Figure 2.10: \mathbf{B}_1 tips \mathbf{M} by an angle α towards the y -axis.

where M_0 is the magnitude of $\mathbf{M}(t)$, α is the “tip” angle between $\mathbf{M}(t)$ and \mathbf{B}_0 , $i = \sqrt{-1}$, and ϕ is the phase angle of the transverse magnetization at time $t = 0$. This situation is represented in Figure 2.11. The rapidly rotating transverse magnetization creates an RF signal proportional to $|M_{xy}(t)|$ that induces a signal in an antenna⁶ located outside the sample. This is the measured signal in MRI. From (2.3) and (2.4), $M_{xy}(t)$ is dependent on the P_D of the tissue. P_D is therefore a source of contrast, as different tissue types have different P_D values. From (2.4), $|M_{xy}(t)|$ is maximized with an α of 90 degrees. A 90-degree α -pulse⁷ is therefore commonly used to tip the magnetic moment $\mathbf{M}(t)$ entirely into the transverse plane. The remainder of this discussion will assume a 90-degree α -pulse is used.

After applying the α -pulse, the individual nuclear spins are aligned around the y -axis, with the transverse components rotating in phase around the z -axis (Figure 2.12). Two processes then act to dampen the RF signal: transverse relaxation and longitudinal relaxation. These processes happen at different rates for different tissue types, so they are the two other sources of signal contrast in MRI, along with P_D .

⁶Sometimes called a resonator

⁷Commonly referred to as a $\pi/2$ pulse because π radians = 180 degrees.

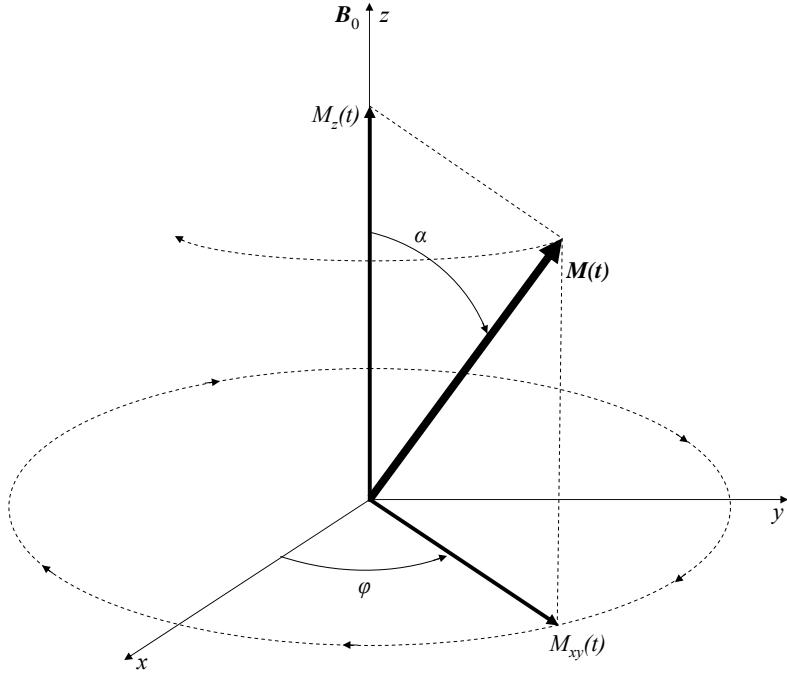


Figure 2.11: The net magnetization vector $\mathbf{M}(t)$, consisting of longitudinal component $M_z(t)$ and transverse component $M_{xy}(t)$, precesses clockwise around the axis of the magnetic field \mathbf{B}_0 . Figure adapted from (Prince and Links, 2006).

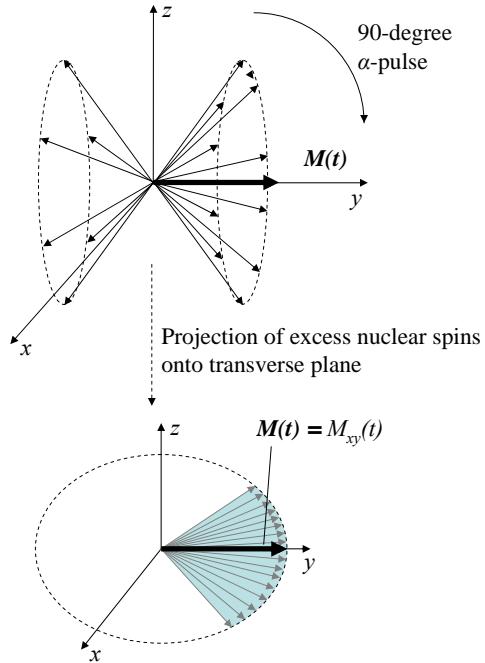


Figure 2.12: After a 90-degree α -pulse, the transverse components of the individual nuclear spins create a net transverse magnetization.

2.2.2.6 Transverse relaxation

Transverse relaxation acts fastest to cause the RF signal to decay. This decay, called free induction decay (FID), is caused by random perturbations due to local interactions between the nuclear spins that constitute $\mathbf{M}(t)$. These random perturbations result in a dephasing of the spins of the nuclei, as the precession of some speed up or slow down. This loss of coherence in the spins causes a reduction in the RF signal. The FID is modeled as an exponential decay with time constant T_2 .

In practice, the signal decays more rapidly than T_2 due to uncontrollable local perturbations in the static magnetic field \mathbf{B}_0 . Decay due to these static perturbations is modeled with time constant T'_2 . Together, T_2 and T'_2 effects form the actual T_2^* decay of the transverse magnetization, where

$$\frac{1}{T_2^*} = \frac{1}{T_2} + \frac{1}{T'_2}. \quad (2.5)$$

T_2^* decay thus results in lower signal and lower contrast than T_2 decay (Figure 2.13). Introducing a frame x', y' that rotates around z at the Larmor frequency is useful for illustrating transverse relaxation. Figure 2.14 illustrates the dephasing of the transverse components of the excess up-oriented⁸ nuclear spins that contribute to $M_{xy}(t)$, using this rotating frame.

2.2.2.7 Spin echoes

It is possible to recover the signal from that determined by the T_2^* rate to that determined by the T_2 rate, as T'_2 decay is due to reversible static effects. This recovery is performed by a spin echo RF pulse that rotates the precessing nuclei 180 degrees around the x' axis. From this flipped position, the dephased nuclei will begin to realign at the same speed at which they dephased due to T'_2 effects.

To illustrate this concept, imagine a race between a tortoise and a hare, in which the contestants are told to begin the race upon a starting gun shot, and then return to the starting line upon a second gun shot. For the first leg of the race, the hare will rapidly outgain the tortoise. However, upon the second gun shot, the contestants will turn around,

⁸With respect to the y -axis after a 90-degree α -pulse.

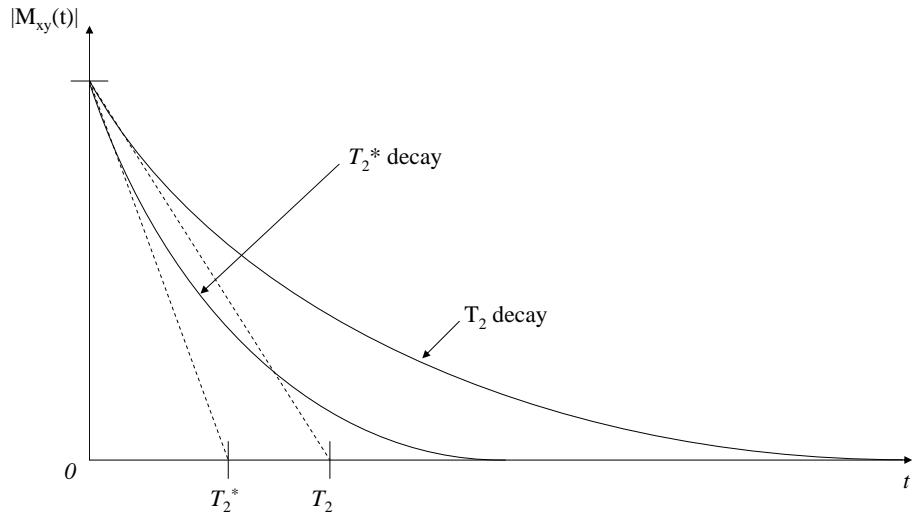


Figure 2.13: Transverse relaxation due to T_2 and T_2^* decay. Figure recreated from (Prince and Links, 2006).

heading back towards the starting line, and the tortoise will have the lead. Assuming both complete the second leg with the same speed they did the first, the hare will finally catch up to the tortoise back at the starting line, resulting in a tie.

In a similar fashion, a spin echo is formed when the spins realign, recovering the dephasing caused by T_2' effects. T_2 decay is not affected by the spin echo, so the echo forms with the amount of dephasing caused by T_2 decay (Figure 2.15). One difference from the race example is that instead of the nuclei reversing their direction and returning at the same speed, the 180-degree pulse flips the nuclei such that they will realign 180 degrees from their original alignment. The time at which the spin echo forms is called the echo time T_E , which can be controlled by specifying the time $T_E/2$ at which the 180-degree pulse is applied.

Although T_2' effects can be recovered by using spin echos, T_2 decay cannot be improved upon due to its random nature. The T_2 parameter is different for different tissue types, so it is a source of contrast in MRI images.

As the transverse relaxation is occurring, there exists another process, longitudinal relaxation, that acts to dampen the RF signal. However, this process typically happens at large enough timescales compared to T_2 decay that its effect on transverse relaxation can be ignored, and the amplitude of the transverse magnetization at T_E approximates the ideal T_2

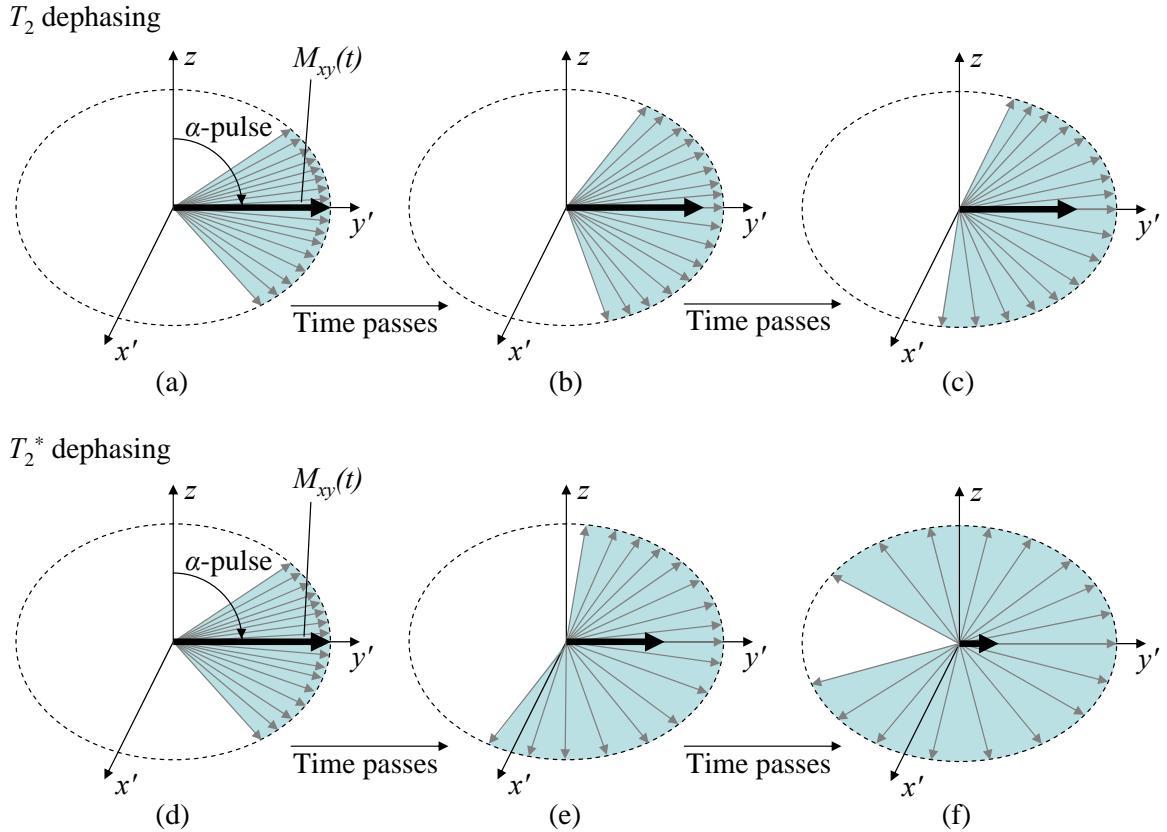


Figure 2.14: T_2 and T_2^* effects cause a dephasing of the transverse magnetism, depicted in the x', y' frame, which rotates at ω_0 around the z -axis. Figure adapted from (Prince and Links, 2006).

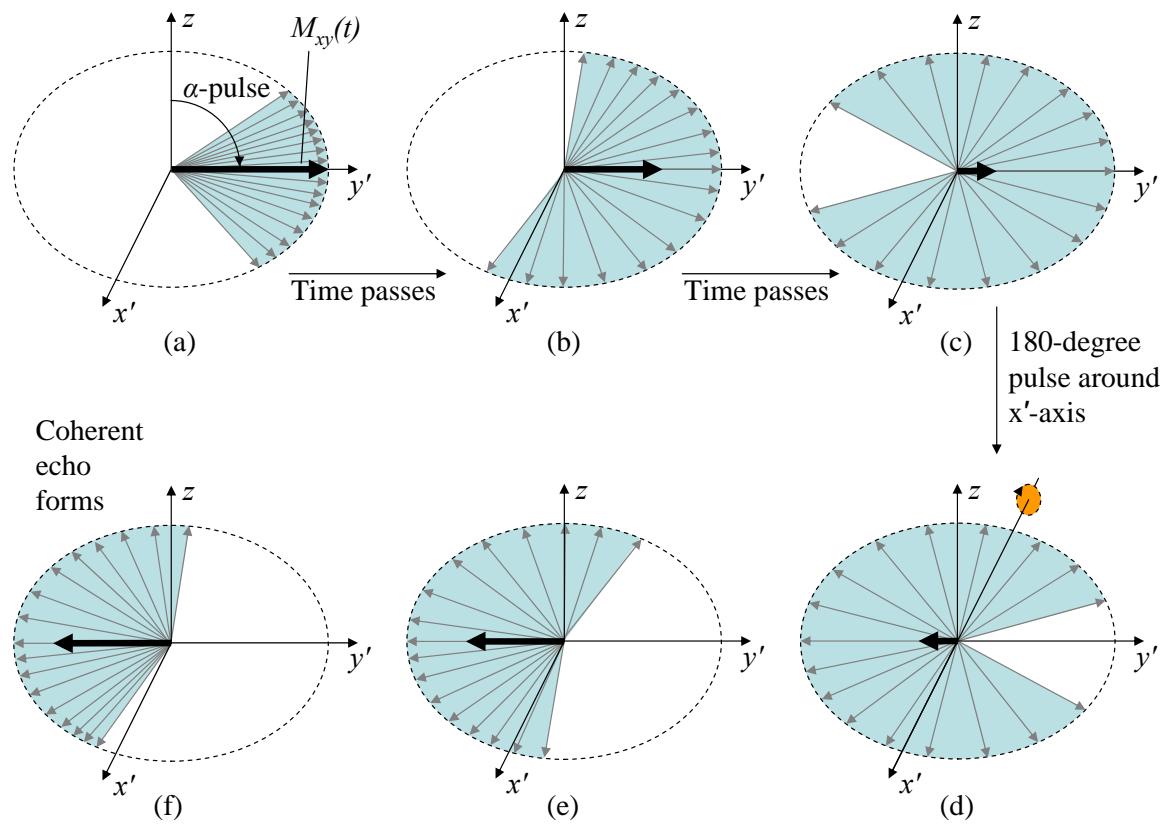


Figure 2.15: Formation of a spin echo, depicted in the x' , y' frame, which rotates at ω_0 around the z -axis. Figure adapted from (Prince and Links, 2006).

decay.

2.2.2.8 Longitudinal relaxation

Longitudinal relaxation is the second relaxation mechanism that occurs after the α -pulse tips $\mathbf{M}(t)$ into the transverse plane. This relaxation occurs as the individual nuclear spins realign with respect to \mathbf{B}_0 at the up and down orientations, pulling the longitudinal magnetization $M_z(t)$ back to its equilibrium value. The relaxation is modeled as a rising exponential with time constant T_1 (Figure 2.16). The T_1 parameter is different for different tissue types,

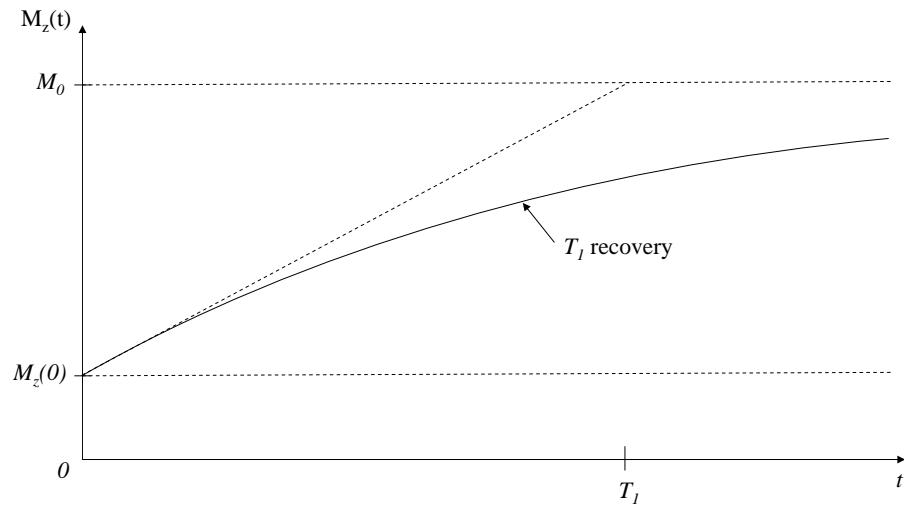


Figure 2.16: Longitudinal relaxation due to T_1 recovery. Figure recreated from (Prince and Links, 2006).

so it is the third source of contrast in MRI images, along with T_2 and P_D .

2.2.2.9 Contrast mechanisms

For tissues in the body, the relaxation times are in the ranges $250 \text{ ms} < T_1 < 2500 \text{ ms}$ and $25 \text{ ms} < T_2 < 250 \text{ ms}$. Usually, $5T_2 < T_1 < 10T_2$, and for all materials, $T_2 < T_1$. By controlling the tip angle α , the echo time T_E , and the pulse repetition interval T_R between successive α -pulses, different tissue contrasts can be generated in an MRI image. The three basic classes of MRI images are P_D -weighted, T_1 -weighted, and T_2 -weighted, determined by the most significant contrast mechanism used in each image.

Contrast	Timing Parameters
P_D	Long T_R , no spin echo or short T_E
T_2	Long T_R , $T_E \approx T_2$
T_1	$T_R \approx T_1$, no spin echo or short T_E

Table 2.1: MRI Contrast Generation Timing Parameters

For P_D -weighted contrast, the signal should be proportional to the number of hydrogen nuclei in the sample. From (2.3), the magnitude of the equilibrium net magnetization M_0 is proportional to P_D . To measure M_0 , $\mathbf{M}(t)$ is tipped over into the transverse plane with an α -pulse, and then the RF signal is measured quickly, before it can decay due to T_2 effects. P_D -weighted contrast can therefore be obtained by using a long T_R , which allows $\mathbf{M}(t)$ to return fully to equilibrium, and either no echo or a short T_E to minimize T_2 decay.

For T_2 -weighted contrast, differences in the transverse relaxation times of different tissues must be apparent. T_2^* effects cause the transverse magnetization to decay too rapidly to observe such differences. Spin echoes must therefore be used. There is a tradeoff in the value of T_E used. A short T_E will give a strong signal, as the signal will not have decayed much, but low contrast, as the difference between signals with different T_2 parameters increases over time. Conversely, a long T_E will give good contrast, but a weak signal. In practice, T_E should be approximately equal to the values of T_2 of the tissues being imaged. To obtain maximum signal, T_R should be long enough to allow $\mathbf{M}(t)$ to return fully to equilibrium.

For T_1 -weighted contrast, differences in the longitudinal relaxation times of different tissues must be apparent. T_1 -contrast timings are similar to P_D contrast timings, except that T_R should be shorter, so that the longitudinal component has not returned to equilibrium. By using a T_R approximately equal to the values of T_1 of the tissues being imaged there will be a difference in the longitudinal magnetization between different tissue types immediately before T_R . Immediately after T_R , the magnetization will be tipped over into the transverse plane. Differences in the transverse magnetization will therefore be due to differences in T_1 . The signal is then measured immediately with no spin echo, or with a short T_E , to minimize T_2 decay. The timing parameters for the three contrast mechanism are summarized in Table 2.1, and examples of T_1 -weighted, T_2 -weighted, and P_D -weighted MRI slice images of a brain are shown in Figure 2.17.



Figure 2.17: T_1 -weighted, T_2 -weighted, and P_D -weighted MRI slice images of a brain, generated using (BrainWeb, 2007).

2.2.3 Image Formation

So far, this discussion has concentrated on the NMR process as it relates to generating a single signal from a volume. However, to create an image, the NMR signal must be spatially encoded to create a series of samples. To do so, three orthogonal magnetic gradient coils are used to change the strength of \mathbf{B}_0 in a spatially-dependent manner. For three gradient coils with strengths G_x , G_y , and G_z , the main field is given by

$$\mathbf{B}_0 = (B_0 + G_x x + G_y y + G_z z) \hat{z}. \quad (2.6)$$

Note that these fields do not change the direction of \mathbf{B}_0 across space, but do give it a different magnitude at different locations. From the Larmor equation (2.2), this spatially-varying magnetic field will result in a spatially-dependent distribution of Larmor frequencies. It is the creation of this distribution of Larmor frequencies, also known as frequency encoding, that enables the formation of an MRI image. In standard 2D slice-based MRI,⁹ frequency encoding is used for slice selection. It is also used, along with a related technique called phase encoding, to build up the image of the selected slice in Fourier space.

⁹3D MRI is also possible, in which an entire volume can be collected at once.

2.2.3.1 Slice selection

The first step in spatially encoding the NMR signal is slice selection. To perform slice selection, one of the magnetic gradient coils applies a gradient field to \mathbf{B}_0 . This discussion will assume a z -gradient is being applied, but either of the other two coils could be used, depending on the application. Applying such a gradient changes the strength of \mathbf{B}_0 along the z -axis (Figure 2.18). By applying the RF-excitation α -pulse over a range of frequencies v_1

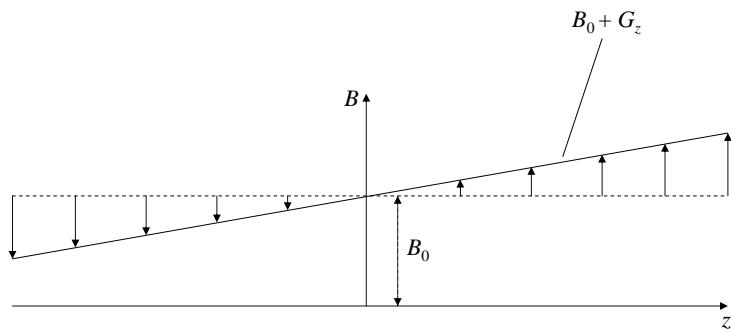


Figure 2.18: Applying a z -gradient to \mathbf{B}_0 . Figure adapted from (Prince and Links, 2006).

to v_2 while G_z is being applied, only the slice in z corresponding to this frequency range is tipped over into the transverse plane (Figure 2.19). Thus, only spins in this slice contribute to the NMR signal. G_z is removed once the alpha pulse has been applied. To obtain an image from this slice, it is necessary to further encode the NMR signal using x and y gradients.

2.2.3.2 Slice frequency encoding

Assuming the slice is thin enough that there is a negligible difference in frequency across the slice thickness due to the z -gradient, once the α -pulse has been applied and FID has begun, all the spins precess at the same frequency. The resulting NMR signal is therefore a sinusoid with this frequency, assuming the time over which the NMR signal is sampled is small enough that T_2 and T_2^* effects are negligible. The amplitude of the signal is summed over the entire slice, so there is no way to determine any spatial information (Figure 2.20). To spatially encode the signal, another frequency encoding gradient must be applied.

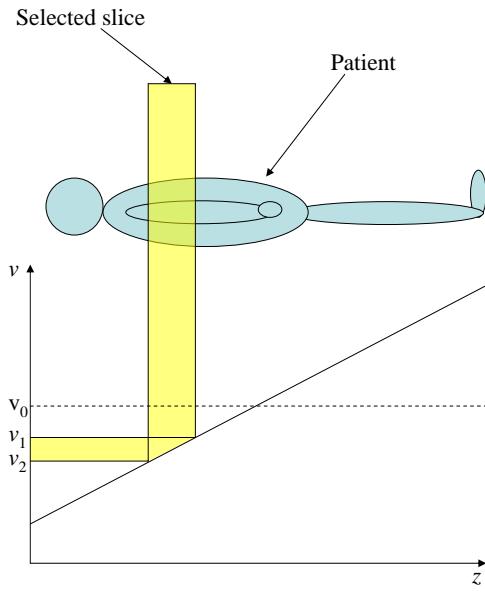


Figure 2.19: Slice selection using frequency encoding. Figure adapted from (Prince and Links, 2006).

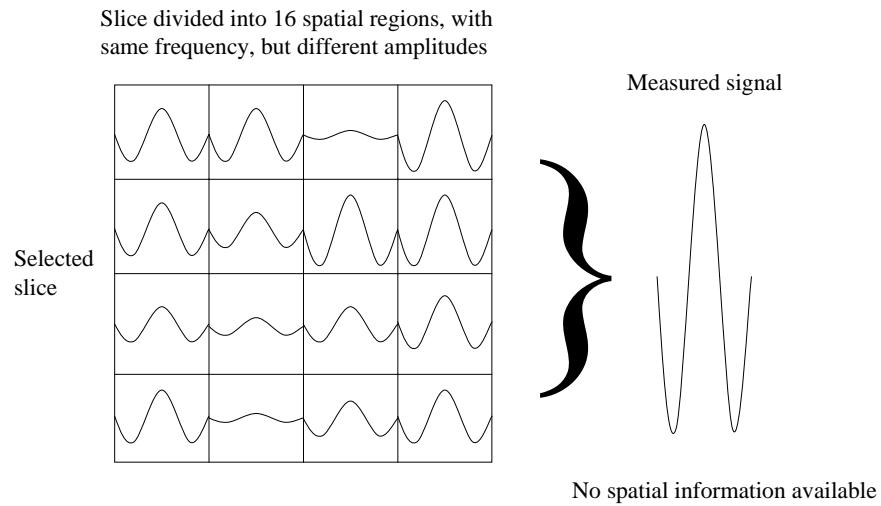


Figure 2.20: The summed amplitudes of the signals in the selected slice give no spatial information.

Applying the G_x gradient field results in different Larmor frequencies along the x -axis¹⁰.

The resulting NMR signal is therefore a summation of sinusoids at different frequencies. The amplitude at each frequency is the summation of the signal across the x -position corresponding to this frequency. Thus, the spatial signal in x is indexed by frequency. Therefore, the signal represents a 1D Fourier transform of the slice being imaged. By performing an inverse Fourier transform of the signal, the amplitudes at the various frequencies can be recovered, and the spatial distribution of the amplitude of the signal along the x -axis can be determined (Figure 2.21).

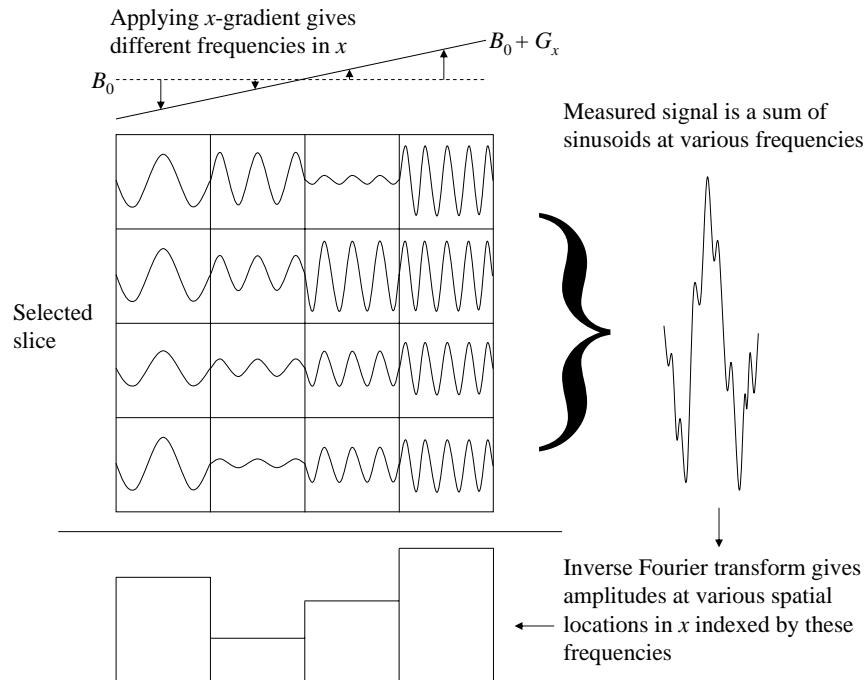


Figure 2.21: Frequency encoding enables spatial encoding of the signal along the x -axis. The spatial information is recovered via a Fourier Transform.

The signal is usually sampled 128 or 256 times before performing the inverse Fourier transform¹¹. Only spatial information in the x -direction is determined using frequency encoding as described above. To obtain a 2D image of the slice, it is necessary to further encode the signal in the y -direction.

¹⁰This is referred to as the read-out axis.

¹¹This will result in an image with a resolution of 128 or 256 in the x direction. Powers of two are used because of their convenience when using the Fast Fourier Transform (FFT) algorithm.

Simply performing frequency encoding by applying a y -gradient as well as the x -gradient will not work, as these gradients will add, and each position in x, y will not have a unique frequency. Instead, a technique called phase encoding is used.

2.2.3.3 Slice phase encoding

Phase encoding is used to spatially encode the signal in the y -direction. To do so, multiple pulse sequences consisting of an α -pulse, a phase-encoding gradient, and a frequency-encoding gradient are performed. The phase-encoding gradient is applied for a fixed amount of time via the G_y gradient field immediately after the α -pulse, and immediately before the frequency-encoding gradient. Because the strength of the G_y gradient field changes along the y -axis, applying the gradient for a fixed amount of time causes different locations in y to precess a different amount, thus imparting a spatially dependent distribution of phases along the y -axis.

For a given location in the slice, increasing the strength of the G_y gradient field a fixed amount each pulse sequence steps the phase a fixed amount each time. Locations where the G_y gradient field is stronger will receive larger steps in the phase change. Different positions in y will therefore have different rates of change of phase, which is equivalent to them having different frequencies. Phase encoding therefore performs a sampled frequency encoding in the y -direction (Figure 2.22). Using phase-encoding to perform a sampled frequency encoding in the y direction does not interfere with the frequency encoding in x , as the measured frequencies and phases during each pulse sequence are independent.

For each slice, the data sampled during the frequency encoding portion of each pulse sequence is stored in a 2D matrix called k -space representing the 2D Fourier transform of the slice image¹². Each frequency-encoding step fills a horizontal row in k -space, with the strength of the phase-encoding gradient determining the row (Figure 2.23).

Usually 128 or 256 pulse sequences are performed for each slice¹³. Once k -space has been filled, the MRI image for this slice is computed via the inverse Fourier Transform.

¹²In physics, waves are described by an amplitude and a wave number, or frequency. The wave number is traditionally denoted by the letter “k,” thus the term k -space.

¹³This will result in an image with a resolution of 128 or 256 in the y direction. Powers of two are used because of their convenience when using the Fast Fourier Transform (FFT).

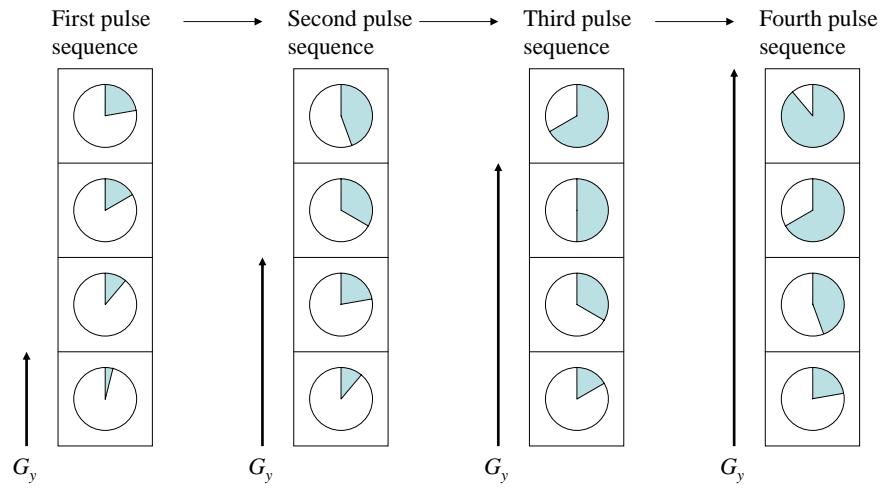


Figure 2.22: For a given location in x , increasing the phase-encoding G_y gradient field each pulse sequence results in a sampled frequency encoding in the y -direction.

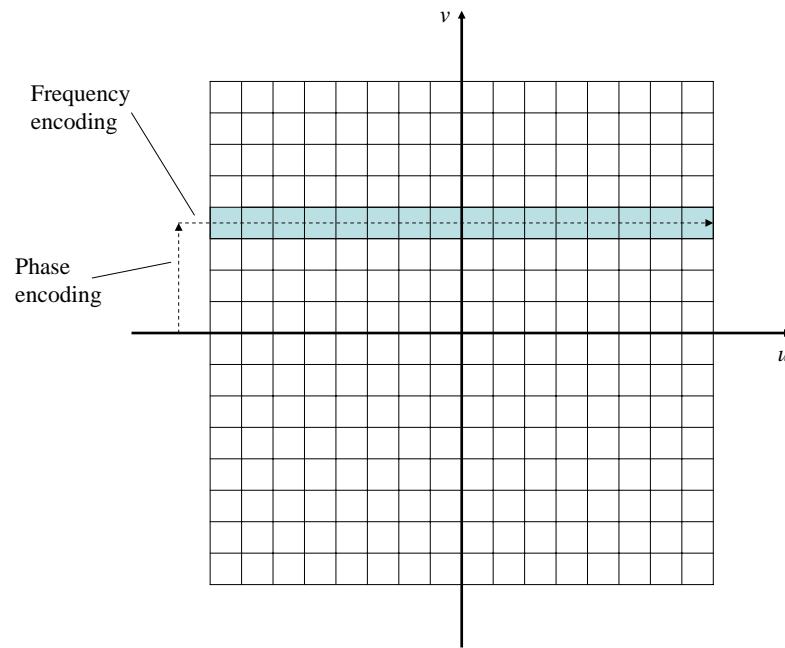


Figure 2.23: The signal resulting from each pulse sequence is stored as a horizontal row in k -space.

2.3 Computed Tomography

2.3.1 History

The first commercially viable CT machine was conceived in 1967 and publicly displayed in 1972 by Godfrey Newbold Hounsfield. Allan McLeod Cormack independently invented a similar process, and they shared the 1979 Nobel Prize in Medicine. There has been continuous technical development and clinical use since its invention. CT has many diagnostic purposes, including diagnosing lung pathology, heart pathology, abdominal pathology, and complex bone fractures. A slice of a CT image of a knee is shown in Figure 2.24. Because CT uses



Figure 2.24: A slice of CT image of a knee.

a collection of x-ray projection images to reconstruct a 3D volume, a discussion of x-ray radiography is necessary to understand it.

2.3.2 X-Ray Radiography

2.3.2.1 X-rays

X-rays are electromagnetic radiation with wavelengths less than 10 nanometers (nm) and frequencies over 30 petahertz (PHz). Visible light has wavelengths from around 380 nm to 750 nm, so x-rays have much smaller wavelengths, and much higher frequencies. X-rays are therefore more energetic than visible light, as $E_{photon} = \hbar\nu$, where \hbar is Planck's constant and ν is the photon frequency.

2.3.2.2 X-ray source

An x-ray tube is used as the source in x-ray radiography (Figure 2.25). In an x-ray tube,

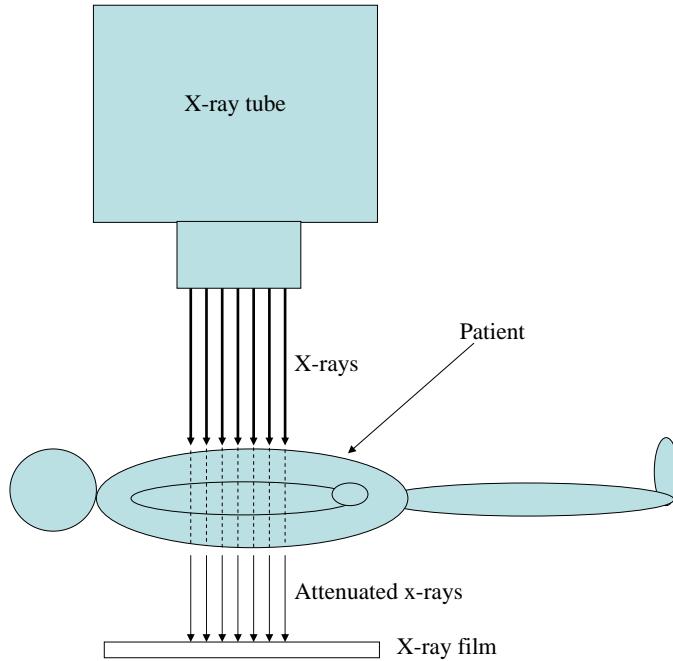


Figure 2.25: X-ray tube.

a thermoionic source¹⁴ is heated to produce electrons. These electrons are accelerated via an electric field towards an x-ray source¹⁵. The electrons interact with the x-ray source, producing x-rays. These x-rays are either absorbed by lead shielding, or emitted through a lead collimator grid, which further absorbs x-rays to provide a more parallel beam. The x-rays pass through the patient, being attenuated differently by different tissues. A photographic film¹⁶ images the x-rays after they have passed through the body.

¹⁴A thermoionic substance emits charged particles when heated. A tungsten coil is typically used.

¹⁵Typically tungsten for most applications. Molybdenum is used for lower energy applications, such as mammography.

¹⁶Digital detectors are becoming more common in certain x-ray radiography applications, and are used in CT imaging.

2.3.2.3 Attenuation

The fact that x-rays are attenuated differently by different tissue types is the reason that x-ray radiography produces useful images. For an image $g(x, y)$, the x-ray imaging equation can be approximated by

$$g(x, y) = I_0 e^{-\int \mu(x, y, z) dz}, \quad (2.7)$$

where I_0 is the effective energy of the x-ray source¹⁷ and $\mu(x, y, z)$ is the linear attenuation coefficient¹⁸ of the 3D distribution of tissue in the body. The x-ray image is therefore an array of line integrals of attenuation along the direction of the x-ray beam. Differences in these line integrals are the source of contrast in the image (Figure 2.26). More-dense tissue,

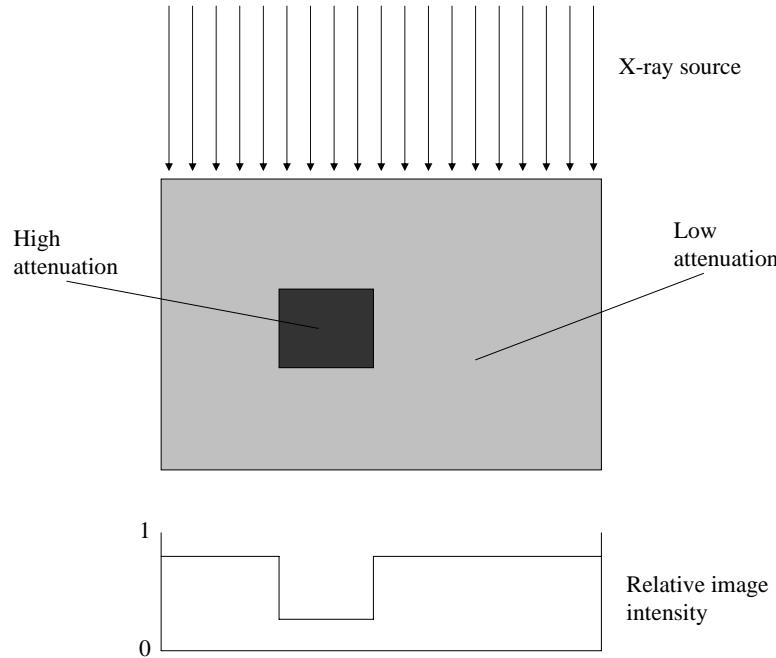


Figure 2.26: Differences in x-ray attenuation provide image contrast.

such as bone, tends to attenuate more than less-dense tissue, such as fat. An x-ray image can

¹⁷There is actually a spectrum of x-ray energies produced by the x-ray source, with the maximum energy in kilo-electronvolts (keV) equal to the peak kilo-voltage (kVp) used to accelerate the electrons in the x-ray tube. Lower energy x-rays are typically filtered out to achieve a smaller range of higher-energy x-rays, with the effective energy given as an average of the resulting energies.

¹⁸The linear attenuation coefficient of a material is the fraction of photons absorbed per unit length for a photon beam passing through the material.

therefore be thought of as a 2D projection of the 3D radiodensity distribution in the patient. Because the image consists of line integrals, all depth information is lost. Interpreting x-ray images correctly takes special training and knowledge of the anatomy being examined. Even with proper training, important anatomical features can be masked. To obtain a full 3D image of the radiodensity of the patient, CT is used.

2.3.3 Tomography

The word “tomography” comes from the Greek words “*tomos*,” meaning slice, and “*graph*,” meaning picture. In CT, the patient lays on a flat bed which is moved in steps through a circular scanner that generates a series of one-dimensional (1D) x-ray projections through the slice of the patient at that step (Figure 2.27). The 3D volume is generated by reconstructing

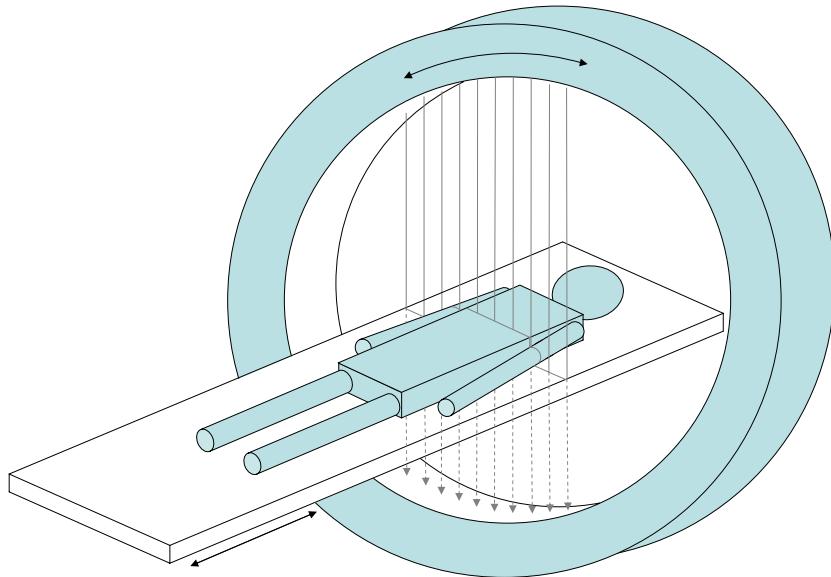


Figure 2.27: Diagram of a CT scanner. The patient is moved horizontally through the circular scanner.

each 2D slice of the volume from the series of 1D x-ray projections collected at each slice. To understand how a series of 1D projections can be reconstructed to create a 2D image, it is necessary to discuss a mathematical formulation known as the Radon transform.

2.3.3.1 The Radon transform

For a single slice $f(x, y)$, the CT scanner gives a series of 1D projections $p(t)$ from various angles. These projections can be parameterized together over the angle θ to give a 2D function $p(t, \theta)$. This situation is depicted graphically in Figure 2.28. This image illustrates that

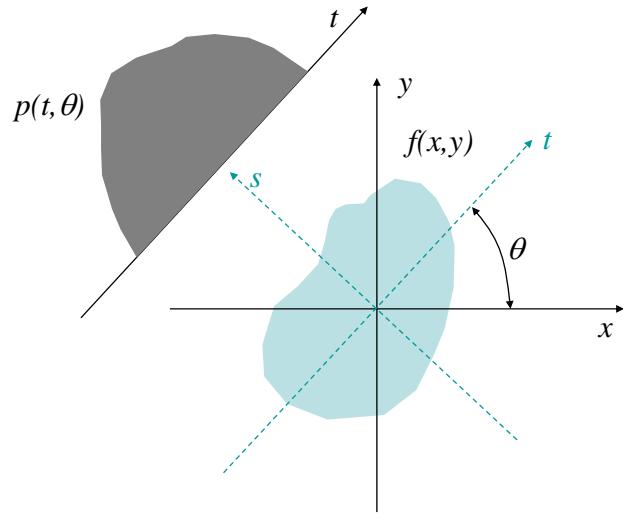


Figure 2.28: The Radon transform. Figure recreated from (Lalush, 2006).

$$p(t, \theta) = \int_{-\infty}^{\infty} f(s, t) ds, \quad (2.8)$$

where s and t are a rotating frame determined by θ . Because s, t is a rotating frame, the following relationships are true:

$$t = x \cos \theta + y \sin \theta \quad (2.9)$$

$$s = -x \sin \theta + y \cos \theta. \quad (2.10)$$

Combining (2.8) with (2.9) and (2.10) gives the Radon transform,

$$p(t, \theta) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) \delta(x \cos \theta + y \sin \theta - t) dx dy, \quad (2.11)$$

where δ is the Dirac delta function.

The problem that must be solved is to find $f(x, y)$ given $p(t, \theta)$. Formulating the acquired CT data as a Radon projection¹⁹ enables this problem to be solved, as shown by the Fourier-slice theorem²⁰.

2.3.3.2 The Fourier-slice theorem

The Fourier-slice theorem relates the Radon transform of $f(x, y)$ to the Fourier transform of $f(x, y)$. This relationship demonstrates that the Radon transform is invertible, and gives a method for determining $f(x, y)$ from $p(t, \theta)$.

The Fourier transform $\mathcal{F}(u)$ of a function $f(x)$ can be written

$$\mathcal{F}(u) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x) e^{-i(ux)} dx. \quad (2.12)$$

From 2.8 and 2.12, the Fourier transform $\mathcal{P}(\omega_t, \theta)$ of $p(t, \theta)$ with respect to a single frequency variable ω_t can be written

$$\mathcal{P}(\omega_t, \theta) = \int_{-\infty}^{\infty} \left[\int_{-\infty}^{\infty} f(s, t) ds \right] e^{-i(\omega_t t)} dt. \quad (2.13)$$

Reordering the integrals gives

$$\mathcal{P}(\omega_t, \theta) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(s, t) e^{-i(\omega_t t)} ds dt. \quad (2.14)$$

Because s and t are a rotation of x and y , a change of variables can be used to replace $f(s, t)$

¹⁹The acquired x-ray data is not in fact a true Radon projection. The Radon projection requires a line integral of attenuation $\int \mu(x, y, z) dz$. From (2.7), it is clear that the line integral of attenuation can be obtained from the x-ray projection $g(x, y)$ as $\int \mu(x, y, z) dz = -\ln \frac{g(x, y)}{I_0}$.

²⁰The Fourier-slice theorem is also known as the projection-slice theorem.

with $f(x, y)$. Combined with 2.9, $\mathcal{P}(\omega_t, \theta)$ can be written

$$\mathcal{P}(\omega_t, \theta) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-i\omega_t(x \cos \theta + y \sin \theta)} dx dy. \quad (2.15)$$

The 2D Fourier transform $\mathcal{F}(u, v)$ of a function $f(x, y)$ can be written

$$\mathcal{F}(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-i(ux + vy)} dx dy. \quad (2.16)$$

It is therefore apparent that 2.15 is equivalent to the 2D Fourier transform of $f(x, y)$ if $u = \omega_t \cos \theta$ and $v = \omega_t \sin \theta$.

Therefore,

$$\mathcal{P}(\omega_t, \theta) = \mathcal{F}(\omega_t \cos \theta, \omega_t \sin \theta), \quad (2.17)$$

which is the Fourier-slice theorem. This relationship states that the 1D Fourier transform of a projection of an object is a slice of the 2D Fourier transform of the object. This slice is a line passing through the origin of the 2D Fourier transform of the object at the angle corresponding to that projection (Figure 2.29).

The slice image can therefore be reconstructed by taking the 1D Fourier transform of each projection, inserting them into the correct slice of the 2D Fourier transform matrix, and taking the inverse 2D Fourier transform of this matrix. Although theoretically this method works, it is seldom used in practice. The Fourier method requires interpolating polar data onto a Cartesian grid. Locations near the origin will be sampled more than locations far from the origin. This will lead to sampling problems at high spatial frequencies.

2.3.3.3 Simple backprojection

Another possible method for reconstruction is called simple backprojection. In simple backprojection, each 1D projection is projected back across the image and accumulated. The resulting image is then normalized by dividing by the number of pixels in the image. A simple example is shown in Figure 2.30.

From Figure 2.30, it is clear that simple backprojection reconstructs an image that is

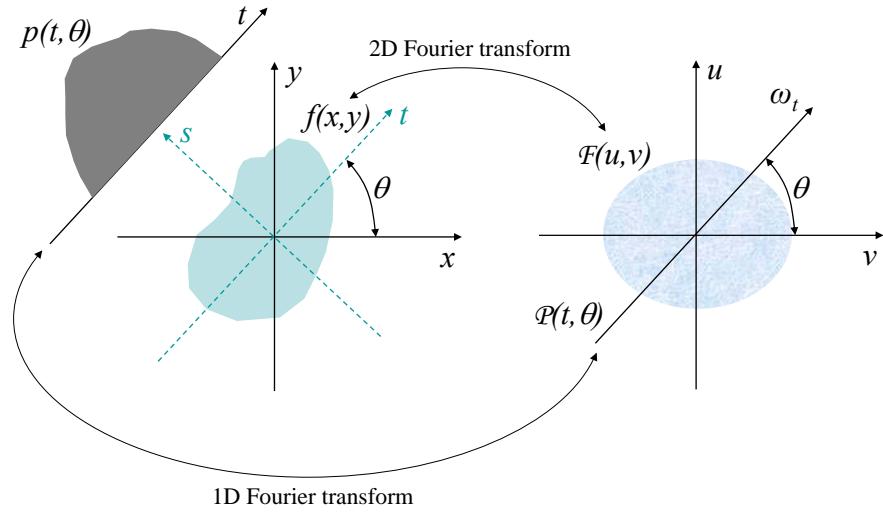


Figure 2.29: Graphical representation of the Fourier-slice theorem. The 1D Fourier transform of a projection of an object is a slice of the 2D Fourier transform of that object. Figure adapted from (Prince and Links, 2006).

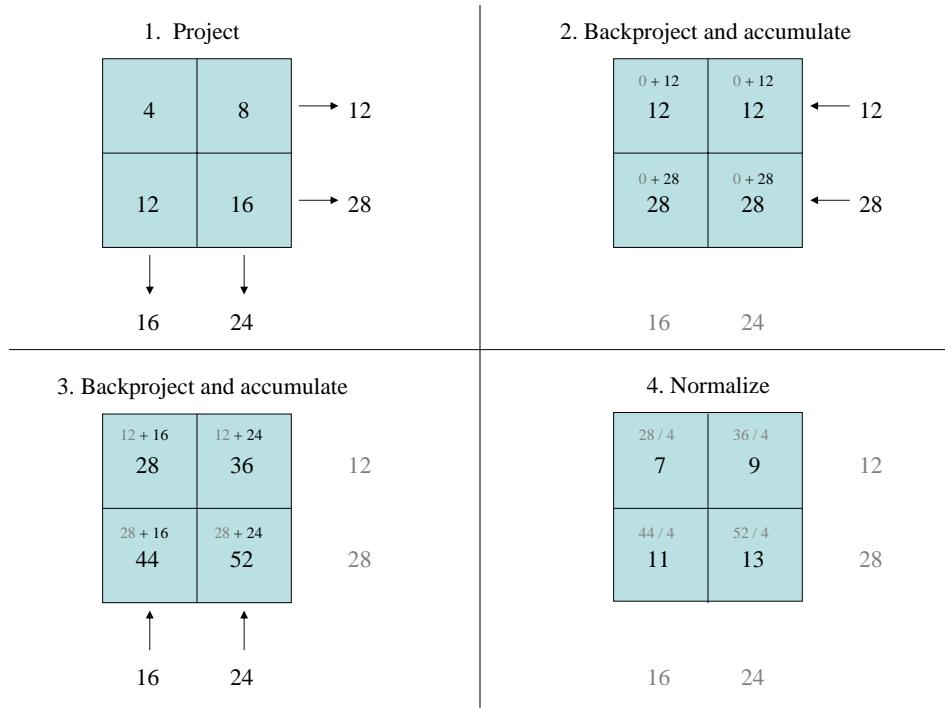


Figure 2.30: Simple backprojection example.

close to the original data, but not exactly correct. The lowest value is increased by 3, the second-lowest value is increased by 1, the second-highest value is decreased by 1, and the highest value is decreased by 3. This illustrates that the result of simple backprojection is a smoothed version of the original data.

The exact relationship between the original data and the smoothed backprojection reconstruction can be shown mathematically. The estimate of the true image given by simple backprojection $\hat{f}_{sbp}(x, y)$ can be written

$$\hat{f}_{sbp}(x, y) = \int_0^\pi p(t = x \cos \theta + y \sin \theta, \theta) d\theta. \quad (2.18)$$

This equation describes the accumulation of the backprojected data over the set of angles of the projections. It does not include the normalization step, which simply consists of division by a constant. Replacing $p(t, \theta)$ with its 1D Fourier transform gives

$$\hat{f}_{sbp}(x, y) = \int_0^\pi \int_{-\infty}^{\infty} \mathcal{P}(\omega_t, \theta) e^{i\omega_t t} d\omega_t d\theta. \quad (2.19)$$

Rewriting in terms of x and y gives

$$\hat{f}_{sbp}(x, y) = \int_0^\pi \int_{-\infty}^{\infty} \mathcal{P}(\omega_t, \theta) e^{i\omega_t(x \cos \theta + y \sin \theta)} d\omega_t d\theta. \quad (2.20)$$

The Fourier transform in polar coordinates of any function $f(x, y)$ can be written

$$f(x, y) = \int_0^\pi \int_{-\infty}^{\infty} \mathcal{F}(\omega_r, \theta) e^{i\omega_r(x \cos \theta + y \sin \theta)} |\omega_r| d\omega_r d\theta. \quad (2.21)$$

Comparing 2.20 and 2.21 shows that the simple backprojection estimate is equivalent to applying a $\frac{1}{|\omega_t|}$ filter to the Fourier reconstruction method. This $\frac{1}{|\omega_t|}$ filter suppresses high spatial frequencies, and is the source of the smoothing displayed in Figure 2.30.

2.3.3.4 Filtered backprojection

To obtain the correct reconstruction using backprojection, a technique called filtered backprojection is used. In filtered backprojection, an $|\omega_t|$ filter, known as a ramp filter because of its shape, is applied to cancel out the $\frac{1}{|\omega_t|}$ filter (Figure 2.31).

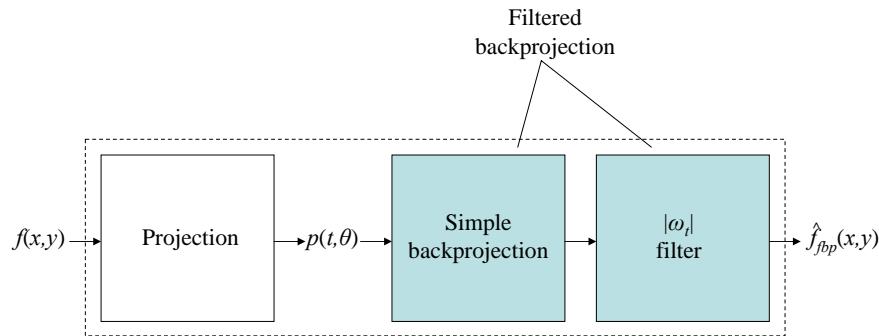


Figure 2.31: Filtered backprojection schematic. Figure adapted from (Lalush, 2006).

There are various ways to apply the filter in filtered backprojection. One way is to perform simple backprojection, and then apply a 2D ramp filter in the Fourier domain. Another way is to apply a 1D ramp filter to each projection in the Fourier domain, and then perform simple backprojection. A third way is to convolve each projection with the inverse Fourier transform of the ramp filter in the spatial domain, and then perform simple backprojection. This method is called convolution backprojection, and is often used in practice for faster computation.

One important practical consideration is that a simple ramp filter will enhance noise in real images. For this reason, the ramp filter is usually apodized, or rolled off to zero, at an appropriate frequency for a given application.

2.4 Medical Images

2.4.1 Anatomy

Because MRI and CT are used to image human anatomy, it is useful to discuss some of the terminology used in anatomy that relate to medical images.

2.4.1.1 Directions

Portions of the body nearer the head are referred to as superior; those nearer the feet as inferior. Portions of the body nearer the front are referred to as anterior; those nearer the rear as posterior. Portions of the body near the midline are referred to as medial, those nearer the sides as lateral (Figure 2.32).

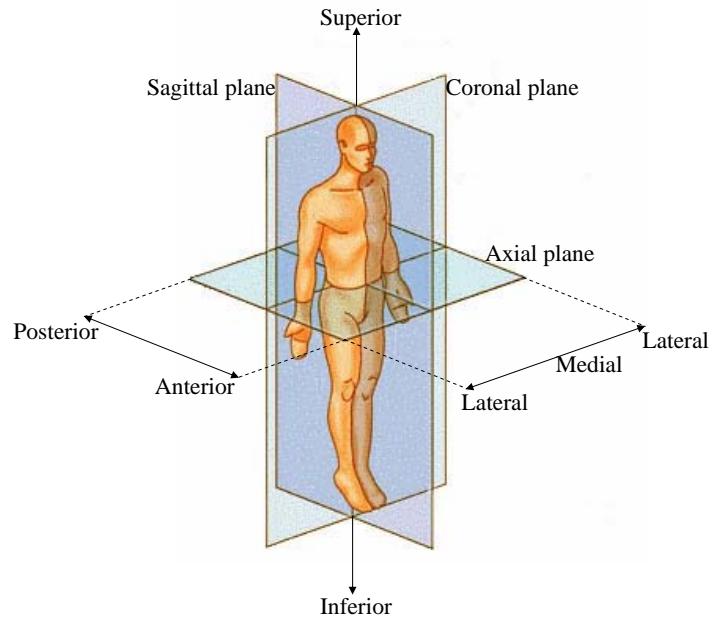


Figure 2.32: The three anatomical reference planes (SEER, 2007).

2.4.1.2 Reference planes

There are three basic reference planes used in anatomy: axial²¹, coronal²², and sagittal (Figure 2.32). Assuming a patient standing on their feet, an axial plane is perpendicular to the main axis of the body and parallel to the ground, separating superior from inferior. A coronal plane is perpendicular to the ground, separating anterior from posterior. A sagittal plane is perpendicular to the ground, separating right from left.

2.4.2 Medical Image Display

Physicians typically view CT and MRI images by looking at individual slices, usually with a control for moving back and forth through the slices in order. When viewing axial images, the convention is to position the image as if the viewer were standing at the feet of a reclining subject, looking towards the head.

The slice image is usually displayed in greyscale, although other colormaps can also be used. Typically, the physician can control the window/level of the greyscale display. The window/level²³ settings determine how raw image values are mapped to display intensities. The level determines the midpoint of the range of image intensities that is mapped to the midpoint of the display intensity, whereas the window determines the size of the range of values mapped to the range of values of the display. Therefore $level - \frac{window}{2}$ maps to the lowest display intensity, and $level + \frac{window}{2}$ maps to the highest display intensity, with a linear mapping of intermediate values.

For example, a CT image might have a dynamic range of 2^{10} bits, or 1024 values. A typical computer display can only display 2^8 , or 256 values. The window/level settings determine which subset of the image intensities map to which output intensities (Figure 2.33). Figure 2.34 gives an example of different window/level settings for a sagittal slice of a knee CT. Window/level controls are also useful with images that have the same dynamic range as the display, such as most MRI images, for enhancing features at different intensity ranges in the

²¹An axial plane is also referred to as a transverse or horizontal plane.

²²A coronal plane is also referred to as a frontal plane.

²³The terms window and level are synonymous with contrast and brightness.

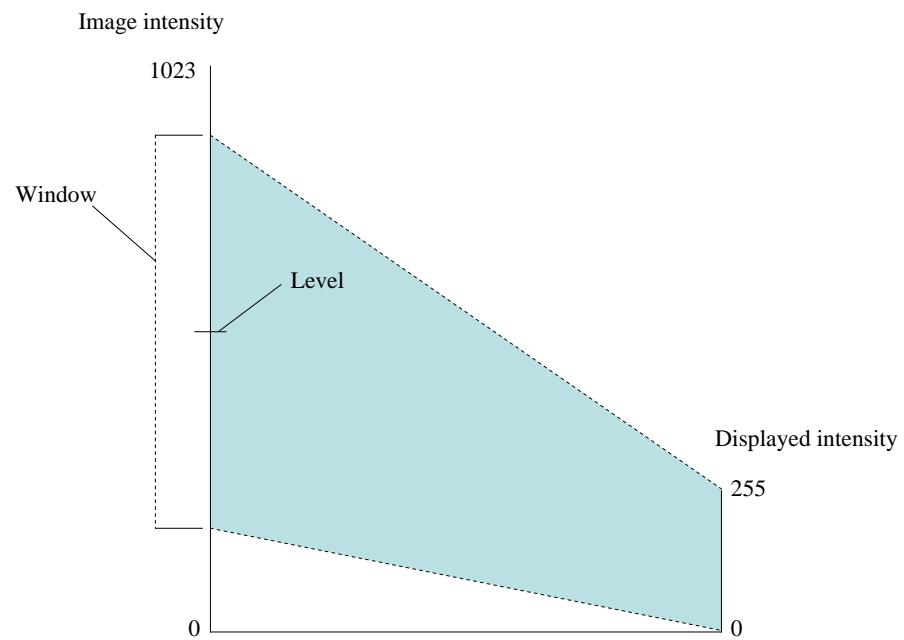


Figure 2.33: Window/level controls map image intensities to display intensities.

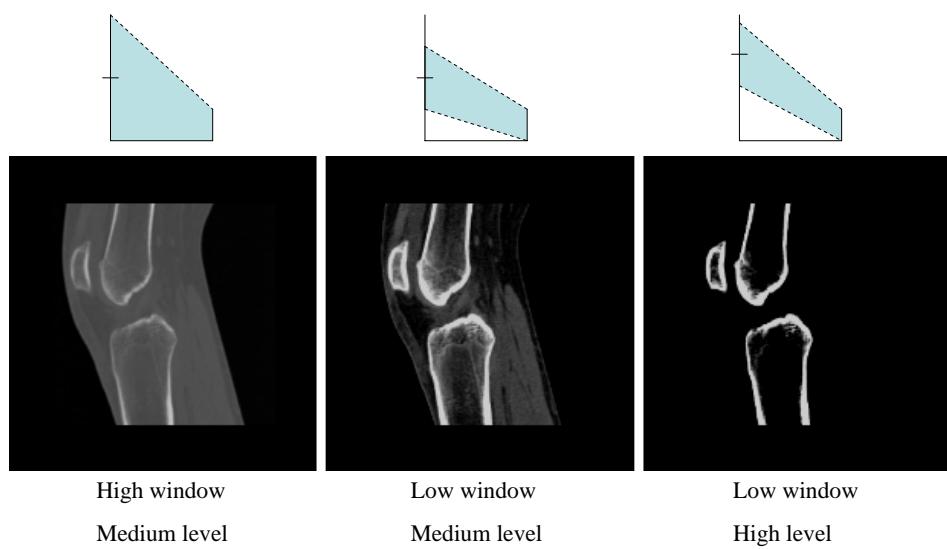


Figure 2.34: Different window/level settings applied to a sagittal slice of a knee CT.

image.

In addition to viewing the stored planar²⁴ slice images, a technique called multi-planar reconstruction (MPR) can be used to display slices at any position and orientation in the volume. Typically, MPR is used to display the canonical axial, coronal, and sagittal slices of the volume (Figure 2.35). Pan and zoom controls are also usually provided for each slice

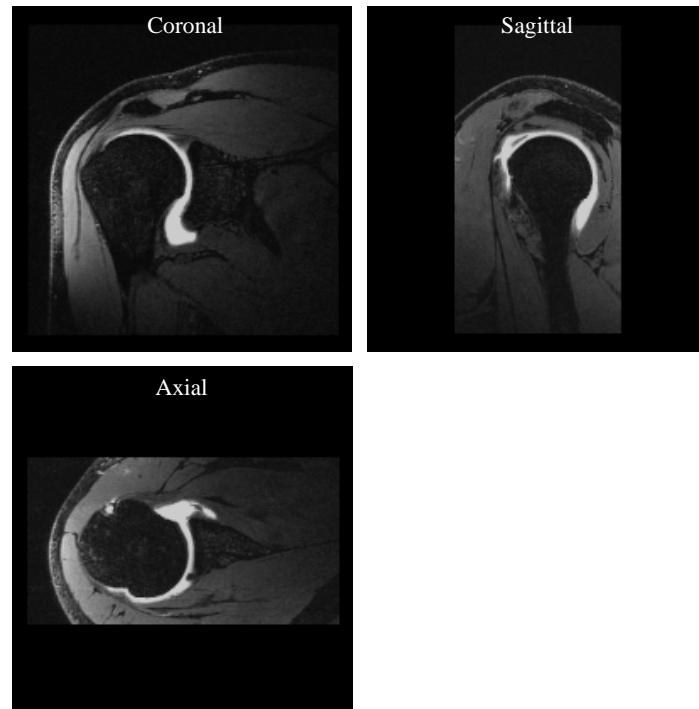


Figure 2.35: MPR used to display axial, coronal, and sagittal slices of a shoulder MRI.

image to enable closer views of features of interest.

2.4.3 Comparison of Magnetic Resonance and Computed Tomography Images

MRI and CT are both used to image anatomy, and both also collect volumetric data. However, each collects different anatomical information, and are therefore useful for different applications. The characteristics of the data collected also differ. In addition to the usefulness of the data generated, other factors, such as cost and invasiveness, can contribute to the physician's decision to use MRI or CT for a given application.

²⁴The term planar is used because the image correspond to one of the standard reference planes.

2.4.3.1 Anatomical information

The three sources of contrast in MRI are the proton density, the T_1 relaxation constant, and the T_2 relaxation constant of different tissues. In general, MRI gives good contrast resolution between soft tissue types and is important for brain imaging, as it can differentiate between white matter, grey matter, and cerebrospinal fluid (CSF). MRI is also used for imaging muscle and musculoskeletal connective tissue, among many other applications. MRI is a very flexible imaging modality because various protocols are available, consisting of different pulse sequences and timings tuned for specific applications. Moreover, MRI can provide multi-contrast images, comparable to multi-spectral color images.

The source of contrast in CT is the linear attenuation coefficient of different tissues, which is directly related to the radiodensity. Because most soft tissues have similar linear attenuation coefficients, CT gives poor soft tissue contrast. It is commonly used for diagnosing complex bone fractures and looking for high-density masses, such as kidney stones or certain kinds of tumors. Contrast between healthy tissue and pathology, e.g. tumor, is often achieved by injecting a specific contrast agent into the patient.

2.4.3.2 Image characteristics

Because any combination of the three gradient coils can be used for slice selection, MRI images can be taken along any axis. Typically, either axial, coronal, or sagittal slices are taken, depending on the application. Usually, the in-plane spatial resolution will be the same in x and y , and this resolution will often be better than the between-plane resolution. Isotropic voxels that are the same resolution in all three directions are possible with certain scan protocols. The spatial resolution depends on the specifics of the protocol being used, but in general, a longer scan time gives a better signal-to-noise ratio (SNR), which enables smaller voxels and better spatial resolution.

In CT, the data is always generated as axial slices²⁵, as the patient lays horizontally while being moved through the scanner, and the projections are collected perpendicular to the patient axis. As with MRI, the in-plane spatial resolution will typically be the same in

²⁵Thus CT is sometimes referred to as computed axial tomography (CAT).

x and y , and this resolution will often be better than the between-plane resolution, although isotropic voxels are possible.

Most MRI images are $256 \times 256 \times X$ arrays of 8-bit data, where X is the number of slices taken. The intensities generated in MRI data are generally not consistent between scans and depend on the specifics of the pulse sequences and pulse timings used, so absolute comparison of the intensities between MRI images is not possible.

Most CT images are $512 \times 512 \times X$ arrays of data. CT data is usually 12 or 16-bit. CT data is normalized via the CT number, measured in Hounsfield Units (HU). The CT number h_{voxel} of a given voxel with measured value μ_{voxel} is computed as

$$h = 1000 \left(\frac{\mu_{voxel} - \mu_{water}}{\mu_{water}} \right). \quad (2.22)$$

Each CT machine must be calibrated frequently using phantoms with known material properties, such as air and water, to achieve accurate conversion. Table 2.2 gives CT numbers for some common materials.

Material	Minimum Value (HU)	Maximum Value (HU)
Bone	400	1000
Soft tissue	40	80
Water	0	0
Fat	-100	-60
Air	-1000	-1000

Table 2.2: CT numbers for common materials

Both MRI and CT data are capable of producing sub-millimeter resolutions. There is a tradeoff between scan time and resolution in most cases, which is further complicated in CT by the fact that increased scan time means increased x-ray exposure for the patient.

2.4.3.3 Image artifacts

Various artifacts can occur in both MRI and CT imaging. In MRI, the most common is geometric warping, which arises when the strength of the magnetic field is not uniform across the entire field of view. This artifact is worsened when using fast switching gradients to decrease imaging time. The effect is that of a low frequency change in intensity across the image.

It can usually be reduced using bias field correction software. Ghosting is another common artifact, caused by patient motion. It is conceptually similar to motion blur in photography, and has a similar effect. Motion blur is usually not a problem when imaging the extremities, but motion due to breathing or heart beat can be problematic for some applications. Another common artifact is wraparound, caused by aliasing in the phase-encoded direction due to an object lying outside of the field of view. Wraparound can usually be avoided with correct MRI scan settings.

In CT, streaking artifacts can be caused by beam hardening. Beam hardening occurs when low energies in the x-ray beam are absorbed more than high energies. This causes a change in the x-ray spectrum as it passes through the subject. Because the linear attenuation coefficient is dependent on the x-ray energy, this change in the x-ray spectrum leads to inconsistencies between projections at different angles. Streak artifacts result from these inconsistencies, occurring near high-density material, such as bone. A related artifact is photon starvation, in which the x-ray beam is mostly absorbed by the subject before reaching the detector. Fewer photons reaching the detector leads to a decreased signal-to-noise ratio (SNR). This noise leads to inconsistencies between projections, resulting in streaking artifacts. Again, photon starvation occurs near high-density material. Both beam hardening and photon starvation cause a great deal of streaking artifacts around metal, a common material for medical implants, because metal absorbs most of the x-ray energy. Both beam hardening and photon starvation can be alleviated by performing iterative reconstruction techniques. These techniques are rarely used in practice, however, because they take longer to compute than filtered backprojection. Motion artifacts also occur in CT, but can be corrected somewhat by estimating rigid-body transforms of the patient, and using this estimation to translate the projections accordingly before reconstruction.

2.4.3.4 Scan time and cost

Selecting an imaging modality to use for a certain application does not just depend on how well it works. Economic factors are also a major issue. Although MRI scans are more flexible than CT scans for many applications, CT is typically much faster and less expensive

than MRI. State-of-the-art spiral CT scanners, which move the patient continuously through the scanner while scanning instead of stepping through slice by slice, can perform a full-body scan in less than 30 seconds, whereas typical MRI scans take on the order of 30 minutes to complete.

CHAPTER 3

BACKGROUND AND RELATED WORK

3.1 Volume Graphics

Volume graphics is the process of producing a 2D image from a 3D volume consisting of volume elements called voxels. A voxel is the 3D analog of a pixel in a 2D image. Each voxel has a spatial extent forming a rectangular prism, and a value that is the physical quantity being measured (Figure 3.1 a). This data value can either be thought of as a constant value

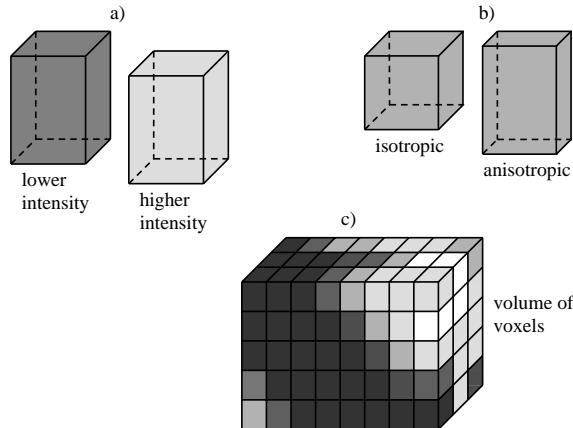


Figure 3.1: Voxels are the elements that comprise a volume.

throughout the voxel, as an average over the voxel, or as the value at the center of the voxel. The third conceptualization is useful when performing interpolation on the volume. If the voxel has the same spatial extent in all three directions it forms a cube, and is referred to as an isotropic voxel. Anisotropic voxels, in which the spatial extent along at least one of the axes differs, are also possible (Figure 3.1 b). Anisotropic voxels are common in medical imaging because the in-plane resolution of a slice (x, y) often differs from the between-plane resolution (z). The collection of voxels is organized into a regular 3D grid that is a discretized

representation of the continuous 3D signal being imaged (Figure 3.1 c). There are two major types of techniques in volume graphics: surface rendering¹ and volume rendering².

3.1.1 Surface Rendering

Surface rendering is a two-step process. The first step is extracting surfaces from the volumetric data, and the second step is rendering these surfaces. The surfaces are usually isosurfaces of a constant value, created using an algorithm such as marching cubes (Lorensen and Cline, 1987). Isosurfaces can be generated either from the original data, or from segmented data. The extracted surfaces are typically rendered as polygonal meshes using standard computer graphics techniques accelerated by graphics hardware (Figure 3.2)³.



Figure 3.2: Isosurface rendering of a knee from CT data.

3.1.2 Volume Rendering

Volume rendering techniques do not rely on surface extraction, making it possible for every voxel to contribute to the final image. The method of volume rendering used in this dissertation is ray casting (Levoy, 1988; Sabella, 1988; Levoy, 1990). With this technique, a ray is cast

¹Surface rendering is also referred to as indirect volume rendering.

²Volume rendering is also referred to as direct volume rendering.

³This image, along with others in this section, is rendered using VolView, a volume visualization program developed by Kitware, and available at www.volview.org.

into the volume for each pixel in the final image. Color and opacity is then accumulated along each ray in four steps: sampling, transfer function evaluation, shading, and compositing.

3.1.2.1 Sampling

When ray casting, a virtual camera generates rays for each pixel in the final 2D image. These rays traverse the volume, sampling the value at equidistant points along each ray (Figure 3.3). Samples are generated either by using a nearest-neighbor approach, which gives the

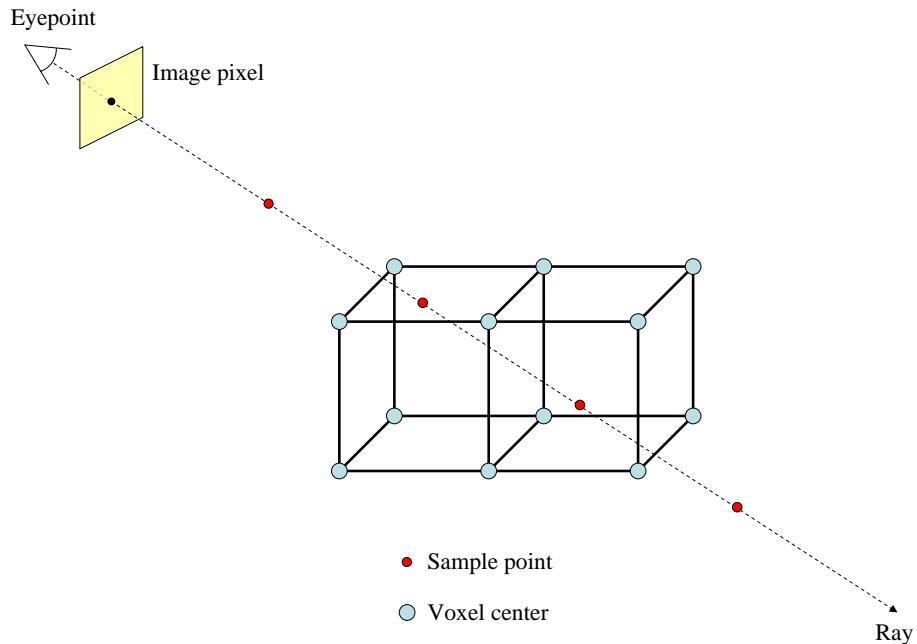
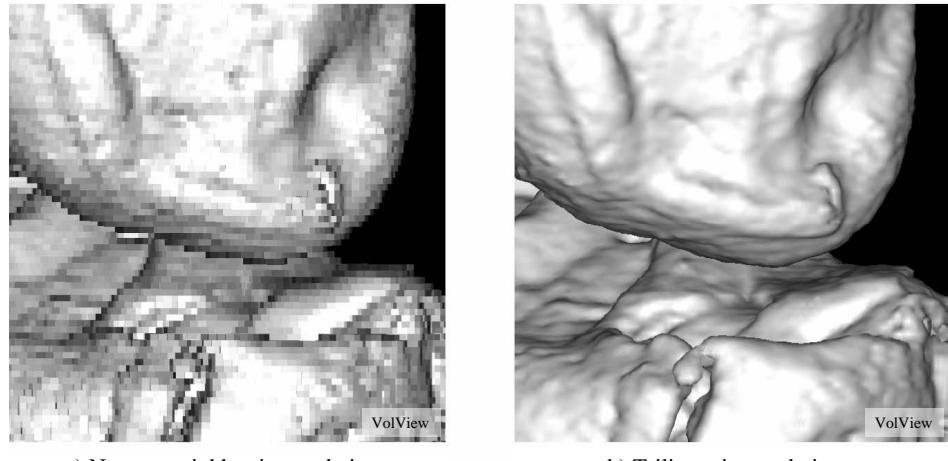


Figure 3.3: The volume is sampled at equidistant points along each ray.

value of the voxel center nearest the sample point, or by interpolating between multiple voxel values. Nearest-neighbor sampling is fast, but results in visual artifacts that make the underlying voxel grid visible (Figure 3.4 a). Trilinear interpolation, which computes a weighted average of the eight voxel centers nearest the sample point, is the most common interpolation approach, slightly decreasing rendering speed while greatly improving rendering quality (Figure 3.4 b). Given voxel centers with values $[V_{000}, V_{100}, V_{010}, V_{110}, V_{001}, V_{101}, V_{011}, V_{111}]$ and a sample point given as fractional voxel coordinates (x, y, z) (Figure 3.5), the interpolated



a) Nearest-neighbor interpolation

b) Trilinear interpolation

Figure 3.4: Volume rendering of a knee from CT data. Nearest neighbor interpolation results in artifacts making the underlying voxel grid apparent. Trilinear interpolation is often used to improve quality.

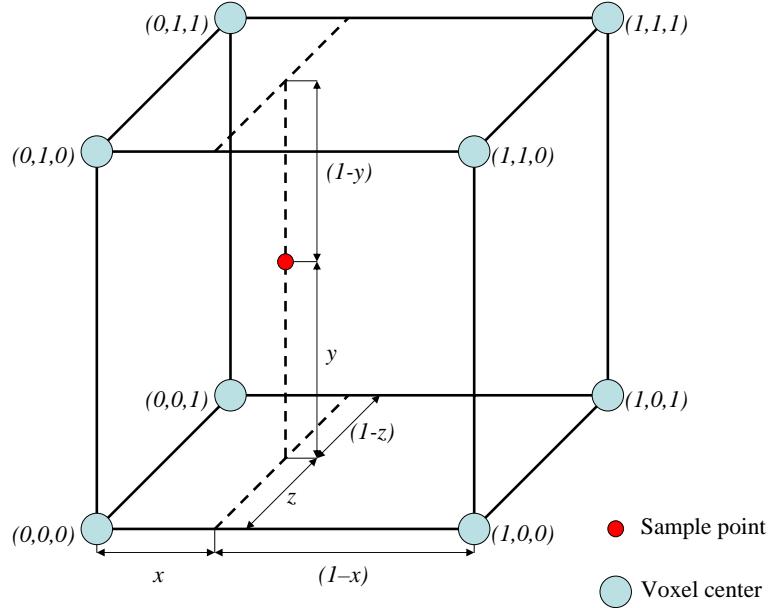


Figure 3.5: Trilinear interpolation.

value V_{xyz} is given as

$$\begin{aligned}
 V_{xyz} = & V_{000}((1-x)(1-y)(1-z)) + \\
 & V_{001}((1-x)(1-y)(-z)) + \\
 & V_{010}((1-x)(-y)(1-z)) + \\
 & V_{011}((1-x)(-y)(-z)) + \\
 & V_{100}((-x)(1-y)(1-z)) + \\
 & V_{101}((-x)(1-y)(-z)) + \\
 & V_{110}((-x)(-y)(1-z)) + \\
 & V_{111}((-x)(-y)(-z)). \tag{3.1}
 \end{aligned}$$

The sampling distance along the ray is a tradeoff between image quality and rendering speed. Significant artifacts can be generated if an insufficient sampling distance is used (Figure 3.6).

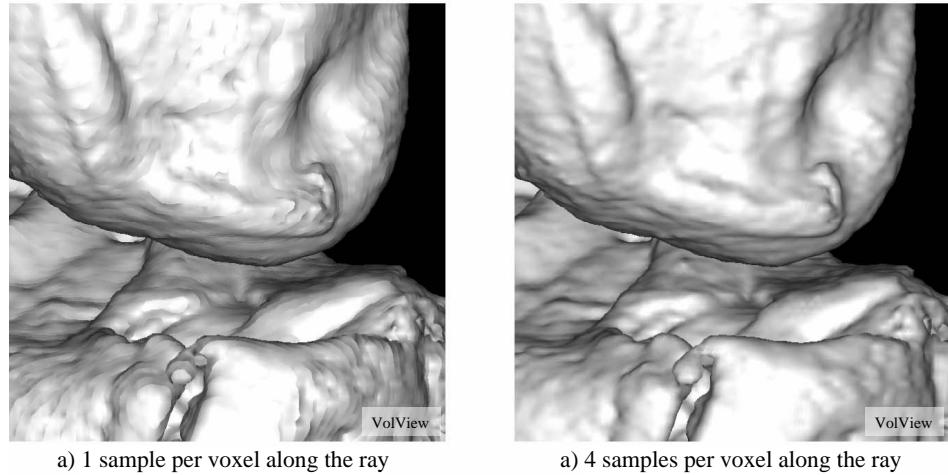


Figure 3.6: Volume rendering of a knee from CT data. Sampling artifacts occur if insufficient samples are taken along the ray.

3.1.2.2 Transfer function evaluation

Once the value is sampled, it is passed through a color transfer function to determine the color of the sample (Figure 3.7). A typical color transfer function control enables the user

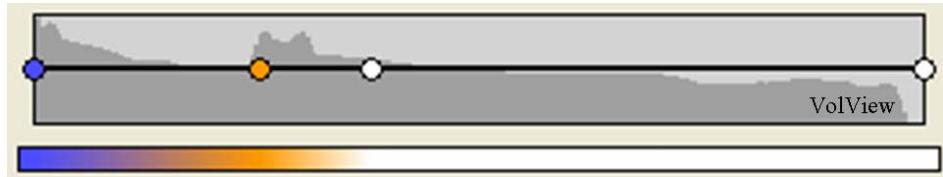


Figure 3.7: Example color transfer function showing control points, interpolated colors, and a histogram of the data values.

to select the color at control points, with the color being linearly interpolated between the control points.

The value of the sample is also passed through a scalar opacity transfer function to determine the opacity of the sample (Figure 3.8). This opacity is used to determine how much



Figure 3.8: Example scalar opacity transfer function showing control points, piece-wise linear function, and a histogram of the data values.

this sample occludes samples farther along the ray. The opacity transfer function is critical in determining what structures of the volume are visible. Figure 3.9 shows the result of applying different color and opacity transfer functions to a knee CT data set.

In addition to a scalar opacity transfer function, a gradient opacity transfer function is often used to highlight surfaces in the data. For a volumetric scalar field $f(x, y, z)$, the gradient $\nabla f(x, y, z)$ is a derived vector field, an approximation of which is typically precomputed for

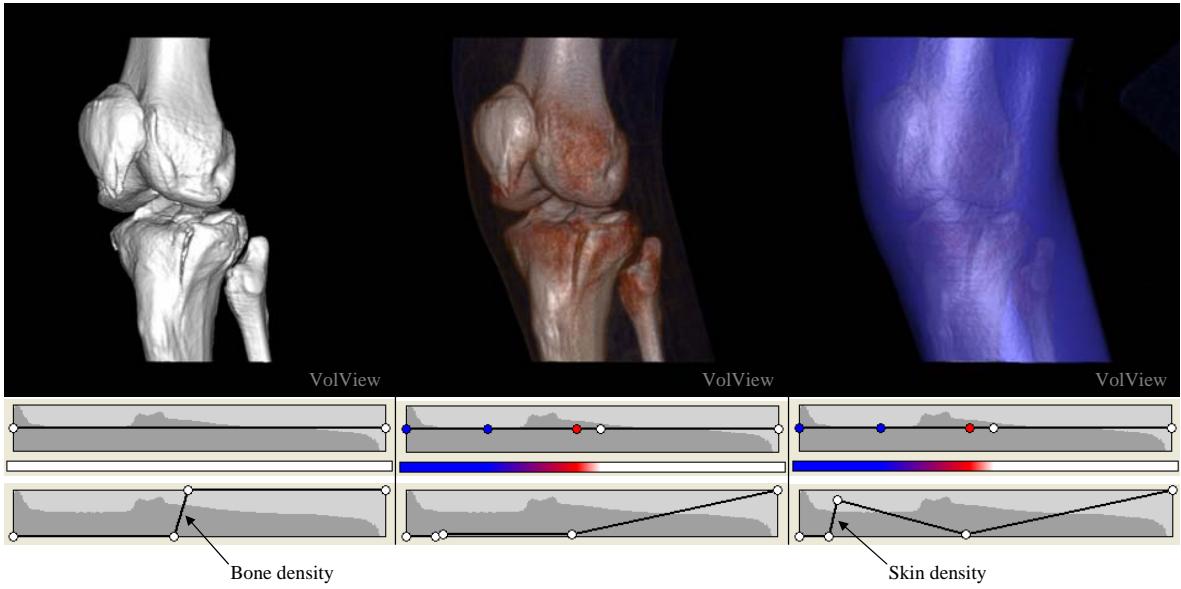


Figure 3.9: Volume renderings of a knee from CT data. The color and scalar opacity transfer functions used are shown beneath each image. The transfer functions affect what materials in the volume are rendered, and how they are rendered.

each voxel in the volume using finite differences as

$$\begin{aligned} \nabla f(x, y, z) \approx & \\ & \left[\frac{1}{2} (f(x+1, y, z) - f(x-1, y, z)), \right. \\ & \left. \frac{1}{2} (f(x, y+1, z) - f(x, y-1, z)), \right. \\ & \left. \frac{1}{2} (f(x, y, z+1) - f(x, y, z-1)) \right]. \end{aligned} \quad (3.2)$$

The gradient opacity transfer function is typically set to give higher opacity where there is a strong gradient in the volume. A strong gradient is usually a good indicator of where surfaces exist. The gradient opacity modulates the scalar opacity to determine the overall opacity of the sample. Figure 3.10 shows the result of applying different gradient transfer functions to a knee CT data set with the same color and scalar opacity transfer functions.

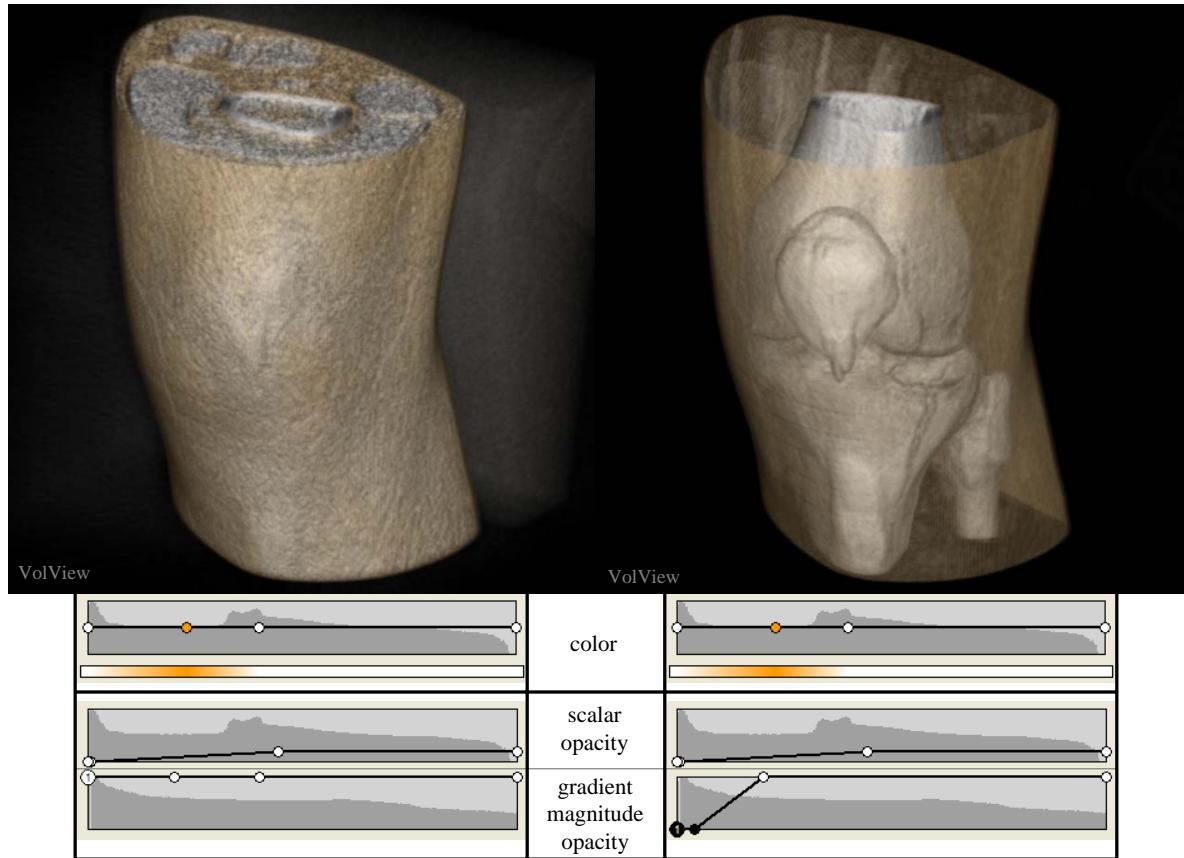


Figure 3.10: Volume renderings of a knee from CT data. The same color and scalar opacity transfer functions are used for both images, but different gradient opacity transfer functions are used. The image on the right shows how a gradient opacity transfer function can be used to highlight surfaces in the data.

3.1.2.3 Shading

The color generated from the color transfer function is optionally shaded based on any virtual lights in the scene. Typically, the Phong illumination model is used (Phong, 1973). The normal $N(x, y, z)$ used in calculating the lighting at each sample point is calculated from the gradient at the sample point as

$$N(x, y, z) = \frac{\nabla f(x, y, z)}{|\nabla f(x, y, z)|}. \quad (3.3)$$

3.1.2.4 Compositing

The compositing stage blends each sample's color and opacity with the color and opacity already accumulated along the ray. When beginning front-to-back ray traversal, the accumulated color C_{acc} is set to zero and the remaining opacity α_{rem} is set to one. At step i along the ray, the compositing is performed as

$$C_{acc} = C_{acc} + (C_i \alpha_i) \alpha_{rem} \quad (3.4)$$

$$\alpha_{rem} = (1 - \alpha_i) \alpha_{rem}, \quad (3.5)$$

where C_i is the i th sample's color, and α_i is the i th sample's opacity. Early ray termination is often employed once the remaining opacity reaches zero to increase rendering speed, as all further voxels along the ray will be completely occluded.

3.1.2.5 Other volume rendering methods

Ray casting is not the only form of volume rendering, although all consist of the same basic steps. A data flow diagram of the general volume rendering algorithm is shown in Figure 3.11.

The various volume-rendering techniques other than ray casting include splatting (Westover, 1989), shear-warp factorization (Cameron and Undrill, 1992; Lacroute and Levoy, 1994), graphics-hardware-accelerated texture mapping (Drebin et al., 1988; Cullip and Neumann, 1993; Cabral et al., 1994; Dachille et al., 1998), and pre-integrated volume rendering (Engel

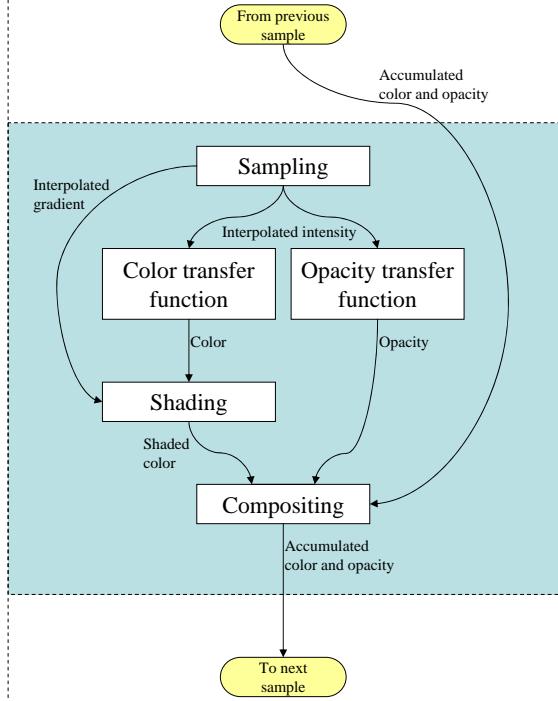


Figure 3.11: Data flow in volume rendering. The blue box is repeated once for each sample contributing to a pixel. This whole process is repeated for each pixel in the final image.

et al., 2001). These techniques are generally used to improve rendering performance over ray casting, taking advantage of specialized hardware and data structures, and may use either front-to-back or back-to-front compositing (Porter and Duff, 1984).

3.1.3 Comparison of Surface Rendering and Volume Rendering

As its name suggests, surface rendering is limited to rendering surfaces. This involves making a binary decision about whether or not a given voxel contributes to the final image. Either a surface passes through a given voxel or it does not. This binary classification does not allow for the display of weak or fuzzy surfaces. Volume rendering can render isosurfaces if wanted (Lin and Ching, 1996; Parker et al., 1998; Choi et al., 2000), but does not require such a binary classification, and can therefore render weak or fuzzy surfaces as well (Figure 3.9). In addition, atmospheric effects (where there is no surface) can be rendered using volume rendering. Although volume rendering can render isosurfaces, it can also render surfaces that consist of a range of values.

Achieving correct transparency using surface rendering requires sorting the surfaces in depth before each frame, a costly procedure that considerably increases rendering time, or use of a technique called depth peeling that can achieve order-independent transparency via multi-pass rendering (Mammen, 1989; Diefenbach, 1996). Volume rendering automatically handles transparency correctly with the compositing step, as the samples along the ray are always sorted by depth.

The surface extraction necessary for surface rendering is often a slow procedure, whereas volume rendering requires no such surface extraction. For example, changing the isosurface level is a costly procedure, whereas changing the transfer function in volume rendering has very little overhead. However, rendering the extracted surface can be faster. Measuring the tradeoff in rendering speed between the two techniques is difficult, as it depends on the scene being rendered, the hardware being used, and any acceleration schemes employed. However, both can achieve interactive rates for typical views generated from MRI and CT data.

3.1.4 Determining the Transfer Function

One major drawback of volume rendering is that designing a transfer function to render what the user wishes to see can be unintuitive and difficult. Interactive (Kindlmann and Durkin, 1998; Kniss et al., 2001) and automatic (Kniss et al., 2003) techniques have been developed for the construction of multidimensional transfer functions that incorporate first and second derivatives to define surfaces (Hladuvka et al., 2000; Kindlmann et al., 2003). Such work has led to an algorithm to automatically estimate opacity transfer functions to display tissue interfaces in noisy 3D ultrasound data (Honigmann et al., 2003). A parallel-coordinates interface that enables the user to efficiently search the large-dimensional parameter space, keeping track of the most effective settings and indicating nearby likely candidates is described in (Tory et al., 2005). Effective transfer function design is not the goal of this dissertation, but complements it well, as the occlusion-handling techniques described here are independent of transfer function design.

3.2 Dealing with Occlusion in Volume Graphics

When viewing volume data, unimportant portions of the volume often occlude areas of interest. The main advantage of the techniques presented in this dissertation is that they automatically deal with this issue. Previous work on dealing with occlusion in volume graphics include illustrative techniques and focus-and-context techniques.

3.2.1 Illustrative Techniques

A surface-rendering technique for view-dependent transparency that aims to automatically produce translucent surfaces similar to technical illustrations is described in (Diepstraten et al., 2002). Later work describes techniques for automatically producing breakaway and cutaway illustrations of nested surfaces (Diepstraten et al., 2003) (Figure 3.12 a). These

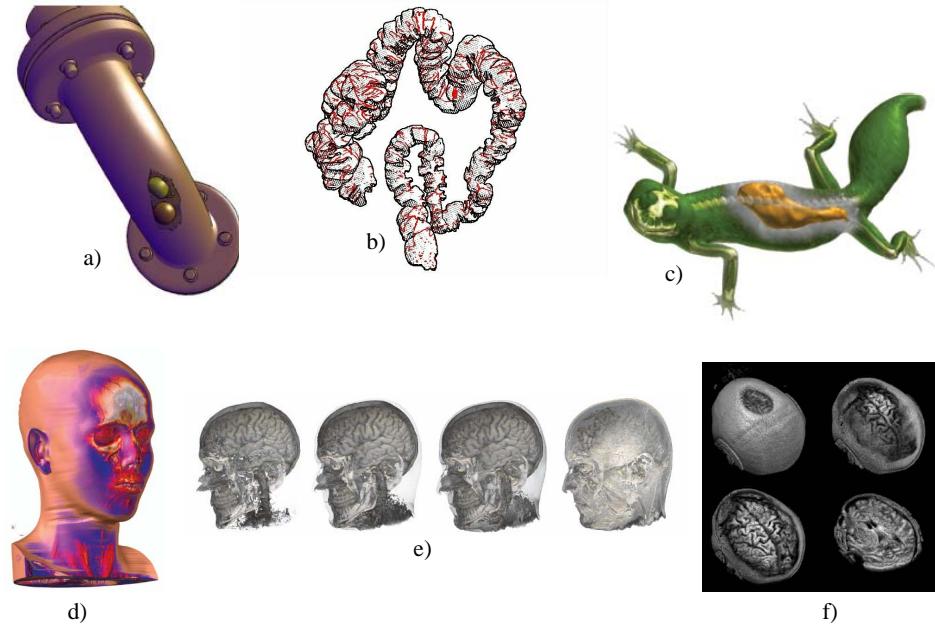


Figure 3.12: Various occlusion handling techniques in volume graphics. a) Illustrative cutaway illustrations (Diepstraten et al., 2003). b) Illustrative display of hidden isosurface structures (Fischer et al., 2005). c) Importance-driven volume rendering (Viola et al., 2004). d) Context-preserving volume rendering (Bruckner et al., 2006). e) Opacity-peeling (Rezk-Salama and Kolb, 2006). f) Confocal volume rendering (Mullicka et al., 2000).

illustrations remove portions of geometry that occlude the interior surfaces as opposed to

rendering those portions as a translucent material. Similar work applied to isosurface rendering is found in (Fischer et al., 2005). These techniques are useful for displaying outer and inner structures when the surfaces have already been extracted from the volume (Figure 3.12 b). The techniques in this dissertation work with volumes in which no surface extraction is performed.

3.2.2 Focus-and-Context Techniques

A class of related focus-and-context volume rendering techniques has been developed to reveal inner structures of volumes while retaining some outer structures to show context. The work of (Viola et al., 2004), produces importance-driven volume rendering that highlights features of interest and automatically subdues the display of occluding objects using one of several possible techniques (Figure 3.12 c). This technique is effective for pre-segmented volumes in which there exists a hierarchy of importance among objects, and where interactive rendering is not important. The techniques presented in this dissertation are designed to work rapidly with unsegmented volumes in which no importance hierarchy is available.

Another technique for viewing inner structures is the work of (Bruckner et al., 2006), which selectively reduces the opacity of regions using a function of shading intensity, gradient magnitude, distance to the eye point, and previously accumulated opacity. This technique is used to reduce the opacity of surface in large, flat regions, which is effective in enabling inner structures to show through in these areas (Figure 3.12 d). The techniques presented in this dissertation are designed to display opaque contiguous surfaces, which are important for medical diagnosis.

(Rezk-Salama and Kolb, 2006) presents an opacity-peeling technique to reveal inner structures. In this work, some number of fully-opaque layers of material is peeled away. This work is shown to be effective for visualizing brain structures (Figure 3.12 e). For the applications addressed in this dissertation, the opacity-peeling technique described could achieve results similar to those of volumetric depth peeling (VDP) described in Chapter 4, including its issues with artifacts due to insufficient surface separation, although VDP correctly handles transitioning from occluded viewpoints to non-occluded viewpoints. This dissertation also addresses

artifacts due to insufficient surface separation.

Another technique for revealing inner structures is the work of (Mullicka et al., 2000), which employs a depth-dependent focal region. This work proves effective for viewing the inner structures of the brain from MRI data (Figure 3.12 f). The techniques described in this dissertation address applications such as virtual arthroscopy, where the optimal separation between surfaces is generally not at a fixed depth.

3.3 Volume Rendering of Medical Images

Volume rendering has been used to generate 3D views of medical images since its description in (Levoy, 1988), and has been demonstrated to improve effectiveness in clinical settings compared to 2D views (Hemminger et al., 1995; Zuiderveld et al., 1996). Especially relevant to this dissertation are volume rendering systems for rendering joint surfaces, and volume rendering systems for performing fly-throughs of various parts of the anatomy.

A volume-rendering-based interactive virtual colonoscopy navigation system has been developed (Wan et al., 1999). Later work includes a combined surface and volume rendering approach for indicating which regions of the surface have been surveyed, and which have not yet been examined (Kaufman et al., 2005). Such systems constrain the viewpoint to positions within the colon, providing limited fields of view. This limitation can be especially problematic in the colon, due to its many complex folds. To address this issue, virtual colon dissection techniques have been developed that unroll the colon and slice it open, laying it flat. Although this approach is effective in reducing errors due to occlusion and perspective, it introduces geometric distortion to the colon surface. By incorporating the work presented in this dissertation, virtual colonoscopy systems could be augmented to provide viewpoints of the interior of the colon from exterior positions without geometric distortion (Figure 3.13), although less surface area would be visible than with the dissection technique.

Volume visualization systems have also been effective in studying aortic aneurysms (Tam et al., 1997). The feasibility of 3D rendering of joint surfaces and pathology using MRI and CT data sets has been demonstrated by several authors (Applegate, 1998; Weishaupt et al., 1999; Lee et al., 2004; Sahin et al., 2004). Such techniques provide real-time interaction and

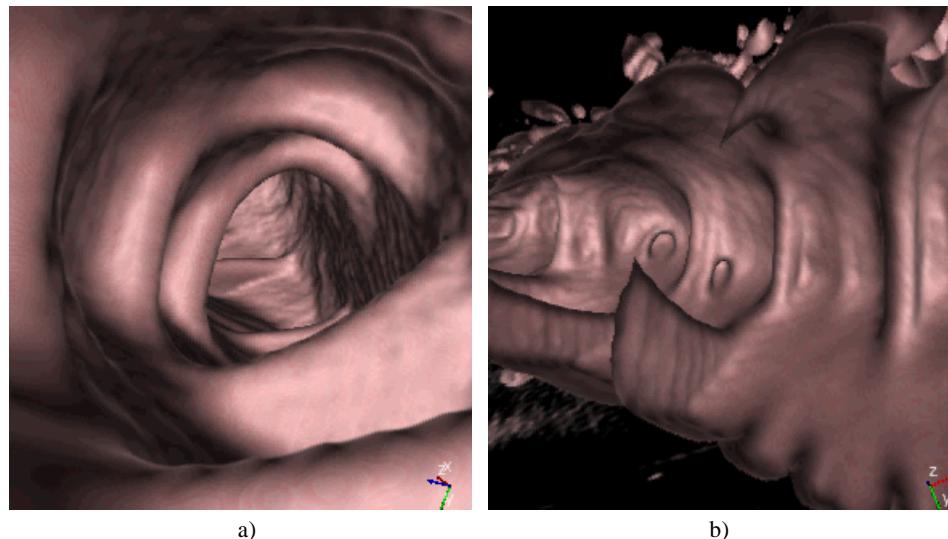


Figure 3.13: Traditional virtual colonoscopy a) restricts the viewpoint to positions within the colon. Virtual colonoscopy with flexible occlusion rendering b) enables viewpoints exterior to the colon, giving a wider view of the interior surface.

evaluation of joints for virtual arthroscopy, but are also limited to constrained viewpoints. The work in this dissertation is designed to enable optimal viewpoints to aid physicians in quick, accurate diagnosis.

CHAPTER 4

FLEXIBLE OCCLUSION RENDERING

4.1 Motivating Example

The goal of flexible occlusion rederring (FOR) is to enable view-dependent rendering of occluded objects in a volume for displaying useful views of MRI and CT data to aid physicians in diagnosis. For example, when using volume rendering to perform virtual arthroscopy (described in more detail in Chapter 5) from a shoulder MRI, a view of the entire shoulder socket is desired for rapid evaluation of the cartilage surface for defects (Figure 4.1). However,

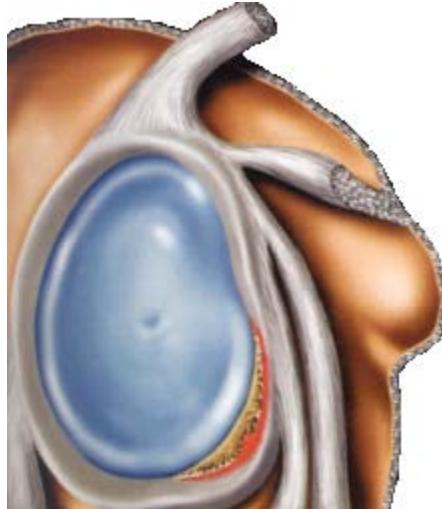


Figure 4.1: Textbook illustration of a shoulder socket (Stoller et al., 2004). In a real shoulder, this viewpoint would be occluded by the humeral head.

views from inside the humeral head that would otherwise show the entire socket are not possible, as the cartilage surface of the humeral head occludes the socket (Figure 4.3, Camera 1). Adjusting the volume rendering transfer function to reduce the opacity of the cartilage surface of the humeral head will also reduce the opacity of the cartilage surface of the shoulder

socket, making the resulting image very difficult to interpret (Figure 4.2).

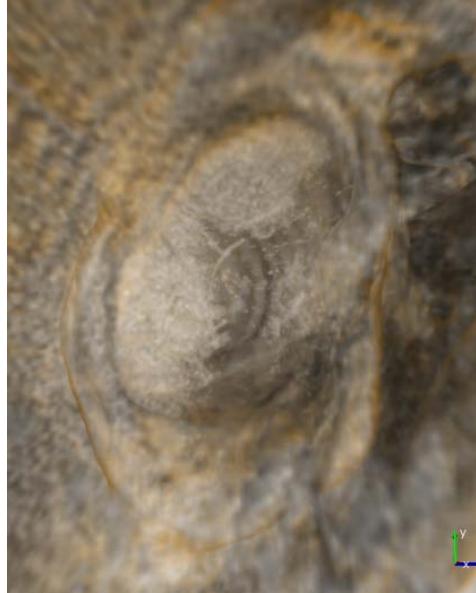


Figure 4.2: Shoulder rendered from MRI data. Rendering the cartilage surfaces with partial transparency results in a confusing image.

Therefore, with standard volume rendering, views of the shoulder socket are limited to those from positions within the fluid-filled joint capsule (Figure 4.3, Camera 2). Because of this limitation, the physician must evaluate the socket from viewpoints either at extreme angles or very close to the surface, much like evaluating the surface of the hand with the palm touching the tip of the nose (Clarke, 2004). A method for removing the occluding material of the humeral head, enabling unobstructed views of the shoulder socket, is required.

4.1.1 The Desired System

What is desired is a system that requires minimal supervised preprocessing of the data, does not suffer from segmentation artifacts, is robust in areas where segmentation of objects is difficult, handles self-occlusion, enables view-dependent determination of the object of interest, and correctly handles transitions from occluded viewpoints to unoccluded viewpoints.

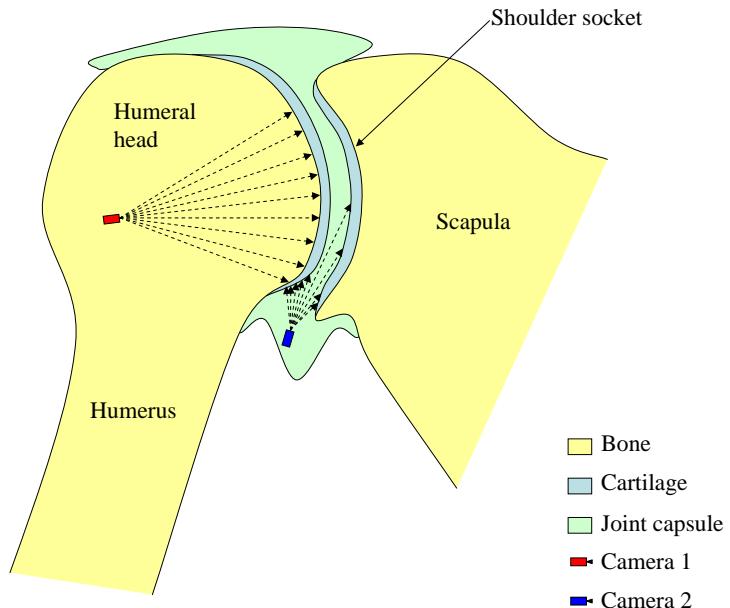


Figure 4.3: Shoulder diagram, coronal slice. The cartilage surface of the humeral head occludes the shoulder socket from viewpoints within the humeral head, and viewpoints within the joint capsule give limited views of the shoulder socket.

4.1.2 Potential Solutions

4.1.2.1 Clipping plane

One potential solution for removing the occluding material of the humeral head is to apply a clipping plane. A clipping plane removes all occluding material between the viewpoint and the plane. The plane can either be placed at a certain distance in front of the viewpoint, or at a predefined position and orientation in the volume. There are two major drawbacks to using a clipping plane for removing the occluding material.

The first drawback is that the clipping plane must be continually redefined by the user when moving the viewpoint. For a clipping plane placed a certain distance in front of the viewpoint, this distance must be adjusted based on the distance from the camera to the shoulder socket. For a clipping plane at a predefined position and orientation in the volume, the position and orientation must be adjusted to give the best effect depending on the viewpoint.

The more fundamental drawback of the clipping plane approach is that, due to the ge-

ometry of the ball-and-socket joint, no single clipping plane is sufficient to render the entire socket (Figure 4.4). This drawback is not specific to the shoulder, as many joints and other

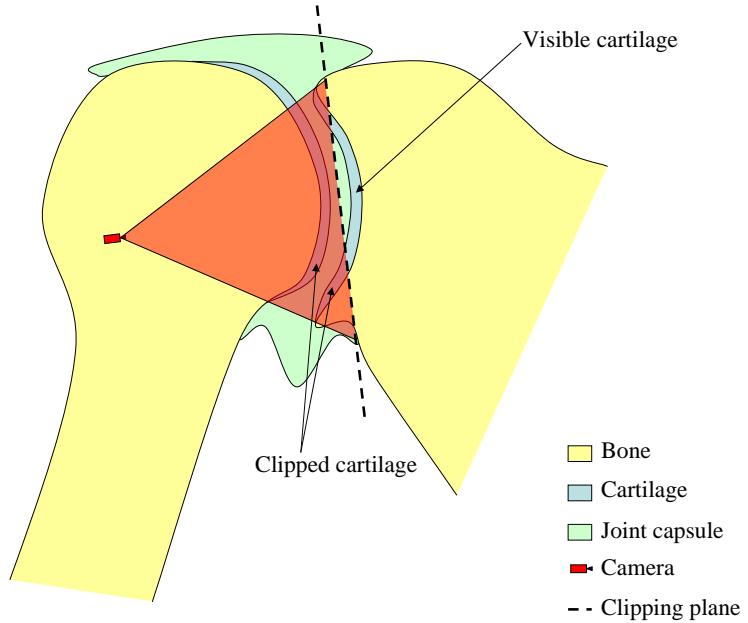


Figure 4.4: No single clipping plane will show the entire socket.

anatomical features of interest have even more complex geometries than that of the shoulder. FOR essentially implements an arbitrarily-shaped, data-dependent clipping surface, thus removing this limitation.

4.1.2.2 Segmentation

Another possible solution for removing the occluding material of the humeral head is to pre-segment the volume. Segmentation is the partitioning of the volume into multiple regions corresponding to objects in the volume. Accurate segmentation of the humeral head can enable its removal from the volume.

One way to perform segmentation is to manually segment the region of interest (ROI). Manual segmentation is an extremely time-consuming and tedious task, requiring the physician to draw ROIs on each slice of the volume. With the large number of medical data sets generated daily, the time taken to perform such a segmentation is prohibitive (Clarke, 2004).

Various automatic segmentation algorithms also exist. Voxel-based segmentation methods, such as thresholding, attempt to classify each voxel into a tissue-type based on voxel intensities. Other methods incorporate different metrics, such as texture. After voxel classification, region-growing techniques can be performed to group the volume into objects.

Model-based methods, such as active contour models¹ (ACM), active shape models (ASM), and active appearance models (AAM), deform an object boundary to fit the image data (Kass et al., 1988; Cootes and Taylor, 1998; Cootes et al., 1992). Level set methods are similar to ACM, but model the evolving boundary implicitly as a higher-dimensional surface (Osher and Sethian, 1988; Sethian, 1996).

Automatic segmentation methods can be computationally expensive for large volumes and often require fine tuning of parameters. Obtaining a good segmentation might therefore require a few user-guided iterations, and may still require some manual modification. Most automatic segmentation techniques require numerous pre-segmented data sets to learn from to aid in classification, so these training data sets must be created manually. Additionally, for the purposes of object removal, an error in the segmentation that joins two objects together results in either both objects being shown, or neither. This problem can arise with the humeral head and shoulder socket when there is not good separation between the two surfaces (Figure 4.5).

Assuming a good segmentation, simply removing the humeral head is still problematic for a virtual arthroscopy application. During the same analysis, it is often desirable to view the socket from positions within the humeral head, and to view the humeral head from within the scapula. To do so with segmentation would require an accurate segmentation of both the humerus and the scapula, and a means to determine which to view each frame of the rendering.

For certain geometries, even a perfect segmentation does not enable view-dependent display of objects of interest. For any non-convex surface, there are views from which the surface will occlude itself (Figure 4.6 a). These irregular surfaces occur in locations such as bone fractures and folded surfaces. Similarly, no segmentation is able to render the interior surfaces of

¹Active contour models are also referred to as snakes.

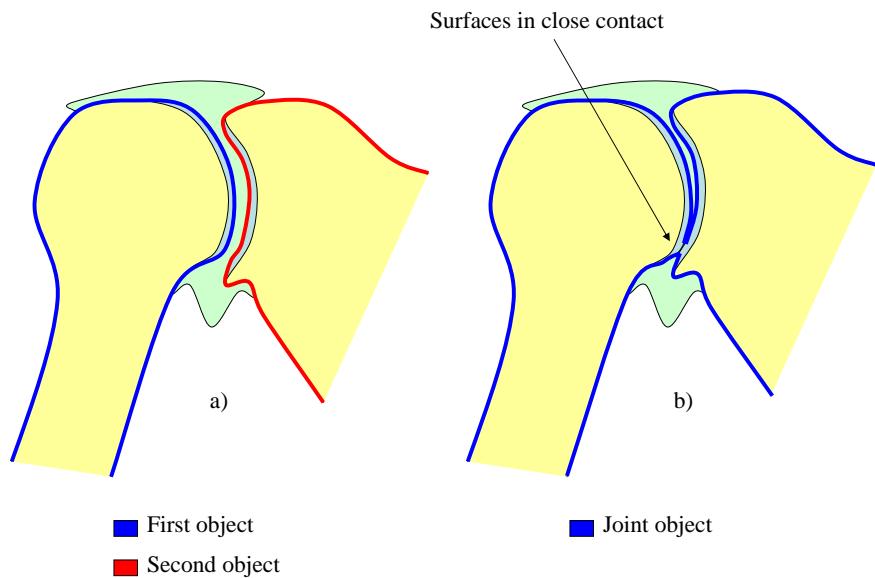


Figure 4.5: A good segmentation a) results in two separate objects. Surfaces in close contact b) can result in separate objects being treated as a single object.

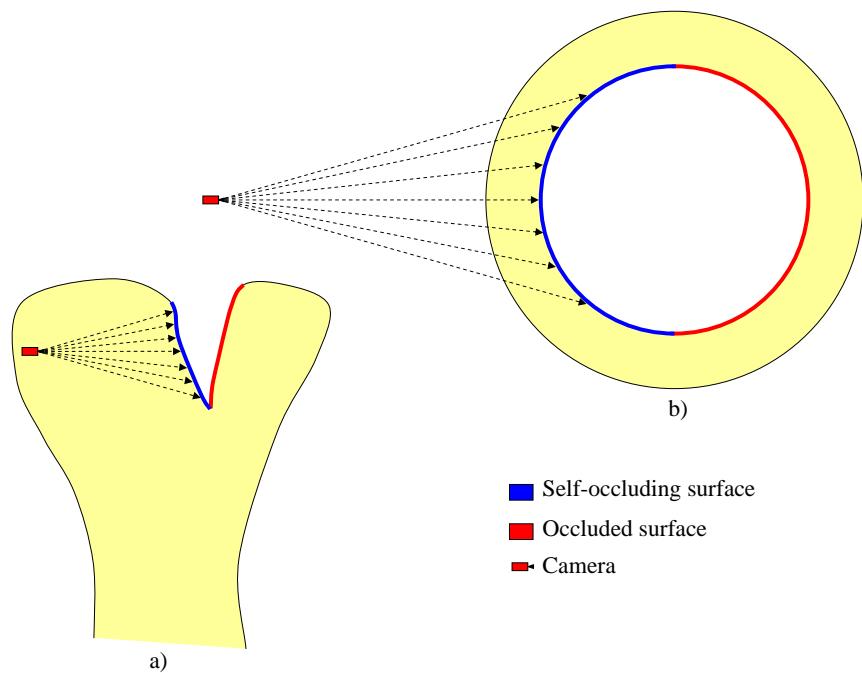


Figure 4.6: Two examples of self occlusion.

objects from arbitrary viewpoints outside the object (Figure 4.6 b). Our collaborators want to view the interior of the ureters and renal collecting systems from exterior positions in a virtual ureteroscopy application (described in Chapter 6).

4.1.2.3 Isosurfaces

The other potential solution for removing the occluding material of the humeral head involves defining an isosurface at the correct isovalue to represent the surfaces of interest. Any isosurface approach will inherit the issues with surface-based rendering discussed in Chapter 3, as well as problems with surfaces occluding each other.

A potential solution to the occlusion problem is to use back-face culling. If the isosurfaces are rendered using back-face culling, polygons that face away from the viewpoint are not rendered. Mathematically, if the polygon normal is N and the vector from the viewpoint to the polygon is V , the polygon is culled if $N \cdot V > 0$. This approach will work for rendering the interior of ureter surfaces, however there are common configurations in which backface culling is insufficient. In Figure 4.7 a, backface culling works, as the surface normals of the humeral head isosurface point away from the camera. However, Figure 4.7 b shows a configuration that can also arise with MRI or CT data. In this case, simple back-face culling does not remove the entire surface of the humeral head, as there are front-facing polygons of the humeral head between the viewpoint and the socket.

To remove the front-facing polygons of the humeral head isosurface, depth peeling can be used. Depth peeling is a framebuffer-based technique that can achieve order-independent transparency when rendering polygons (Mammen, 1989; Diefenbach, 1996; Everitt, 2002). From viewpoints within the humeral head, setting the first depth layer of polygons to fully transparent renders the humeral head isosurface fully transparent, revealing the shoulder socket surface. However, this technique will require the user to enable and disable depth peeling when maneuvering into and out of the joint capsule, as the shoulder socket isosurface is made transparent when the humeral head isosurface does not occlude it. FOR enables moving seamlessly between the humeral head and the joint capsule, which is common when performing surveys of the socket and then inspecting areas of interest more closely.

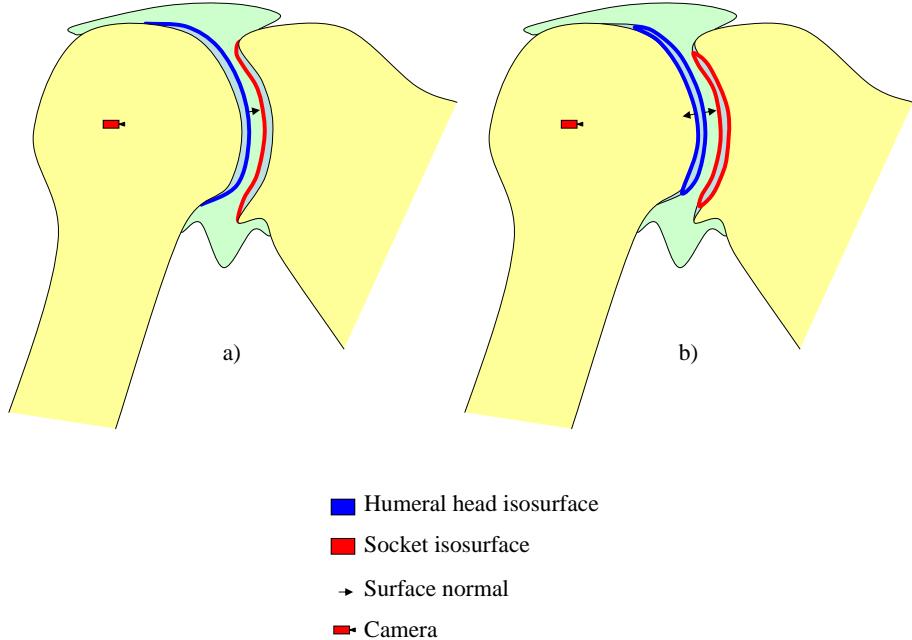


Figure 4.7: Situations where a) backface culling works because the normal for the humeral head isosurface points away from the camera, and b) backface culling does not work because portions of the humeral head isosurface point towards the camera.

4.2 Volumetric Depth Peeling

Volumetric depth peeling (VDP) is the simplest implementation of FOR. It shares its name with (Nagy and Klein, 2003), which involves texture-based isosurface rendering with depth peeling. However, the implementation of VDP described here is extended to the more general case of ray-based volume rendering. It retains full transfer function control, enabling the full range of volumetric effects to be applied, including effects that simulate directionally-illuminated surface rendering. VDP is an extension to standard ray-cast volume rendering that resets accumulated color and opacity along each ray when features in the volume indicating a separation between occluding objects is found. Figure 4.8 shows a standard volume rendering of a shoulder joint compared with a rendering using VDP.

With the shoulder joint example, an MRI image displays the bright fluid-filled joint capsule defining the space between the humeral head and the shoulder socket (Figure 4.9). In standard

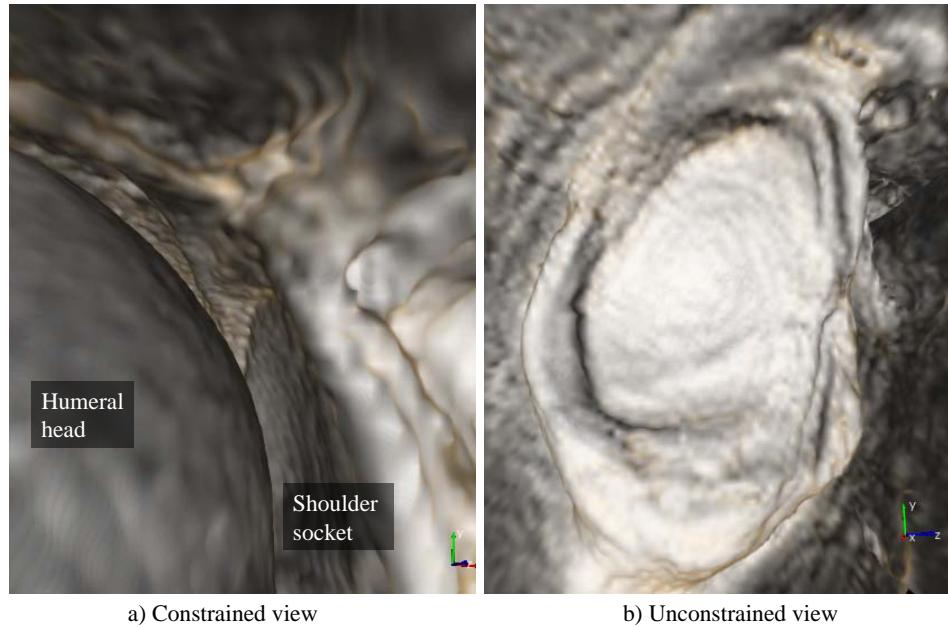


Figure 4.8: A standard volume rendering a) of a shoulder from MRI data is constrained to viewpoints within the fluid-filled joint capsule between the humeral head and the shoulder socket. Using VDP b) enables survey views of the entire socket by automatically removing the occluding material of the humeral head.

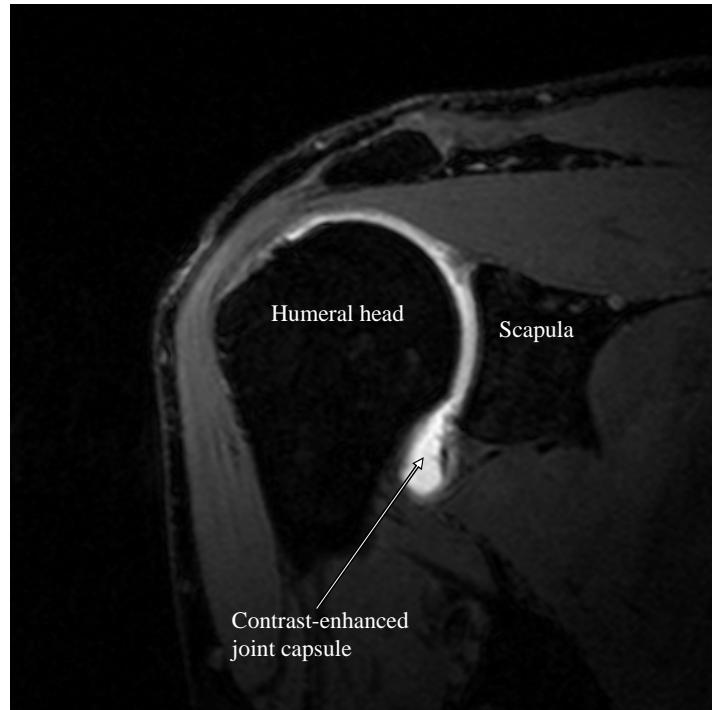


Figure 4.9: Coronal slice of a shoulder MRI. The bright fluid-filled joint capsule defines the space between the humeral head and the shoulder socket.

MRI joint trauma diagnosis, a fluid called a contrast agent² is often injected into the joint capsule to give a brighter signal and increase the separation between joint surfaces. Most of the MRI data sets used in this dissertation include such contrast enhancement, however the term fluid will be used instead of contrast agent to retain generality.

VDP works by determining when each ray encounters the fluid-filled joint capsule, resetting the accumulated color and remaining opacity the first time it is encountered. The effect of resetting the accumulated color and remaining opacity is to clip away all of the occluding material between the viewpoint and the far side of the joint capsule. The simplest method to determine when the separating fluid is encountered is a fixed-value threshold.

²Gadolinium-based contrast agents are commonly used.

4.2.1 Fixed-Threshold Method

4.2.1.1 Algorithm

In standard ray-cast volume rendering, a ray is cast into the volume for each pixel in the rendered image. As the ray traverses the volume, it accumulates color and opacity based on the volume rendering transfer function. For VDP with a fixed threshold, the threshold is set to a value above which represents fluid in the MRI image. When traversing each ray, the first time the current data value is greater than or equal to the threshold, the accumulated color and remaining opacity are reset. This removes the occluding material, making the shoulder socket visible (Figure 4.10). Psuedocode for this algorithm is given in Figure 4.11.

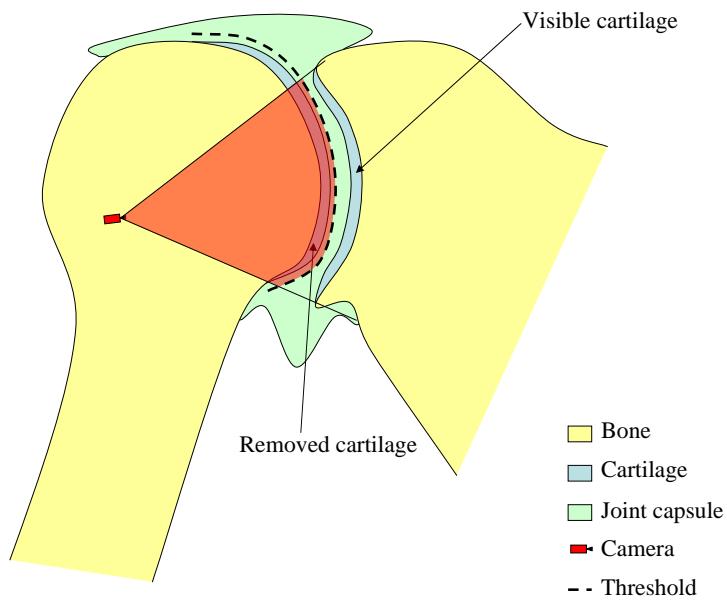


Figure 4.10: With VDP, the accumulated color and remaining opacity between the viewpoint and the threshold are reset, making the entire socket visible.

In addition to the case of separate objects occluding each other, the fixed-threshold VDP algorithm works for self-occluding surfaces. For example, in virtual ureteroscopy of CT data a high-density contrast agent is introduced within the ureters and renal collecting systems to aid in visualizing their interior surfaces for small tumors and stones. It is desirable to enable views of the interior surfaces of the renal collecting system from exterior viewpoints to obtain

```

rayPosition ← rayBegin;
accumulatedColor ← (0.0,0.0,0.0);
remainingOpacity ← 1.0;
reachedThreshold ← false;

while rayPosition < rayEnd do
    // Sampling
    sampleValue ← InterpolateValue(rayPosition);
    // /////////////////////////////////
    // Fixed-Threshold VDP
    if !reachedThreshold then
        if sampleValue >= threshold then
            accumulatedColor ← (0.0,0.0,0.0);
            remainingOpacity ← 1.0;
            reachedThreshold ← true;
        end
    end
    // /////////////////////////////////
    // Transfer function evaluation
    sampleColor ← ColorTransferFunction(value);
    sampleOpacity ← OpacityTransferFunction(value);

    // Shading
    sampleGradient ← InterpolateGradient(rayPosition);
    shadedColor ← DoShading(sampleColor, sampleGradient);

    // Compositing
    accumulatedColor ← accumulatedColor + shadedColor * sampleOpacity *
    remainingOpacity;
    remainingOpacity ← (1.0 – sampleOpacity) * remainingOpacity;
    // Increment position
    rayPosition ← IncrementRayPosition(rayPosition);
end

```

Figure 4.11: Fixed-threshold VDP within a standard volume rendering implementation.

survey views. The same fixed-threshold VDP method will work in this case, as shown in Figure 4.12.

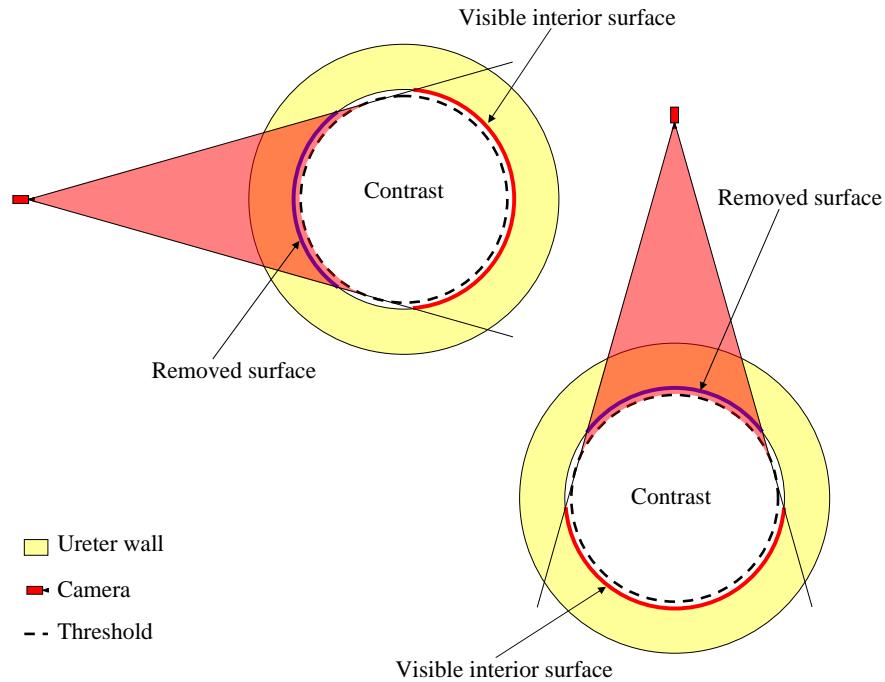


Figure 4.12: Tube structures. Using VDP, the far interior surface is visible from any angle.

4.2.1.2 Selecting the threshold

Figure 4.13 shows the same MRI image shown in Figure 4.9, but with a black-body radiation color map and an appropriate window/level applied to aid in distinguishing bone, cartilage/soft tissue, and fluid. A diagram of an idealized intensity profile along a ray such as the one indicated in Figure 4.13, starting in the humeral head, passing through the joint capsule, and ending in the scapula, is shown in Figure 4.14. The indicated tissue-type boundaries are approximate, so samples at the boundaries should be weighted less than samples well within the tissue boundaries when rendering. Therefore, to render the cartilage surface using volume rendering, a ramped scalar opacity transfer function is used (Figure 4.15). Such a transfer function gives full opacity to samples that are well within the range of cartilage values, and increasingly less opacity to samples as their value moves away from the central region of the range of cartilage values.

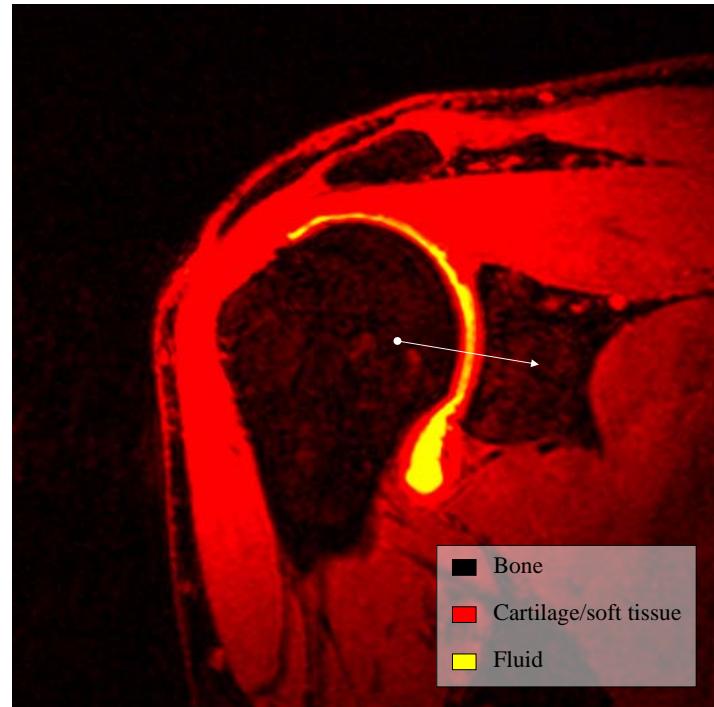


Figure 4.13: Shoulder MRI. With the black-body radiation color map and an appropriate window/level applied, the tissue types are easier to distinguish visually.

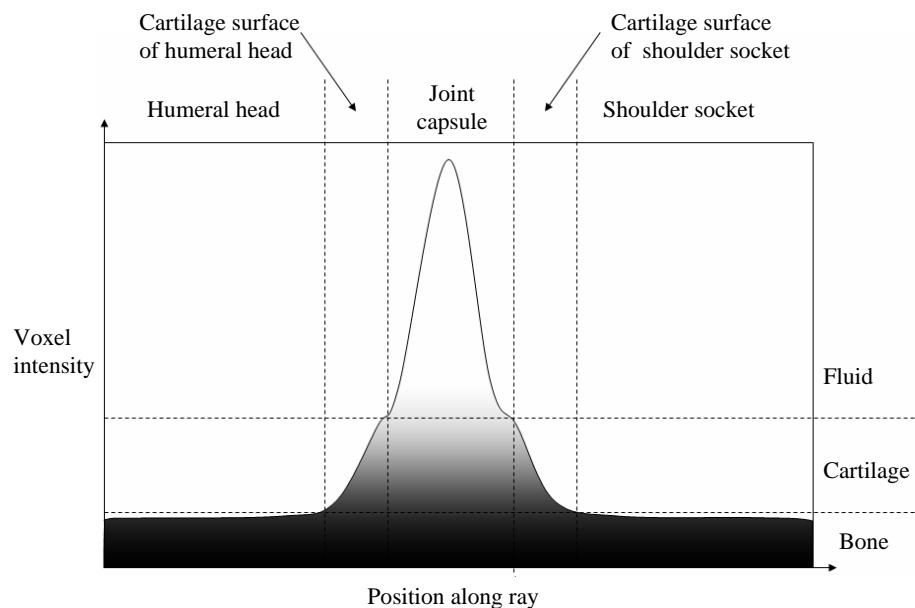


Figure 4.14: An idealized MRI intensity profile of a ray starting in the humeral head, passing through the joint capsule, and ending in the scapula.

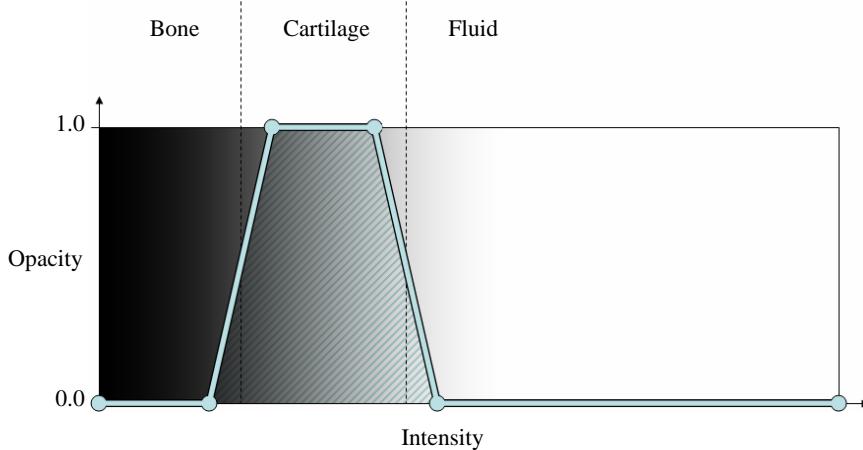


Figure 4.15: A typical scalar opacity transfer function used to display cartilage surfaces.

The VDP threshold for resetting can be set independently from the transfer function settings. Figure 4.16 shows a diagram of three different thresholds. The low threshold is set within the range of cartilage values, the medium threshold is set to the maximum cartilage value, and the high value is set above the range of cartilage values. The low threshold resets too early, causing part of the cartilage surface of the humeral head to be rendered. The medium threshold resets when exiting the cartilage surface of the humeral head, exposing the socket. The high threshold also exposes the surface of the socket, but along some rays the threshold is not reached. VDP renderings from the same viewpoint with different thresholds are shown in Figure 4.17. The artifacts circled in red are caused by the threshold not being reached along some rays, due to lower concentrations of fluid where the cartilage surfaces are close together or touching. Because the threshold is not reached, the near cartilage surface of the humeral head is rendered in these locations. The optimal setting for the threshold is therefore at the maximum value representing cartilage in the transfer function. Such a setting ensures that the ray has exited the cartilage surface of the humeral head, while introducing as few artifacts as possible. Note that the fixed-threshold VDP technique differs from depth and opacity peeling techniques in that it correctly renders the first surface encountered from viewpoints within the joint capsule.

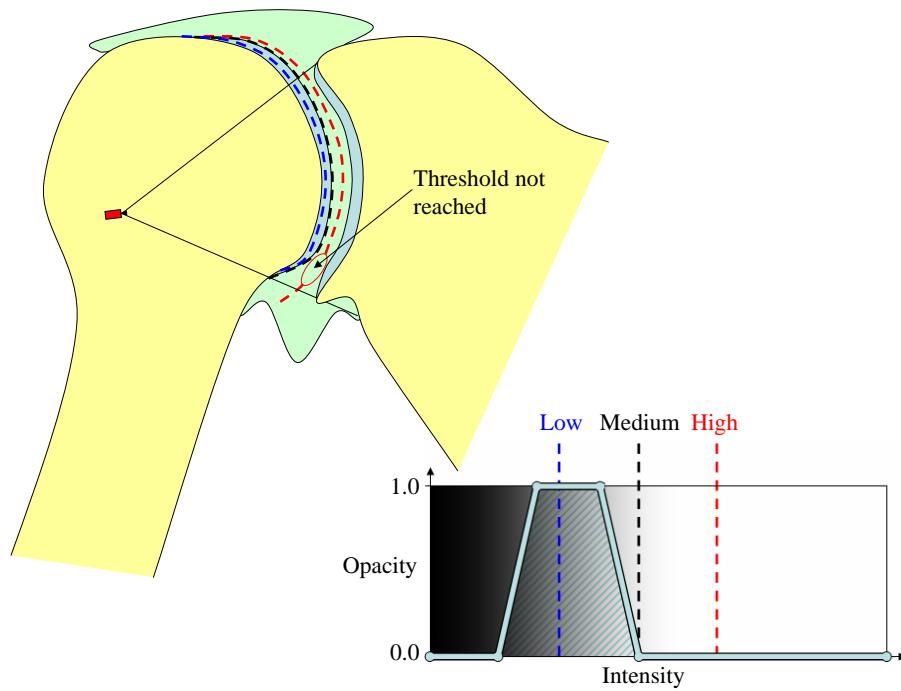


Figure 4.16: A low threshold, shown in blue, results in the cartilage surface of the humeral head being rendered. A medium threshold, shown in black, reveals the shoulder socket. A high threshold, shown in red, is not reached along all rays.

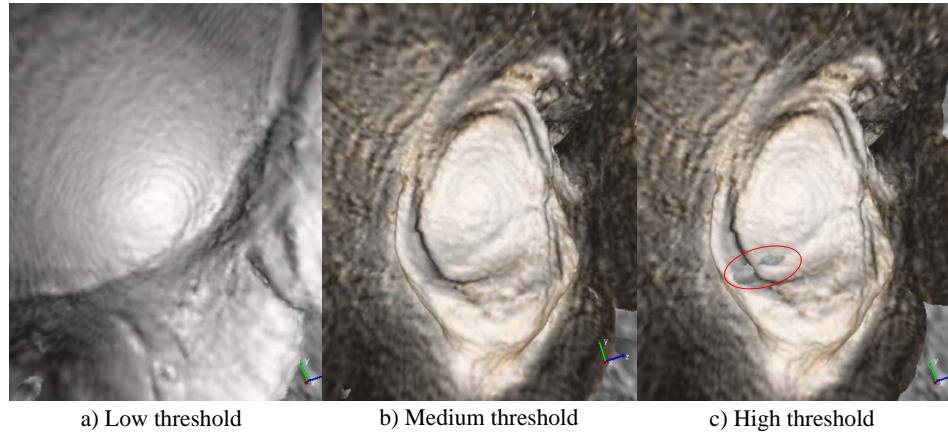


Figure 4.17: VDP renderings of shoulders from MRI data. A low threshold a) results in the near side of the cartilage surface of the humeral head being rendered. A medium threshold b) results in a good rendering of the shoulder socket. A high threshold c) results in artifacts, circled in red, due to the threshold not being reached.

4.2.1.3 Extensions

Various extensions to the basic fixed-threshold VDP method are possible. One such extension is to modulate the accumulated color and opacity when reaching the threshold, instead of resetting. Doing so can achieve a transparency effect for the near surface, rendering the far surface fully opaque although both surfaces are the same material. Figure 4.18 shows two renderings of a contrast-filled rubber glove used as a medical phantom in the virtual ureteroscopy study described in Chapter 6. Figure 4.18 a) displays the exterior surface of the glove

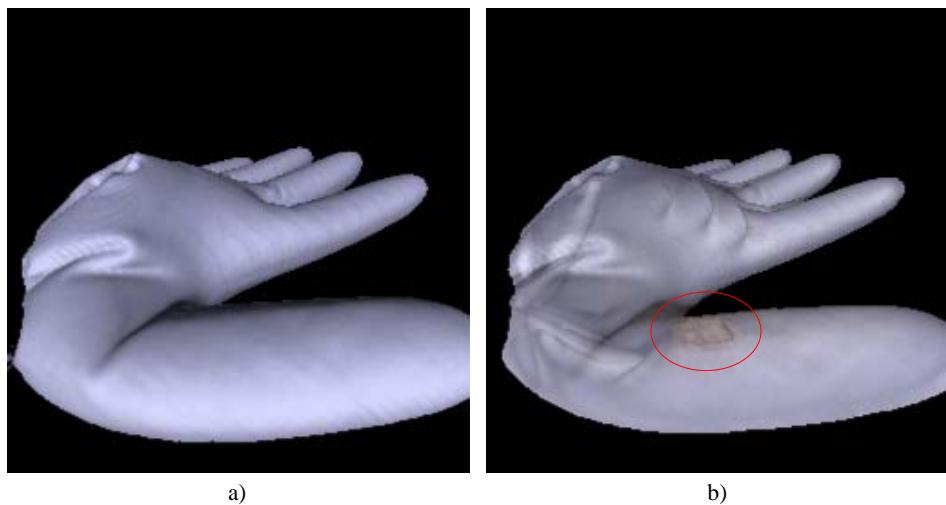


Figure 4.18: VDP renderings of a contrast-filled rubber glove used as a medical phantom in a virtual ureteroscopy study. Using full opacity a) renders the outer surface, as with standard volume rendering. Modulating the opacity b) achieves a transparency effect. Note the small mass circled in red that can be seen on the inner surface.

using standard volume rendering. Figure 4.18 b) enables VDP to display the inner surface of the glove, but modulates the accumulated color and opacity of the outer surface instead of resetting. Different color transfer functions are also used for the outer and inner surfaces, although they are the same material.

Another choice when rendering is what to do along rays that do not reach the threshold. Depending on the application, these rays can either be rendered as with normal volume rendering to provide context, or set to the background color to remove distraction (Figure 4.19). One problem with rendering the rays that do not reach the threshold is that a lack of

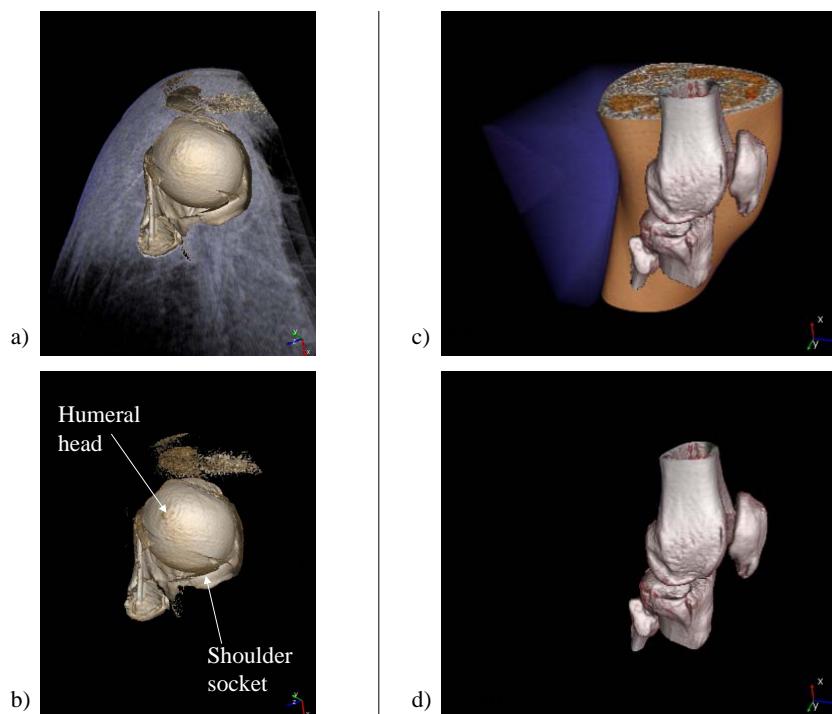


Figure 4.19: A top-down view of a shoulder rendered from MRI data a), b), and a view of a knee rendered from CT data c), d). Rays that do not reach the threshold can be rendered as with normal volume rendering a), c) to provide context, or set to black b), d) to remove distraction.

occlusion cues for determining the relative depths of objects can lead to confusion. However, using opacity modulation helps alleviate this problem (Figure 4.20). In this manner, VDP can

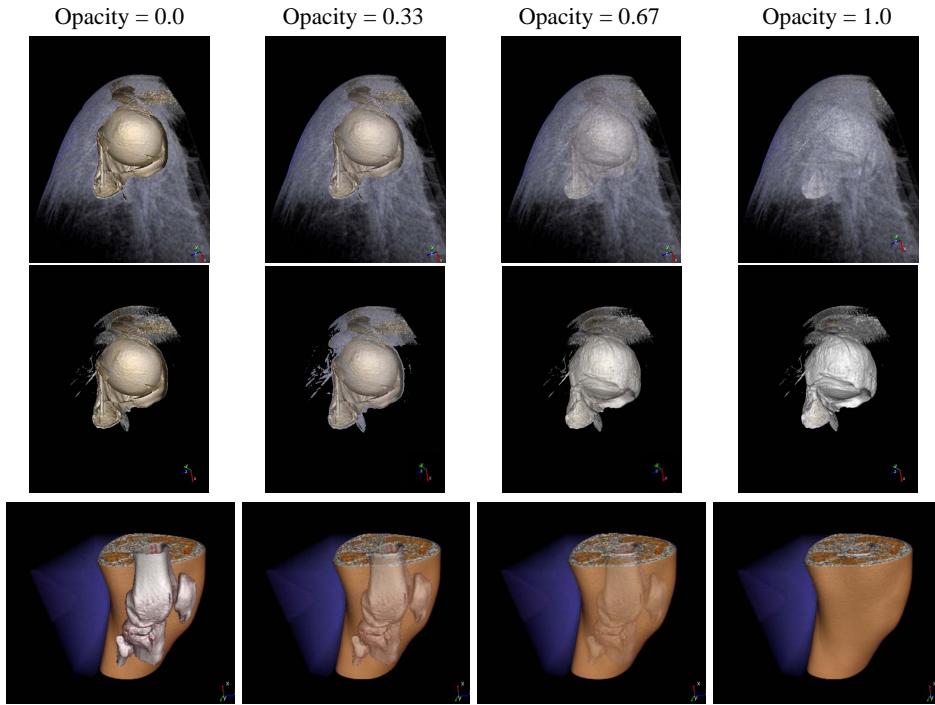


Figure 4.20: Accumulated color and opacity modulation combined with display of rays that don't reach the threshold. The first row shows increasing opacity for a top-down view of a shoulder rendered from MRI data. The second row shows the same shoulder with a different transfer function applied to render the outer surface of the joint capsule. The third row shows a knee rendered from CT data. In all cases, full opacity has the same effect as not performing VDP.

achieve effects similar to importance-driven volume rendering (Viola et al., 2004) described in Chapter 3 without requiring pre-segmentation. Only simple blended transparency is currently implemented, however.

Such focus-and-context techniques are possible with VDP to enable survey views of the data. However, the most useful form of VDP for the applications addressed in this dissertation is to only render rays that reach the threshold, and to remove all of the accumulated color and opacity when reaching the threshold. In this case, only the data on the far side of the threshold will contribute to the final image. A performance increase over the standard VDP algorithm is therefore enabled by only accumulating color and opacity when the threshold has been reached. Pseudocode for this variant of VDP is given in Figure 4.21. Most of the

```

rayPosition ← rayBegin;
accumulatedColor ← (0.0,0.0,0.0);
remainingOpacity ← 1.0;
reachedThreshold ← false;

while rayPosition < rayEnd do
    // Sampling
    sampleValue ← InterpolateValue(rayPosition);
    // /////////////////////////////////
    // Skip-to-Threshold VDP
    if !reachedThreshold then
        if sampleValue >= threshold then
            reachedThreshold ← true;
        end
    end
    if reachedThreshold then
        // Transfer function evaluation
        sampleColor ← ColorTransferFunction(value);
        sampleOpacity ← OpacityTransferFunction(value);
        // Shading
        sampleGradient ← InterpolateGradient(rayPosition);
        shadedColor ← DoShading(sampleColor, sampleGradient);
        // Compositing
        accumulatedColor ← accumulatedColor + shadedColor * sampleOpacity *
        remainingOpacity;
        remainingOpacity ← (1.0 – sampleOpacity) * remainingOpacity;
    end
    // ///////////////////////////////
    // Increment position
    rayPosition ← IncrementRayPosition(rayPosition);
end

```

Figure 4.21: Skip-to-threshold VDP.

remaining images in this dissertation employ some sort of skipping technique, modified for the particular FOR algorithm used.

For the applications presented in this dissertation, finding the first separating feature has proven useful, however finding multiple separations or the last separation are also possible, depending on the application.

The extensions described above for fixed-threshold VDP can be applied to all variants of FOR presented in this chapter.

4.2.1.4 Limitations

Selecting a threshold for VDP is problematic for shoulders where the two surfaces are in close contact. Figure 4.22 a) shows a rendering of a shoulder exhibiting hole artifacts circled in red. These artifacts occur in areas where the threshold is not reached, which causes no accumulation of color and opacity using the skip-to-threshold VDP algorithm. Reducing the threshold to eliminate the artifacts, concomitantly reducing the maximum value of the cartilage in the transfer function, results in eroded areas of the socket, circled in blue in Figure 4.22 b). In cases such as this, there is no setting for the threshold that both avoids hole artifacts and prevents erosion of the surface.

Figure 4.23 shows a slice from the MRI data used to generate these images. An intensity profile along the ray indicated in a), where the cartilage surfaces appear to be touching is shown in b). Although the intensity peak is below the threshold, a clear peak exists.

In areas where the cartilage surfaces are close to touching, each voxel contains contributions from both cartilage and fluid. Volume averaging therefore causes such voxels to have values higher than those of voxels containing just cartilage, and lower than voxels containing just fluid. The peak therefore represents the highest concentration of fluid along the ray, ignoring the presence of noise. Because of this, the peak is the optimal location at which to reset the accumulated color and opacity, as it is the most likely point of separation between the two surfaces. To find the intensity peak representing the optimal point of separation along each ray, a technique more sophisticated than the fixed threshold must be used.

A key insight is that finding the separation between the regions of interest in the volume

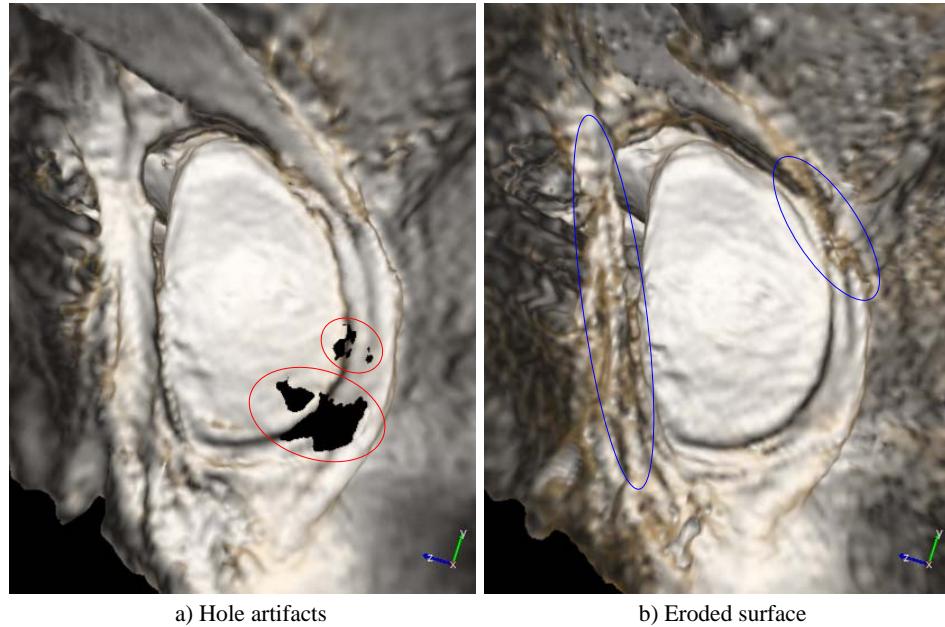


Figure 4.22: Fixed-threshold VDP limitations. Hole artifacts due to the threshold not being reached a) are circled in red. Eroded areas of the surface due to an inaccurate cartilage definition in the transfer function b) are circled in blue.

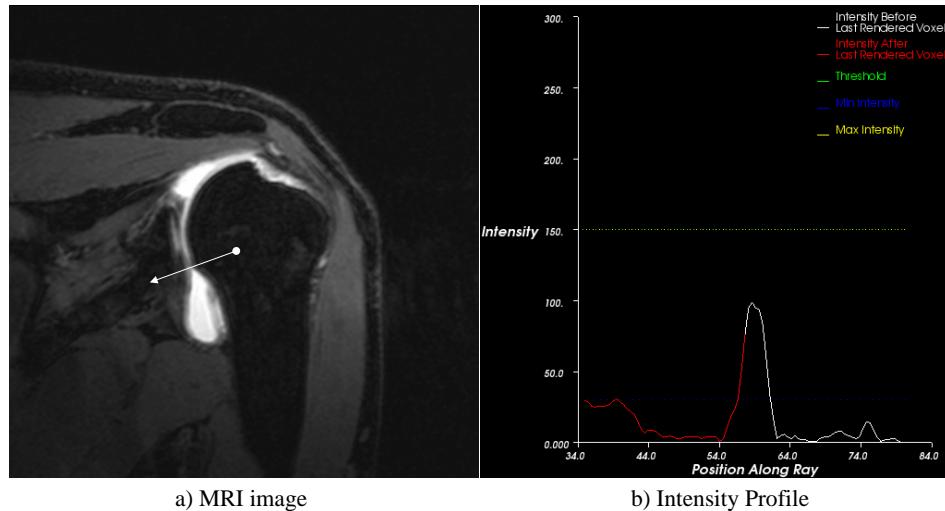


Figure 4.23: Intensity profile along a ray where the threshold is not reached.

is the key to optimal occlusion determination, rather than determining occlusion based on the definition of the surfaces rendered. The benefits of such an approach are twofold: the occlusion determination and the transfer function defining the surfaces can be independently optimized, and the viewpoint can be moved seamlessly from occluded positions to unoccluded positions while still displaying the surface of interest, as no counting of surfaces is performed.

Hole artifacts when using a fixed-threshold VDP can also occur when rendering ureters for virtual ureteroscopy. This can occur in areas where the ureter narrows, as there will be a smaller concentration of contrast. Finding the intensity peak along the ray is also desirable in this case.

4.2.2 Intensity Peak Determination Methods

4.2.2.1 Global maximum

One possible method for finding the fluid's intensity peak is to search for the global maximum value along the ray. This technique does not always work, however, as there are often cases in which multiple layers of fluid are present along a ray. If the maximum value along a ray lies in a layer behind the first, the ray will reset in that layer of fluid. This can cause unwanted removal of portions of the surface of interest.

4.2.2.2 First local maximum

Instead, what is needed is to find the first local intensity maximum corresponding to the peak of interest. In the ideal case, this can be performed by simply testing the current sample's value against the value of the next sample. If the current sample's value is greater than the next sample's value, then it represents an intensity peak. This approach finds the intensity peak from viewpoints inside the humeral head, viewpoints inside the cartilage surface of the humeral head, and viewpoints inside the joint capsule on the near side of the intensity peak. It also works from viewpoints inside the joint capsule on the far side of the true intensity peak, as the first sample will be considered an intensity peak, causing immediate resetting of the accumulated color and opacity. This algorithm therefore enables navigation from the humeral head to positions within the joint space, while always rendering the socket surface

of interest.

A major drawback to this implementation is that it is sensitive to noise. Defining a peak in this manner will cause small noise spikes to trigger the resetting of color and opacity. One method of finding the true intensity peak is to use two thresholds to select peaks of a given height and a given intensity.

4.2.2.3 Dual-threshold first local maximum

(Canny, 1986) describes a 2D edge-detection algorithm³ that uses two thresholds to link strong edges possessing high gradient magnitudes with weak edges possessing lower gradient magnitudes, removing weak edges that are not connected to strong ones.

When detecting intensity peaks along rays, two thresholds can be used to select peaks of a large enough magnitude and to eliminate spurious peaks. The algorithm works by finding the first local maximum, then continuing forward until a sample value is found that is a specified amount, determined by the first threshold parameter, below the peak. This threshold reduces sensitivity to noise spikes by finding peaks of a specified relative height. If such a value is found, the accumulated color and opacity is reset. Ray traversal is started again from the peak.

To further ensure that spurious peaks are not identified, a lower threshold value is used to eliminate peaks with absolute intensities below this value. This can be useful because the signal-to-noise ratio is lower at lower intensities, and there may also be inhomogeneities in the signal at lower intensities. The lower threshold is independent of the transfer function setting, unlike with fixed-threshold VDP, and a wide range of values for the lower threshold usually produce the same result, making it much more robust than fixed-threshold VDP. Pseudocode for this algorithm is given in Figure 4.24, and an intensity profile illustrating the two thresholds is shown in Figure 4.25.

Figure 4.26 compares rendering using the fixed-threshold VDP and the local maximum VDP algorithm. The holes are successfully filled in, however some artifacts are introduced in the area circled in red. These artifacts occur because the gradient is not well defined at the

³This algorithm is commonly referred to as Canny edge detection.

```

rayPosition ← rayBegin;
accumulatedColor ← (0.0,0.0,0.0);
remainingOpacity ← 1.0;
reachedThreshold ← false;
foundMax ← false;

while rayPosition < rayEnd do
    // Sampling
    sampleValue ← InterpolateValue(rayPosition);

    // /////////////////////////////////
    // First Local Maximum VDP
    if sampleValue >= lowThreshold then
        reachedThreshold = true;
    end
    if reachedThreshold and !foundMax then
        // Going uphill
        if sampleValue > maxValue then
            maxValue ← sampleValue;
            savedPosition ← rayPosition;
        end
        // Going downhill
        else if sampleValue <= maxValue - threshold then
            accumulatedColor ← (0.0,0.0,0.0);
            remainingOpacity ← 1.0;
            rayPosition ← savedPosition;
            foundMax ← true;
            continue;
        end
    end
    // /////////////////////////////////

    // Transfer function evaluation
    sampleColor ← ColorTransferFunction(value);
    sampleOpacity ← OpacityTransferFunction(value);

    // Shading
    sampleGradient ← InterpolateGradient(rayPosition);
    shadedColor ← DoShading(sampleColor, sampleGradient);

    // Compositing
    accumulatedColor ← accumulatedColor + shadedColor * sampleOpacity *
    remainingOpacity;
    remainingOpacity ← (1.0 – sampleOpacity) * remainingOpacity;

    // Increment position
    rayPosition ← IncrementRayPosition(rayPosition);
end

```

Figure 4.24: First local maximum VDP.

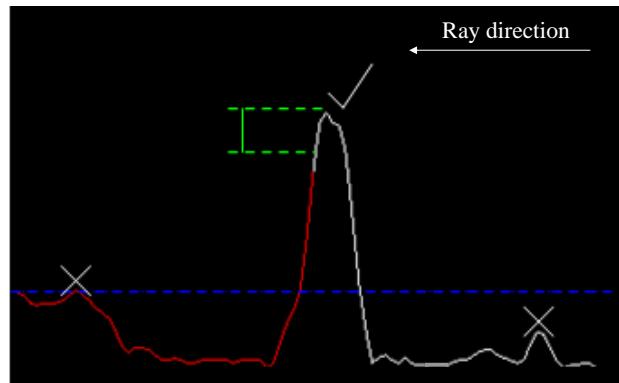


Figure 4.25: First local maximum VDP thresholds. There is one threshold, in blue, above which the maximum must lie and a second one, in green, that is the distance down from the maximum.

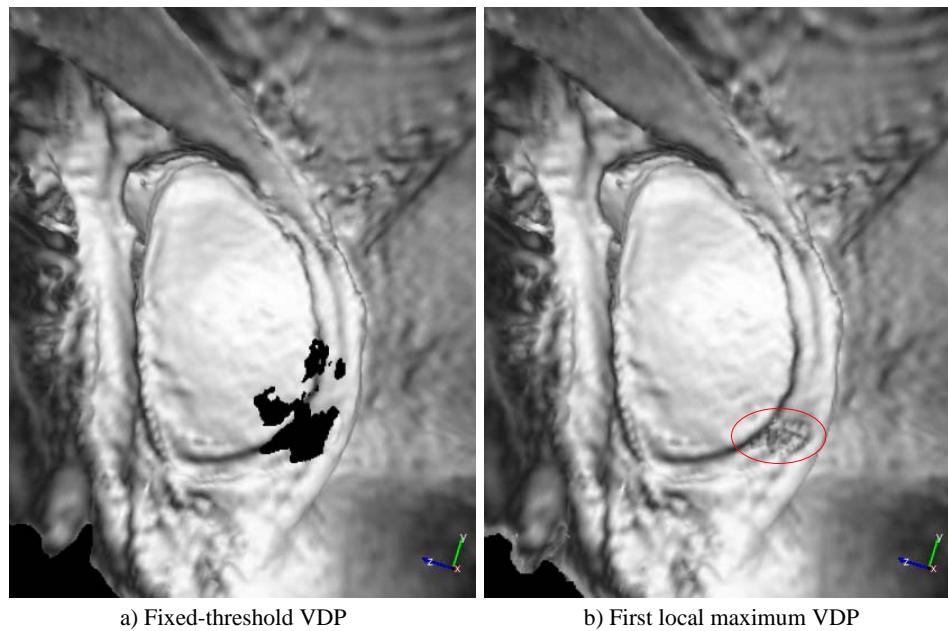


Figure 4.26: A rendering of a shoulder from MRI data using fixed-threshold VDP exhibiting hole artifacts. Using a local maximum algorithm to determine the intensity peak along the ray fills in the holes. An area with low gradient is circled in red.

peak, which results in inconsistent surface normals being generated when the peak lies within the range of values specified by the transfer function. One possible solution is to step forward a fixed amount after finding the peak before starting to accumulate color and opacity again. This may achieve nicer-looking results in some cases, but is an ad-hoc solution, as the amount to step forward is difficult to determine, and can erode the surface.

A better choice is to employ a gradient opacity transfer function that reduces contributions of voxels with small gradients (Figure 4.27). The use of gradient opacity transfer functions to

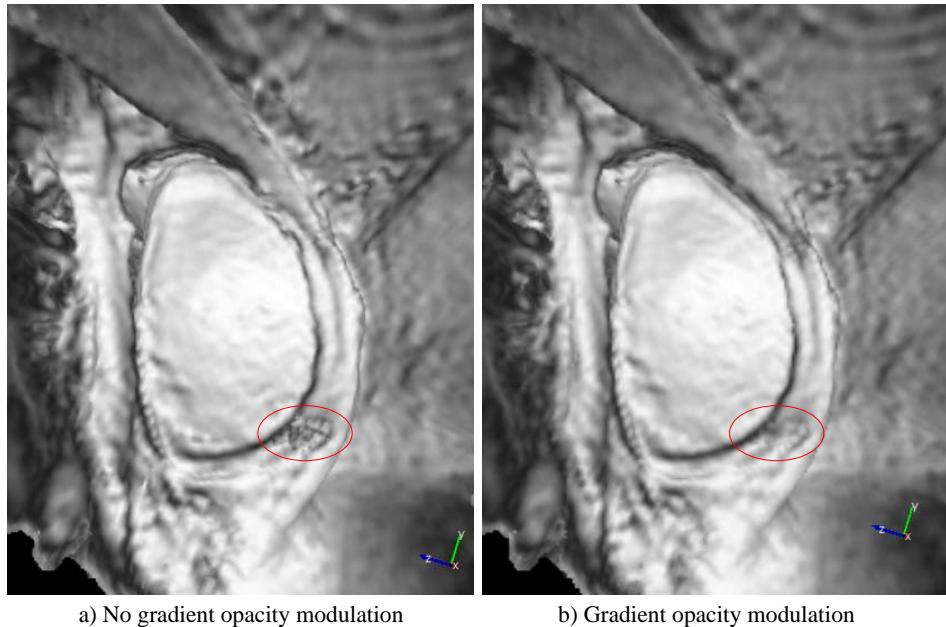


Figure 4.27: Artifacts can occur in areas of low gradient a) at the intensity peak along each ray. Using a gradient opacity transfer function b) can reduce these artifacts.

reduce the opacity of voxels with small gradients is suggested in (Levoy, 1988) to better define surfaces in the volume. This approach therefore improves the surface definition in general, while still keeping the transfer function settings separate from the occlusion determination.

Areas with a small intensity peak can represent places where the surfaces are actually touching, and are the same areas that are problematic for segmentation techniques. These areas can be highlighted in various ways by adjusting the color transfer function based on the relative or absolute intensity at the peak to convey the fact that the surface displayed here is

less certain than in other areas (Figure 4.28).

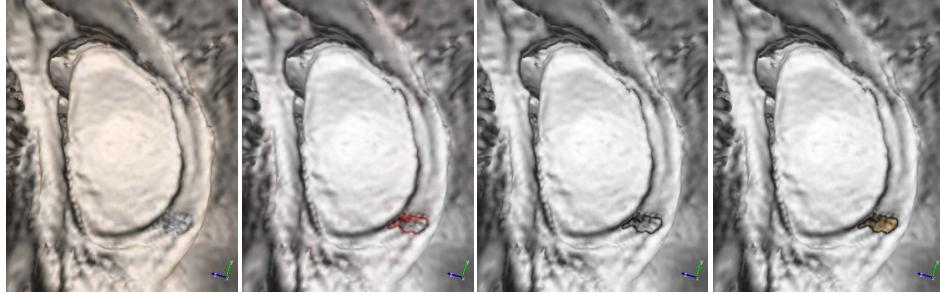


Figure 4.28: Areas where the surfaces are touching can be highlighting in various ways by adjusting the color transfer function.

The point of introducing the dual thresholds is to deal with noise in the data. However there are other methods than can be employed, both to remove noise and to introduce a notion of scale, discussed in the next section.

4.3 Noise and Filtering

The standard way of reducing noise from a signal processing standpoint is to apply a low-pass filter. A Gaussian filter is commonly used. The mathematical formula for an isotropic 2D Gaussian $g(x, y)$ is

$$g(x, y) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{x^2+y^2}{\sigma^2}\right)}, \quad (4.1)$$

where σ is the standard deviation of the Gaussian. The Gaussian is very similar to the ideal smoothing filter for 2D edge detection described in (Canny, 1986), and is often used in practice as a replacement. In Canny edge detection, the original image is smoothed, a simple edge detector is applied, and then intensity peaks representing edges are detected. FOR performs a similar task along each ray, but without the edge detection step.

The Fourier transform of a Gaussian is another Gaussian, but with an inverse standard deviation. Filtering by multiplying with a Gaussian in the frequency domain is therefore

equivalent to convolution with a different Gaussian in the spatial domain. The σ of the Gaussian convolution kernel controls its width, and therefore the amount of amount of smoothing performed on the data. Smoothing to remove noise also smoothes other high frequency data, such as small bumps and edges. Therefore, simply smoothing the data to remove noise and then rendering the smoothed data it is not optimal, as this will smooth away small features in the data that are of interest to the physician (Figure 4.29).

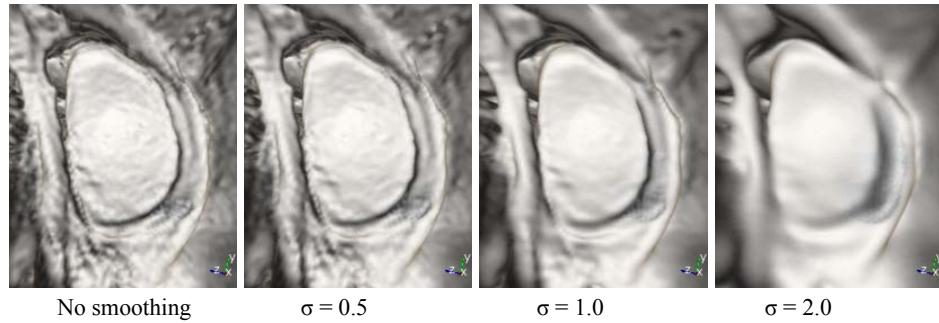


Figure 4.29: Renderings of a shoulder socket from MRI data. The standard deviation of the Gaussian kernel used to smooth the data increases from left to right.

Although using smoothed data is not desirable for rendering, it has two properties that are desirable for determining occlusion. The first is noise removal for finding intensity peaks representing the separation between surfaces. By smoothing, the average over a local 3D neighborhood at each voxel is taken into account, rather than just along a 1D ray. The second desirable feature is control over the scale of objects being separated. For example, with a bone fracture, it might be desirable to skip over multiple small fragments to see the surface beyond them.

FOR takes advantage of the fact that the best data set used for rendering might not be the best data set used for determining occlusion. FOR uses the smoothed data set to locate separating surfaces, while using the original data set to render.

4.4 Flexible Occlusion Rendering

With FOR, an auxilliary data set is used for determining occlusion, while rendering the original data. The VDP methods discussed both fit into the FOR framework. The more general term FOR will be used instead of VDP for the remainder of the dissertation.

4.4.1 Fixed-threshold FOR

Fixed-threshold VDP can be thought of as a specific instance of FOR that uses a thresholded data set for occlusion determination⁴. In this case, the test to perform for resetting along the ray becomes a simple test, as the thresholded data set is a binary mask. Because the thresholding operation is so inexpensive, it is easier to perform the computation while rendering, without computing a separate thresholded dataset⁵.

4.4.2 First local maximum FOR

First local maximum VDP does not use a smoothed version of the original data set. First local maximum FOR uses a smoothed version of the original data set at some scale. The local maximum peak-finding algorithm is performed on the smoothed data set. If the scale desired is that of the original data, no auxilliary data set needs to be generated, although it is possible to always use a secondary data set for simplicity of implementation⁶.

These two data sets are useful for virtual arthroscopy from MRI data and virtual ureteroscopy from CT data. A different but related type of auxilliary data set is necessary for the third application presented in this dissertation, knee fracture visualization from CT data.

⁴The implementation of fixed-threshold VDP described previously is not exactly the same as using a thresholded dataset, as the thresholding operation is being performed on a tri-linearly interpolated value along the ray, not on the actual voxel values. Using nearest-neighbor interpolation for the threshold would have the same result as a thresholded dataset.

⁵There may be some systems in which memory access is cheap enough that reading from a thresholded dataset and doing a binary comparison will be cheaper than performing a comparison on the sample value and the threshold value.

⁶There may be some performance decrease when implemented as an auxilliary data set, as two data values must be generated for each sample point

4.4.3 Knee fracture visualization

For diagnosing knee fractures, CT is often used, as it gives excellent bone contrast. An idealized diagram of a knee as seen using CT is shown in figure 4.30. The hard outer surface

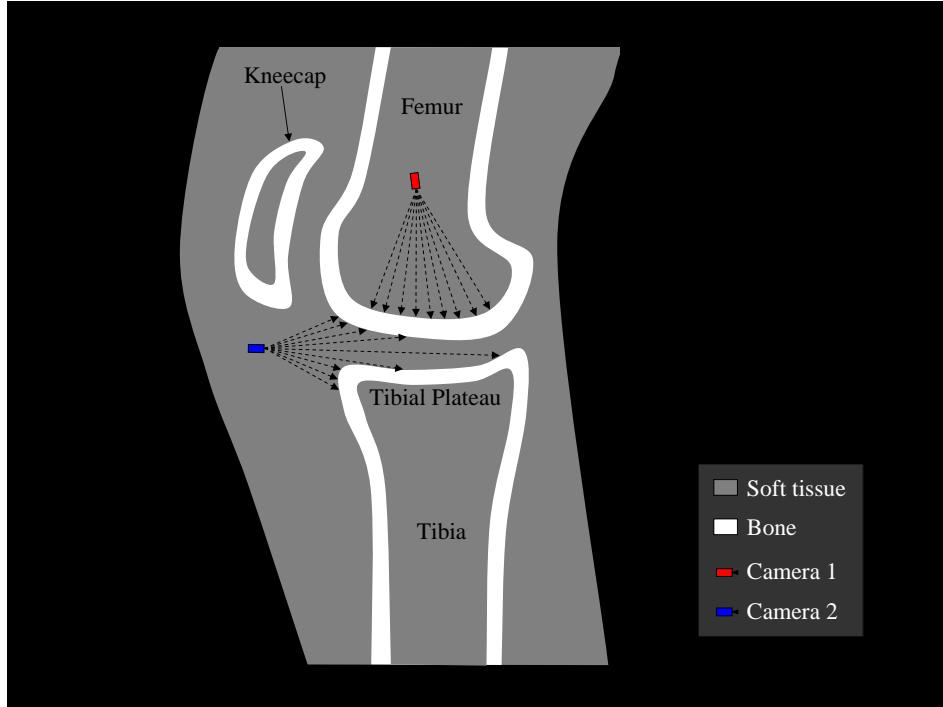


Figure 4.30: Diagram of a sagittal slice of a knee CT, showing the desired viewpoint for a tibial plateau fracture (Camera 1), and the types of viewpoints possible with standard volume rendering (Camera 2).

of the bones in the joint have a very high intensity, whereas the inside of the bone has a lower intensity, in the same range as that of the soft tissue and fluid present in the joint.

The most common type of knee fracture diagnosed using CT data is a tibial plateau fracture. Figure 4.30, Camera 1 shows the view most useful for diagnosing such a fracture. The occluding material of the femur makes such views impossible with standard volume rendering. Figure 4.30, Camera 2 shows a view available with standard volume rendering.

4.4.3.1 Problems with FOR

Because the space between the surfaces of the joint has a lower range of intensities than that of the bone surfaces to be rendered, a possible choice for a data set to define the separation used to determine occlusion using FOR is to apply a lower threshold to the data. In this

case, all values below the threshold are set to 1, and all values above the threshold are set to 0. Defining the thresholding in this manner inverts the intensities, making the separation the highest intensity, as with virtual arthroscopy and virtual ureteroscopy. This approach only works for viewpoints just within the hard bone surface of the femur (Figure 4.31, Camera 2), which is better than standard volume rendering, but far from optimal. For viewpoints on the

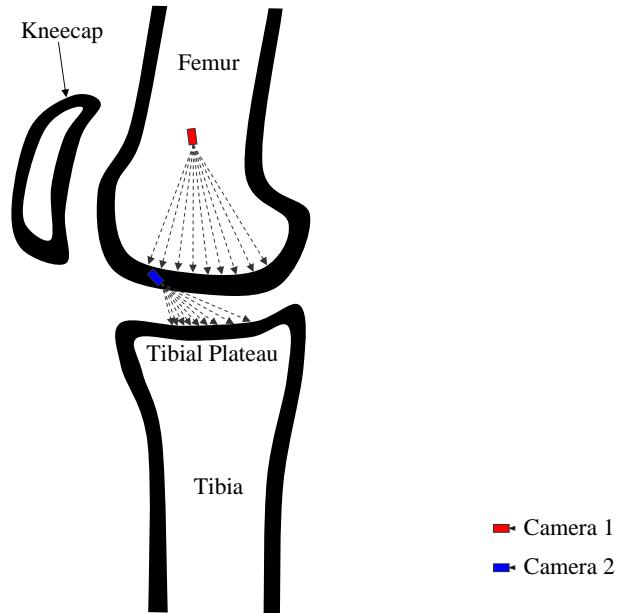


Figure 4.31: Diagram of a thresholded data set for knee fracture visualization. Only views from within the hard bone surface of the femur (Camera 2) are possible.

inside of the bone, the accumulated color and opacity is immediately reset, and the inside surface of the femur is rendered, giving the same result as standard volume rendering (Figure 4.31, Camera 1). The same problem occurs if attempting to use the first local maximum FOR algorithm.

4.4.3.2 Flood-filled mask

Performing a flood fill operation on the thresholded data enables FOR to correctly determine the separation used to reset accumulated color and opacity (Figure 4.32). Before performing the flood fill, an axial slice of pixels set to 0 is added above and below the volume.

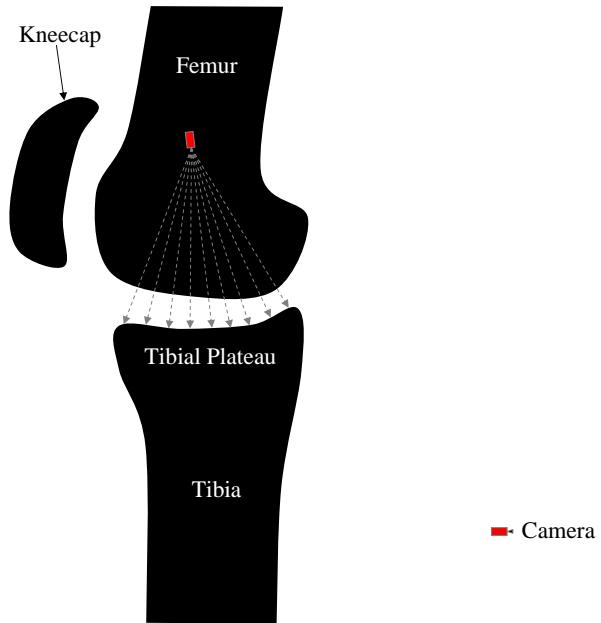


Figure 4.32: Diagram of a thresholded and flood-filled data set for knee fracture visualization. Views from within the femur show the surface of the tibial plateau.

This ensures that the interiors of the bones, which have been thresholded to a value of 1, are closed off from the edges of the image. The flood fill operation then sets to 0 all voxels with value 1 that are not connected to the edge of the image by other voxels with value 1. In this manner, the interior of the bones is filled in, resulting in a mask of the bones and the separation between the bones.

With the flood-filled mask, FOR can be employed to reset the accumulated color and opacity whenever a 1 is encountered along the ray in the mask, which represents the separation between the surfaces. This enables seamless transitions from viewpoints exterior to the bones being rendered to viewpoints within the bones being rendered, while always displaying the far surface of interest.

A problem similar to that encountered when using fixed-threshold VDP arises with this binary mask. The threshold must be tightly coupled to the transfer function used to render the surfaces because the resetting occurs immediately when entering the separation mask. Overestimating the separation by setting the threshold too high can cause the resetting to

happen early, which causes the near surface to be rendered (Figure 4.33). Conversely, under-

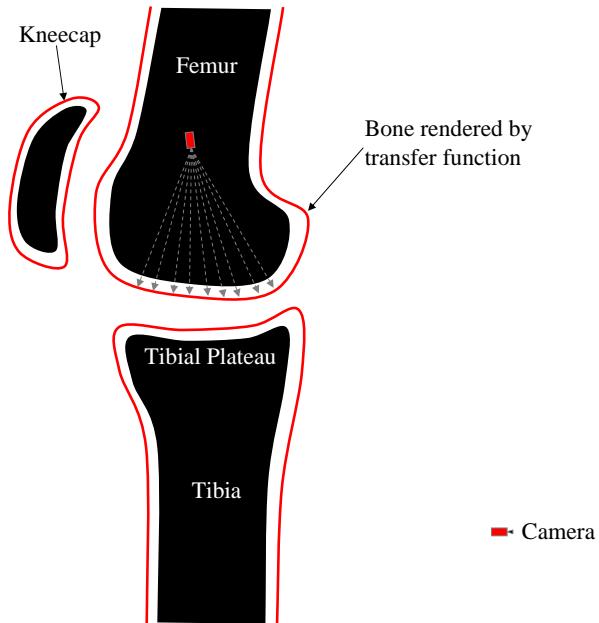


Figure 4.33: Overestimating the separation by setting the threshold too high can cause the resetting to happen early, which causes the near surface to be rendered.

estimating the separation by setting the threshold too low can cause hole artifacts (Figure 4.34). This is especially problematic because the thresholding and flood filling is performed as a preprocess, and cannot be recomputed easily.

4.4.3.3 Flood-filled mask at scale

A more robust method for finding the separation incorporates thresholding, flood-filling, and smoothing. By smoothing the thresholded and flood-filled data, intensity peaks are formed near the center of the separation (Figure 4.35).

This process can be seen as an approximation of a medial axis transform of the separation. The first local maximum method can therefore be used to determine intensity peaks for resetting. This method is more robust to an overestimate of the separation, as it will reset towards the center of the separation. It can also help with small holes, as smoothing the edges of the hole will create small intensity peaks that can be found with a low threshold setting.

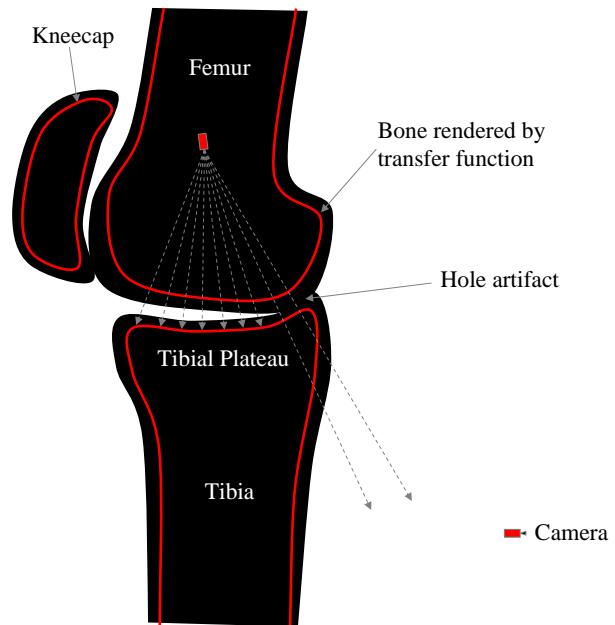


Figure 4.34: Underestimating the separation by setting the threshold too low can hole artifacts.

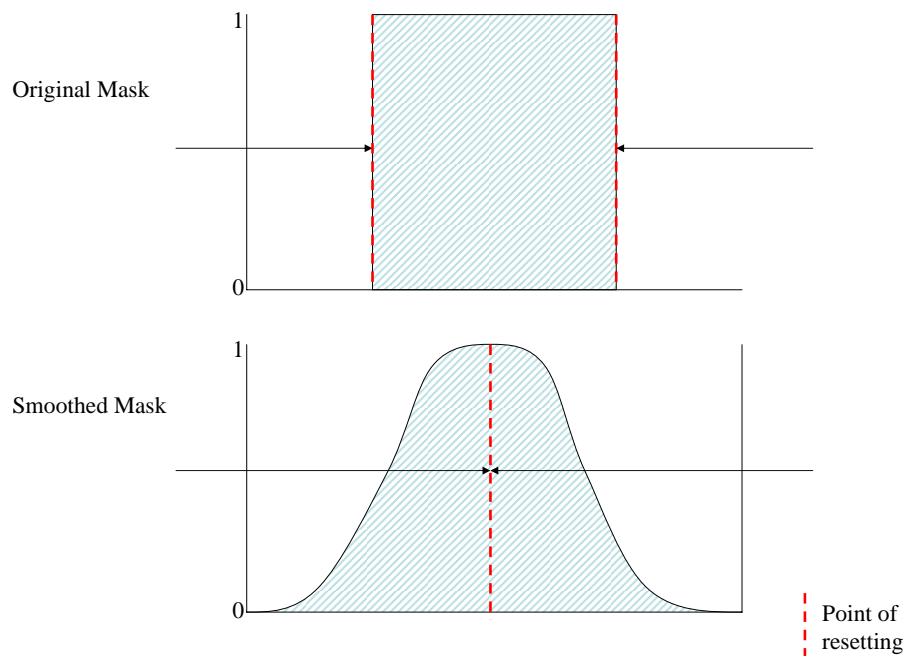


Figure 4.35: Smoothing the mask forms intensity peaks.

Results of applying FOR in this manner are given in Chapter 7.

4.5 Recap

The key insights and features in FOR are summarized as follows:

- Finding separating features between objects, rather than the boundaries of the objects themselves, enables more robust determination of undesired occlusions.
- The location of these separating features in data sets that have noise and partial-volume effects can be performed robustly by finding intensity peaks of a minimum height and a minimum intensity.
- These features can be found during the rendering process, without a need for pre-segmentation.
- These features can be found using the same dataset used for rendering, or a separate dataset optimized for occlusion determination.
- Finding the separation enables seamless transition from occluded viewpoints to occluding viewpoints.
- The scale of the dataset used for feature-detection can be adjusted without altering the resolution of the data being rendered.
- For the applications presented in this dissertation finding the first separating feature has proven useful, however finding multiple separations or the last separation, are also possible, depending on the application.

CHAPTER 5

VIRTUAL ARTHROSCOPY

This chapter presents the results of using FOR of MRI data to perform diagnosis of joint pathology, technically referred to as internal derangement of the joint, via virtual arthroscopy. An overview of joint anatomy is given, with an emphasis on the shoulder joint, the most common joint used in this study. A presentation of standard techniques used to diagnose internal joint derangements is then given, including diagnostic clinical arthroscopy, MRI slice viewing, and virtual arthroscopy using standard volume rendering, before presenting virtual arthroscopy using FOR.

5.1 Joint Anatomy

In human anatomy, a joint is a location where two or more bones make contact. Joints in which there exists a space between the cartilage-covered surfaces of the bone are called synovial joints. A thin synovial membrane encloses the joint, secreting synovial fluid that helps lubricate joint articulation. The major synovial joints in the human body are the ankle, knee, hip, wrist, elbow, and shoulder. Common components of synovial joints include the following:

- Bones.
- Articular cartilage, formed on the contact surfaces of bones.
- The synovial capsule, a collagenous structure enclosing the joint.
- The synovial membrane, which lines the synovial capsule and secretes synovial fluid.
- Synovial fluid, which lubricates the joint.

- Tendons, which connect bones to muscle.
- Ligaments, which connect bones to other bones.

In this dissertation, the shoulder has received the most attention for the purposes of virtual arthroscopy using FOR.

5.1.1 The Shoulder

Two of the major bones of the shoulder are the humerus, or upper arm, and the scapula, or shoulder blade. The main shoulder articulation is a ball-and-socket joint, in which the humeral head is the ball, and the glenoid cavity of the scapula is the socket (Figure 5.1). The humeral head and glenoid are covered with articular cartilage¹ to provide cushioning.

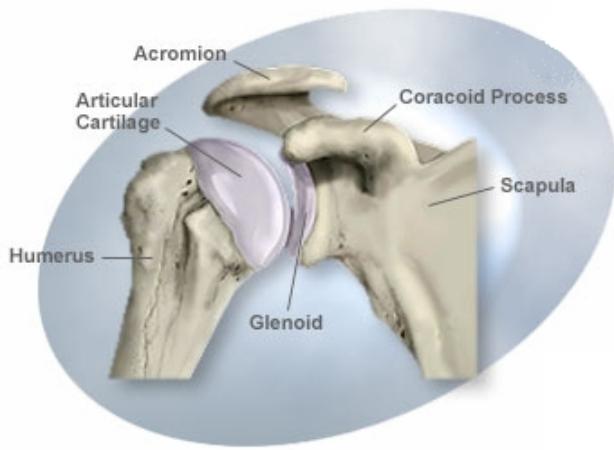


Figure 5.1: Anterior view of the interior of a shoulder, demonstrating the ball and socket (SCOI, 2007).

Articular cartilage has an extremely low coefficient of friction, allowing the ball to rotate smoothly in the socket. The joint capsule surrounds the ball-and-socket joint, protecting it and containing the synovial fluid that lubricates the joint (Figure 5.2). The glenoid is a relatively shallow socket, with the humeral head being held in place by various ligaments and tendons. A cartilage ring around the glenoid called the labrum acts like a gasket, helping to stabilize the humeral head while still allowing a large range of motion. The labrum is made

¹Articular cartilage is also referred to as hyalan cartilage.

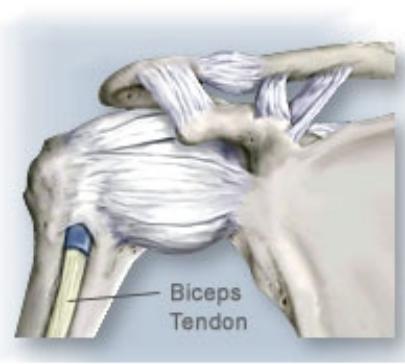


Figure 5.2: Anterior view of a shoulder, showing the joint capsule surrounding the joint (SCOI, 2007).

of fibrocartilage, which is more flexible and tough than articular cartilage, but has a higher coefficient of friction. Figure 5.3 shows a lateral view of the shoulder socket with the humeral head removed to reveal the glenoid and surrounding tissue. Injuries, or internal derangements, of the shoulder are common, including damage to the articular cartilage surfaces, and various tears of the labrum and ligaments. Some common derangements are described below.

5.1.2 Common Derangements

5.1.2.1 Bankart lesion

A Bankart lesion is a tear of the labrum which may be caused during an anterior shoulder dislocation. As the shoulder pops out of joint, it can tear a part of the anteroinferior labrum. This injury can result in a sense of instability, catching sensations, repeated dislocations of the shoulder, and pain. Tears of other regions of the labrum and associated ligaments may occur as well.

5.1.2.2 Hill-Sachs lesion

Another derangement that can occur during an anterior shoulder dislocation is a Hill-Sachs lesion, which often accompanies a Bankart lesion. A Hill-Sachs lesion is an impaction fracture of the humeral head surface that occurs when the humeral head strikes the anterior edge of the glenoid during dislocation. Both the articular cartilage surface and the underlying bone

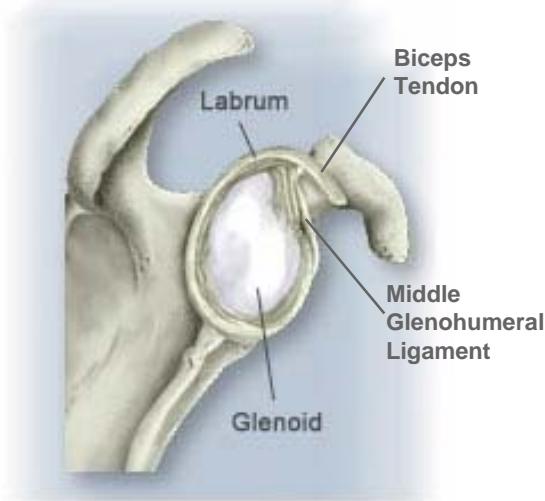


Figure 5.3: Lateral view of the glenoid and surrounding tissue. The humeral head is removed to enable such a view (SCOI, 2007).

are damaged in such an injury.

5.1.2.3 SLAP tear

A superior labrum from anterior to posterior (SLAP) tear occurs in the area near where the biceps tendon connects to the labrum. Such tears can result in a gap between the labrum and the glenoid. Contrast-enhanced MRI is useful for diagnosing such tears, as the contrast agent will highlight the gap.

5.2 Standard Joint Derangement Diagnosis

The first stage in diagnosing internal joint derangements is through a clinical exam, in which the physician forms a diagnosis based on patient history and a physical exam. Once the clinical exam is complete, it may be necessary to employ other techniques to confirm or deny the physician's diagnosis.

5.2.1 Diagnostic Clinical Arthroscopy

Arthroscopy is the process of inserting a small fiber-optic camera called an arthroscope into a joint through a small incision in the skin, enabling viewing of the interior of the joint (Figure 5.4). Arthroscopy is useful for diagnosis of derangements within joints and for performing

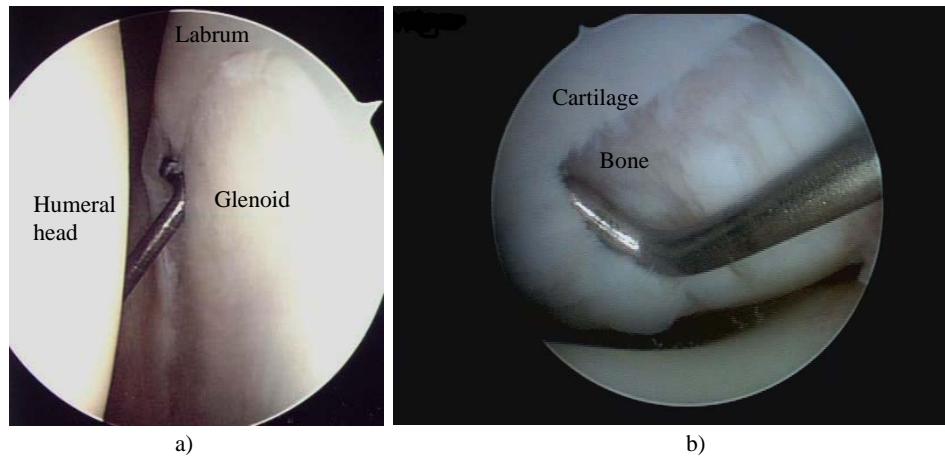


Figure 5.4: Shoulder arthroscopy a), showing the humeral head, glenoid, and labrum (Duggan, 2007), and arthroscopy of the author’s knee b) showing a tear of the articular cartilage surface of the femur. Arthroscopic probes used to inspect the joints are visible.

surgery to treat such internal derangements. The advantages of arthroscopy are that the physician can obtain very high-resolution images of the joint surfaces, and can use arthroscopic probes to push and pull on structures to get tactile feedback. The disadvantages of arthroscopy are that it is an expensive, invasive surgical procedure requiring skilful manipulation of arthroscopic tools confined within the cramped area between the joint surfaces called the joint space. For pre- and post-operative diagnosis of joint derangements, it is less invasive to view MRI images of the joint².

5.2.2 Magnetic Resonance Imaging

MRI images showing intensity variation based on tissue type, as detailed in Chapter 2, provide an alternate, less-invasive, method to display joint surfaces. By viewing slices of MRI

²Contrast-enhanced CT may also be used.

data along the three orthogonal axes (Figure 5.5), clinicians can diagnose joint pathologies without the use of invasive arthroscopic procedures. A fluid containing an imaging contrast

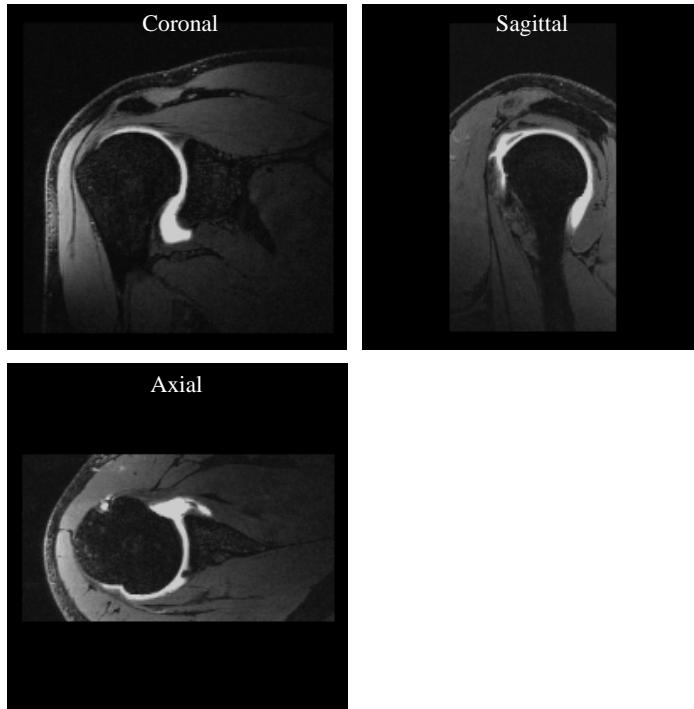


Figure 5.5: Three orthogonal slices of a shoulder MRI.

agent is often injected into the joint to aid in diagnosis. The contrast agent improves contrast between tissue boundaries and helps increase separation between the surfaces in the joint. In MRI imaging, a gadolinium-based contrast agent is usually used.

5.2.2.1 MRI sequences used

Multiple MRI sequences are usually acquired to aid in diagnosis, including T1- and T2-weighted sequences. A sequence routinely acquired by colleagues at UNC Radiology that is of special importance for virtual arthroscopy is a contrast-enhanced water excitation dual echo in the steady state (WEDESS) sequence.

WEDESS results in suppression of fat signal, yielding intense signal from long T2 substances like water (Clarke, 2006). Its benefits for both MRI slice viewing and virtual arthroscopy are that it can obtain isotropic voxels with a high spatial resolution of about 0.6 mm while still retaining a high signal-to-noise ratio, and that the injected contrast agent appears

very bright, giving good contrast between it and the tissue in the joint.

5.2.2.2 Advantages

MRI slice viewing is less invasive and less expensive than arthroscopy, only requiring the patient to undergo an MRI scan lasting about 30 minutes, rather than a surgical procedure. Another advantage MRI has over arthroscopy is that an entire volume is acquired, so the physician has information about what is behind the surfaces visible in arthroscopy.

5.2.2.3 Disadvantages

Although joint pathologies can be determined from MRI slice viewing, diagnosis requires tedious manual selection of slices, along with a mental reconstruction of 3D structures in the joint by correlating features between the three slices. Because an understanding of the 3D shape of the surfaces in the volume is helpful in understanding the joint anatomy, displaying these surfaces to the physician is beneficial for quick and accurate diagnosis of joint derangements. Virtual arthroscopy enables real-time evaluation of joint surfaces in 3D, removing the tedious slice manipulation and 3D mental reconstruction required with MRI slice viewing.

5.2.3 Standard Virtual Arthroscopy

Virtual arthroscopy is the 3D rendering of joint surfaces from MRI or CT data sets for diagnosis of joint pathologies (Applegate, 1998; Weishaupt et al., 1999; Lee et al., 2004; Sahin et al., 2004). Commercial systems such as the Siemens Syngo software running on the Leonardo workstation use standard volume rendering to enable virtual arthroscopy fly-throughs. With this flythrough system, the user can position the virtual camera in the joint space and move to obtain views similar to those obtained in clinical arthroscopy (Figure 5.6).

5.2.3.1 Advantages

Standard virtual arthroscopy is no more invasive than MRI slice viewing, as it uses MRI data for rendering, and is less invasive than clinical diagnostic arthroscopy. Because it

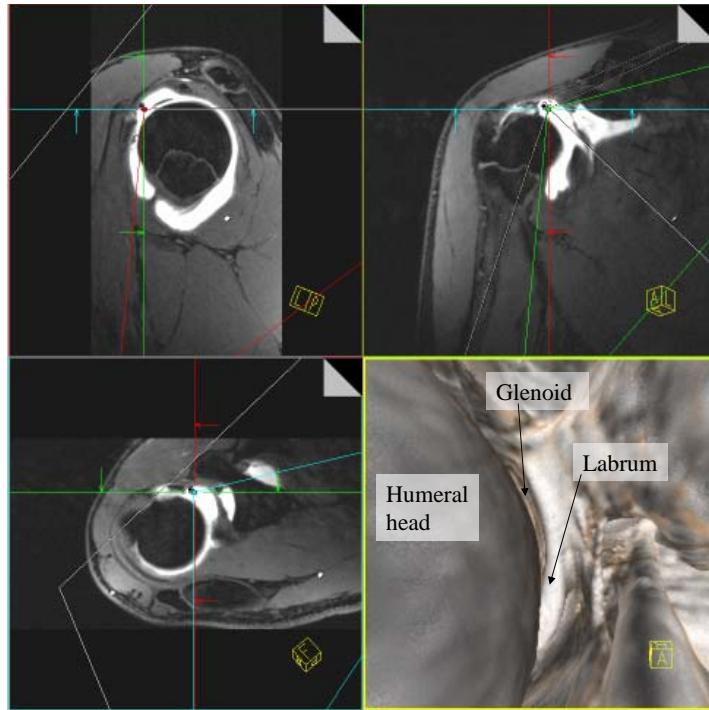


Figure 5.6: Shoulder virtual arthroscopy flythrough from MRI data using Siemens Syngo software, illustrating a constrained viewpoint.

displays the surfaces in the joint directly in 3D, it is a more direct and intuitive way of understanding these surfaces than MRI slice viewing.

5.2.3.2 Disadvantages

Although it is less invasive and requires no surgery, virtual arthroscopy shares with clinical diagnostic arthroscopy the limitation that the virtual camera is confined to viewpoints within the joint space. The transfer function of the volume rendering is set to render voxels within the range of values of the surfaces in the joint, so placing the camera outside of the joint space results in the surfaces occluding each other. Because of this limitation, the physician must evaluate the joint surfaces from viewpoints at extreme angles, often very close to the surface. This process is much like evaluating the surface of the hand with the palm touching the tip of the nose (Clarke, 2004), and necessitates tedious maneuvering of the virtual camera to find the best available positions from which to evaluate the surface. Thus, a mental reconstruction of the surface from these limited viewpoints is required. Standard virtual arthroscopy thus

adds the burden of navigating the joint to the task of evaluating the images for derangement.

Flexible occlusion rendering (FOR) was developed at the request of radiologists to enable unconstrained placement of the virtual camera for unobstructed views of entire areas of interest in a joint by automatically culling occluding voxels between the viewpoint and the far side of the joint capsule. FOR provides views resembling textbook cutaways, enabling rapid surveying of the joint for pathology (Figure 5.7).

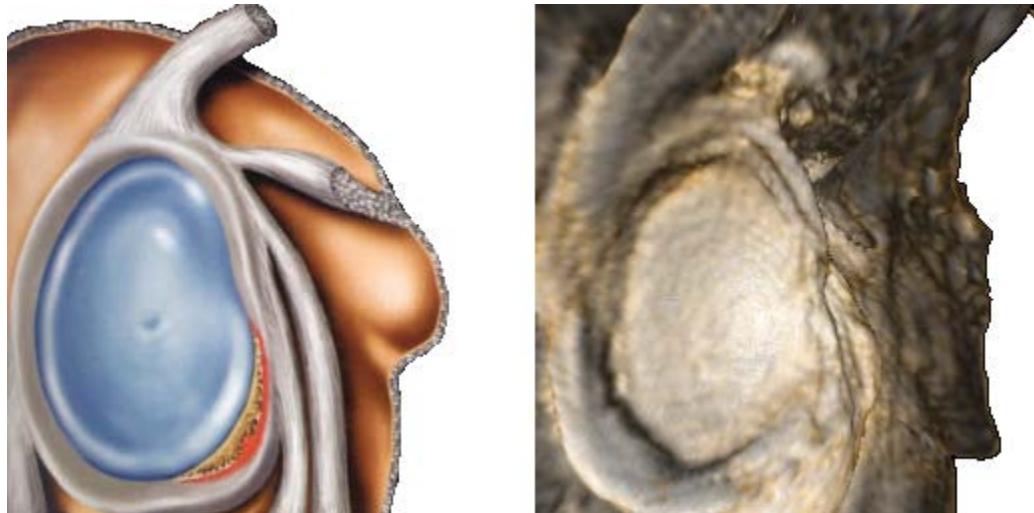


Figure 5.7: Cutaway illustration of a shoulder socket from a textbook (Stoller et al., 2004) compared with FOR used to render a shoulder socket from MRI data.

5.3 Flexible Occlusion Rendering for Virtual Arthroscopy

5.3.1 Shoulder virtual arthroscopy

Using the FOR techniques described in Chapter 4, useful views of the shoulder joint for performing diagnosis of internal derangements can be rendered that are not possible using standard volume rendering. Figure 5.8 shows a survey view of a shoulder socket that has been injured during a dislocation. This view is created by placing the virtual camera inside the humeral head, using FOR to remove the occluding material of the humeral head. A view such as this is useful for performing an initial survey of the shoulder socket to check for internal derangements, and is not possible using standard volume rendering, as the surface of the humeral head will occlude the shoulder socket from such a viewpoint. In this case,

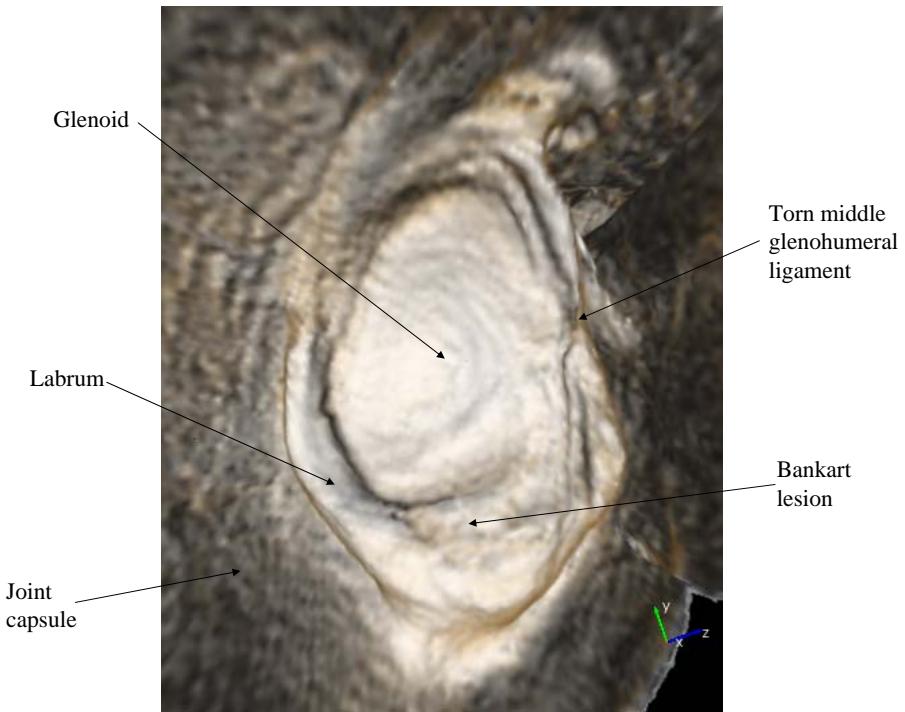


Figure 5.8: FOR survey view of a shoulder socket injured during a dislocation.

the physician is immediately able to diagnose a torn middle glenohumeral ligament and a Bankart lesion.

5.3.1.1 Torn middle glenohumeral ligament

Figure 5.9 shows closer views of the torn ligament. On the left is a view created using standard volume rendering from within the joint capsule, with the torn ligament circled in red. Because the virtual camera is constrained to viewpoints within the joint capsule, the tear must be viewed at an extreme angle very close to the tear, making it difficult to find and diagnose. On the right is a view created using FOR, with the same tear circled in red. In this view it is much easier to see the torn ligament and place it in context with neighboring anatomical structures.

Figure 5.10 shows closer views of the Bankart lesion, with the lesion circled in red.

5.3.1.2 Bankart lesion

Figure 5.10 shows closer views of the Bankart lesion. On the left are views created using

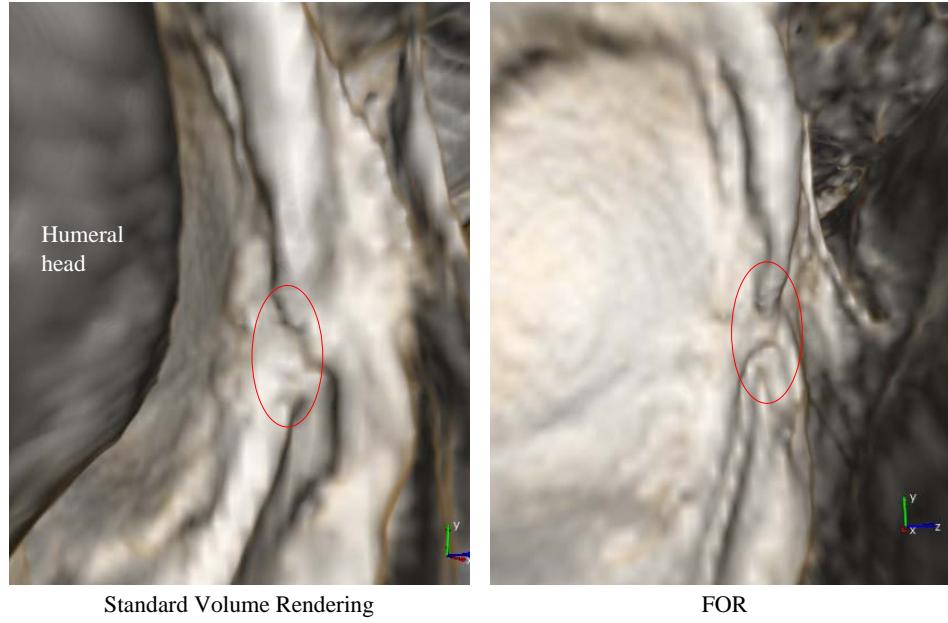


Figure 5.9: Comparison of ligament tear views, with the tear circled in red.

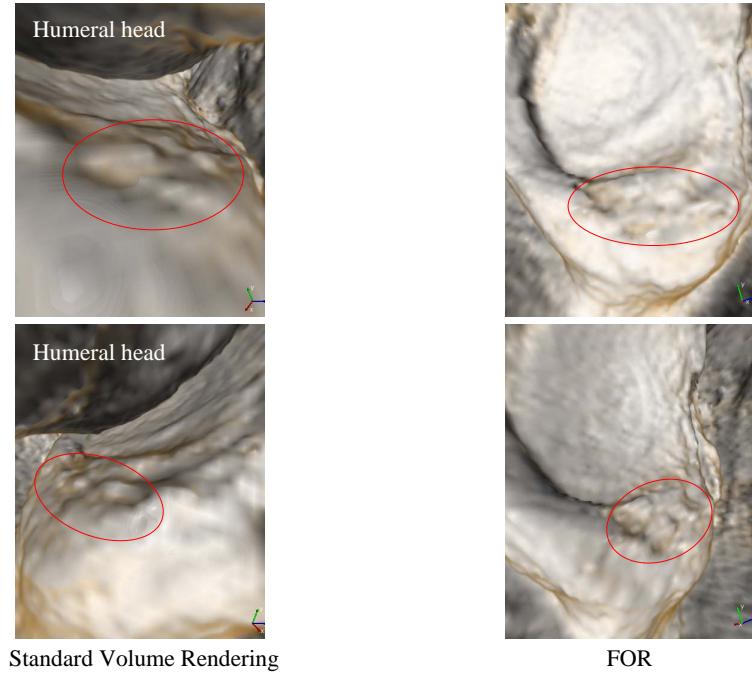


Figure 5.10: Comparison of Bankart lesion views, with the lesion circled in red.

standard volume rendering. As with the view of the torn ligament, the camera is forced to be either at an extreme angle, or extremely close to the lesion. On the right are views created with FOR, enabling unconstrained viewpoints to obtain more revealing views of the lesion.

5.3.1.3 Hill-Sachs lesion

Moving into the scapula and turning around gives a survey view of the humeral head when using FOR to remove the occluding material of the glenoid and labrum, a view not possible with standard volume rendering (Figure 5.11).

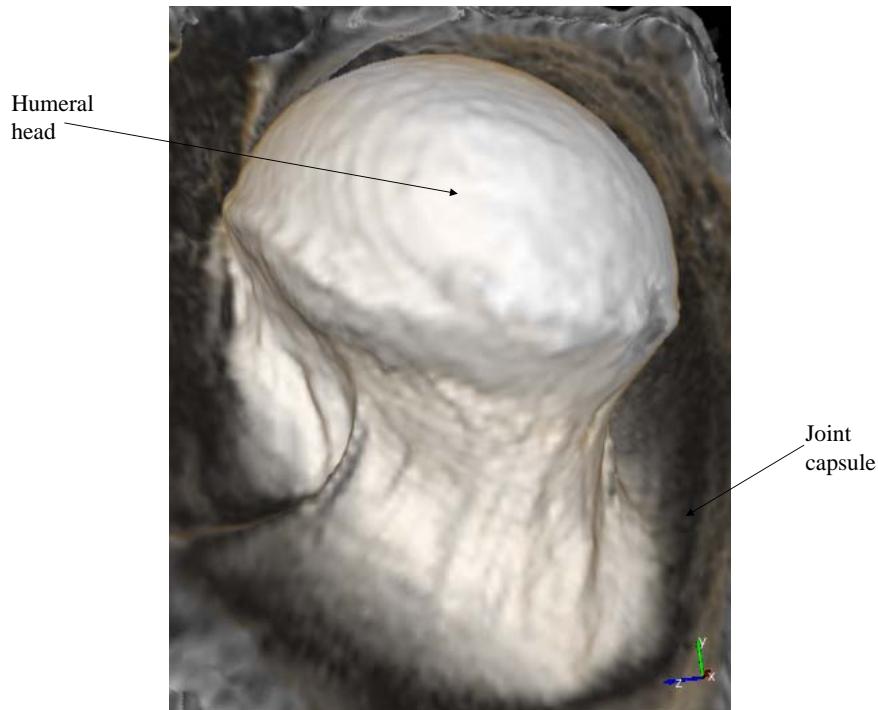


Figure 5.11: View of the humeral head from within the scapula using FOR.

Rotating around the humeral head to a different position reveals a Hill-Sachs lesion (Figure 5.12). On the left is a view created using standard volume rendering from within the joint capsule. The lesion, circled in red, is very hard to distinguish because the viewpoint is so close to it, and the lesion is so large and shallow. The lesion is also hard to find using standard volume rendering, as the physician has to navigate across the entire surface of the humeral head for inspection. On the right is a view created using FOR, with the Bankart lesion circled in red. This view is easy to obtain using FOR, as the physician can rotate around the whole

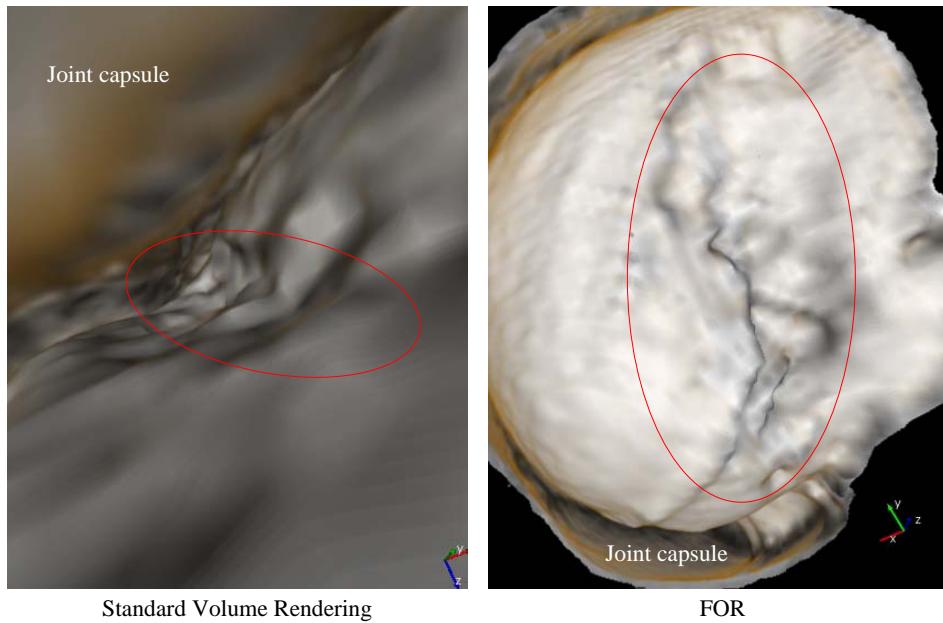


Figure 5.12: Comparison of Hills-Sachs lesion views, with the lesion circled in red.

humeral head maintaining a large field of view at all times. The view of the Bankart lesion generated by FOR gives a much better appreciation of its magnitude than views generated with standard volume rendering.

This view is also superior to using MRI slice viewing for diagnosis. An orthopedist colleague was shown MRI slices of this data, followed by a short movie of the lesion generated using FOR. He stated that, were it his patient, he would change his surgical approach based on the improved 3D perspective available using FOR versus the information gained from MRI slice viewing.

5.3.1.4 SLAP tear

Figure 5.13 shows a survey view of a different shoulder. The large gap between the upper-left portion of the glenoid and the labrum is a SLAP tear, and is immediately recognizable from this survey view.

Figure 5.14 shows closer views of the SLAP tear. The top row illustrates the utility of FOR verus standard volume rendering, as the physician can get a much better appreciation

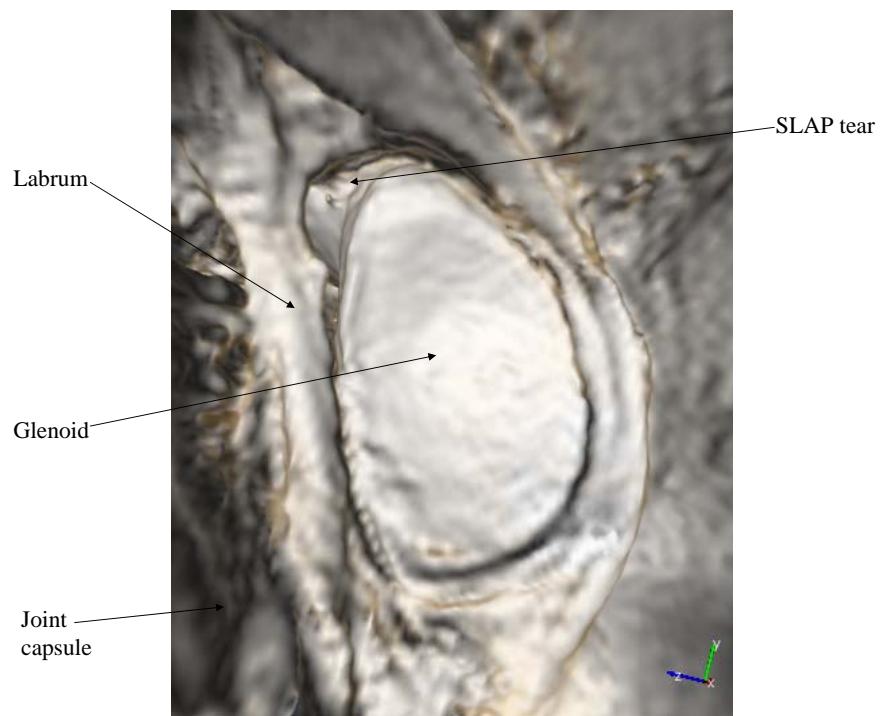


Figure 5.13: FOR survey view of a shoulder socket exhibiting a SLAP tear.

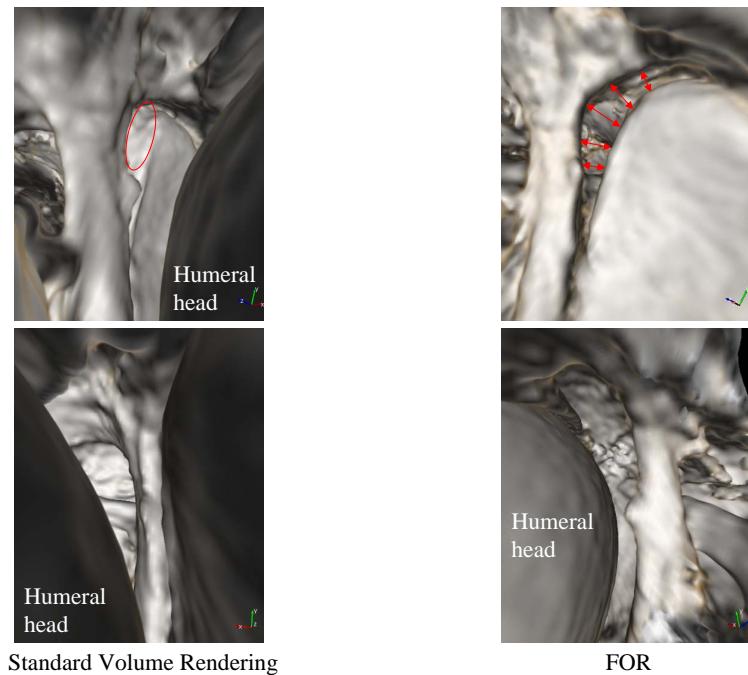


Figure 5.14: Comparison of SLAP tear views, with the tear indicated in red.

of the size and shape of the tear using FOR, versus the view from an extreme angle forced by being constrained to the joint space. The bottom row demonstrates how FOR can be used to obtain views of the inner surface of the torn labrum from viewpoints within the scapula, as well as from viewpoints within the humeral head, whereas standard volume rendering is confined to viewpoints within the joint space.

5.3.1.5 Multiple shoulders

Figure 5.15 shows survey views of 15 shoulder sockets rendered using FOR from MRI data. In all cases, the occluding material of the humeral head is successfully removed to display the

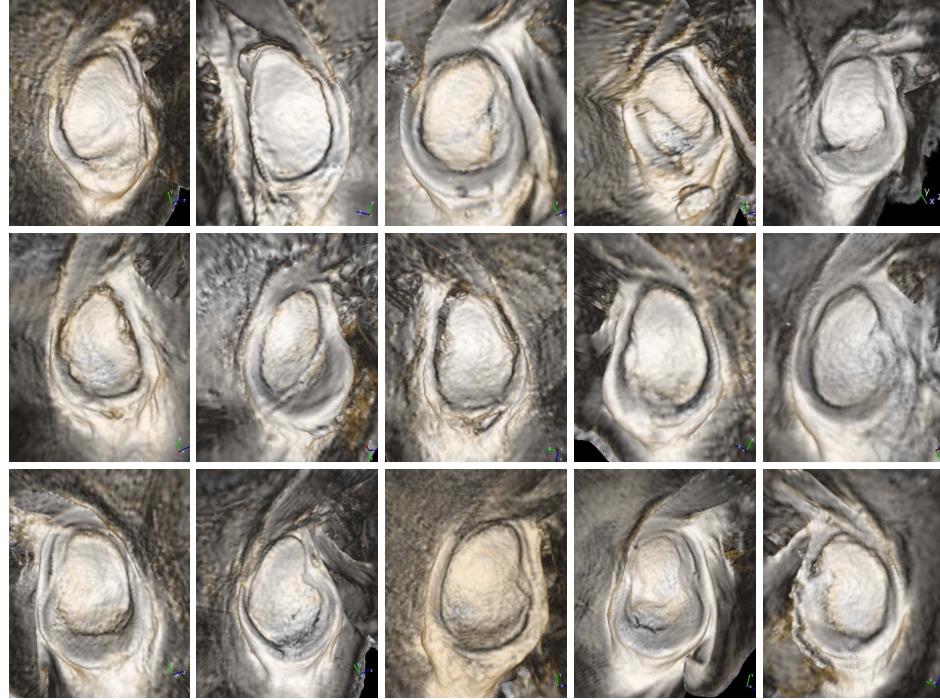


Figure 5.15: Multiple shoulders rendered using FOR.

shoulder socket surface.

5.3.1.6 Selecting scale

Using FOR, the scale of the features used to determine occlusion can be selected. This is helpful for reducing noise in the occlusion determination, removal of small structures, and reduction of rendering unwanted grazing edges. Figure 5.16 shows the result of using

a larger smoothing of the occlusion data to estimate portions of the socket rendered from a viewpoint within the scapula. Note that the surface of the humeral head is rendered at the

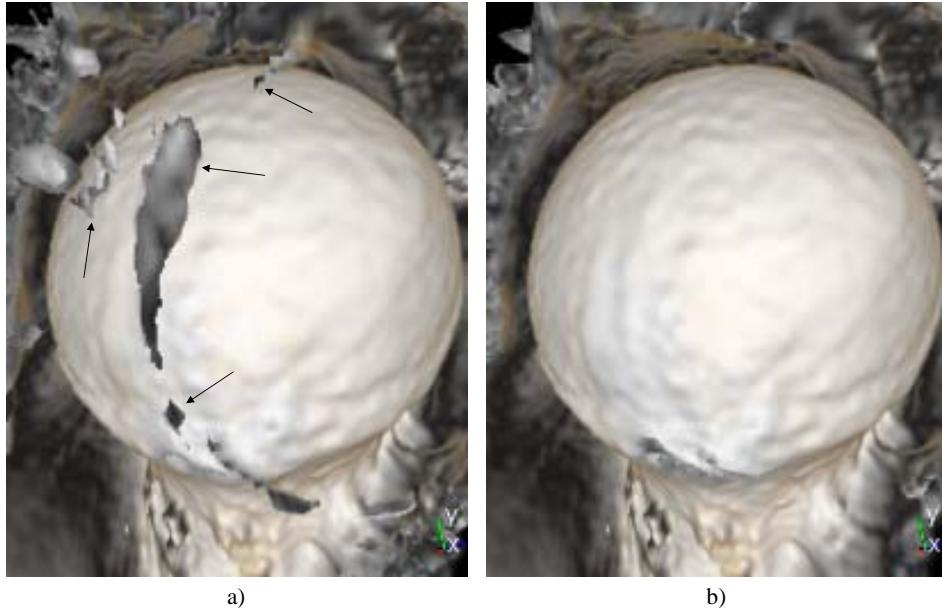


Figure 5.16: Occluding portions of the shoulder socket are indicated in a). Increasing the smoothing of the occlusion data enables elimination of these structures b), revealing the entire humeral head.

same resolution in both images, showing the advantage of decoupling rendering and occlusion determination by using separate data sets for each.

5.3.1.7 Discussion

FOR was able to successfully render the shoulder socket from viewpoints within the humeral head in all 31 shoulder MRI data sets studied. Because MRI intensities are not measured on an absolute scale, rendering settings are not in general transferable from one data set to another, requiring the user to perform some adjustment of parameters in all cases³. However, some data sets required more fine-tuning of rendering parameters than others.

The major factors that seemed to influence how much tuning was required were the amount of separation between the shoulder and humeral head and the presence of artifacts in the MRI

³Typically, around 10 seconds is required to adjust parameters for an experienced user.

image. The resolution of the data affected the amount of detail in the renderings, but did not seem to affect the ability of FOR to render the surface of interest.

When operating, the typical first step is to reduce the lower threshold until there are no holes in the surface of the humeral head. The transfer function settings are then adjusted to achieve a good result. If there appear to be any spurious occluders, due to either noise or inhomogeneities in the humeral head, the scale and threshold parameters can then be adjusted.

All of these parameters can be adjusted interactively, making the process quite rapid. However, automation to achieve reasonable results is desirable, and is a possibility for future work.

5.3.2 Other Joints

In addition to virtual arthroscopy of shoulders, FOR has been used for virtual arthroscopy of the ankle, knee, hip, wrist, and elbow (Figure 5.17).

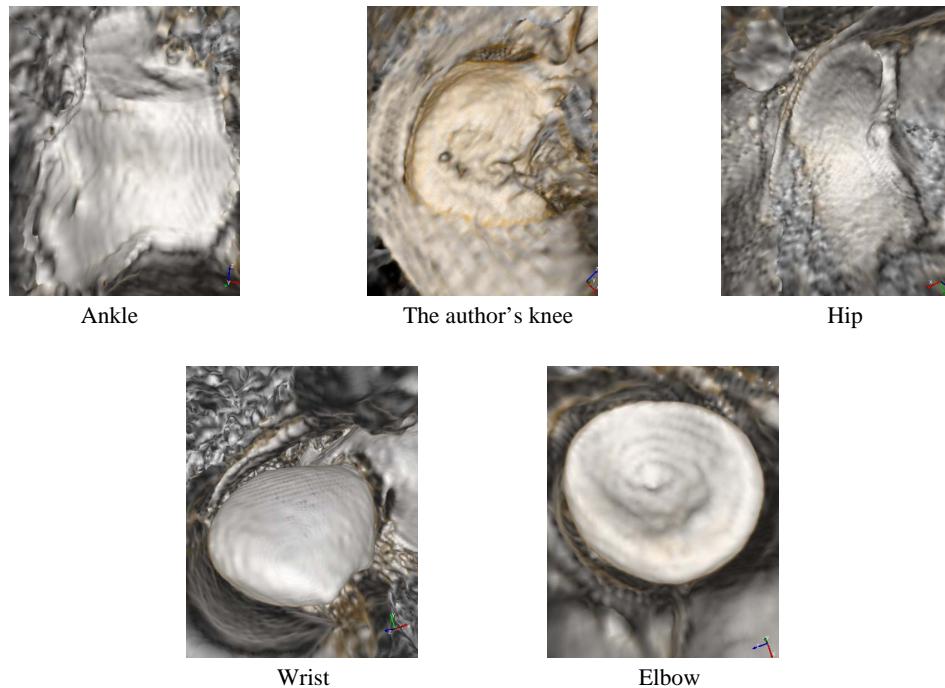


Figure 5.17: Other joints rendered using FOR.

FOR has provided useful views of joints during virtual arthroscopy. These views are not

possible with standard volume rendering because of occlusion. By automatically culling occluding material, FOR enables rapid survey of joints for internal derangements, providing useful anatomical information to physicians in a more rapid and intuitive manner than with MRI slice viewing. These views can augment MRI slice viewing by enabling rapid determination of features of interest for further inspection, and a more accurate understanding of the 3D shape of these features. Although it has proven anecdotally very useful, FOR has yet to undergo clinical trials to prove its effectiveness for virtual arthroscopy.

CHAPTER 6

VIRTUAL URETEROSCOPY

This chapter presents the results of a study using FOR to enable virtual ureteroscopy of the renal collecting system¹ (Fielding et al., 2006). An overview of the anatomy involved is given, followed by a presentation of standard techniques used to diagnose pathology in the renal collecting system, before presenting virtual ureteroscopy using FOR.

6.1 The Renal Collecting System

In human anatomy, the renal collecting system transports waste filtered by the kidneys out of the human body. It is composed of the renal calyces, renal pelvis, ureters, bladder, and urethra. The parts of the renal collecting system of most interest with respect to the virtual ureteroscopy study are the renal calyces and renal pelvis of the kidneys, and the ureters (Figure 6.1).

6.1.1 The Kidneys and Ureters

The kidneys are organs that filter waste, such as urea, from the blood, and excrete them along with water as urine. There are two kidneys in a normal human, one on the left side of the body and one on the right. The waste filtered by each kidney is funneled into spaces called the renal calyces, and then collects in the renal pelvis. Each ureter is a muscular tube that transports urine from a renal pelvis to the bladder.

¹The term virtual pyelescopy is used in the referenced paper. Pyelescopy is fluoroscopic observation of the pelvis and calyces of the kidney via an injected contrast medium. FOR uses volume rendering to simulate views of the surfaces of the renal collecting system from contrast-enhanced CT data. These views can be either from interior viewpoints, simulating clinical ureteroscopy, or from exterior viewpoints, while still displaying the interior surface. The term virtual ureteroscopy therefore seems more appropriate.

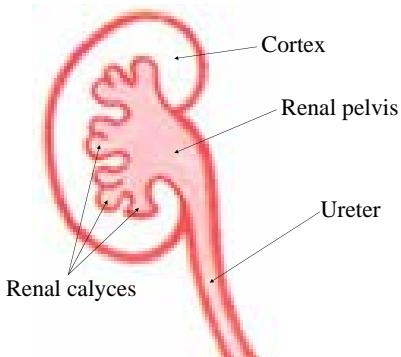


Figure 6.1: Kidney diagram.

6.1.2 Common Pathologies

Some common pathologies of interest for virtual ureteroscopy are kidney stones, tumors, and deformed calyces.

6.1.2.1 Kidney stones

Kidney stones² are solid calcifications that form in the renal calyces, renal pelvii, and ureters. They are usually not noticed until they start to block the flow of urine, causing pain. Often, kidney stones are passed without the need for intervention. However, some degree of intervention can be necessary, including using ultrasound to try to break up the stone into smaller pieces to make them easier to pass, ureteroscopy to remove the stone, and open surgery to remove the stone.

6.1.2.2 Tumors

A tumor is an abnormal growth of tissue, and can form in the renal calyces, renal pelvi, and ureters. Early detection of tumors is highly desirable, as they may be, or may become, cancerous.

6.1.2.3 Deformed calyces

Deformed calyces can be symptomatic of a number of different diseases of the kidney. Displaying the 3D shape of the calyces directly is therefore beneficial.

²Kidney stones are also referred to as renal calculi.

6.2 Standard Ways of Diagnosing Pathology in the Renal Collecting System

Two standard ways of surveying the renal collecting system for stones and tumors are diagnostic ureteroscopy and CT urogram slice viewing.

6.2.1 Ureteroscopy

In ureteroscopy, a small fiber-optic camera called a ureteroscope is introduced to the body through the urethra. The ureteroscope is then maneuvered past the bladder into one of the ureters. Then the ureteroscope is used to navigate through the interior of the ureter, into the renal pelvis and renal calyces, while inspecting for stones and tumors. Ureteroscopy is an expensive and uncomfortable procedure, so less-invasive techniques are preferable.

6.2.2 CT Urogram Slice Viewing

CT urogram slice viewing is a less-invasive technique for diagnosing pathology of the renal collecting system (Figure 6.2). During CT urogram slice viewing, the physician reviews three orthogonal sets of slices of the collected volume, inspecting the slices for abnormalities in the renal collecting system. Usually, two CT scans are taken, one with contrast-enhancement and one without. The non-contrast-enhanced CT scan is useful for detecting stones, as the stones typically have a very high density, appearing very bright in CT. They therefore contrast well with the soft tissue of the renal collecting system. For the contrast-enhanced CT scan, a contrast agent is injected into the patient, which is then filtered by the kidney into the renal collecting system. Contrast enhancement is useful for detecting tumors, as it causes the interior of the renal collecting system to appear bright, making it possible to find small surface irregularities representing tumors.

A problem with CT urogram slice viewing is that it requires tedious manual selection of hundreds of slices to survey the entire renal collecting system. Also, an understanding of 3D shape is difficult to obtain from slice viewing. Such an understanding can be useful in recognizing surface abnormalities and characterizing stones and tumors.

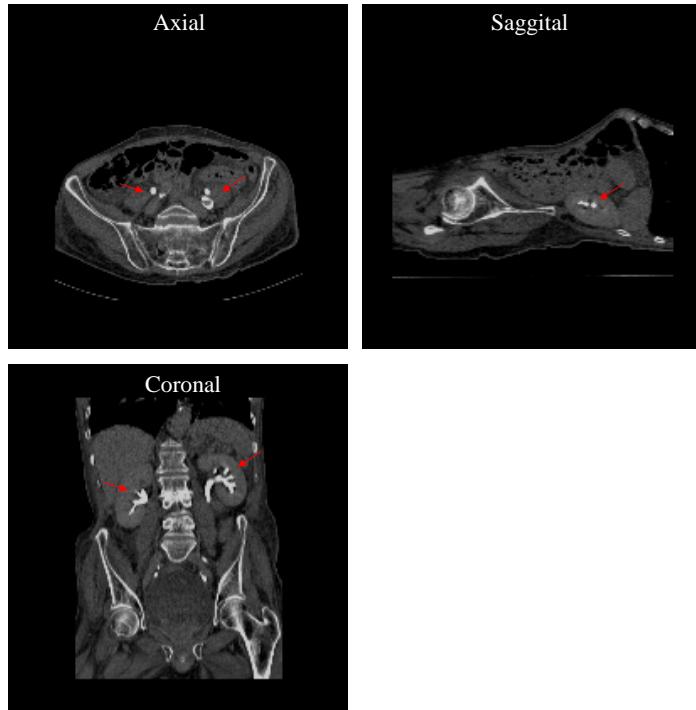


Figure 6.2: Three orthogonal CT slices of the renal collecting system. The contrast-enhanced renal calyces and renal pelvii are indicated by the red arrows.

6.3 Flexible Occlusion Rendering for Virtual Ureteroscopy

Colleagues in radiology wished to be able to obtain 3D survey views of the renal collecting system, displaying the inner surface from exterior positions to show large areas of the inner surface in a single view. A study using FOR to enable virtual ureteroscopy was therefore performed. The study consisted of two parts, an *in vitro* portion using three contrast-filled latex glove phantoms, and an *in vivo* portion using ten CT scans, with two kidneys per scan, resulting in twenty kidneys. All images were obtained using a Siemens 16 slice multidetector CT scanner.

6.3.1 Glove-Phantom Study

Latex gloves were chosen as the phantom because the digits of the glove resemble the tube-like renal calyces, and the palm resembles the renal pelvis. Three gloves were filled with dilute water-soluble contrast material with a CT number around 1100 Hounsfield Units (HU), similar to that of the renal collecting system in the excretory phase of a CT urogram. Each

glove was placed in an agar bath and scanned individually. Objects of air, water, and calcific density ranging from 4 to 20 mm in diameter were glued to the interior surface of the digits of the gloves.

For the study, the thumb was defined as digit A. Glove 1 contained no objects, Glove 2 contained a calcium-density seashell measuring 20x10x8 mm in digit B and a stone measuring 4x7x4 mm in digit D. Glove 3 contained a mixed air and soft-tissue-density object measuring 10x6x5 mm in digit A and a soft-tissue-density object measuring 9x4x7 mm in digit E. Figure 6.3 shows a comparison of standard volume rendering and FOR used to render Glove 2. Standard volume rendering displays the outer surface of the glove, whereas FOR removes the

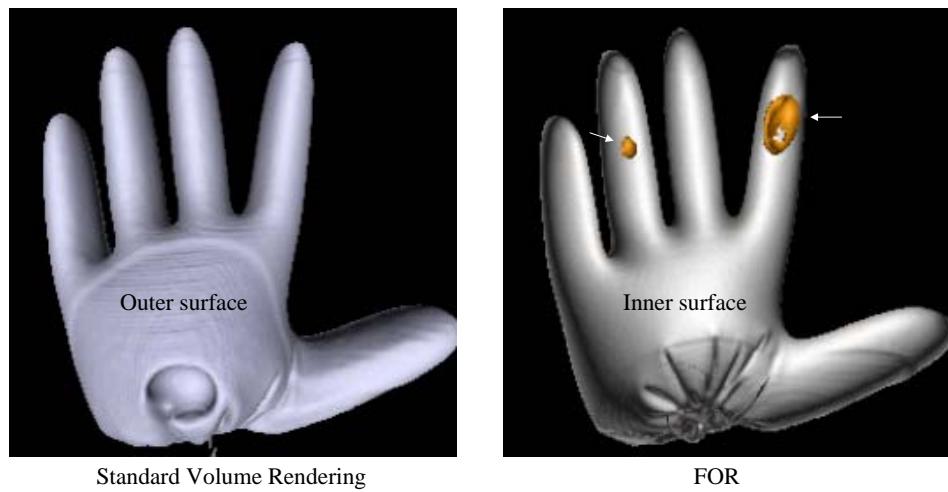


Figure 6.3: Contrast-filled latex glove phantom. Standard volume rendering displays the outer surface of the glove. FOR removes the occluding outer surface to display the inner surface, revealing a small stone and a seashell placed inside.

occluding outer surface to display the inner surface, revealing a small stone and a seashell.

The author was given the number of objects in each glove, but the positions and densities of the material were unknown. All objects were identified by the author, without any intervention from radiologist colleagues, and there were no false-positive results. Additionally, the unlimited camera angle allowed the operator to view each glove in its entirety, both internally and externally, enabling rapid identification of objects.

Figure 6.4 shows a glove phantom scanned after the described study. These views show a soft-tissue-density object, akin to a tumor, along with a high-density object, akin to a stone,

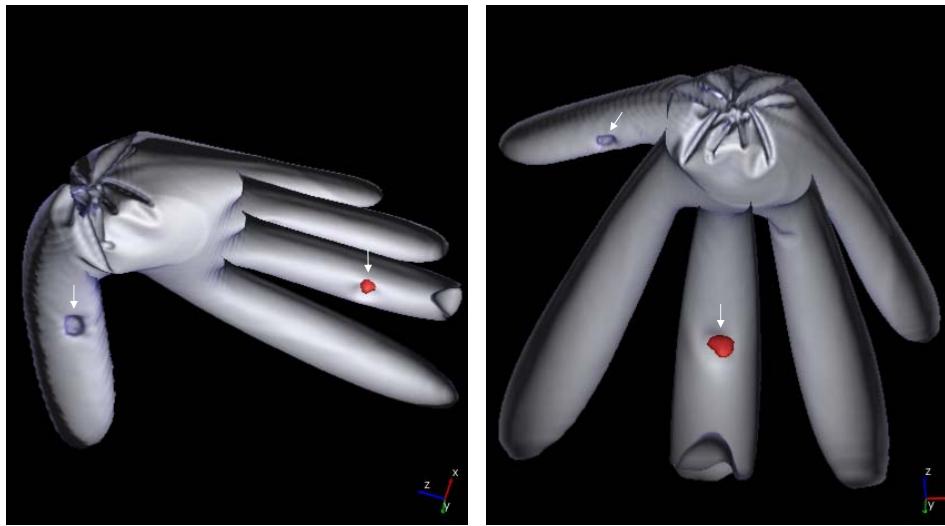


Figure 6.4: Two views of a glove phantom rendered using FOR. A soft-tissue-density object is shown in blue, and a high-density object is shown in red.

in the same rendering.

6.3.2 Contrast-Enhanced CT Study

For the *in vivo* portion of the study, images from ten consecutive patients who underwent CT scanning as part of routine clinical care were compiled. Both kidneys were reviewed for each patient, for a total of twenty kidneys. Use of this anonymized patient data was approved by our institution's internal review board.

During the study, the radiologist chose specific calyces, one within each kidney, for review based on abnormal anatomy or presence of stone. Because identification of particular calyces was difficult without training in radiology, review of the *in vivo* FOR renderings was done in consensus with a radiologist. No tumors were identified on source images or ureteroscopy, and none were identified using FOR. FOR was applied successfully to CT images of 19 kidneys (Figure 6.5). In addition, successful flythroughs of all ureters were performed.

Every data set that could be successfully viewed with CT slice viewing could be successfully viewed using FOR. However, multiple metal clips within one kidney resulted in significant artifacts, making it impossible to clearly identify the walls of the calyces, either via CT slice viewing or viewing 3D renderings of the data. Fifteen normal calyces, three deformed calyces

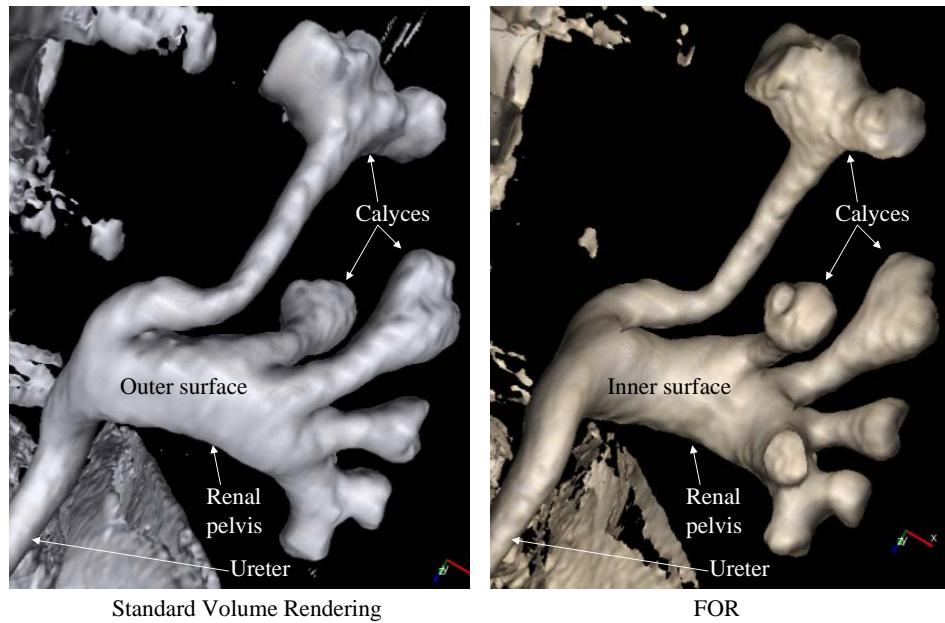


Figure 6.5: Renal collecting system rendered from CT data. Standard volume rendering displays the outer wall, whereas FOR removes the occluding material of the outer wall, displaying the interior surface.

(Figure 6.6), and one site of extrinsic compression caused by a crossing vessel were clearly shown.

Two stones, one 3 mm and the other 9 mm, were not identified. These stones were only visible in the non-contrast CT urograms, as their HU value was in the range of the contrast used. As the study was only performed on the contrast-enhanced CT urograms, the stones could not be rendered. All contrast-filled portions of the ureter could be virtually traversed. The minimal ureteral diameter measured on source images was 3 mm.

6.3.3 Conclusion

The ultimate goal of virtual pyeloscopy is to replace surveillance ureteroscopy. In this small pilot study, only anatomic abnormalities were studied because upper-tract tumors are relatively uncommon and none were identified during the acquisition time frame. FOR is a promising technique for enabling fast and accurate inspection of the renal collecting system for pathology, although more extensive testing is required.

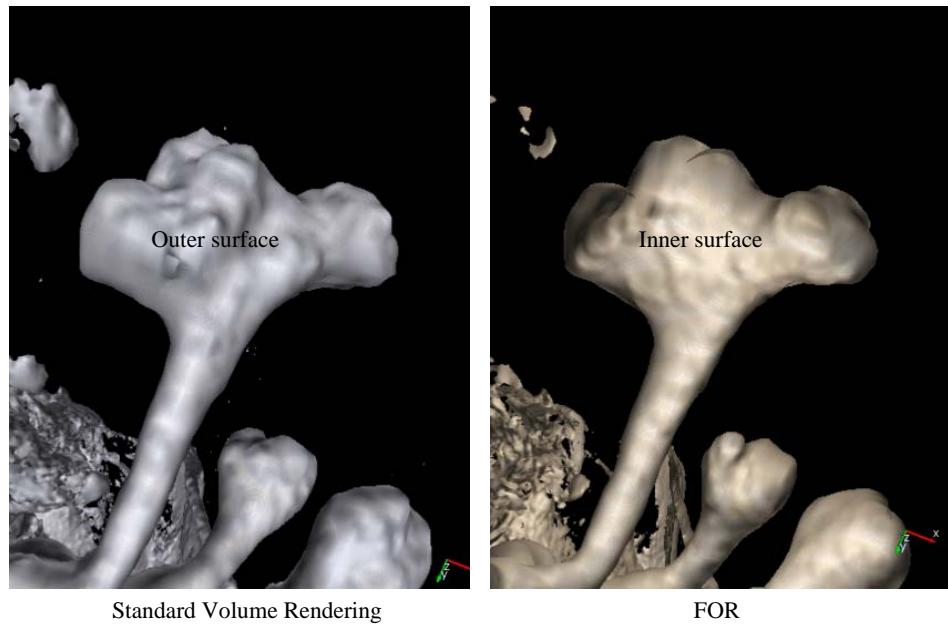


Figure 6.6: Deformed calyx rendered from CT data. Standard volume rendering displays the outer wall, whereas FOR removes the occluding material of the outer wall, displaying the interior surface.

Future work might include combining the non-contrast-enhanced and contrast-enhanced CT urograms to enable detection of stones along with inspection for tumors. This would require an accurate registration of the two data sets, however. Also, iterative CT reconstruction techniques could be applied to data sets to attempt to reduce metal artifacts.

CHAPTER 7

KNEE FRACTURE VISUALIZATION

This chapter presents preliminary results using FOR to render knee fracture surfaces from CT data. An overview of the knee-bone anatomy is given, followed by a presentation of the standard methods for viewing knee fractures. The results of applying FOR to visualize two knee fractures is then presented.

7.1 The Knee

The three bones in the knee are the patella, femur, and tibia, also referred to as the kneecap, thigh bone, and shin bone (Figure 7.1). The fibula, or calf bone, connects to the femur at the

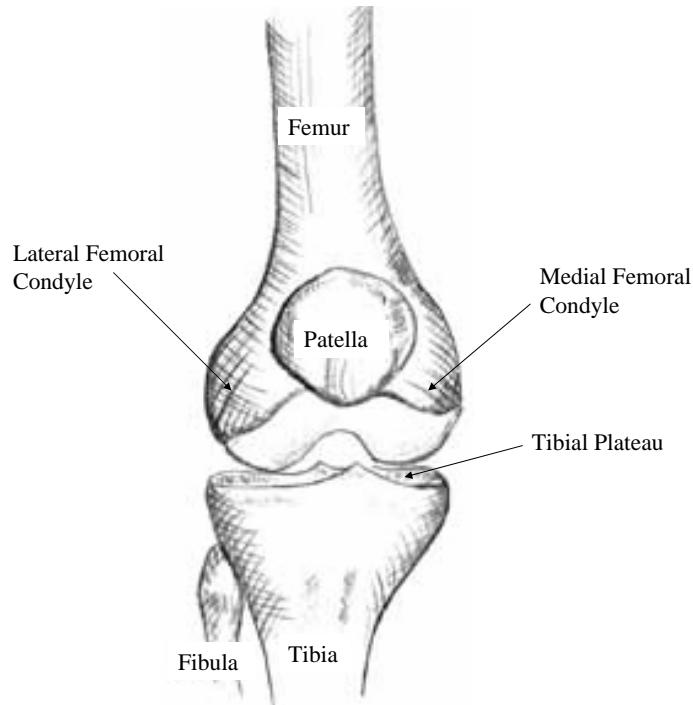


Figure 7.1: Diagram of the bones in the knee (KNEEguru, 2007).

main articulation of the knee via the lateral collateral ligament, and to the tibia by a separate small joint. However, is not technically considered part of the knee joint.

The knee can be subdivided into the femoro-patellar component and the femoro-tibial component. The bottom of the femur forms two lobes called the lateral femoral condyle and the medial femoral condyle. The femoral condyles articulate with the upper portion of the tibia, called the tibial plateau, to form the femoro-tibial compartment.

The knee supports almost the entire weight of the human body, and is therefore susceptible to injury, including fractures. A common knee fracture is a tibial plateau fracture.

7.1.1 Tibial Plateau Fractures

There are two types of arrangements of bone tissue: cortical bone and cancellous bone. Cortical bone is very dense, forming the hard outer surface, whereas cancellous bone is spongy, filling the interior. A fracture is a breaking apart of the cortical bone surface.

A common mechanism for a tibial plateau fracture is when one or both of the femoral condyles impacts the tibial plateau, causing damage to the articular surface. The femoral condyles may also be damaged during the impact.

Fractures may be displaced or non-displaced. In a non-displaced fracture, the surface of the bone is cracked, but the bones remain in their proper position and alignment. Non-displaced fractures can usually be treated without surgery. Displaced fractures are more severe, as the surfaces of the bones in the joint are shifted, usually requiring surgery for realignment.

7.2 Standard Ways of Viewing Tibial Plateau Fractures

The two standard ways of viewing tibial plateau fractures are planar x-ray imaging and CT.

7.2.1 X-Ray Imaging

X-ray imaging is a useful modality for viewing bone fractures, because cortical bone attenuates x-rays much more than surrounding tissue. Therefore, x-ray imaging offers good contrast for displaying fractures of the cortical bone. The major problem with x-ray imaging is that

it forms a 2D projection of the 3D volume being imaged (Figure 7.2). All depth information



Figure 7.2: X-ray image of a knee (Georgetown Pathology, 2007).

along the direction of projection is lost. There are two issues with this loss of information. The first is that the projection can cause small fractures to be difficult to diagnose if a lot of high-density structures overlap along the line of projection. This causes a loss of contrast sometimes referred to as contrast pile-up. The second issue is that the loss of 3D information makes surgical planning for correcting displaced tibial plateau fractures difficult.

7.2.2 CT Slice Viewing

Because CT is reconstructed from multiple x-ray images, it retains the good cortical bone contrast found in x-ray imaging. It also eliminates the problems with 2D x-ray projection because reconstructing a 3D volume eliminates the contrast pile-up problem, and gives spatial information along all three axes.

CT scans of bone fractures are typically viewed by looking at slices of the volume (Figure 7.3). This requires the physician to manually select and review many slices of the volume, mentally reconstructing the 3D shape of the bones to gain an appreciation of the location of



Figure 7.3: Coronal slice of a knee CT, indicating a tibial plateau fracture. The patella and fibula are not present in this slice.

fractures, and the amount of displacement involved. A good understanding of the amount of displacement is desired when planning surgery to repair the joint. A more direct way of viewing the CT data involves reconstructing 3D images of the bone surfaces. The next section presents the results of applying FOR to study two tibial plateau fractures.

7.3 Flexible Occlusion Rendering for Knee Fracture Visualization

The FOR images in this section use the flood-filled mask technique described in Section 4.4.3.2. The occlusion masks are created as an automated preprocess using Matlab, with the scale of the mask smoothing adjustable during rendering.

7.3.1 Knee 1

Using volume rendering, the cortical bone surfaces of the knee joint can be rendered in 3D (Figure 7.4). The physician can move the virtual camera to obtain better views for inspecting

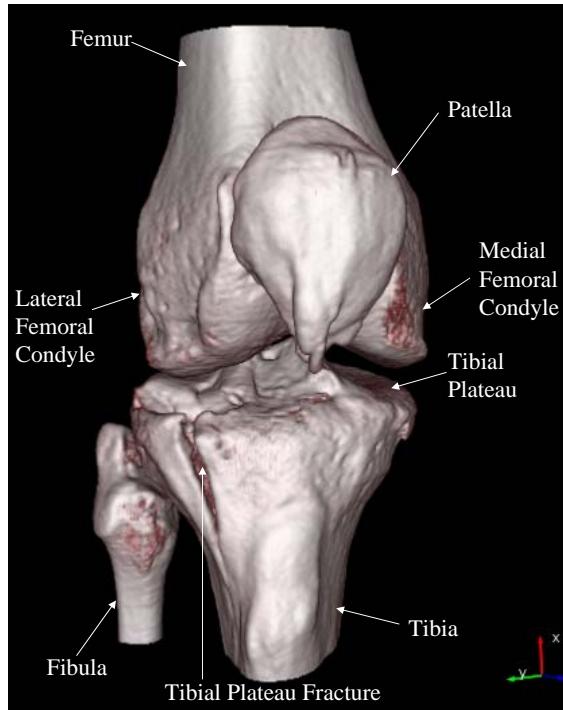


Figure 7.4: Survey view of a knee using volume rendering of CT data.

the tibial plateau for fractures, however these views are restricted to viewpoints outside of the bones in the joint, as the bones will otherwise occlude each other. Figure 7.5 shows a closer view of the damage caused by a tibial plateau fracture. The crack formed by the fracture is obvious, but the amount of displacement is difficult to determine from such an angle. Using standard volume rendering, the lateral femoral condyle will occlude the tibial plateau from angles better suited to determine the amount of displacement.

FOR can render the same views possible with standard volume rendering, but also enables views augmenting these standard views. Such views can render the surface of the tibial plateau from viewpoints within the femur, making the amount of displacement more apparent (Figure 7.6). Such viewpoints are desired by colleagues in radiology to better understand the geometry of large fractures, and to enable inspection of the entire tibial plateau surface for smaller fractures. The physician can also fly the virtual camera into the tibia to inspect the femoral condyles for damage (Figure 7.7).

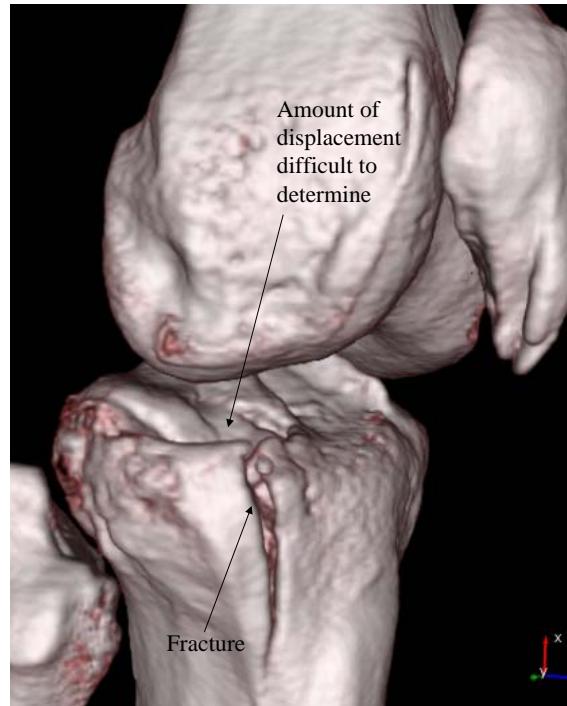


Figure 7.5: Volume rendering of a tibial plateau fracture from CT data. The crack formed by the fracture is obvious, but the amount of displacement is difficult to determine from this angle.

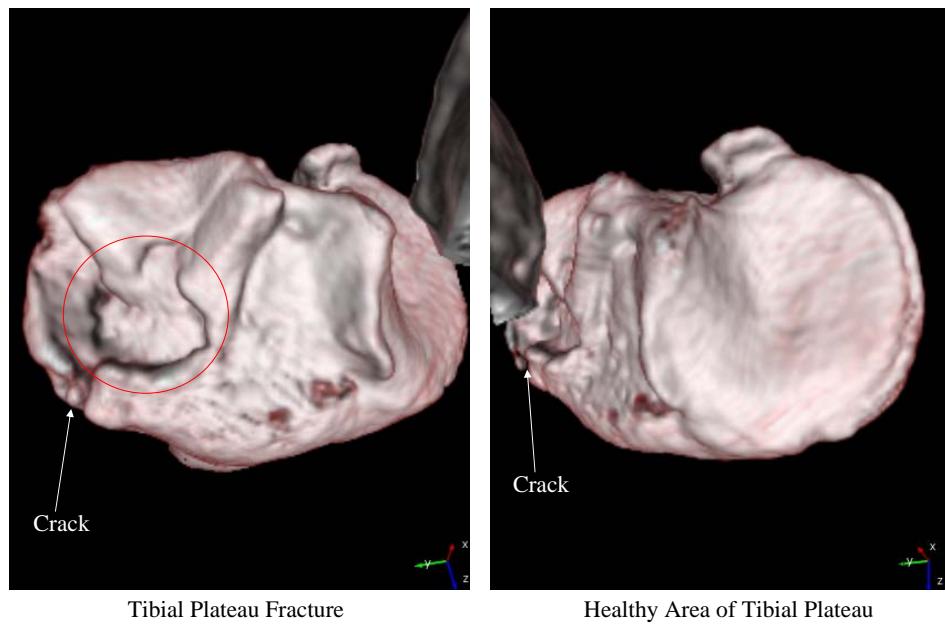


Figure 7.6: FOR used to render a tibial plateau fracture, with a large amount of displacement circled in red.

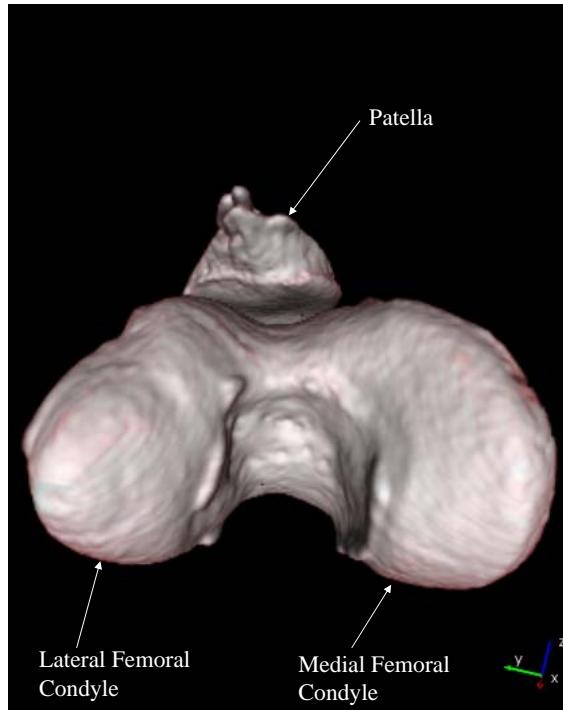


Figure 7.7: A view of the femur from within the tibia, rendered using FOR. No visible damage was caused to the femur during this fracture.

7.3.1.1 Grazing effects

Figure 7.6 exhibits some grazing effects due to the geometry of the femur, in which rays reach a local maxima in the occlusion data between two parts of the femur. This causes portions of the femoral condyles to be rendered. Adjusting the amount of smoothing performed on the occlusion data can help alleviate this effect, without altering the resolution of the rendering of the tibial plateau surface (Figure 7.8).

7.3.2 Knee 2

FOR has also been successfully applied to a second knee-fracture data set. During this fracture, the tibial plateau was smashed into numerous pieces. Figure 7.9 shows a survey view of the knee.

Using FOR, the surface of the tibial plateau fracture can be investigated, revealing extensive damage (Figure 7.10). The surgery to repair a tibial plateau fracture is like a 3D jigsaw puzzle, as the surgeon needs to know how to fit the broken pieces back together. This is very

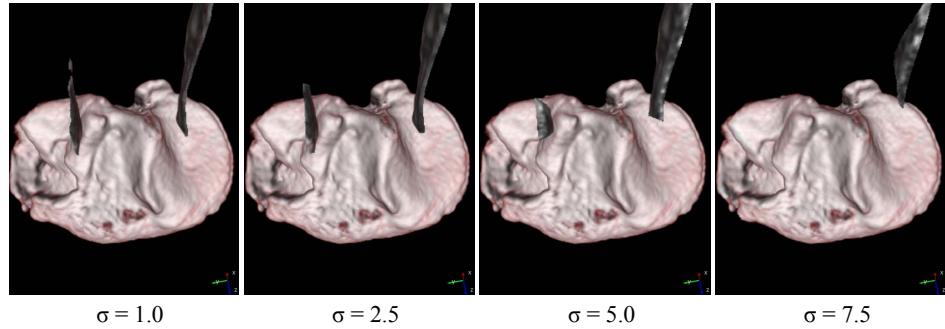


Figure 7.8: Different amounts of smoothing applied to the occlusion data help to alleviate femoral condyle grazing effects without altering the resolution of the rendering of the tibial plateau surface.



Figure 7.9: Survey view of a knee that suffered a severe tibial plateau fracture. The vertical rods are part of a splint surrounding the leg, rendered because they have a density similar to that of cortical bone.

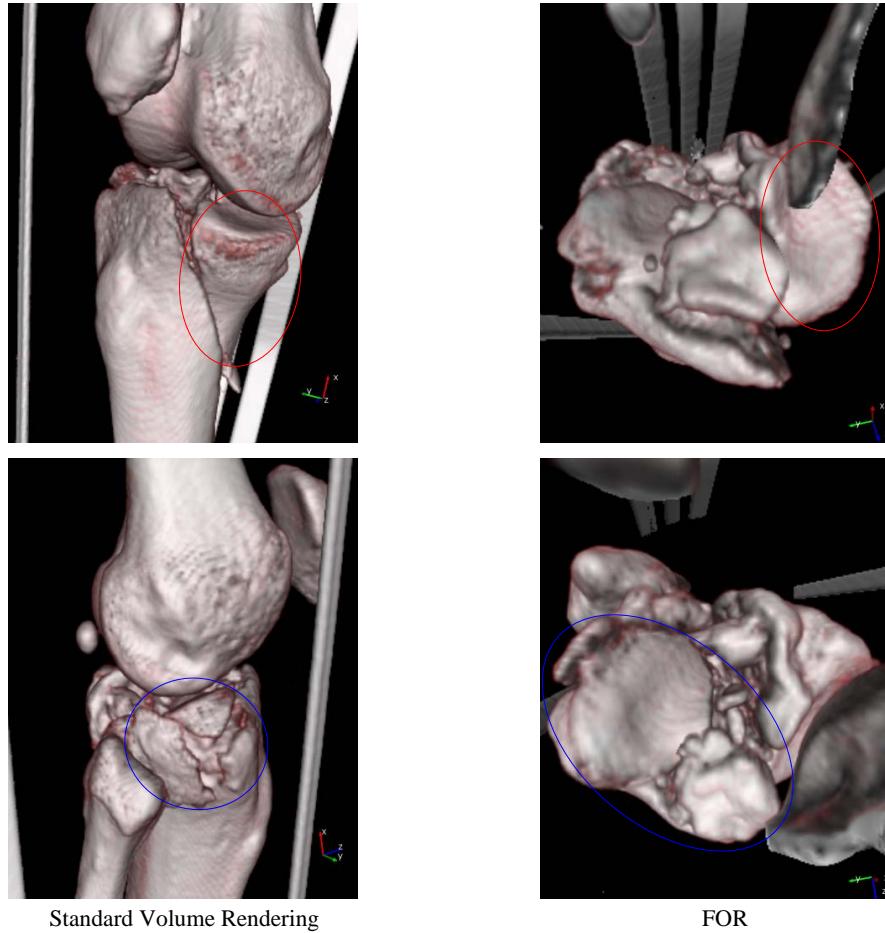


Figure 7.10: Standard volume rendering of a severe tibial plateau fracture compared with FOR. Related features are indicated by colored circles.

difficult with CT slice viewing, as the portions of the surface to fit together may not appear in the same slice. The views available with FOR are expected to be helpful for physicians when planning surgical reconstruction, as the entire surface of the tibial plateau can be surveyed, and the 3D shapes of the fractured pieces can be determined.

7.3.3 Limitations

The flood-filled occlusion masks used are currently generated as a pre-process. Because CT data is measured in absolute units automatic setting of the threshold was used. For the first knee, the generated masks were successful for all bones in the joint. However, the flood fill on the second knee was only successful for the femur, patella, and fibula. The flood fill failed on the tibia because it is broken into large chunks. Views of the tibia from inside the femur were successful, but views of the femur from inside the tibia were occluded by large chunks of the tibia. Experimenting with smoothing and morphological techniques after thresholding may prove fruitful. Other methods of automatically creating the masks could also be investigated, bearing in mind that the masks need not be very accurate, as the smoothing performed on the masks enables robust occlusion determination.

7.4 Conclusion

Preliminary studies using two CT scans of tibial plateau fractures demonstrate FOR's ability to provide useful views of the surface of the tibial plateau for inspection of fractures. The views generated using FOR could potentially help physicians understand the geometry of fractures better than with CT slice viewing or standard volume rendering. Clinical trials are needed to prove this utility, however.

CHAPTER 8

CONCLUSION

8.1 Results

FOR is an extension to standard ray cast volume rendering that enables view-dependent display of otherwise-occluded features in 3D volumes. The goal of this work is to enable more rapid and accurate diagnosis of pathology from three-dimensional (3D) medical images. The key insights and features in FOR are summarized as follows:

- Finding separating features between objects, rather than the boundaries of the objects themselves, enables more robust determination of undesired occlusions.
- The location of these separating features in data sets that have noise and partial-volume effects can be performed robustly by finding intensity peaks of a minimum height and a minimum intensity.
- These features can be found during the rendering process, without a need for pre-segmentation.
- These features can be found using the same dataset used for rendering, or a separate dataset optimized for occlusion determination.
- Finding the separation enables seamless transition from occluded viewpoints to occluding viewpoints.
- The scale of the dataset used for feature-detection can be adjusted without altering the resolution of the data being rendered.

- For the applications presented in this dissertation finding the first separating feature has proven useful, however finding multiple separations or the last separation are also possible, depending on the application.

By modulating the accumulated color and opacity upon determining features of interest, FOR can achieve effects similar to importance driven volume rendering (Viola et al., 2004) without requiring pre-segmentation and selection of an importance hierarchy. By resetting the accumulated color and opacity upon determining separation features, FOR enables removal of occluding features.

FOR has been applied to three different applications: virtual arthroscopy of joints from MRI data (Borland et al., 2005; Borland et al., 2006), virtual ureteroscopy of the renal collecting system from contrast-enhanced CT data (Fielding et al., 2006), and knee fracture visualization from CT data. In all three cases, FOR enables useful views of the data that are not possible using standard volume rendering.

International Patent Application No. PCT/US2006/009331, *Methods, systems, and computer programming products for processing three-dimensional image data to render an image from a viewpoint within or beyond an occluding region of the image data*, has been filed by the inventors, encapsulating portions of the FOR framework. A second provisional patent application has also been filed, covering the remainder of the full FOR description.

8.2 Limitations and Future Work

8.2.1 MRI Data

The major limitation with rendering MRI data using FOR, as with the virtual arthroscopy application described in this dissertation, is that significant tuning of FOR parameters is necessary to achieve good results because MRI intensities are not measured on an absolute scale. Techniques for normalizing MRI images so that intensities are comparable between different data sets would be beneficial for creating default rendering settings, reducing the amount of tuning necessary.

Currently, the same MRI sequence used for rendering is used to generate the occlusion

data. However, it may be the case that different MRI sequences might be optimal for determining occlusion and rendering.

8.2.2 CT Data

A limitation with rendering CT data using FOR is that the rendering will not work properly in the presence of significant metal artifacts. FOR shares this limitation with any CT-based method that relies on filtered backprojection to reconstruct the data, including display of CT slices. Iterative reconstruction methods may prove useful in data sets exhibiting metal artifacts.

For visualizing knee fracture CT data using FOR, the major limitation is that the flood fill operation used to generate the occlusion masks can fail with large fractures. Experimenting with different filtering methods and morphological operations to create the masks may prove useful. Other rough segmentation schemes to create the masks may also prove useful, as FOR is robust to inaccuracies in the occlusion mask data.

8.3 Future Directions

FOR has provided promising results in the three applications tested. However, more extensive trials are needed to prove clinical validity. FOR may also prove useful in a variety of other medical applications, such as virtual colonoscopy. The application of FOR to visualizations of hip fractures and heart valves from CT data is already underway.

Combining the accumulated color and opacity resetting to display surfaces with opacity modulation techniques to reveal structures hidden behind surfaces may be effective for indicating hidden pathology.

FOR may also prove useful for the display of non-medical 3D volumes. Figure 8.1 shows FOR applied to a CT scan of an engine block, with variable accumulated color and opacity modulation.

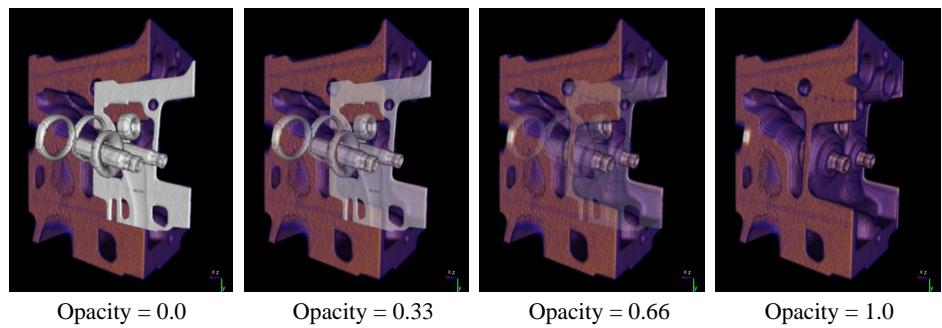


Figure 8.1: Engine block rendered using FOR, with variable opacity modulation to reveal inner structures.

APPENDIX A

INNERSPACE

Innerspace is a program developed to enable viewing medical images using FOR, and was built using open-source, cross-platform libraries. Innerspace can read Digital Imaging and Communications in Medicine (DICOM) files, commonly used for storing MRI and CT data, using the Grass roots DiCoM (GDCM) library. The wxWidgets library is used for the graphical user interface (GUI), and FOR is implemented within the volume rendering framework of the Visualization ToolKit (VTK). Innerspace uses wxVTKRenderWindowInteractor to enable wxWidgets and VTK to interact with each other. A screenshot of the Innerspace main window is shown in Figure A.1.

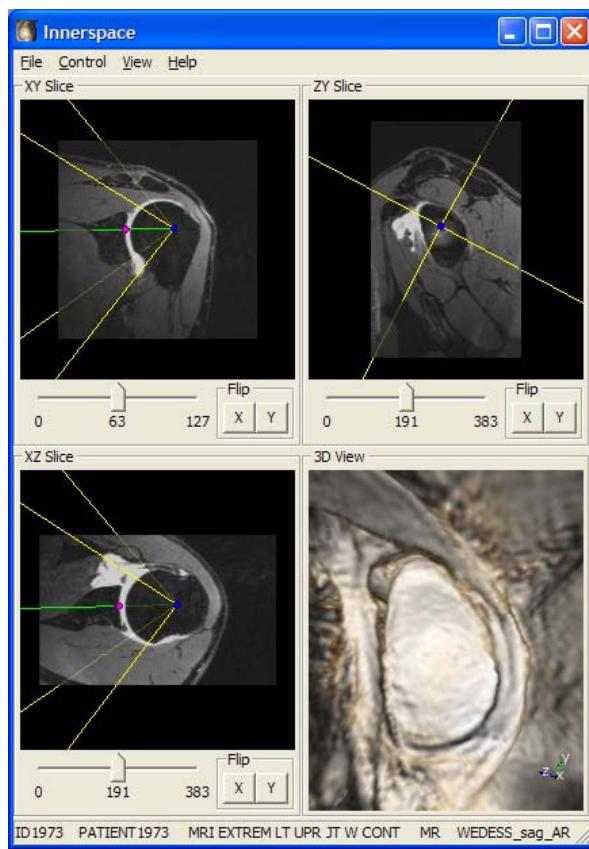


Figure A.1: Screenshot of the Innerspace main window.

A.1 Interface

Innerspace consists of a main window displaying three orthogonal slices of the data and a 3D rendering of the data. There are also buttons to flip the image in X and Y. The main window has four menus: File, Control, View, and Help.

A.1.1 Loading Data

Data is loaded by choosing Open from the File menu. A dialog for opening a file then appears. DICOM data can be loaded either by selecting an individual DICOM slice image in a directory containing the rest of the slices of the data, or by selecting a DICOMDIR file. the DICOMDIR file is used to archive multiple patients with multiple DICOM series per patient. If a DICOMDIR file is selected, a dialog for selecting the DICOM series to be loaded appears (Figure A.2). All DICOM series are presented, organized by patient. For MRI

Patient ID	Patient Name	Study Description	Series Modality	Series Description
ID1973	PATIENT1973	MRI EXTREM LT UPR JT W CONT	MR	SHOULDER/PD_T2_SE_AXIAL
ID1973	PATIENT1973	MRI EXTREM LT UPR JT W CONT	MR	Coronal FS Arthrogram
ID1973	PATIENT1973	MRI EXTREM LT UPR JT W CONT	MR	Sagittal FS Arthrogram
ID1973	PATIENT1973	MRI EXTREM LT UPR JT W CONT	MR	Axial FS Arthrogra
ID1973	PATIENT1973	MRI EXTREM LT UPR JT W CONT	MR	WEDESS_sag_ARTHRO_ISO
ID1975	PATIENT1975	MRI EXTREM RT UPR JT W CONT	MR	RT AXIAL FS
ID1975	PATIENT1975	MRI EXTREM RT UPR JT W CONT	MR	RT COR FS
ID1975	PATIENT1975	MRI EXTREM RT UPR JT W CONT	MR	SHOULDER/PD_T2_TSE5_CORONA
ID1975	PATIENT1975	MRI EXTREM RT UPR JT W CONT	MR	RT SAG FS
ID1975	PATIENT1975	MRI EXTREM RT UPR JT W CONT	MR	SHOULDER/WE_DESS_ARTHRO_CO
ID1977	PATIENT1977	MRI EXTREM LT UPR JT W+WO CONT	MR	LEFT_T1_FS_AX
ID1977	PATIENT1977	MRI EXTREM LT UPR JT W+WO CONT	MR	SHOULDER/PD_T2_TSE5_CORONA
ID1977	PATIENT1977	MRI EXTREM LT UPR JT W+WO CONT	MR	LEFT_T1_FS_COR
ID1977	PATIENT1977	MRI EXTREM LT UPR JT W+WO CONT	MR	LEFT_KNEE/LEFT_WEDDESS
ID1977	PATIENT1977	MRI EXTREM LT UPR JT W+WO CONT	MR	LEFT_T1_FS_SAG2

Figure A.2: DICOM series selection dialog.

data, WEDESS sequences are highlighted in blue, as they are the sequences used for virtual arthroscopy. Double-clicking on the series will load it into Innerspace. Innerspace can also load DICOM files in .ima format, and VTK's .vti file format.

Once loaded, the volume can be saved in .vti format. This can be useful for loading the volume into Kitware's VolView and ParaView programs. A separate occlusion volume for use with FOR can also optionally be loaded by choosing Open Occlusion Data from the File

menu. Screenshots of the 3D view can be saved in TIFF format by selecting Save Image from the File menu.

A.1.2 Interacting with the Data

A.1.2.1 Slice views

Once the data has been loaded, the three slice views show slices through the three orthogonal planes of the data, and the 3D view shows a 3D rendering of the data. The slice views display an optional camera widget, representing the location of the virtual camera used to render the 3D scene (Figure A.3). Displaying the camera widget can be turned on and off in

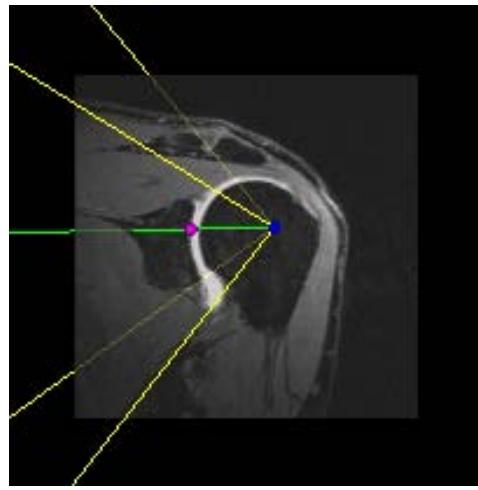


Figure A.3: Camera widget. The blue sphere represents the camera position, the magenta sphere represents the camera focal point, the green line is a center line through the position and focal point, and the yellow lines represent the view frustum.

the Slice Views Controls window (Figure A.4), available in the Controls menu. The camera



Figure A.4: Slice Views Controls window.

position can be moved by left-clicking and dragging the blue sphere of the camera widget. The virtual camera used to render the 3D scene will be automatically updated, as will the camera widgets in the other two slice views. Similarly, the focal point of the virtual camera, which defines the point that the camera rotates around and zooms into, can be moved with the magenta sphere. Both the position and focal point can be moved by dragging the green line, representing the center line through the position and focal point, and the view angle can be adjusted by dragging the yellow lines which represent the view frustum of the camera.

The slice shown in the slice views can be set to correspond to the camera position or the camera focal point through the Slice Views Controls window. The slice shown can also be controlled with a slider beneath each slice view. The camera position or focal point in all slices and the 3D view will be automatically updated as the slice slider is adjusted.

The window/level setting of each slice is controlled by left-clicking and dragging on the image in one of the slice views. Moving left and right controls the window, or contrast, and moving forwards and backwards controls the level, or brightness. The same window/level is automatically applied to all three slice views, and can be reset by hitting the ‘r’ key. The threshold value used in FOR can also be displayed in the slice views, displaying all values above the threshold as bright white, controlled by a checkbox in the Slice Views Controls window.

A.1.2.2 3D view

A 3D rendering of the data is shown in the 3D view, with an axis gnomon representing the orientation of the virtual camera. Left-clicking and dragging the mouse will rotate the camera around the center of focus. Holding the Ctrl key while left-clicking and dragging will spin around the center line. Right-clicking and dragging the mouse forward and backward will zoom into the center of focus and back. Middle-clicking and dragging will translate. Holding the Shift key while left-clicking and dragging will also translate. Pressing the ‘u’ key will rotate the camera to an upright position. All changes to the camera are automatically updated in the camera widgets and slice positions in the slice views.

Pressing the ‘p’ key will perform a picking operation at the position indicated by the

mouse cursor in the 3D view. The picking operation sets the center of focus to the last voxel that contributes opacity along the ray emanating from the selected pixel in the image. The camera will then animate the rotation to the new center of focus. Pressing the ‘z’ key will also perform a picking operation, but will also zoom in a fixed amount. Pressing the ‘b’ key will zoom back out. Picking is a useful tool for selecting features in the 3D view to review in the slice views, as the slices can be automatically set to the focal position. A spherical marker representing the last picked position can be toggled by pressing the ‘m’ key.

Picking with the ‘o’ key will display an intensity profile along the ray (Figure A.5). The

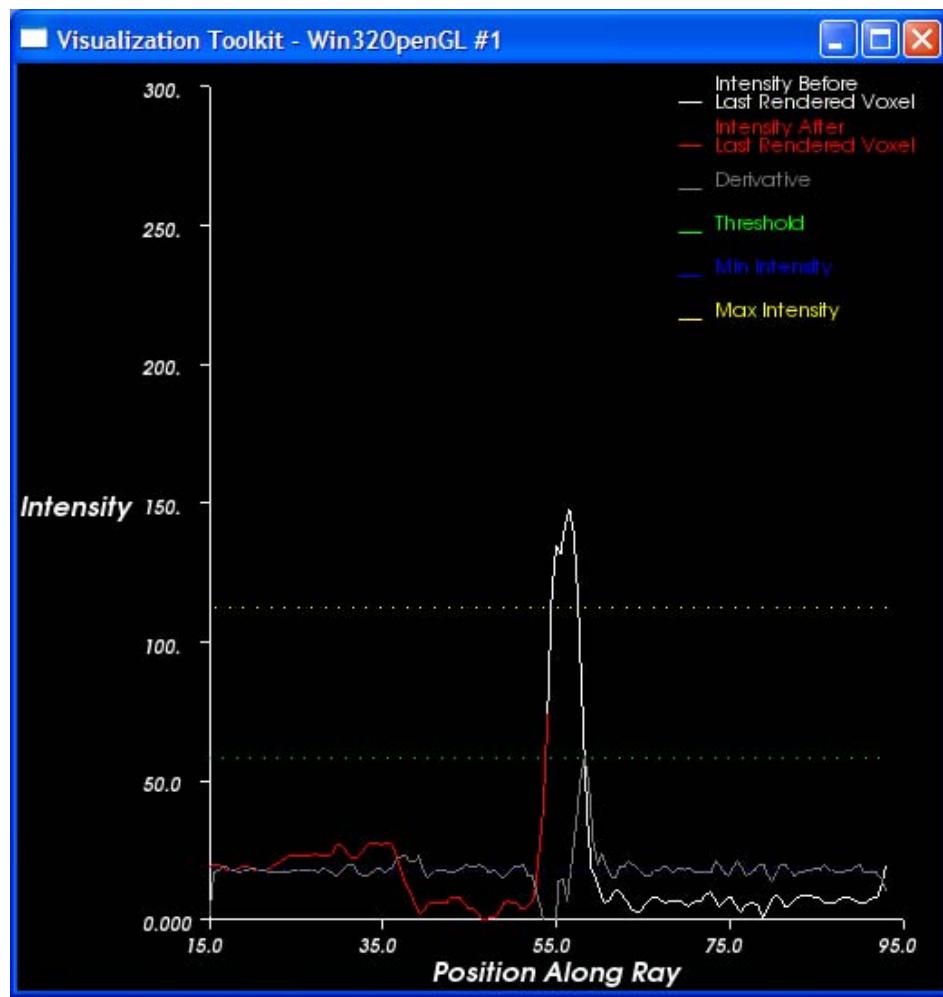


Figure A.5: Intensity profile.

intensity profile tool proved useful when designing algorithms to determine occlusion features along the ray.

All changes to the camera, such as rotating in the 3D view and moving the camera widgets in the slice views, or rendering settings are performed by rendering the image at a lower resolution to achieve interactive rates. A higher quality rendering is then rendered when interaction is complete.

A TIFF image of the the current 3D view can be saved by hitting the ‘s’ key. An image called ‘InnerspaceImageX.tif’ will be saved in current directory, where ‘X’ is a number incremented from 0 for each image saved.

AVI movies of the 3D view can also be recorded. Hitting the ‘r’ key will start recording a movie. When recording the movie, the user can interact at normal rates. The camera settings are saved each time they are changed, along with a timestamp. When the ‘r’ key is hit again, Innerspace will replay the series of camera settings, recording high quality renderings of each view as frames in the movie. The movie will be saved as ‘InnerspaceMovieX.avi’ in the current directory, where ‘X’ is a number incremented from 0 for each movie saved. Hitting the ‘c’ key when recording will cancel the movie.

A.1.2.3 FOR controls

The FOR settings and transfer function definition are controlled through the VDP Controls window (Figure A.6), available in the Controls menu. With these controls, FOR can be turned on or off, with standard volume rendering being used when turned off, and the FOR mode can be set to fixed-threshold, local maximum using the original data, or local maximum using occlusion data. The FOR lower threshold and hysteresis¹ parameters can also be set.

The scalar opacity transfer function can be controlled with the Min Material, Max Material, and Transfer Function Ramp sliders. Alternately, the scalar opacity transfer function can be set using the transfer function widget at the bottom of the window. The FOR threshold can also be set here, using the green bar. The triangle on the bar controls the modulation of the accumulated opacity when an occlusion feature is reached. The gradient opacity transfer function is controlled with the Gradient Opacity Ramp Start and Gradient Opacity Ramp

¹The hysteresis parameter refers to the noise-reducing threshold that selects peaks of the specified relative height.

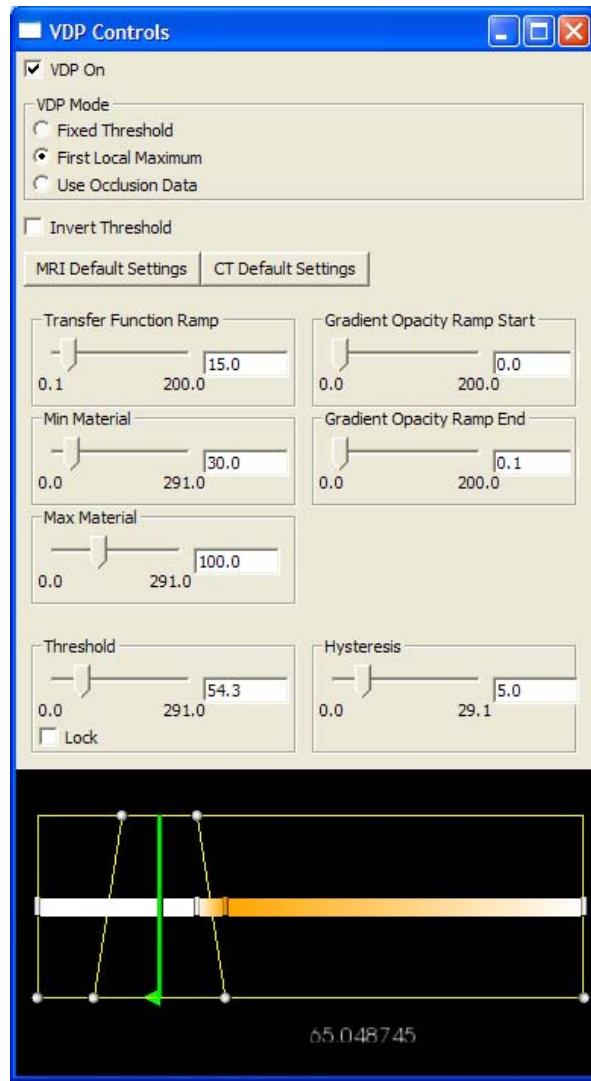


Figure A.6: VDP Controls window.

End sliders.

The settings for the data used for occlusion determination are controlled through the Occlusion Data Controls window (Figure A.7), available in the Controls menu. The radius

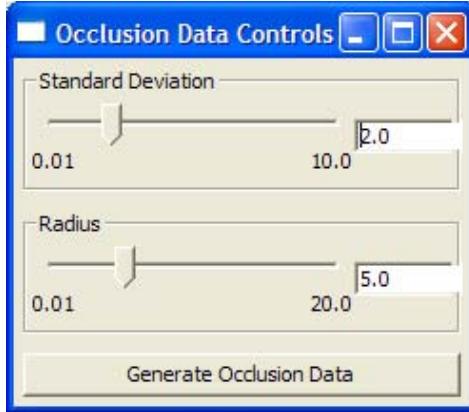


Figure A.7: Occlusion Data Controls window.

of the convolution kernel and standard deviation of the Gaussian within the kernel used to smooth the occlusion data can be set here². If separate occlusion data has been loaded, the smoothing will be applied to it. Otherwise, it will be applied to the original data.

When adjusting the FOR settings, the 3D view updates interactively with a low resolution rendering, and renders a higher resolution image when the interaction is complete.

A.1.2.4 Rendering controls

The settings of the volume renderer are controlled through the Rendering Controls window (Figure A.8, selected from the Controls menu. The version of VTK's software volume renderer to use can be selected, either a standard volume renderer, or a version that uses fixed-point numerical representations and implementing space skipping. Sample distance along the ray can be selected, with different values possible when interacting versus not interacting. The image sample distance can also be selected, although 1 sample per ray is common. The frames per second (FPS) when interacting and not interacting can be set. These two param-

²The radius of the kernel is one-half the width of the kernel, in pixels. An isotropic kernel is used, so it is also one-half the height. The standard deviation of the Gaussian controls the width of the Gaussian used, so a large kernel radius is needed to adequately represent a large Gaussian. However, the convolution is faster with smaller kernels. The kernel radius could be set automatically from the standard deviation of the Gaussian, obviating the need for the kernel radius control.

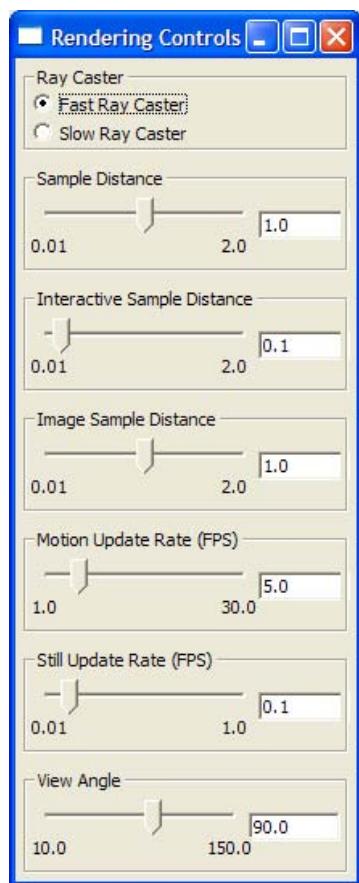


Figure A.8: Rendering Controls window.

eters enable a higher-framerate, lower-quality image when interacting, and a lower-framerate, higher-quality image when not interacting. The view angle of the virtual camera can also be adjusted here.

A.1.2.5 Miscellaneous

The arrangement of the three slice views and the 3D view can be controlled in the View menu. The views can be arranged 2 By 2, the three slice views in a row over the 3D view, or the three slice views in a column to the left of the 3D view. Each individual view can also be selected to be the only view shown. A Help menu informs the user of the controls for interacting with the slice views and the 3D view.

A.2 VTK Extensions

Some extensions to VTK were necessary to implement Innerspace. These extensions were made to incorporate FOR into VTK’s volume rendering framework, and add some of the widgets used for user interaction.

A.2.1 FOR

Subclasses of vtkVolumeRayCastMapper and vtkVolumeRayCastFunction implement FOR using VTK’s standard software ray cast volume renderer. Subclasses of vtkFixedPointVolumeRayCastMapper and vtkFixedPointVolumeRayCastHelper have also been implemented. These classes implement a software ray cast volume renderer that utilizes fixed point arithmetic and space skipping to accelerate rendering speed.

A.2.2 Widgets

Subclasses of vtk3DWidget were written to implement the camera widgets and the transfer function widget. The camera widget takes a vtkCamera widget as an input, enabling it to interact directly with a virtual camera in a 3D scene. Two other camera widgets can also be added, enabling it to update those camera widgets when necessary.

The basic transfer function widget enables creation of a piecewise linear scalar opacity transfer function by adding, deleting, and manipulating control points. It also enables creation of a color transfer function, by adding, deleting, and manipulating control points. The color is linearly interpolated between control points.

A subclass of the basic transfer function widget was also written, adding controls for FOR. These controls consist of a vertical bar that can be moved to set the threshold, and a triangular handle on the bar that is moved up and down to set the amount of opacity modulation performed on the accumulated color and opacity when a separation features is encountered. A subclass of the FOR transfer function widget adds the ability to set the color control points of the color transfer function using a wxColourDialog.

A.2.3 Performance

Innerspace can render typical MRI and CT data sets at interactive rates. Exact performance numbers vary based on hardware, data set, virtual camera position, and rendering settings. Table A.1 gives the rendering performance for a typical shoulder socket view of a 384x384x128 MRI data set, using 10 samples per voxel along each ray, rendered using the fixed point volume ray cast mapper with space skipping on a 3.0 GHz dual processor, dual core Dell Precision 690 with 4.0 GB of RAM.

Image Resolution	Rendering Speed
200x200	7-8 fps
300x300	4.9-5.4 fps
400x400	3.5-3.8 fps

Table A.1: Innerspace Rendering Performance

These performance measures are for full resolution renderings, but lower resolution versions are rendered when interacting to keep a desired frame rate.

Implementing FOR using ray casting on programmable graphics hardware can result in performance improvements. A graphics processing unit (GPU) implementation of fixed-threshold FOR resulted in an 8x performance improvement for the shoulder in Table A.1 when rendered using an nVidia 8800GTX graphics card (Quammen, 2006).

BIBLIOGRAPHY

- Applegate, G. R. (1998). Three-dimensional MR arthrography of the shoulder: An intraarticular perspective. *American Journal of Radiology*, 171(1):239–241.
- Borland, D., Clarke, J. P., Fielding, J. R., and Taylor II, R. M. (2006). Volumetric depth peeling for medical image display. In Erbacher, R. F., Roberts, J. C., Gröhn, M. T., and Börner, K., editors, *Visualization and Data Analysis 2006*. Proceedings of SPIE-IS&T Electronic Imaging, SPIE Vol. 6060, 606004.
- Borland, D., Clarke, J. P., and Taylor II, R. M. (2005). Volumetric depth peeling for virtual arthroscopy. *Electronic Imaging*, 16(2).
- BrainWeb (2007). <http://www.bic.mni.mcgill.ca/brainweb/>.
- Bruckner, S., Grimm, S., Kanitsar, A., and Gröller, M. E. (2006). Illustrative context-preserving exploration of volume data. *IEEE Transactions on Visualization and Computer Graphics*, 12(6):1559–1569.
- Cabral, B., Cam, N., and Foran, J. (1994). Accelerated volume rendering and tomographic reconstruction using texture mapping hardware. In *Proceedings of the 1994 Symposium on Volume Visualization*, pages 91–98.
- Cameron, G. G. and Undrill, P. E. (1992). Rendering volumetric medical image data on a SIMD-architecture computer. In *Proceedings of the Third Eurographics Workshop on Rendering*, pages 135–145.
- Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–698.
- Choi, J. J., Shin, B.-S., Shin, Y. G., and Cleary, K. (2000). Efficient volumetric ray casting for isosurface rendering. *Computers and Graphics*, 24(10):661–670.
- Clarke, J. P. (2004). personal communication.
- Clarke, J. P. (2006). personal communication.
- Cootes, T. F., Edwards, G. J., and Taylor, C. J. (1992). Active appearance models. In *Proceedings of British Machine Vision*, pages 266–275.
- Cootes, T. F. and Taylor, C. J. (1998). Active shape models—smart snakes. In *Proceedings of European Conference on Computer Vision*, pages 484–498.
- Cullip, T. J. and Neumann, U. (1993). Accelerating volume reconstruction with 3d texture hardware. Technical Report TR93-027, Department of Computer Science at the University of North Carolina, Chapel Hill.
- Dachille, F., Kreeger, K., Chen, B., Bitter, I., and Kaufman, A. (1998). High-quality volume rendering using texture mapping hardware. In *Proceedings of the ACM SIGGRAPH/EUROGRAPHICS Workshop on Graphics Hardware*, pages 69–76.

- Diefenbach, P. J. (1996). *Pipeline rendering: Interaction and realism through hardware-based multi-pass Rendering*. PhD thesis, Department of Computer Science, University of Pennsylvania.
- Diepstraten, J., Weiskopf, D., and Ertl, T. (2002). Transparency in interactive technical illustration. *Computer Graphics Forum*, 21(3):317–325.
- Diepstraten, J., Weiskopf, D., and Ertl, T. (2003). Interactive cutaway illustrations. *Computer Graphics Forum*, 22(3):523–532.
- Drebin, R. A., Carpenter, L., and Hanrahan, P. (1988). Volume rendering. In *Proceedings of SIGGRAPH*, pages 65–74.
- Duggan, A. (2007). <http://www.tardis.ed.ac.uk/~ajcd/tales/shoulder.html>.
- Engel, K., Klaus, M., and Ertl, T. (2001). High-quality pre-integrated volume rendering using hardware-accelerated pixel shading. In *Proceedings of the ACM SIGGRAPH/EUROGRAPHICS Workshop on Graphics Hardware*, pages 9–16.
- Everitt, C. (2002). Interactive order-independent transparency. Technical report, Nvidia Corporation.
- Fielding, J. R., Borland, D., Lee, K. H., Clarke, J. P., Wallen, E., Pruthi, R., and Taylor II, R. M. (2006). Virtual pyeloscopy using volumetric depth peeling. *Academic Radiology*, 13:759–763.
- Fischer, J., Bartz, D., and Strasser, W. (2005). Illustrative display of hidden iso-surface structures. In *Proceedings of IEEE Visualization*, pages 663–670.
- Georgetown Pathology (2007). <http://www.georgetown.edu/departments/pathology/main/cases/case1.html>.
- Gibson, J. J. (1979). *The Ecological Approach to Visual Perception*. Houghton Mifflin.
- Griffiths, D. J. (2004). *Introduction to Quantum Mechanics*. Prentice Hall, second edition.
- Hemminger, B. M., Molina, P. L., Braeuning, M. P., Detterbeck, F. C., Egan, T. M., Pisano, E. D., and Beard, D. V. (1995). Clinical applications of real-time volume rendering. In Kim, Y., editor, *Medical Imaging 1995: Image Display*. Proceedings of SPIE, SPIE Vol. 2431.
- Hladuvka, J., König, A., and Gröller, M. E. (2000). Curvature-based transfer functions for direct volume rendering. In *Proceedings of Spring Conference on Computer Graphics and its Applications*, pages 58–65.
- Honigmann, D., Ruisz, J., and Haider, C. (2003). Adaptive design of a global opacity transfer function for direct volume rendering of ultrasound data. In *Proceedings of IEEE Visualization*, pages 489–496.
- Interrante, V., Fuchs, H., and Pizer, S. M. (1997). Conveying the 3d shape of smoothly curving transparent surfaces via texture. *IEEE Transactions on Visualization and Computer Graphics*, 3(2):98–117.

- Kass, M., Witkin, A., and Terzopoulos, D. (1988). Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331.
- Kaufman, A. E., Lakare, S., Kreeger, K., and Bitter, I. (2005). Virtual colonoscopy. *Communications of the ACM*, 48(2):37–41.
- Kindlmann, G. and Durkin, J. W. (1998). Semi-automatic generation of transfer functions for direct volume rendering. In *Proceedings IEEE Symposium on Volume Visualization*, pages 79–86.
- Kindlmann, G., Whitaker, R., Tasdizen, T., and Möller, T. (2003). Curvature-based transfer functions for direct volume rendering: methods and applications. In *Proceedings of IEEE Visualization*, pages 513–520.
- KNEEguru (2007). http://www.kneeguru.co.uk/html/knee/knee_anatomy/bones_front02.html.
- Kniss, J., Ikits, S. P. M., Lefohn, A., Hansen, C., and Praun, E. (2003). Gaussian transfer functions for multi-fields volume visualization. In *Proceedings of IEEE Visualization*, pages 497–504.
- Kniss, J., Kindlmann, G., and Hansen, C. (2001). Interactive volume rendering using multi-dimensional transfer functions and direct manipulation widgets. In *Proceedings of IEEE Visualization*, pages 255–262.
- Lacroute, P. and Levoy, M. (1994). Fast volume rendering using a shear-warp factorization of the viewing transformation. In *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques*, pages 451–458.
- Lalush, D. S. (2006). BMME 560, Medical Imaging II Course Slides, University of Chapel Hill.
- Lee, W., Kim, H. S., Kim, S. J., Kim, H. H., Chung, J. W., Kang, H. S., Hong, S. H., and Choi, J. Y. (2004). CT arthrography and virtual arthroscopy in the diagnosis of the anterior cruciate ligament and meniscal abnormalities of the knee joint. *Korean Journal of Radiology*, 5(1):47–54.
- Levoy, M. (1988). Display of surfaces from volume data. *IEEE Computer Graphics and Applications*, 8(3):29–37.
- Levoy, M. (1990). Efficient ray tracing of volume data. *ACM Transactions on Graphics*, 9(3):245–261.
- Lin, C.-C. and Ching, Y.-T. (1996). Efficient volumetric ray casting for isosurface rendering. *The Visual Computer*, 12(10):515–526.
- Lorensen, W. E. and Cline, H. E. (1987). Marching cubes: A high resolution 3d surface construction algorithm. *Computer Graphics*, 21(4):163–169.
- Mammen, A. (1989). Transparency and antialiasing algorithms implemented with the virtual pixel maps technique. *IEEE Computer Graphics and Applications*, 9(4):43–55.

- Mullicka, R., Bryanh, R. N., and Butmana, J. (2000). Confocal volume rendering: Fast segmentation-free visualization of internal structures. In Mun, S. K., editor, *Medical Imaging 2000: Image Display and Visualization*. Proceedings of SPIE, SPIE Vol. 3976.
- Nagy, Z. and Klein, R. (2003). Depth-peeling for texture-based volume rendering. In *Proceedings of 11th Pacific Conference on Computer Graphics and Applications*, pages 429–433.
- Osher, S. and Sethian, J. A. (1988). Fronts propagating with curvature dependent speed: Algorithms based on hamilton-jacobi formulations. *Journal of Computational Physics*, 79:12–49.
- Parker, S., Shirley, P., Livnat, Y., Hansen, C., and Sloan, P.-P. (1998). Interactive ray tracing for isosurface rendering. In *Proceedings of IEEE Visualization*, pages 233–238.
- Phong, B. T. (1973). *Illumination of Computer-Generated Images*. PhD thesis, Department of Computer Science, University of Utah.
- Porter, T. and Duff, T. (1984). Compositing digital images. *Computer Graphics*, 18(3):253–259.
- Prince, J. L. and Links, J. M. (2006). *Medical Imaging Signals and Systems*. Pearson Prentice Hall.
- Quammen, C. (2006). Volumetric depth peeling on graphics hardware. Final project writeup, COMP 870 Advanced Image Generation, The University of North Carolina at Chapel Hill.
- Rezk-Salama, C. and Kolb, A. (2006). Opacity peeling for direct volume rendering. *Computer Graphics Forum*, 25(3):597–606.
- RHL (2007). <http://www.uab.edu/reynolds/MajMedFigs/Rontgen.htm>. Reynolds Historical Library, University of Alabama at Birmingham.
- Sabella, P. (1988). A rendering algorithm for visualizing 3d scalar fields. In *Proceedings of SIGGRAPH*, pages 51–58.
- Sahin, G., Dogen, B., and Demirtas, M. (2004). Virtual MR arthroscopy of the wrist joint: A new intraarticular perspective. *Skeletal Radiology*, 33(6):9–14.
- SCOI (2007). <http://www.scoi.com/sholanat.htm>. Southern California Orthopedic Institute.
- SEER (2007). http://training.seer.cancer.gov/module_anatomy/unit1_3_terminology2_planes.html. Surveillance Epidemiology and End Results, National Cancer Institute.
- Sethian, J. A. (1996). A marching level set method for monotonically advancing fronts. In *Proceedings of National Academy of Sciences*.
- Stoller, D., Tirman, P., and Bredella, M. (2004). *Diagnostic Imaging Orthopaedics*. Amirsys.
- Tam, R. C., Healey, C. G., Flak, B., and Cahoon, P. (1997). Volume visualization of abdominal aortic aneurysms. In *Proceedings of IEEE Visualization*, pages 43–50.

- Tory, M., Potts, S., and Moller, T. (2005). A parallel coordinates style interface for exploratory volume visualization. *IEEE Transactions on Visualization and Computer Graphics*, 11(1):71–80.
- Viola, I., Kanitsar, A., and Gröller, M. E. (2004). Importance-driven volume rendering. In *Proceedings of IEEE Visualization*, pages 139–146.
- Wan, M., Tang, Q., Kaufman, A., Liang, Z., and Wax, M. (1999). Volume rendering based interactive navigation within the human colon. In *Proceedings of IEEE Visualization*, pages 397–400.
- Ware, C. (2004). *Information Visualization: Perception for Design*. Morgan Kaufmann Publishers, second edition.
- Weishaupt, D., Wildermuth, S., Schmid, M., Hilfiker, P. R., Hodler, J., and Debatin, J. F. (1999). Virtual MR arthroscopy: New insights into joint morphology. *Journal of Magnetic Resonance Imaging*, 9(6):757–760.
- Westover, L. (1989). Interactive volume rendering. In *Proceedings of the Chapel Hill Workshop on Volume Visualization*, pages 9–16.
- Zuiderveld, K. J., van Ooijen, P. M. A., Chin-A-Woeng, J. W. C., Buijs, P. C., Olree, M., and Post, F. H. (1996). Clinical evaluation of interactive volume visualization. In *Proceedings of IEEE Visualization*, pages 367–370.