

2014

MODEL-BASED 3-D RECONSTRUCTION OF TRANSPARENT OBJECTS FROM A SINGLE IMAGE

Wiehrs Lewis Collier
University of Rhode Island, wlewiscollier@gmail.com

Follow this and additional works at: https://digitalcommons.uri.edu/oa_diss

Terms of Use

All rights reserved under copyright.

Recommended Citation

Collier, Wiehrs Lewis, "MODEL-BASED 3-D RECONSTRUCTION OF TRANSPARENT OBJECTS FROM A SINGLE IMAGE" (2014). *Open Access Dissertations*. Paper 260.
https://digitalcommons.uri.edu/oa_diss/260

This Dissertation is brought to you by the University of Rhode Island. It has been accepted for inclusion in Open Access Dissertations by an authorized administrator of DigitalCommons@URI. For more information, please contact digitalcommons-group@uri.edu. For permission to reuse copyrighted content, contact the author directly.

MODEL-BASED 3-D RECONSTRUCTION
OF TRANSPARENT OBJECTS FROM A SINGLE IMAGE

BY

WIEHRS LEWIS COLLIER

A DISSERTATION SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY
IN
COMPUTER SCIENCE

UNIVERSITY OF RHODE ISLAND

2014

DOCTOR OF PHILOSOPHY DISSERTATION
OF
WIEHRS LEWIS COLLIER

APPROVED:

Dissertation Committee:

Major Professor Jean-Yves Hervé

Lutz Hamel

Lenore Martin

Joan Peckham

Nasser H. Zawia
DEAN OF THE GRADUATE SCHOOL

UNIVERSITY OF RHODE ISLAND

2014

ABSTRACT

This dissertation describes an interdisciplinary research project that spans computer-vision, imaging of biological nanoparticles via cryogenic transmission electron microscopy (*cryo-tem*), and bio-informatics / proteomics. This research introduces a new method for reconstructing *individual* biological particles imaged via *cryo-tem* with typical accuracies of 88% to 98%. This individual particle reconstruction (IPR) technique is based on a hidden Markov model derived from the underlying biological structure. The processing presented in this dissertation reconstructs each individual particle from a *single* image, which is in contrast to current approaches that can require tens of thousands of images to create an ensemble representation of a particle class. The presented approach also has an internal intermediate result structure that provides confidence measures and can lead to updates of the model based upon further statistical analysis. As this interdisciplinary research represents a new area of reconstruction processing, this abstract serves as both an executive summary of the research and as an introduction to concepts that may be unfamiliar to practitioners of one (or more) of the disciplines.

The main computer-vision focus of this dissertation presents the IPR algorithm, which is a new model-based technique for reconstructing a transparent object from a single image. Model-based 3-D object reconstruction algorithms operating on 2-D projection images require a model that provides the basis for understanding the representation of the imaged object in the scene. Existing *cryo-tem* reconstruction approaches rely upon averaging of classified images to reduce noise. Improvements in these iterative approaches occur by omitting those images that do not meet the latest reconstruction classification constraints. Another approach uses statistical methods in order to determine a most probable fit to averaged

images of a particle class' members. But, again, this processing reconstructs an ensemble representation. The IPR processing presented here uses a higher-order hidden Markov model schema, with levels of states and durations in the states, to define relationships between the image and the candidate reconstruction object in three dimensions without the need for image averaging. IPR processing also allows for a non-iterative solution that grows only linearly in time with reconstruction resolution. Experiments on projection images of simulated transparent objects, with added noise modeled from actual *cryo-tem* images, allowed for reconstruction of each object from a single image. These reconstructions matched the original voxel space up to 98% when compared on a voxel-by-voxel basis.

The imaging modality used for this research is *cryo-tem*, which is a mainstream method for molecular level analysis in nanotechnology and proteomics. *Cryo-tem* has achieved this status because flash-vitrification can capture the physical state and phase of unstained target particles in the projection imagery. Transparent imaging modalities (e.g. *cryo-tem*) can offer more reconstruction options than imaging of opaque objects since both the external shell and the internal particle structure contribute to the overall projection image. The *cryo-tem* reconstruction problem previously has been solved by various methods that average images. These strategies can utilize up to 50,000 distinct images of the same type of particle. While these methods provide a detailed ensemble representation for a single particle class, the resulting depiction is *not* a reconstruction of an individual particle. Besides the desire for reconstruction of individual particles, the *cryo-tem* imaging technique itself has constraints that drive the selected processing goals of this research. The electron beam used in *cryo-tem* leads to time-dependent sample damage that escalates as the electron-beam energy is increased to provide higher and higher resolutions of the particles in the image. The presented approach meets

the goal of requiring only a single high-resolution image of each particle in order to reconstruct each individual particle.

The bio-informatics / proteomics component of this research provides insights into the requirements for models used in the IPR processing. Features of the model arise from the underlying biological structure of the object of interest. These biological structure features include items such as information about the molecular composition, which is obtained by orthogonal and routine analytical techniques. Given their importance to health issues and their basic 3-part composition, low density lipoprotein (LDL) particles were used as the basis family of transparent nanoparticles. The processing approach at the core of this research uses a current biological hypothesis of the details of the LDL macroassembly structure to derive the higher-order HMM schema. A simplified initial approach to modeling the composition of the LDL particles used the three distinct states {phospholipid head groups, lipids/cholesterol, and protein} as well as a fourth state for the buffer solution. These states were used because it is well known that these components exist in quite different electron density and/or localization regimes. After initial trials, the algorithm was further refined by including the duration that each of the multiple parallel Markov chains stayed in each state as each chain traversed through the particle model. IPR processing of model sizes versus target sizes was performed over the expected radius range for LDL particles. These trials resulted in voxel to voxel matching accuracies greater than 88% from a single 2-D image, even with typical *cryo-tem* image noise added to each target's simulated projection image. More importantly, these reconstructions matched the actual target geometry parameters within a pixel of the actual values. This accuracy implies that a histogram of these parameters (e.g. radius, height, etc.) could be generated to glean a better understanding of the particle structures. It is

expected that the results of this research can be used to help produce statistical data for a population of real images of LDL particles. The ultimate goal for this research is that correlations of these LDL sizes and shapes to related known health markers could lead to an improved diagnostic strategy to predict the risk of coronary heart disease / arteriosclerosis. And, while not directly addressed, the innovative method described here could have other uses for analyzing orthographic projection images of transparent objects. These uses range from general proteomic studies and quality assurance for drug delivery nanoparticles to dental x-rays and other medical and manufacturing inspection imaging opportunities.

ACKNOWLEDGMENTS

As with all major efforts, there are many people to acknowledge for helping me reach this milestone. Luckily this isn't the Oscars where a band will start playing if I go on too long.

I must start with my major advisors Dr. Jean-Yves Hervé and Dr. Lenore Martin. Who would have thought that the little project from your computer vision and bioinformatics courses would turn into such a monster? Your encouragement, guidance, inspiration, and support were absolutely needed and much appreciated. We may have to keep up our evening meetings lest we not know what to do with our Wednesday or Thursday nights.

Dr. Hervé, while the end result may not be what you were thinking of at the time, thank you for suggesting that I investigate hidden Markov models. From my description of how I could “see” that the information was there, but a linear solution would be even sparser than the images we would have to process, your experience matched hidden Markov modeling as a candidate methodology. Your guidance in this and in other methodologies to study and our discussions about these and other matters were insightful and much appreciated.

Dr. Martin, thanks for your presentation of the problem and all of the underlying biological underpinnings. I am sure you agree with me when I say that projects like this are the epitome of collaborations between biological and computer scientists. Along the way we got to play with tinker toys and Mr. Potato Head's rings at ACM-BIB-2013. And, of course, putting a cholesterol projection in terms of Oreo cookies and their fat-filled creme fillings only seems appropriate. Thanks for explaining all the biology terms and mechanisms during our discussions about how this all works.

I also need to thank Dr. Joan Peckham and Dr. Lutz Hamel for their contributions to my committee and this effort. Besides many insightful conversations over the years, Dr. Peckham's advice to start writing early was essential. Dr. Hamel's knowledge of machine learning was also greatly appreciated, even though I was unable to utilize SVM's for this effort.

To all of my defense committee members, I thank you for your time in reading and reviewing my research and this dissertation. Your comments and discussions have helped to make this a better document and will help guide the continuance of this research.

As with any project of this duration, there have been colleagues who have impacted this research. Mr. William DiSanto and Mr. Peter Morley provided direct support with some of the coding and in many discussions that ultimately helped lead to a better understanding of the problem. Mr. Jacob Rooney was also instrumental in coding with wxWidgets as well as helping to set up the software environments and keeping me up to date with the latest versions of wxWidgets and Microsoft Visual Studio. Gentlemen, many thanks!

I must also thank the team at Capstone Visual. While I didn't miss too many Friday lunches at Wood River, and I always led the charge to play frogger as we "hit the deck", I counted on you all to carry the load in the office whilst I toiled away at this project, and you did.

Thanks to Jeff Lynne, The Move, Electric Light Orchestra, The Traveling Wilburys, and all his fellow music makers for all the wonderful songs; and to David Nesbit for introducing us way back when. Many hours of this fine music was consumed, but no musicians were harmed, during the conduct of this research and the writing of this dissertation.

I also extend a hearty thank you to the staff of the Computer Science Department for all they do to keep the department functioning, and to help students such as myself navigate the maze. A special thanks goes to Ms. Lorraine Berube for all her help. Between scheduling times and rooms for CSC 305, 505, 509, and 522, and making sure I filled out all the correct forms for the oral and written Ph.D. exams, she always had time to chat and keep me up to speed on the happenings around campus and within the department. I will certainly miss our chats. I'm glad to have met her and to have graduated before she retired ☺. PS, I hope you enjoyed your vacation down-under.

And, finally, I must also acknowledge my “girls” Zara and Hannah. Your journeys to Bat-Mitzvah and Black-Belt provided me with the inspiration to begin this project. The successful completion of your experiences encouraged me along the way. And your achievements in your academic and personal lives buoyed me to finish this.

DEDICATION

*For my Leopard Lady,
Cydney.*

*I appreciate all of your love, support, patience,
and everything else you did for our family
that helped me accomplish this.*

Love, your Lewy Bear...

PREFACE

This dissertation was produced in \LaTeX and \BibTeX using \MiKTeX version 2.9 with the build option [$\LaTeX \Rightarrow \text{PS} \Rightarrow \text{PDF}$]. The \LaTeX *svn-multi* package was used to glean SVN tracking information. The style `urithesis.cls` as provided in [1] was used to provide the \LaTeX and \BibTeX formatting details.

This revision of the dissertation has the following properties:

Author	LCollier
Revision	149
Rev Date	2014-01-10 12:54:07 -0500 (Fri, 10 Jan 2014)
Print Date	Fri 10 th Jan, 2014 13:02
urithesis.cls Version	1.13
urithesis.cls Date	2011/08/23

TABLE OF CONTENTS

ABSTRACT	ii
ACKNOWLEDGMENTS	vi
DEDICATION	ix
PREFACE	x
TABLE OF CONTENTS	xi
LIST OF TABLES	xvi
LIST OF FIGURES	xvii
CHAPTER	
1 Introduction	1
1.1 Model-Based Vision Processing	3
1.1.1 Transparent Objects	5
1.1.2 Associated Problems	6
1.2 Biological Impetus	6
1.2.1 Cryogenic Transmission Electron Microscopy	7
1.2.2 LDL Particles	8
1.3 Problem Statement	10
1.3.1 Geometry Orientation	10
1.3.2 Parameters	11
2 Review of Literature	15
2.1 General 3-D Reconstruction Techniques	15

	Page
2.1.1 Stereographic Reconstruction	16
2.1.2 Reconstruction From Multiple Views	17
2.1.2.1 Shape From Surface Properties	18
2.1.2.2 Space Carving	20
2.1.2.3 Tomography	21
2.2 Existing <i>Cryo-tem</i> Transparency Reconstruction Techniques	22
2.2.1 Underlying SPR Issues	23
2.2.1.1 Fourier Slice Theorem	24
2.2.1.2 Contrast Transfer Function	24
2.2.2 SPR Algorithms	25
2.2.2.1 Angular Reconstitution	25
2.2.2.2 Single particle tilt series	26
2.2.2.3 Random Conical Tilt	26
2.2.3 Available Software	27
2.2.3.1 EMAN	27
2.2.3.2 IMAGIC	29
2.3 Prior LDL Shape Studies	29
2.4 Computer Vision and Models	32
2.4.1 Solid Model Matching	33
2.4.2 Geons and Superquadrics	34
2.4.3 Transparent Model Matching	39
2.5 Statistical Models	41
2.5.1 Bayesian Method	42

	Page
2.5.2 Hidden Markov Models	43
2.5.2.1 HMM Problem 1	49
2.5.2.2 HMM Problem 2	52
2.5.2.3 HMM Problem 3	55
2.5.2.4 Other Types of HMM	55
3 Methodology	58
3.1 HMM Reconstruction Processing Concepts	59
3.1.1 Basic FOHMM Reconstruction Processing	61
3.1.2 Simple HOHMM Processing	64
3.1.3 Training the HMM	64
3.2 Extended HMM Reconstruction Processing	66
3.2.1 Initial State Statistics	66
3.2.2 State to State Statistics	67
3.2.3 Flattening the HOHMM	71
3.2.4 State to Observation Statistics	74
3.2.5 Alpha Pass	80
3.2.6 Beta Pass	83
3.2.7 Gamma Pass	84
3.2.8 Most Probable State Sequence	85
3.2.9 Merging State Sequences	86
4 Findings	88
4.1 Hidden Markov Modeling	88
4.1.1 First Order HMM	89

	Page
4.1.2 Simple Extended HOHMM Processing	93
4.1.3 Extended HOHMM Processing	95
4.1.4 Sequence Direction	96
4.1.5 State Sequence Merging	99
4.2 Noise	101
4.2.1 No Noise	101
4.2.2 Single Training Run	101
4.2.3 Multiple Training Runs	102
4.2.4 Modeled Training	102
4.3 Geometry Parameters	106
4.3.1 Size Changes	106
4.3.2 Density Changes	108
4.3.3 Shape Changes	108
5 Conclusion	111
5.1 Technology Remarks	111
5.2 Future Efforts	113
5.3 Impact	114
LIST OF REFERENCES	116
 APPENDIX	
A Processing Details	125
A.1 ARTEMIS Software Framework	125
A.1.1 genMRC	127
A.1.2 cvt2MRC	127

	Page
A.1.3 genProjections	127
A.1.4 genObservations	128
A.1.5 hmmSolver	133
A.2 Simulation of Projection Images	135
A.2.1 Orthographic Projections	139
A.2.2 Projection Image Pixel Values	140
A.3 Linear Reconstruction Representation	143
B Constructions	147
B.1 Image Rotation Limits	147
B.2 Tuple To Tuple Probabilities	148
Index	152
BIBLIOGRAPHY	153

LIST OF TABLES

Table		Page
1	Hidden Markov Model Notation	46
2	Extended HMM State Example	69
3	45° Reconstruction Voxel Accuracy	99
A.4	<code>genObservations</code> Software Specification Synopsis	129
A.5	Matrix Solution Parameters	145
B.6	Example Tuple to Tuple Chains	149

LIST OF FIGURES

Figure		Page
1	Example LDL <i>Cryo-tem</i> Image	7
2	Small Angle <i>Cryo-tem</i> Image Acquisition	12
3	Problem Definition Examples	13
4	Geometry Orientation	14
5	2010 LDL <i>Cryo-tem</i> Reconstruction	31
6	Simple Disc Model and Projection Image	39
7	Representative Hidden Markov System	45
8	Forward Algorithm Lattice Meshes	50
9	Single Image Reconstruction Geometry	60
10	First Order HMM Statistics Generation	62
11	Example State-to-Observation Statistics	65
12	Higher Order HMM Statistics Generation	68
13	Expanded HOHMM Trellis Structure	70
14	Extended HOHMM Processing Directions	75
15	Bilateral Interpolation for HMM Observation Generation	76
16	Model State Limitations	78
17	Rotated Reconstruction Spaces	80
18	Single Fan of Forward α -Pass Trellis	81
19	Single Fan of Backwards β -Pass Trellis	83
20	First Order Vertical HMM Reconstruction	89
21	First Order Horizontal HMM Reconstruction	90

Figure		Page
22	Simple HMM Processing Results	91
23	Extended Reconstruction Results for Simple Discoid Model . .	94
24	Rotated Reconstruction Results for Simple Discoid Model . .	95
25	Extended Reconstruction Results for Simple Discoid Model . .	96
26	Extended Reconstruction Results	98
27	Projection Image at 45° with Noise Added	100
28	Merged Reconstruction	100
29	Reconstruction with Noise	103
30	Histograms of Observations	103
31	Simulated HMM Observation Probabilities	104
32	Enhanced Simulated HMM Observation Probabilities	106
33	Different Training vs. Processing Radii Matching	107
34	Warped Disc Model	109
35	Warped Target Images	109
A.36	Artemis Processing Flow	126
A.37	Sub-voxels Needed for Observation Independence	130
A.38	Observation Independence	132
A.39	Simulated Target, Voxel Space, and Geometry Orientation . .	135
A.40	Generation of Projection Images from a Simulated Target . .	137
A.41	Extension of m_0 to M_0	138
A.42	Simple Discoidal Models	142
A.43	Simple Discoidal Projection Images	142
A.44	Simulated Projection Image Creation	143

Figure		Page
B.45	Image Rotation Limits	147

CHAPTER 1

Introduction

The research presented in this dissertation studied the problem of 3-Dimensional (3-D) reconstruction of an individual transparent object from a single 2-Dimensional (2-D) image. The term “individual” is used instead of “single” to prevent confusion with the existing “single particle reconstruction” (SPR) method name. In IPR processing, only a single orthographic projection image of the transparent object is available to be used in the reconstruction of the original object. Since the expected *cryo-tem* imaging modality results in increased damage to the target on each successive image acquisition, it is very desirable to perform the reconstruction with just a single image.

This single projection approach is in contrast to the currently available 3-D reconstruction techniques for transparent objects. These methods include acquisition and processing of many thousands [2, 3] of images of an ensemble of assumed similar targets taken at varying spherical angles. These techniques allow for a blended representation of all the targets to be reconstructed via methods such as SPR or tomography. In these methods, the abundance of images allows for noise reduction through traditional image processing means such as averaging. For averaging to work, however, a statistically significant sample is required of the same or similar targets. The research presented in this dissertation was directed towards the special case where only a single image of the same individual target is available.

The availability of just a single projection image may be due to image acquisition and/or processing needs. The limit to one image can come from issues such as damage to the target from the imaging modality or an inability to control the range of available projection imaging angles. Besides limiting the number of

images to one, beam damage may also lead to increased noise in successive images if the target and the surrounding medium are deformed by the act of imaging. Another significant reason for the limit of one image is when the non-uniformity of the targets does not allow for acquisition of an appropriate set of randomly oriented images. In all of these scenarios, the requisite thousands of images are not available or processing such as averaging may result in an improper understanding if discrete measurements such as a histogram of geometric parameters for these objects are required.

The single image of each target, however, does not mean that useful information is unavailable. The transparent nature of the targets means that each image can contain details about the internal structure of the target. The ability to see into and *through* the target makes this method of analysis more useful than simple sparse methods such as stereo imaging of opaque objects. This reconstruction also may allow for more detailed information to be gleaned from the samples. These additional parameters of individual particles can be more useful than ensemble metrics from other analysis methods such as light scattering or computer vision processing such as particle analysis. Thus, 3-D reconstruction of transparent objects from a single image can provide useful information.

The transparent nature of the object and the imaging acquisition technique can make other processing techniques intractable. Even if multiple images are available, simple registration processing becomes much more difficult since the image acquisition damage to the target, that increases with each image, leads to differing similarity measures in each successive image. If a single image is used for the reconstruction processing, the V^3 transparent voxels, the N state choices for each voxel, and the M possible values for each pixel in the projection image allow for many combinations that of states that result in the same observation

value. However, with just a single image, there are no extra variables against which to solve the under-determined equation. Existing approaches also expect noise reduction through averaging and other processing that is unavailable with just a single image.

This dissertation is organized in the following manner. The remainder of this chapter provides a more detailed introduction to the research described in this dissertation. The discussion begins with an overview of model-based vision processing in Section 1.1. Since there is a strong biological impetus for this research, Section 1.2 provides additional information regarding the nature of the underlying image acquisition and its uses. The chapter concludes in Section 1.3 with a formal statement of the transparent object reconstruction problem to be solved and the expected parameters to be gleaned from the reconstruction. Chapter 2 reviews related literature. Chapter 3 presents the algorithm developed for this effort. Chapter 4 discusses the finding of this research. Chapter 5 provides concluding remarks and lists possible next steps for continuing the research.

1.1 Model-Based Vision Processing

Generally speaking, the information gathering and processing performed by the human visual system in order to perceive the world in three dimensions has not been matched yet by digital machine processing that is known as computer vision. In human and computer vision, the reception of images and the processing of the information embedded in the images leads to a 3-D reconstruction of the scene. This 3-D reconstruction provides both relative location information for the objects and an understanding of the objects themselves. The complexity of 3-D reconstruction is further affected by both the viewer's innate understanding of the scene and the objects in the scene as well as by the "optics" through which the scene is observed.

3-D reconstruction relies on an understanding of the world being viewed and of the imaging system being used to view the world. Many people remember the narrowing of train tracks from a first art class in perspective, or the road and characters disappearing into the distance in a Saturday morning cartoon. But how this works in human vision has been very hard to implement in computers. Human brains implicitly “know” the size of objects from a given range of distances based on a learned set of rules. When the view of “known” objects appear smaller, the brain “understands” that they are farther away than when they appear larger. By using a model of the items in the image, the human visual processing system infers size, shape, and distance information from just the two images received through our eyes.

A second reason for the complexity of 3-D reconstruction via computer vision is the manner in which the image is received by the processor. As noted by Trumbull [4], directors of 3-D movies must take into account the fact that humans are forced to focus on a flat plane when viewing a movie. Since this is contrary to the expected depth of focusing that normal human vision enjoys, the movie maker must adjust the objects in the image so that when the eye focuses on the flat screen, objects appear in the correct location, with the correct scale, and proper proportions so that the human vision processing system “sees” the scene as it would in the *real* 3-D world. This adjustment in movies can be accomplished through a combination of both sensors and processing. Note that this processing, which can cost hundreds of millions of dollars and can take decades of computer processing time is, in fact a form of 3-D reconstruction. 3-D systems such as this often use a pair of cameras to capture two simultaneous images of the scene. The scene is then reconstructed into three dimensions within the computer so that it can then be projected back on the screen. This level of processing in humans is more amazing

when one considers that the human vision system absorbs all the 3-D information through the eyes and performs all of this processing in the brain in real-time.

The culmination of 3-D reconstruction requirements is the need for a model against which the vision processor can process its observed data. In human vision processing, the brain stores the models of known artifacts in the real world. In computer vision, the processor must be presented with models of the artifacts or the algorithm must be based upon a model by which the observed data can be analyzed so that the reconstruction processing can occur. For computer vision processing, this equates to a nearly infinite set of tests to determine the distance based on the geometric calculations while the human vision system “just does it.”

1.1.1 Transparent Objects

There are, however, some areas in which computer vision can match, or even exceed human vision. By taking advantage of imaging means other than visible light, non-human sensors can see through objects that are normally opaque. For example, X-rays and electron beams can penetrate many objects of interest to capture the internal structure of those objects. Humans and computers can view the 2-D projection images but computers hold an advantage for this type of processing. Since generally there is no model of the object in the brain, humans cannot “see” the 3-D structure of these transparent objects.

Computers also have no innate understanding of the internal structure but they are better suited for the intense and repetitive analysis required for this type of 3-D reconstruction. Computers can execute algorithms on a training data set that generate the model of the transparent object. This generated model can then be applied in order to preform a 3-D reconstruction of the transparent object. Or, many different supplied models can be tested against the observed structure in order to provide for a “best fit” reconstruction. In a recent paper [5], over

50,000 2-D projection images were analyzed in order to create a representative 3-D reconstruction of a biological particle. Clearly humans are not suited for this level of repetitive vision processing so there are ample opportunities for computer vision to perform 3-D reconstruction of transparent objects. The problem then becomes one of finding the best model for the reconstruction and an efficient means of computing and applying the model.

1.1.2 Associated Problems

There are many problems that fall into this category of processing. Most any imaging regime that results in a small number of images (e.g. mammograms, dental X rays, nanoparticle studies, or proteomics) is a candidate. But, since a model is expected, certain examples may not be as suitable. Mammograms, while offering a view into a transparent region, may rely more on the cells detected rather than the mammary mass itself as the target of interest. If a model of cancer cells is known, then the tissue mass could be considered the suspension medium and the parameters of the cancerous region could be reconstructed. Dental x-rays would follow the same model with air serving as the solution. Or, with a proper model, caries could be suspended in the enamel. The use of a model based approach for providing geometric parameters for an ensemble of particles is the thrust of the research. Biological particles can be considered a sub-class of nanoparticles, so the focus of the research is directed towards assisting in this arena.

1.2 Biological Impetus

Since the motivation for this research has its basis in proteomics, an understanding of the biological underpinnings provides an appreciation of the constraints on the solution space. This section summarizes the biological basis for this research. These constraints are used in Section 1.3 to present the generalized problem.

1.2.1 Cryogenic Transmission Electron Microscopy

A classic example of the reconstruction of non-uniform transparent objects is imaging of biological and nanoparticles via cryogenic electron microscopy (*cryo-tem*). See [6] for a historical and theoretical review of *cryo-tem*. (Note that this dissertation uses the term *cryo-tem* instead of the alternative *cryo-em* in order to emphasize the transparent nature of *transmission electron microscope* imaging.) In this processing, the targets are flash-frozen from their natural state so that structure may be preserved. An example from [7] of *cryo-tem* of low-density lipoprotein (LDL) particles is shown in Figure 1. In this example, a single LDL particle was

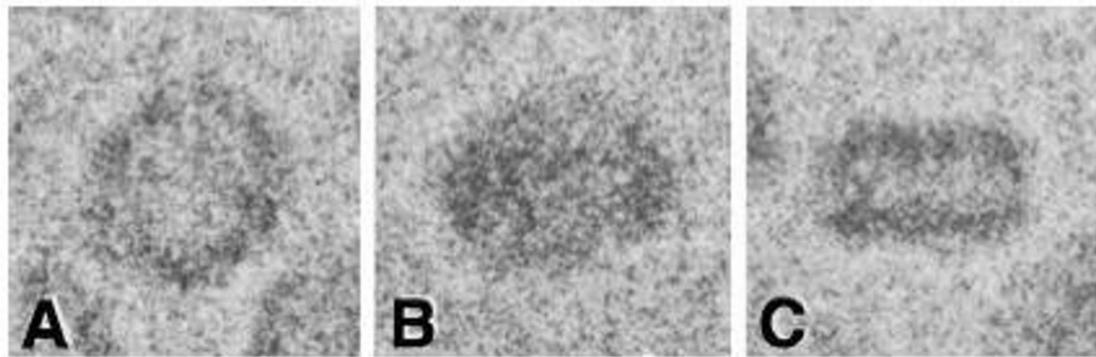


Figure 1. Example LDL *Cryo-tem* Image (from [7])

imaged at relatively large rotation angles. To acquire this set of images, the target was identified and imaged at multiple rotation angles. A diagram of this scenario, but with smaller rotation angles, is shown in Figure 2. Note the requirement to keep the target(s) within the focus region of the apparatus may require translation of the sample in order to preserve focus.

In the small angle case, the *cryo-tem* sample holder is rotated through a limited range of angles. These small angles allow for a reduction in the damage to the biological sample by the electron beam since refocusing may not be required. The small rotation angles, however, limit the amount of information that may be extracted from the orthographic projections of the particles in order to determine

the geometric parameters of the particles. This constraint leads to a natural trade-off between sample integrity and processing requirements. Starting from images acquired with the small-angle rotations, processing can be applied in order to obtain an improved mapping between geometric parameters of a 3-D model and the rotated 2-D projection images. This mapping can be used for the extraction of pertinent geometric information that can lead to an understanding of the internal structure. An obvious desire from this observation is a processing scheme that can use but a single image of the particle so that refocusing is eliminated and varying beam damage between images is eliminated.

1.2.2 LDL Particles

The biologic targets of interest are lipoprotein particles. Lipoprotein particles are a combination of lipids (fat and cholesterol) and proteins. Together, the lipids and proteins assemble into a macromolecular vesicle that acts as a carrier for the distribution of insoluble cholesterol in serum (blood plasma) throughout the bodies of all animals. (LDL) macromolecules serve to carry cholesterol to the body so that the cholesterol can serve its many essential functions. High-density lipoprotein (HDL) acts as a scavenger of cholesterol from the body, providing transport back to the liver and intestines for processing and elimination.

As coronary heart disease (CHD) is the single largest killer of Americans [8], improved means of detecting risk factors *before* arterial obstructions appear could lead to a improvement in quality of life with a reduced cost. It is generally accepted that HDL particles are smaller in average diameter (than LDL) with more protein than fat content while LDL particles are quite a bit larger in diameter (than HDL) with much more fat than protein. Current medical thinking is that large numbers of LDL correlate with poorer cardiovascular health while larger numbers of HDL are considered as a marker for lower risk of arteriosclerosis.

Research over the past decade, however, has suggested that geometric parameters of lipoprotein macromolecules may better correlate with cardiovascular risk than just the particle counts and the associated density of particles in the serum. Newer tests [9] that are based upon Nuclear Magnetic Resonance (NMR) spectroscopy can provide a count of the number of LDL particles in size bins but not actual LDL particle sizes. In general, LDL particles that fall on the large end of the LDL particle size spectrum are tied to better health while smaller LDL particles correlate to worse health. In a study of diabetics [10], changes in the LDL sizes and plasma lipid levels accounted for part of the antiatherogenic effect of fenofibrate in type-2 diabetes. In this study, increases of only 0.98nm in average LDL diameters (which is in the range of %5 to %10 of expected particle size) showed a detectable change in atherogenesis. LDL packages that are at the small end of the overall LDL size spectrum are considered to be an important cause of arterial plaque initiation that can eventually lead to heart attacks and stroke. And somewhat surprisingly, even though conventional wisdom states that HDL is “good” cholesterol, another study [11] showed that high concentrations of smaller sized HDL particles resulted in a 15-fold increase in the risk of heart disease. These studies lead to the conclusion that more is at play than just the size of the particles: shape and/or other geometric properties may be important.

The characteristic parameters of LDL *and* HDL packages are important to a better understanding and prediction of heart disease risk. Geometric parameters other than average diameters or density measurements may one day prove more important to achieving a better understanding of and prediction of heart disease risk. By allowing for efficient generation of a histogram of parameters from LDL and HDL particles found in blood samples, information derived from *cryo-tem* images may be correlated to observed cardiovascular state in order to assist in the

determination of essential relationships between lipoprotein geometry and overall cardiovascular health. Tools that can provide these parameters can also be used for other proteomic studies as well.

1.3 Problem Statement

The problem can be described as performing a 3-D reconstruction of an individual 3-D target in a volume of real space that is imaged via a single orthographic projection through the object, with the object at a random orientation relative to the image plane. The target is not opaque to the imaging technology being used so that the pixels of the projection image are created by passing the imaging energy through the target sample. A representation of this scenario is shown in Figure 3. In this example, the target (blue discoid object) is completely contained in the $v_0 \times v_0 \times v_0$ volume space. The orthographic projections give rise to an image that also completely contains the target projection that is $m_0 \times m_0$ pixels in size.

Figure 3 shows the target as fixed and the image planes of separate projection images as not being co-planar but this is not an important issue in the problem definition. In some cases (e.g. *cryo-tem*) the image plane is fixed and the object is rotated and translated in free space. In other cases (such as dental x-rays), the image plane may move around a fixed target (i.e. a tooth). Thus, the target origin can be considered to be fixed in space and the projection plane can move or the converse where the image plane is fixed and the object is free to rotate around its center.

1.3.1 Geometry Orientation

The geometry to be used for the reconstruction, which is shown in Figure 4, is a standard right-handed x, y, z graphics co-ordinate system. The positive x axis extends to the right and the positive y axis points upwards so the positive z axis

points out of the $x-y$ plane towards the viewer. The 3-D target, shown generically as a discoid, is centered at the origin. Due to the rotation and translation symmetry noted above, this is the same problem as shown in Figure 3. Figure 4 shows a single instance of the same discoid target without the associated reconstruction space. As will be seen, the reconstruction space can be rotated as well so it is removed here for simplicity.

1.3.2 Parameters

The desired solution is a set of densities for the $v_0 \times v_0 \times v_0$ volume space. It is assumed that the density of the non-target portion of the volume is sufficiently different from the target so that the boundary of the target can be distinguished. For some targets, such as dental x-rays, the difference in tooth to air is substantially different. For the *cryo-tem* case, this means that the electron density of the buffer differs from the target, or at least the outer shell of the target. As is seen in Figure 1, this is a fairly reasonable assumption, but care needs to be taken in sample preparation to ensure this constraint is obeyed.

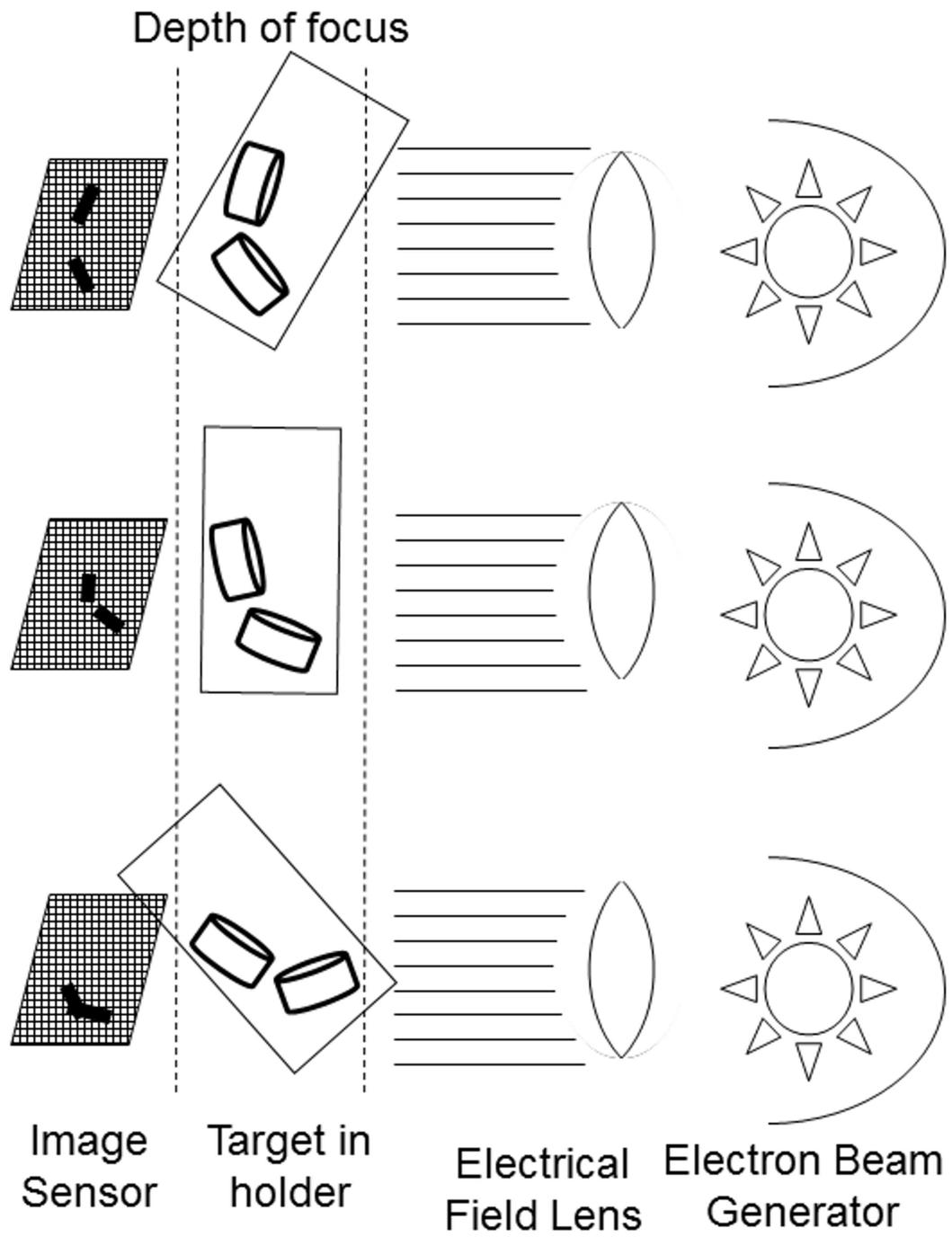


Figure 2. Small Angle *Cryo-tem* Image Acquisition

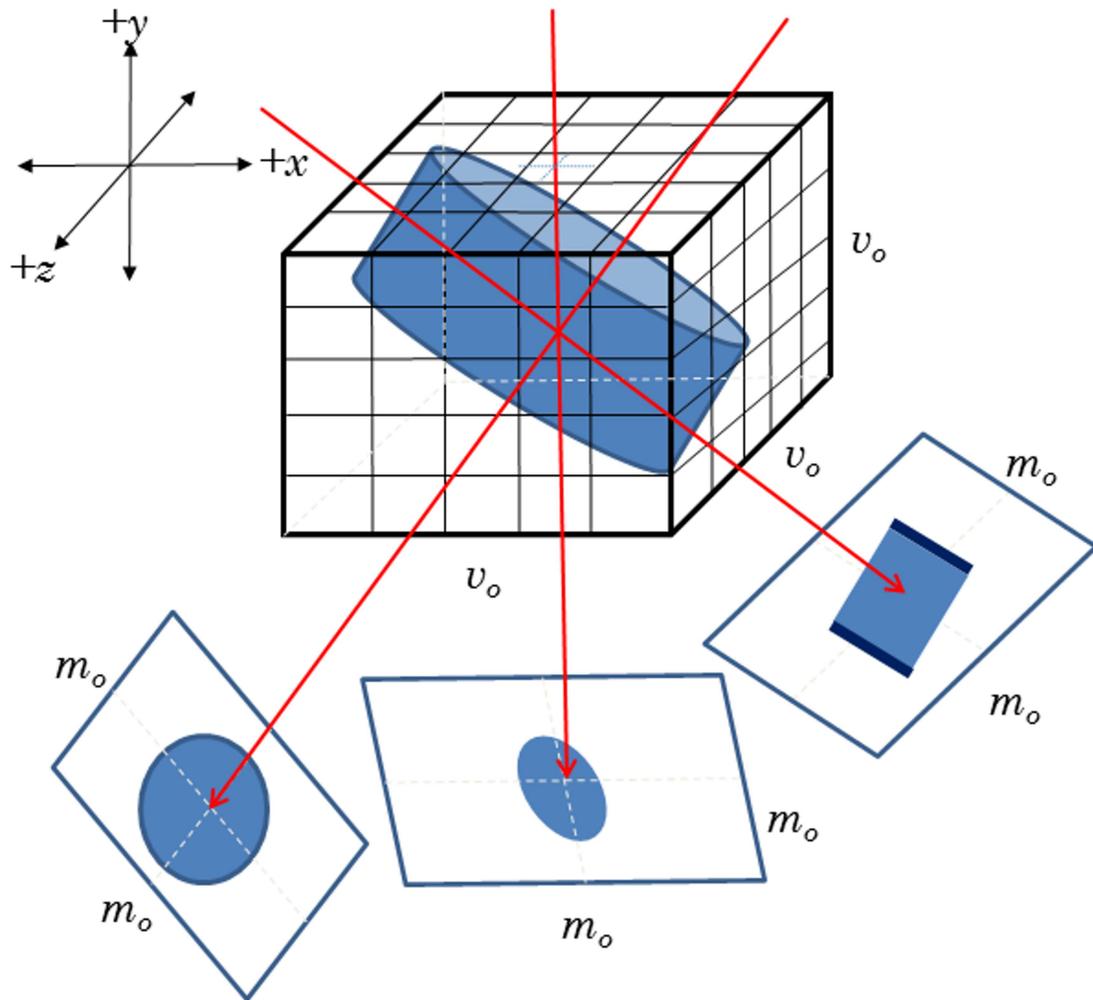


Figure 3. Problem Definition Examples

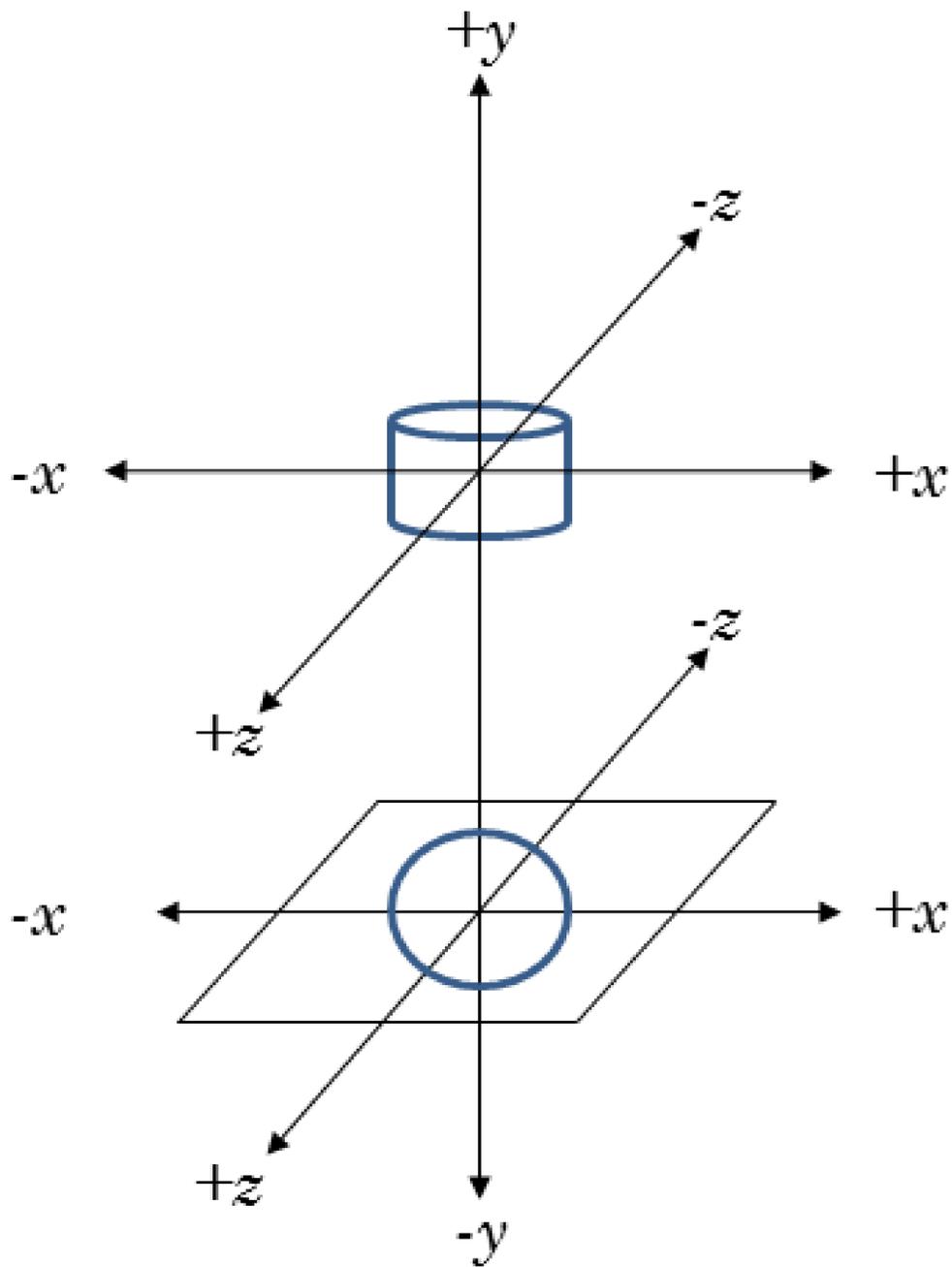


Figure 4. Geometry Orientation

CHAPTER 2

Review of Literature

This chapter discusses the current state of 3-D reconstruction and related technologies. The vision community typically assumes that 3-D reconstruction problems fall into 2 classes : 1) objects can be represented as completely opaque, and the effects both of transparency and aliasing can be ignored, or 2) volumes contain only semi-transparent tissue and the effects of occlusion can be ignored [12]. The research discussed in this dissertation follows the latter class, so the review of prior studies will concentrate on issues related to the semi-transparent schema. However, due to the nature of the reconstruction goals of this project, only the non-occluded class will be considered.

The remainder of this chapter is organized as follows. The review begins in Section 2.1 with more generalized 3-D reconstruction methods for opaque and transparent objects. Since there is a specific imaging technique for the targets of interest, the discussion continues with a review of standard practice *cryo-tem* reconstruction techniques in Section 2.2. Specific studies of the LDL particles themselves are then discussed in Section 2.3. Section 2.4 contains information relating computer vision and model matching in the context of the stated problem. The final section, 2.5, provides a review of statistical reconstruction methods, including hidden Markov models, as required for some of the reconstruction techniques discussed in Chapter 3.

2.1 General 3-D Reconstruction Techniques

When there is but a single larger item (or scene) to be rendered, 3-D representations can be created from many different algorithms. Brief descriptions of many approaches are provided by Ihrke, *et al.* [13]. This section provides an overview

of a few of the approaches more closely related to having a limited set of images such as starting with two pictures (stereo) and progressing to methodologies with more images. An overarching consideration of these approaches is that a prior understanding of the target is not always known. In these cases, measurements from the projections of the object are used to create a representation of the object from mathematical considerations rather than from a model-based approach. Still, it is important to understand these approaches to understand constraints and assumptions that can be used in the model approach.

2.1.1 Stereographic Reconstruction

The conceptually simplest 3-D reconstruction uses only two images. Brains of many living creatures utilize only a pair of eyes through which their world is viewed. Thus, evolution has caused neuron connections that can process the two images to reconstruct (understand) the scene at hand. But, as reported by Freeman [14], electrical activation of these regions can bias an observer's depth estimates. So, as good as animal vision is, the processing can have biases that cause improper localization and shape recognition due to localization estimate errors. Luckily, other processing regions of the brain may handle the distortions to allow for a relatively good level of functionality from just the two images.

The goal of computer vision is to mimic, or exceed, the human processing capabilities without introducing biases from the processing and by correcting errors introduced by the optics. Stereoscopic reconstruction instruments, such as that described in [15], estimate 3-D (x, y, z) objects viewed by a single pair of digital cameras with optics that can change focus and zoom much as human eyes can do. The positions, shapes, and sizes of the objects are generally determined by processing the images. The resultant scene reconstruction can then be considered the computer vision similar to human vision processing.

There are many methods for performing stereographic computer vision processing. In general, stereoscopic depth perception is based on the differences between the left and right half-images of a stereogram [16]. But, difference (or disparity) processing can actually be considered to be a matching function. See [17, 18] for a survey of computer vision matching methods. Sabater [19] discusses meaningful matches in stereo vision and provides a theorem that guarantees an upper bound on wrong matches by building a “faithful statistical background model” for image blocks from the images themselves. The statistical approach helps account for noise and other distortions in the two stereographic images by seeking an answer to the query “What is the probability that given two images and two similar patches in these images, this similarity arises just by chance?” [19]. But, in the end, this method uses a final disparity map across sets of image pairs. As with human vision, this involves a time series of image pairs, which is not available in the *cryo-tem* imaging paradigm.

2.1.2 Reconstruction From Multiple Views

An obvious next step in scene reconstruction is to utilize more than two images. It is believed that humans tend to represent known objects by sets of 2-D views rather than by single 3-D descriptions [20, 21]. Thus, one approach is to use sets of stereographs. Computers can store large numbers of image pairs but even the most powerful computers are somewhat limited by processing time so there is a practical limit to the number of image pairs that can be processed by computer vision techniques. Human physiology is limited to two eyes but computer vision systems do not share this limitation. In a still scene, a single camera can be moved to record many images of the target or multiple synchronized cameras can record simultaneous images of the target from varied viewpoints. Thus, images from multiple viewpoints is a natural approach to be tried in computer vision.

As with general stereo processing, there are many approaches to reconstruction from multiple views. See Seitz *et al.* [22] for a review of algorithms for dense multi-view stereo algorithms and Labatut [23] for a list of multi-view approaches. The following sections describe multiple view techniques that are more applicable to the *cryo-tem* problem.

2.1.2.1 Shape From Surface Properties

A common way to estimate the shape of a 3-D solid object is to use surface properties of the object. These methods involve using measurements derived from 2-D images of the object to create an estimate of the physical 3-D object shape. Common techniques of surface-based methods are estimation of shape from contour, also called shape from silhouette and contour disparities. Related methods utilize slices of the solid object to estimate the shape by connecting the contours (surfaces) of the slices. In all cases, multiple images are required in order to provide a “good” estimate of the original shape.

The idea of shape from silhouette, which is credited to Baumgart [24], involves estimating the visual hull of the original object. The visual hull is defined as the maximal volume consistent with an object’s silhouettes as seen from some set of viewpoints [24, 25]. The visual hull is created by intersecting the solid visual cones formed by back-projecting the target’s silhouette from each camera’s image plane [24]. See Figure 2 in [24] for an example diagram. In 1974, Baumgart estimated 3-D shapes of toys from four silhouette images. As can be seen from Figure 0.1 in [26], the 3-D representations are polygon meshes that show only the basic outline and very little real 3-D depth information. Additional research into shape from contour has been performed since these initial efforts.

A related method for estimating shape from surface properties is to use disparity in the projected contours. Marr and Poggio used the cooperative phenomenon

of the human brain as the basis for a stereo vision processing algorithm [27]. As such, they condensed the solution to the stereo vision problem (also see Section 1.1) to the fusing of two images by finding common points and using those points to find associations between other points in the image [28]. That is, these algorithms work by finding along the outline contour “bumps” (control vertices) or regions that are similar. Since these efforts are directed towards solid objects, only the contours can be utilized since concavities elsewhere in the projection are masked by the overall “shadow” of the projection image. It is only when a concave region is captured in the projection contour that it can be used to help in the reconstruction. This requires an inordinate number of images to ensure that all such features are captured. As such, a highly cooperative processing engine (such as parallel neurons in a brain) is needed to fully solve the stereo vision problem by utilizing just the contours of the projections.

Other methods of shape from contour rely on slices of data. In some cases, true tomography is used to produce image slices of the target then reconstruct the shape from the tomography slice edges stacked edge to edge. In other cases, virtual slices are used. Dong and Hillman [29] incorporated a curve matching process when producing the vertical contours for the object. This approach uses re-sampled B-spline interpolation of the contours to increase the resolution of the curves connecting the layers of the slices. Giblin and Weiss [30] used multiple views from a known camera motion but with a restriction of co-planar viewpoints in order to induce a correspondence between the two profiles. Where as the standard “bumps on the profile” processing assumes similarity, this approach uses a minimization in depth to resolve the ambiguity. This reduces the envelope of planes problem to that of computing the envelope of a family of lines in a single plane.

2.1.2.2 Space Carving

An extension of surface-based estimation techniques is to look at the photographic and lighting nature of the object surface. Kutulakos and Seitz [31] provide an analysis that provides the tightest possible bound on the shape. Space carving starts by using the visual hull as a basis when there is no information available regarding the scene’s radiance. A family of all shapes that are photo-consistent with the photographs is used to define the ambiguity that exists in recovering the 3-D shape from the images. The union of all photo-consistent shapes defines the *photo-hull* of the object. The *photo-hull* can be computed by a simple algorithm that “carves” space until the remaining voxels represent the *photo-hull*.

The basic algorithm consists of carving away voxels that are not consistent with the images. An initial volume that contains the true scene is initialized. Each photograph is projected and the colors and optical rays from each voxel to the sensors are compared via a consistency function. The number of consistency checks is bounded by $N \cdot M$ where N is the number of photographs and M is the number of voxels, but as the number of removed voxels increases, the number of voxels remaining to be checked is reduced from M , so improvements in computation speed from the upper bound are to be expected. This approach, however, is based on consistency of the voxels based on the illumination of the object in the images. Pixels in the photographs, which are generated by reflected light, are used to determine which voxels remain in the solid volume of the reconstruction object.

More recent efforts utilized a probabilistic approach to space carving. Yao and Calway [32] used likelihoods of either opaque or free space to generate a depth mapping. Broadhurst *et al.* [33] used a plane by plane approach to carve pixels from series of 16 and 63 images in order to recreate 3-D reconstructions of highly textured houses that do not have holes in the reconstruction. These approaches

still require many images but the concept of consistency and probabilistic decisions could be useful tools in the reconstruction of particles from *cryo-tem* images.

2.1.2.3 Tomography

A major methodology for reconstructing “solid” objects is to use multiple images that consist of a series of 2-D “slices” of the target. By adjusting the slice depth of field and the focus plane, this computerized tomography (CT) can also be used to provide scans of the scene or object. Classic computer-vision techniques can be used to “find the edges” of the slices so that a 3-D model of the object can be constructed from the 2-D slices joined at the related edges. In this process, volumetric rendering utilizes the edges of each slice in order to connect them on a grid or mesh so that a composite 3-D representation is generated. This technique works well when the target is fairly large with respect to the imaging technology resolution. CT also requires that the target can be held fairly stationary with respect to the imaging device or there must be a registration mechanism so that the slice offsets are known with a degree of accuracy. Since nanoparticles are already thin, this method of generating slices is not applicable.

Another way of performing tomography involves slices that are not co-planar. Electron tomography is a means by which a three-dimensional structure can be reconstructed from a series of images or projections, taken at regular tilt intervals [34]. In their paper, Midgley and Weyland describe the use of the “projection requirement,” which states that the projection is, in principle, a monotonically varying function of some feature of the target. In the case of a bright field *transmission electron microscope*, the assumption is made that the mass-thickness contrast due to elastic (atomic) scattering is dominant [34]. Another recent effort in tomographic reconstruction of transparent objects [35] utilizes hundreds of images and requires the immersion of the object in an index-matched material in order

to reduce surface scattering and refractions. Clearly, biological samples will not be index-matched to the buffer solution. But these examples of *transmission electron microscope* tomography show what can be done with sufficient images of an individual target object.

2.2 Existing *Cryo-tem* Transparency Reconstruction Techniques

The 3-D reconstruction techniques in electron microscopy have led to what is called “single particle reconstruction” (SPR). As noted by Frank [36], this nomenclature can be confusing since the reconstruction process traditionally uses many thousands of projection images. The name is derived from the fact that a single particle class (e.g. a specific virus) is reconstructed and not that pictures of just a single item, are used to perform the reconstruction. It is a set of distinct images, however, that are used to reconstruct a class average from a representative ensemble of distinct particles that represent the class.

Regardless of the lineage of the nomenclature, the problem of solving a SPR is a complex procedure. “The problem of reconstructing a 3-D object from a set of 2-D projections was in principle solved analytically by Radon at the beginning of the twentieth century (Radon, 1917). The principles of 3-D reconstruction were re-invented in the late 1960s and early 1970s, with the introduction of computerized tomography in medicine and 3-D reconstruction techniques in electron microscopy.” [37]. SPR utilizes many aspects of Radon’s principles in order to reconstruct biological particles. See [34] for a treatment of the Radon transform and its application to electron microscopy processing. As is seen, the complexity is not so much in the math, per se, but in the alignment and filtering of all the images that are required to provide coverage of the entire object 3-D spatial extent.

Most research in the area of visualization of biological macromolecules via SPR has utilized a uniformity assumption in order to achieve high-fidelity render-

ing of 3-D representations from multiple 2-D projection images. By assuming a uniform shape and a relatively small range of sizes amongst the target objects, these algorithms convert 2-D images into a 3-D model that is based on sampling many thousands of individual images taken at varying spherical angles around a large population of targets [2, 3]. In essence, these techniques align sample images and average out the imaging noise and variations in the population samples by processing the various pictures of the “same” object into a single result space. See Figure 12 in [37] for a graphical map between the 3-D, 2-D, and 1-D real and Fourier spaces. This technique works well for self-consistent objects such as viruses and other biological constructs, where a single composite image may represent many instances of the same object imaged from many differing view angles.

The following subsections provide details for many of the underlying technologies used in current techniques with an eye towards gleaning issues and ideas that are applicable to the dissimilar target problem. Reviews of many of these topics are provided in [6, 37, 36]. The following sections delve into aspects that are related to the non-uniform target problem. Section 2.2.1 discusses two basic issues in many reconstruction techniques. Section 2.2.2 then cover the most common reconstruction techniques. Section 2.2.3 concludes with a discussion of available software tools that are applicable to 3-D reconstruction from *cryo-tem*.

2.2.1 Underlying SPR Issues

Because all of the methods in this section use *transmission electron microscope* as the imaging method, there are several common issues across all processing methodologies. Two of these common issues are important to research related to a single image: the Fourier central slice theorem and Contrast transfer function. This section provides details of these issues as background for discussion of related reconstruction methods.

2.2.1.1 Fourier Slice Theorem

A first consideration when processing images of a multitude of particles is alignment of the images based on the particle projections within the images. This alignment can be used in many ways but two main objectives are orientation classification for averaging and angular coverage. Averaging is used to reduce the tens of thousands of images to a subset that has reduced noise characteristics. Angular coverage is necessary to prevent “holes” in the reconstruction analysis by ensuring that samples provide coverage across a suitable portion of the full range of spherical angle space.

Alignment of such images can be handled via the Fourier slice theorem. A complete description of the underlying mathematics can be found in [38] and other descriptions can be found in resources such as [39]. The basic idea is that the central section of the Fourier Transform (FT) is a thin slice of the 2-D FT of the orthographic projection image. If the FT is extended toward infinity in length, the central slice in real space collapses to a single line. These slices (the line) can be used to determine the alignment of the images by determining the angle of the slices that match. These slices, with appropriate filtering to limit the oversampled low frequency contribution, can also be used to reconstruct the base image by creating a new FT from the central slices based on alignment angles. By having many thousands of such projections, a suitable reconstruction can be created.

2.2.1.2 Contrast Transfer Function

Images generated via a *transmission electron microscope*, regardless of whether or not the sample is frozen, are not true 2-D projections of the 3-D sample space. Various artifacts and noise are present that degrade the information in the projection images. Noises are induced from many sources but with appropriate care, these can be ignored in the presence of other factors. The two largest contributors

to distortion are the Contrast transfer function (CTF) and the envelope function of the microscope [40, 41]. While the CTF shape (see, e.g., Figure 1 in [42]) depends on several parameters [43] such as defocus, spherical aberration coefficient, source size, and defocus spread, only defocus is a parameter that changes between images.

Despite the fact that information is lost at the CTF’s zero crossings, there are ways to overcome this effect. A simple correction method is to truncate the data at the first zero crossing. More robust techniques have been developed [42, 44] to estimate the CTF so that the true projection image can be recovered. Thus, for the study at hand, acquisition of the proper defocus images should allow for recovery of the underlying projections to be used in the reconstruction processing.

2.2.2 SPR Algorithms

Many algorithms have been developed to perform the basic SPR processing. This section discusses three of the most applicable methodologies.

2.2.2.1 Angular Reconstitution

The angular reconstitution (AR) method, which is summarized in [6], uses the random nature of particles in suspension (which may be frozen) in order to glean the requisite FT slices across a range of spherical angles without requiring a physical tilting of the *cryo-tem* sample. Common line processing, which looks for similar central slice lines, is used in order to align, classify, and average the thousands of projection images. This data is then used to perform a 3-D reconstruction. An iteration step is performed where the reconstructed volume is used to create a new “anchor” set of projection images. Since these anchor images are created from a single 3-D volume, they represent a better frame of reference for assigning rotation angles. The original images are reprocessed against the new reference angles and a new 3-D reconstruction is generated.

Of course the overall angular reconstitution process is not applicable to the single image problem. The projection regeneration processing, however, may have value in the generation of 2-D projections based on our expected sample model.

2.2.2.2 Single particle tilt series

Single particle tilt series (SPTS) often do use just a single particle as a target. By utilizing a much lower electron beam energy, the SPTS method allows for capture of 60 to 280 images of the same particle [45]. The number of images is limited by the stage mechanism, so small angle rotations are used to increase the number of images. In some cases, a 90° flip is performed in order to allow for more images to be acquired. But in all cases, the 60° to 70° span still creates a major hole in the spherical angle coverage of the 3-D FT space.

The SPTS algorithm has other limitations when compared to the larger image set techniques. The large number of rotations also induces rotation and alignment errors due to the mechanical slop in the stage rotation mechanisms. Since achievable resolution is proportional to the electron energy, the lower electron beam energy reduces the detail in the reconstruction. And, even at low dosages, the *cryo-tem* imparts a contrast transfer function (CTF) artifact from aberrations in the focusing lenses and from the defocus used in imaging. Having multiple images with low signal-to-noise (SNR) characteristics makes “seeing” the CTF oscillations hard. Due to all these issues, SPTS provides limited resolution tomographic results.

2.2.2.3 Random Conical Tilt

The random conical tilt (RCT) process takes advantage of the target’s preferred orientation in the sample holder [46]. RCT only takes two images of the sample space: one at some desired rotation angle and the second one at a nom-

inally untilted angle. Since the particles are in the same basic orientation in the untilted image, this image provides a “known” orientation angle for alignment processing. With only two exposures, the electron beam can be more energetic to achieve higher resolutions. In fact, only the rotated image, which has but a single exposure, is used for reconstruction so sample damage is vastly minimized as an input into the reconstruction process.

An alternate version of this is the Albany Zero Tilt (AZT), which uses but a single image of the sample. The alignment and further processing is performed via a series of post processing steps. As with all of the above approaches, the AZT needs images of targets that are basically the same. Thus, targets that do not meet the uniformity requirement are not suitable for these approaches.

2.2.3 Available Software

There is a plethora of image processing software and applications. A leader in the field of *cryo-tem* imaging and reconstruction is the “Baker” research group at UCSD. Besides an introduction to *TEM* [47], the group also lists several programs on the website that can be used for *cryo-tem* image processing [48]. Most of this software, however, is geared towards to 3-D single particle reconstruction of identical viruses from multiple images. Since many samples of a given virus are considered to be essentially the same structure, this technique acquires many projection images from many angles. From these images, this software can build 3-D models. This technique, however, can provide only an average representation for a single class of non-uniform macroassemblies such as LDL particles.

2.2.3.1 EMAN

The EMAN (Electron Micrograph ANalysis) software was introduced in 1999. The goals of the EMAN project [44] were to improve processing efficiency, make

SPR more accessible, and introduce a robust method of performing the CTF correction. EMAN makes SPR more accessible by providing the software as a free download. To meet the processing goals, EMAN uses a three-stage approach to performing the final reconstruction.

The first stage in an EMAN reconstruction is particle selection. Given that tens of thousands of particles are required for the reconstruction, a means of selecting the particles from the micrographs is provided. The `boxer` program is used for manual or semiautomatic selection. The utility has support for manually locating particles in low-contrast images and for handling high-resolution images by breaking them into smaller, more memory friendly, pieces. `boxer` includes options for threshold adjustments for each particle. The processing also has support for projection-rotational orientation of the particles.

The second stage of EMAN processing is the preliminary model generation. The EMAN refinement stage is model based so EMAN needs a starting model. If the images have sufficient SNR, then the refinement loop will converge, but the convergence will occur faster with a better initial model [44]. EMAN provides support for many symmetry cases. These can be used when there is some a priori understanding of the particle being reconstructed. Models for asymmetric particles are supported by generating class averages then aligning them via the Fourier common-lines. In addition, EMAN can read in initial models from other sources; this is a possible use for the results of this research.

The final stage of EMAN processing is to refine the model based on the images. The refinement loop uses the current model to create a uniform distribution of projections. Each particle (2-D image from `boxer`) is associated with one of the projections. The particles are checked for consistency against the reference projection to ensure that they are suitable based on issues such as beam damage,

functional state, or just in a bad region of the sample. This check ensures that a minimum number of projections is available for each reference projection angle. A class average is then created for each projection angle. Since the orientation of each class is known (the reference images were created from the model) the class averages simply can be inserted into the Fourier volume in the correct orientation. Convergence of the modeling is based on the Fourier shell correlation (FSC), which is the normalized cross-correlation coefficient between two 3-D volumes over corresponding shells in Fourier space [49]. EMAN seeks a refinement where the FSC ceases to improve without requiring a specific value.

2.2.3.2 IMAGIC

IMAGIC is a high end environment for the analysis of images, spectra and other multi-dimensional data-sets that is aimed at processing (huge) data sets from (cryo-) electron microscopy, especially in the field of single particle analyses in Structural Biology [50]. IMAGIC is mainly a for-fee set of software costing upwards of €26,300 . IMAGIC, does provide two tools as freeware: `em2em` converts images between typical *transmission electron microscope* formats and `FSC` calculates the FSC of two 2-D images or two 3-D volumes.

2.3 Prior LDL Shape Studies

With a background understanding of *cryo-tem* and its associated processing, a review of some pertinent LDL shape studies can be presented. These studies show how the science of analysis has progressed and how reconstruction from a single image can support the overall science of *cryo-tem* analysis of LDL particles specifically and heterogeneous particles in general.

Initial *cryo-tem* micrograph studies [51] reported that the LDL particles may be more discoid in shape than spherical as previously assumed. Figure 1 shows

example *cryo-tem* images of an LDL particle at rotations of $+45^\circ$, 0° , and -45° . This study measured the diameter and height of the discs but did not present a thickness of the wall as seen in the projection at 0° in Frame A of Figure 1. These studies, however, were difficult to carry out, and did not apply computer vision processing to provide a way to perform measurements on samples from a large number of objects from each image. A mechanized study on multiple images is necessary to evaluate the effect of geometric shape parameters on overall health in a statistically significant patient sample.

Additional efforts by this research group [52, 7] extended their previous results to indicate that LDL was actually divided into sub-classes based on size. Again, simple projections at $+45^\circ$, 0° , and -45° were used to measure the pertinent dimensions. These measurements were done using commercially available software tools on digitally-scanned images of photographs from the *cryo-tem* equipment. These later and more quantitative studies, however, did yield a more clearly defined set of images that showed size ranges.

Other studies have been performed in an attempt to resolve the LDL shape question, but these did not provide actual metrics on the shapes. In one study [53], computer-based analytical methods were applied to *cryo-tem* images by applying a homogeneity requirement on subclasses of LDL particles. In this analysis, approximately 5600 individual LDL particles were manually divided into subclasses. Further classification reduced the data to approximately 4000 particles. These particles were further analyzed and were determined to be generally ellipsoidal in shape. A later study [54] used volume/mass ratios of High Performance Gel Chromatography (HPGC) to indirectly determine LDL shape. This study fitted data based on a geometric model of a discoid object with varying heights rather than directly from images of the macromolecules.

One of the most recent full-featured reconstructions of LDL particles was reported in 2010 [5]. This study reported a reconstructed 3-D volume that shows the shape of LDL particles to be somewhat like a flattened walnut with a slight bump on the narrow end. In this analysis, 8,500 particle images were selected from a pool of 48,000 particle images to be processed. The resultant reconstructed shape is a slightly flattened ellipsoid and a size of approximately 250 Å by 240 Å by 166 Å. The detailed reconstruction from [5] (See Figure 5) also shows 4 internal layers of alternating cholesterol esters (CE) and fatty acyl chains. The

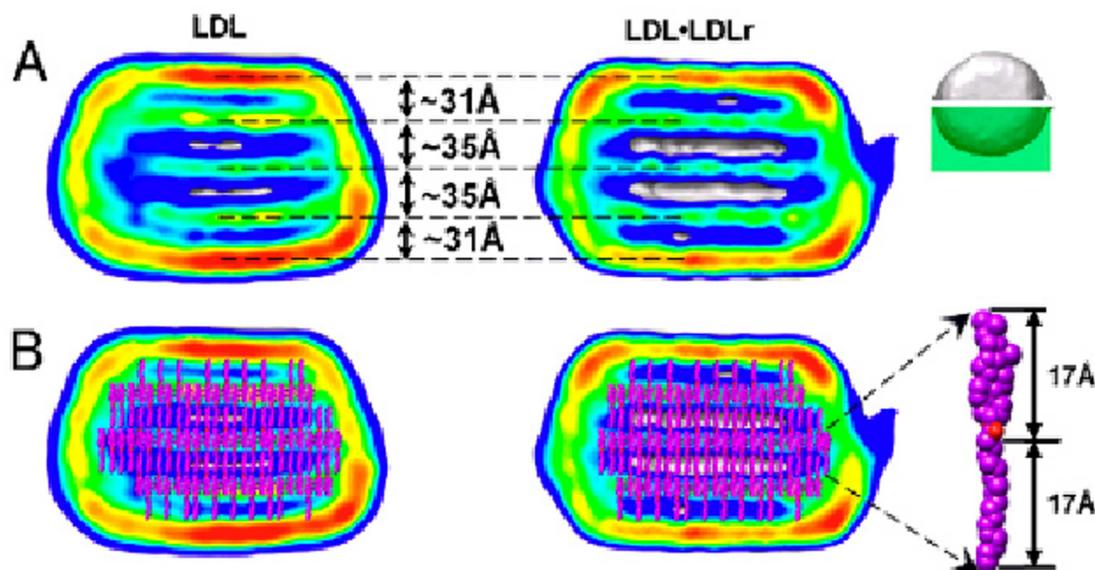


Figure 5. 2010 LDL *Cryo-tem* Reconstruction (from [5])

reconstruction shows a very advanced model of LDL that can be used to understand the structure of the particle. But, this study is a demonstration of a major concern with heterogeneous sample sets: about 40,000 of the particles were *not* included since they were of different size or shape. These studies all used large numbers of images but resulted in but a single class average description of the biological target of interest. These heterogeneous studies do not provide a pathway to understanding the individual parameters of a large sample of proteomic particles.

2.4 Computer Vision and Models

While one objective of computer vision is to mimic the functional requirements of the human vision system, there is no requirement that the algorithm match that used in the brain. The methods presented above are mathematical algorithms that strive to reconstruct the original as closely as possible by performing processing in some closed form. In general, such approaches do not include any a priori information about the target other than general size constraints. Even the initial model used in EMAN is really a first guess rather than an a priori model.

Human vision processing, however, has been described as a two-stage process : concatenated filters followed by a sampling operation [55]. Thus, the images “seen” by the human brain for processing effectively have been digitized by the rods and cones in the eye. While we cannot be sure exactly what happens after the data has been received by the eyes, models can be proposed and tested to help explain the processing. In one example [56], the speed at which objects can be recognized, the timing constraints of neuron firings, and data transmission rates lead to a model of encoded data being processed in a multi-layer network where recognition of certain face parts are merged to provide a recognition capability for human faces. Thus, underlying knowledge of the human vision system is used to create a model of the processing that “simulates” human vision processing capability.

There are many processing model types that can be employed to perform computer vision. See, for example, Prince’s book from 2012 [57] for information about computer vision and machine learning. In general, model-based approaches to object recognition use a database (a library) of modeled objects; for a given set of sensed data, the problem of model-based recognition is to identify and locate the objects from the library that are present in the data [58]. In this approach, information about the target is used to determine if the object in an image is

the search object. Even when there is but a single search object, model-based recognition is NP-complete [58]. The following sections provide information about model approaches that may be useful to the *cryo-tem* reconstruction problem.

2.4.1 Solid Model Matching

A basic idea behind model matching is to measure the correlation of the observed image with projections of possible sample objects. In a one-dimensional sense, this can be accomplished by performing the correlation function between the measured data and the representative function. In a two-dimensional case, as with projection images, a 2-D correlation can be performed between the observed and reference images. This 2-D correlation processing, however, results in a significant processing challenge. The processing itself is performed via some fashion of multiplication of Fourier transforms of the reference and measured images. This computation load increases both as the targets get larger and larger and as the resolution of the images gets higher and higher. But the unbounded portion of the processing burden comes from the need to examine a significant set of reference images to ensure that a positive match is found.

The NP-complete nature of model matching is an issue due to the need to examine many images. Even if the first reference sample matches perfectly, other references must be processed in order to rule out all other possible choices. In general, a 2-D surface of correlation values must be generated in order to determine a best fit for all the reference images. The dimensions of the surface are based upon the number of parameters to be considered. For example, if a simple 3-D reference shape is correlated against a sample image then reference images from multiple spherical angles may be needed in order to help determine what the actual shape is. The number of angles can be limited by any symmetry considerations of the reference target. But, if a detailed estimate of the size, or some other parameter,

is required then the reference images must also range over the desired parameter. By having a range of sizes, reference angles, and other parameters (say coloring), the surface can be constructed over the parameters. The more simple constraints can be correlated in a straightforward manner such as over a range of sizes. The more complicated parameters, such as coloring, may require 16 million attempts for a solid color object described in eight bits for each red, green, and blue sub-color, if an exact match is desired. Thus, the significant complexity comes not from the number of calculations for each image, but rather from all the variants to be considered.

The end result of “simple” model matching (correlation matching) is to determine the best match from the correlation surface. The surface is confined by the search parameters. Thus, a range of sizes, shapes, colors, and other pertinent parameters define the search space. The surface is constructed from the correlation results of each correlation reference image to the sample image. Samples that more closely match will have a higher correlation “score”, samples that do not match as closely will have lower scores. The surface must then be searched from local maximum points. These peaks must then be run through additional levels of processing in order to determine the best fit. Since the correlation processing is a “blind” method, no a priori information is used, so a local peak may, in fact, be a false peak due to the nature of the image juxtapositions and parameters. The post-processing needs to weed out such false peaks. The final output of such processing is then considered a best match, but it is not a definitive statement that the original shape has been determined.

2.4.2 Geons and Superquadrics

As previously noted, study of the human visual system has affected development of many computer vision and reconstruction algorithms. Following the

example of speech perception, where only 55 phonemes lead to coding of tens of thousands of words, Biederman [59] proposed recognition-by-components (RBC) as a way to use just 36 geometrical ions (geons) as the building blocks for generating all complex shapes. The RBC theory poses that geons can be derived from contrasts of five readily detectable properties (curvature, co-linearity, symmetry, parallelism, and co-termination) of edges in a two-dimensional image. Understanding the known occlusions when geons combine to build larger objects helps computer vision identify complex shapes from their constituent geons. The RBC theory is in contrast with the Gestalt theory of human perception in the entirety [60] since RBC leads to a hierarchical identification of the whole from its parts.

The hierarchical RBC construction approach, however, is more in line with how computers solve problems via human programming. Figure 2 in [59] shows the stages of object perception. After edge extraction, detection of nonaccidental properties of the edges and parsing of regions of concavity occur in parallel. These results feed into determination of components, which is followed by matching of components to object representations. The final step is to use all of the preceding information to perform object identification. These steps constitute an algorithm that can be coded for computer execution.

A major factor in RBC is the use of edges. According to RBC, human perception uses discontinuities in the 2-D image to segment the image “painted” on the retina. (An important note is that RBC does not depend on human stereo vision, which is sensible given that stereo vision is not required for identification of objects by humans.) These edges are then used to recognize the objects based on expected transformations back to the 3-D space. Figure 4 of [59] lists the properties that are expected to hold from the 2-D projection back into the original 3-D space. In general, contours are important; noise along the contour (e.g. camouflaging some,

but not too many, of the edge points) is not. Thus, human perception only needs line drawings for edge and object recognition; surface texture and color are not important. If too much of the contour is deleted then the component becomes unrecoverable. Nonrecoverability can also be caused by misleading symmetry or parallelism from the edges.

Geons provide the basis components for hierarchical identification of a whole from its parts. The basis set of 36 geons gives rise to an almost infinite combination of items. Simple shapes constructed from just 3 geons give rise to 154 million combinations [59], but this search is not insurmountable in today's computer technology. The challenge is to find a method of finding the edges in such a way that the geons can be localized regardless of the viewing angle.

Superquadric models [61, 62, 63] have received significant attention in the vision and robotics community because of their compact representation and robust methods of recovery of individual models [64]. A superquadric surface is defined by a relationship of the form shown in Equation 2.4.1

$$F(x, y, z) \equiv \left(\left(\frac{x}{a_1} \right)^{\frac{2}{\epsilon_2}} + \left(\frac{y}{a_2} \right)^{\frac{2}{\epsilon_2}} \right)^{\frac{\epsilon_2}{\epsilon_1}} + \left(\frac{z}{a_3} \right)^{\frac{2}{\epsilon_1}} = 1 \quad (2.4.1)$$

where a_1, a_2, a_3 define the size and ϵ_1 and ϵ_2 define the shape. The parameters for the superquadric are recovered by use of an expression of the form shown in Equation 2.4.2

$$F(x, y, z, \Lambda) = f(x, y, z, a_1, a_2, a_3, \epsilon_1, \epsilon_2, \phi, \theta, \psi, p_x, p_y, p_z) \quad (2.4.2)$$

where the Λ terms represent the model parameters, rotation, and position of the shape. Since all sides are not visible, processing must account for this self-occlusion as well as nonaccidental occlusions from other geons in the scene.

Several variants of superquadric processing allow for determination of geons from 2-D images of 3-D scenes. See Biederman [65] for a review of recent research

in recognition of depth rotated objects. The superquadric algorithm introduced by Solina and Bajcsy [63] supports parametric deformations of the recovery target. In this approach, three-dimensional range points (distances from points in the scene to a common point such as a sensor) are input to a cost function. The cost function is minimized via least squares minimization. An initial estimate of the rough position, orientation, and size of the object is required to begin the gradient descent minimization process. Of course there can be multiple local minima so their approach limited the search space to handle the ambiguity of superquadric models (the same shape can be described with different parameters). Kriegman and Ponce [66] use surfaces to recover objects in monocular images of precisely definable geometric models. Terzopoulos and Metaxas [67] use superquadrics and physical force modeling then use the parameters to look up the shape in a database. Dickinson (et als.) [68] build a database of surface projections with conditional probabilities based on aspects. These probabilities are used to limit the search space. This database can be aided by precisely defining and computing view likelihoods as was done by Weinshall and Werman [69]. Dickinson (et als.) [70], Leonardis (et als.) [64], and Zhang (et als.) [71] all use range data to help determine the superquadric parameters. All of these approaches have merits and constraints when applied to opaque objects in 2-D images.

It is not apparent, however, that geons and superquadrics can be applied to transparent objects imaged via *cryo-tem*. One constraint arises from the algorithms that are used for determining the superquadric parameters via minimization of range data. Since *cryo-tem* utilizes beams that are effectively parallel at the target, there is no common point from which the range data can be based. Other constraints of the RBC approach with *cryo-tem* imaging of transparent objects come from the need for understanding nonaccidental occlusions and by having

symmetry or parallelism that is not misleading. The *cryo-tem* images shown in Figure 1 show several examples of violations. Panel A shows what looks like a circle, which will be interpreted as an infinitely tall cylinder. Panel C shows what looks like two parallel lines that could be construed to be a rectangle, or an infinitely tall box of some kind. Panel B is also misleading. While an edge can be found around the overall shape, this blob has no obvious nonaccidental occlusions and the nature of the image does not lead to obvious internal edges. For this simple case of an expected discoid shape, there are issues. Other shapes are expected to give similar concerns due to the projection imaging of transparent objects.

While the physical geons could be determined from the underlying biological representation, another issue arises from line edges derived from the image that are not expected to be functionally correct. Incorrect edge representations, of course, will affect the RBC approach. A simple model derived from Figure 1 that consists of three solid discs representing the phospholipid head group (red top and bottom plates) with the third disc representing the center fat mixture (green) and a hollow cylinder (blue) representing the protein wrapped around the outside is shown in Figure 6a. The resulting simulated projection image is shown in Figure 6b. While the model structure can be easily seen by the human eye due to the colors, the projection image has ambiguity for the edges. The regularized solution using superquadrics is expected to be a larger cylinder wrapped around the inner cylinder; the true tri-layer stack in the middle would most likely be lost.

The more complicated model shown in Figure 5 could be presented as either stacks of trapezoidal discs or some other more complex shape. The issue with RBC representation is the line edge presentation. Other than the overall boundary, the *cryo-tem* image leads to neither nonaccidental occlusions nor symmetry or parallelism that are not misleading.

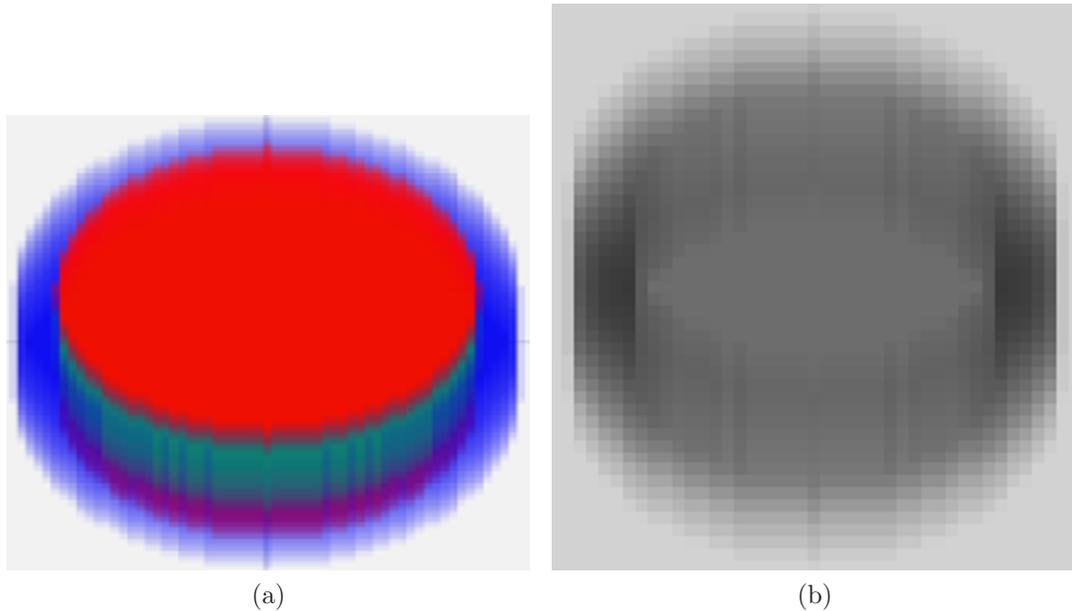


Figure 6. Simple disc model (a) model and (b) projection image.

A final note is that even if superquadrics could find the overall shape, a goal of this research is to glean a representation of the internal density structure. While superquadrics have been used to generate shapes for creating simulated MMM images [72], the RBC approach only supports reconstruction of the bounding surface.

2.4.3 Transparent Model Matching

The match processing gets more complicated when transparent objects are imaged. This is more complex than simple color processing because there are many ways in which a density set could be aligned to create the same density value. Color and lighting could result in a similar complexity but the achievable color levels for each pixel can be constrained by illumination and other consistency metrics. A standard 10-bit camera image could have 1024 possible values so a 3-D reconstruction depth of 128 voxels could have up to 1024^{128} possible combinations. Some combinations may be eliminated from the search space (for example the all zero case), but this generally results in a small reduction of overall possible

combinations. The image matching also cannot, in general, tell the order of the densities when there is a single sample image. Thus, the reference image set may be greatly reduced for a single projection angle since all the combinations collapse because a given set of density values along the projection line may be the same regardless of the order. But this relaxation depends on the physical mechanism of the imaging technology. If the 2-D projection does depend on the order of the voxels of the 3-D transparent object, then this reduction cannot be taken and the complexity remains large.

A final consideration of the model matching complexity is the number of sample images. Reconstruction by graphical models involves comparing projection images to possible shapes that could have made the projection image. This leads to finding the most probable shape, or combination of shapes, that could lead to the outline(s) of the images. The basic goal of multi-view model matching is to look for consistency between the 2-D matching surfaces. This generally requires knowledge of the relative view angles so that the reference images can be generated from the same view point, otherwise, the complete spherical space must be searched as well as the other parameters such as size, shape, etc. Once the surfaces from each view angle have been computed, the correlation peaks must be examined to determine the reference shape that best matches the sample images. One such example would be a 3-D cylinder object. When viewed from the end, the projection would be a circle and when viewed from the side, the projection would be a rectangle. Simple and unique objects may have a unique 3-D shape solution from just a few images. Transparent images, however, return to the combinatoric growth in possible targets since the order of voxel densities now have constraints in 3-D reconstruction space. For example, a projection ray in one view is formed by the physical manifestations of the voxels traversed by the imaging beam through that set of voxels. But, each

voxel in that ray is a fixed value so that relationship must be preserved when each of those voxels is imaged in projections at different rotation angles. This additional level of matching adds to the complexity when transparent objects are reconstructed.

Previous approaches to reconstruction of transparent objects in a non-tomographic method utilizes several images to determine the best match. The probabilistic voxelized volume reconstruction (Poxel) method [12] uses the final color of each pixel in the projection images to determine the weights of the voxel colors. A binary responsibility matrix is generated and an iterative approach is followed to account for colors and opaque voxel. But this approach requires tens of images so the small number of images, much less a single image, that are available for approaching the LDL shape problem will result in a severely under determined linear system. The authors of the Density Sheet Decomposition method [73] have recognized that using as few as two images is equivalent to a severely under-constrained problem. But, their method was developed for visually realistic modeling of gaseous masses such as flame and smoke rather than more physical objects. This technique was demonstrated for large separation angles of the individual views which also is not supported by the *cryo-tem* imaging technique. Therefore, these existing approaches are not suitable to the *cryo-tem* problem with a limited image set.

2.5 Statistical Models

Statistical modeling methods of reconstructing 3-D offer techniques that may not require the exponentially scaled computations of correlation processing nor the extensive collection of images required for averaging methods such as single particle reconstruction. If a relationship can be established between the observables (pixels in a projection image) and the underlying 3-D object configuration (voxels)

then it may be possible to estimate the 3-D structure utilizing the relationship. Some methods, such as Bayesian methods seek to determine the most probable “explanation” of the observations given a physical model of how the observed images are created. An example of this is discussed in Section 2.5.1. Other methods, such as hidden Markov models, utilize a prior understanding (model) of the 3-D target to help find the most probable set of states that explain the observations. This processing is discussed in Section 2.5.2.

2.5.1 Bayesian Method

A recent paper [74] utilizes a Bayesian method for inferring macromolecular structure from SPR microscopy. Rather than modeling the target itself, this method creates an estimation of the 3-D space by modeling how the observed image pixels are created by the electron beam passing through the material. The vertices of the voxels are used as the definition points of the coulomb density of the 3-D space. A tri-lateral interpolation of the vertices along each ray to the pixels is used as a means of connecting the pixel values to the voxels. A probabilistic model for the pixel values from the line integration along the rays is then used to feed an iterative process for maximizing the probability that the vertices give rise to the given pixel values.

The images used for this processing are averaged images, much like those used for standard single particle reconstruction processing. As with SPR, the averaging of the images implies a uniformity of all the particles in the ensemble class. And again, as with SPR, this method assumes that the alignment of the particles is sufficiently accurate to not unduly influence the estimate of the 3-D space. The averaging can also affect the statistics of the pixel values.

A pure Bayesian approach needs to understand the standard deviation of the pixel values in the images so that the probabilities of the voxels can be estimated.

The paper makes an assumption that the standard deviation of the noise is the same for all pixels. As with the rotational alignment, an assumption of images corrected for microscope issues such as CTF makes sense since these are well understood. But, an assumption of constant sigma across all pixels does not seem plausible since the number of electrons received for each pixel changes based on the voxels involved in the pixel. The electron flux will clearly have an effect on the variance of the pixel values. As can be easily measured, and was shown in [75], the variance of the pixel values is not constant. In this study, the sigma of *cryo-tem* pixel images was shown to range from 1 to 14 for a 3×3 region when the mean ranged from 50 to 200. As with the size of the region over which the variance is computed, the number of images in the average also plays an important role in the measured variance. In [74] differing numbers of images used for averaging at various rotation angles was cited as a possible cause of errors in the estimates.

The presented approach also had other constraints that limited its applicability to the problem studied in this dissertation. The paper [74] uses a constraint about there only being small changes in intensity between neighboring grid points. From biological inference, we know that there can be significant changes in coulomb density from voxel to voxel as atoms and molecules with various densities exist side by side. Independence is a primary concern in simplifying statistical computations. While the results shown [74] were fairly good approximate reconstructions, from the above constraints, it is not clear if this method can be used in general practice. And, as with SPR techniques, it does *not* provide details of individual particles.

2.5.2 Hidden Markov Models

While many real-world processes provide measurable outputs, there are also many classes of processing where the direct measurement of the characteristics are not possible. This is comparable to the problem of 3-D reconstruction of trans-

parent objects in that the exact internal structure may not be directly discernible from the images. The underlying information of these systems is hidden from the observation process. In these situations, the goal is to estimate the unmeasurable metrics (or structure) via a model based on things that are observable. Stamp [76] provides an example where past weather is estimated by the size of tree rings. In this scenario, the goal is to ascertain if the average temperatures of winters long ago were either hot or cold. While the average winter temperature cannot be measured in the past, we can measure tree rings from the past. Since the recent weather is known, we can inspect a few recently felled trees and correlate the sizes of the rings (denoted as either small, medium, or large) to the average temperatures (denoted as either hot or cold) in the corresponding recent winters. One can then use a state-machine model that relates tree ring size and weather to estimate the weather as a series of states in the past from tree rings from the past.

A state machine, called a discrete finite automaton (DFA), is defined [77] as a 5-tuple $(S, \Sigma, \delta, s_0, F)$ where

- S is a finite set called the states,
- Σ is a finite set of inputs to the machine called the alphabet,
- δ is the transition function,
- $s_0 \in S$ is the initial state, and
- $F \subseteq S$ is the set of accept states.

Using the formal definition of a DFA, the parts of the machine can be mapped to the tree ring example. The states are the weather values hot and cold so for simplicity we can define $S = \{H, C\}$. The alphabet of inputs to the machine, Σ , is simply the set of years over which the weather is to be estimated. The transition

function δ is defined by the transitions between the states when an input signal is received. For this example $\delta = \{H \rightarrow C, C \rightarrow H\}$. The initial state, s_0 , is unknown since we want to estimate the entire sequence. For this example, however, the set of accept states is empty since we are not computing an answer per se but a series of states.

The DFA notation gives rise to a state machine that can take on any of the allowable states in S but we actually want to estimate the series of states that is a best fit to the observations in some statistical way. Note that the observations, the sizes of the tree rings, do not enter into the definition of the DFA so modifications to the basic DFA are needed in order to include them. One statistical model of such systems is called a Markov process; when the states cannot be directly observed, the process can be represented by a hidden Markov model (HMM).

A HMM is a form of a Bayesian network that has been in use since the late 1960's for a variety of state-machine modeling applications. See [78] for a seminal paper that provides a tutorial on HMMs and their use in speech recognition. In a HMM, the system is modeled as a state machine that gives rise to a set of observed measurements. An representation of an HMM system is shown in Figure 7. The

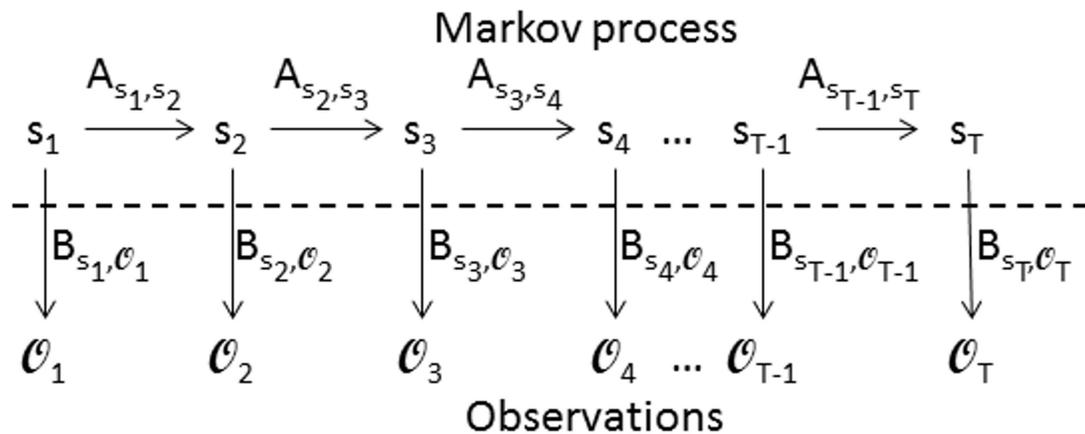


Figure 7. Representative Hidden Markov System

Markov process is denoted by the sequence of unknown states s_t and the corresponding T observations are denoted by \mathcal{O}_t . While the representation of a Markov system is shown as states and observations versus time, this could just as easily be a spatial relationship so the T subscript could be K , a traditional spatial or wave number index. The probabilities of the process transitioning from state to state are denoted by A_{s_i, s_j} and the probability of a given state giving rise to an observation is shown as B_{s_i, \mathcal{O}_t} . The full notation for the HMM is given in Table 1. For the simple case where only the prior state and the next state are considered,

T	=	length of the observation sequence
N	=	number of states in the model
M	=	number of observation symbols
S	=	distinct states of the Markov model $\{s_1, s_2, \dots, s_N\}$
V	=	set of possible observations $\{v_1, v_2, \dots, v_M\}$
A	=	state transition probability matrix
B	=	observation probability matrix
π	=	initial state distribution
\mathcal{O}	=	the observation sequence $\{\mathcal{O}_1, \mathcal{O}_2, \dots, \mathcal{O}_T\}$

Table 1. Hidden Markov Model Notation

the HMM can be considered a first order HMM (FOHMM).

The first order HMM must also satisfy additional requirements.

current state only depends on prior state One could expect that existence in a state is dependent upon several (or all) prior states. The special case of a first order Markov chain is defined as transition to a given state only depends on the prior state. Higher order Markov chains can be defined but the chain of dependencies makes finding a solution harder.

time independence The Markov process probabilities must be stationary over time. Obviously the state machine changes state but the probabilities of changing from state to state do not vary with time.

observation independence The observations of the process are independent.

This constraint is also directed towards the tractability of the solution.

The state machine used in the tree ring example can also be considered a hidden Markov process with a few changes to the problem configuration. First, the only signal in the alphabet is a clock whereby a change in state occurs. We can consider the changing of the year for which the weather is estimated as our clock. All other items in the 5-tuple are as defined above. Second, some probability metrics need to be added to account for the probabilistic nature of the HMM. Hidden Markov models utilize the probability of the system transitioning from state to state and the probabilities of each state giving rise to a discrete observation value to estimate the unobservable states. The state machine is stationary but the state of the machine is expected to change as a function of time. In the tree ring example, the unobservable (hidden) states are the weather conditions of long-ago winters and the observations are the tree rings from the same period.

The HMM state machine would be used in the following way to estimate the weather in years long ago. Known weather states (from more recent years when records exist) can be examined to determine the probability of transitioning from state to state. Tree ring sizes for the same period as well as the known weather states can be studied to provide the probability of a given observation (ring size $\in S, M, L$) for a given weather state (average temperature $\in H, C$). A formal description of a HMM using the notation provided by [76] denotes the state transition matrix A by

$$A = \begin{bmatrix} P\{H \rightarrow H\} & P\{H \rightarrow C\} \\ P\{C \rightarrow H\} & P\{C \rightarrow C\} \end{bmatrix}. \quad (2.5.1)$$

The observation matrix contains the probabilities of each observation based on the

states such that

$$B = \begin{bmatrix} P\{H \rightarrow S\} & P\{H \rightarrow M\} & P\{H \rightarrow L\} \\ P\{C \rightarrow S\} & P\{C \rightarrow M\} & P\{C \rightarrow L\} \end{bmatrix}. \quad (2.5.2)$$

The initial state distribution is

$$\pi = [P\{\text{initial_state} = C\} \quad P\{\text{initial_state} = H\}]. \quad (2.5.3)$$

Each of these matrices is row stochastic since the sum of the probabilities *from a given state* in each row sums to 1.

Note that the model as given does *not* account for all things that could affect the rings. For example, the tree ring sizes could be affected by the density of the forest, or rainfall, or cloudiness. While this could be a limitation of the technique, it can also be an asset. By ignoring the other factors, the most probable state sequence is the sequence that is best explained by the factors of concern. Stamp [76] alludes to this in the example he presents based on [79]. In this example, a set of characters from the English language is modeled with just two states. As may be expected, a 50,000 character sample of text is divided into vowels and consonants. But, the work in [79] was able to decipher results for models with up to 12 hidden states. Thus, other factors do not necessarily affect the results when a smaller number of factors are considered.

Given a notation and statement of the use of the technique, the HMM can be used in three ways [78].

1. Given an observation sequence $\mathcal{O} = \mathcal{O}_1, \mathcal{O}_2, \dots, \mathcal{O}_T$ and a model $\lambda = (A, B, \pi)$, determine the probability of observing the sequence \mathcal{O} given the model λ .
2. Given an observation sequence \mathcal{O} and the model λ , determine the state sequence $S = s_1, s_2, \dots, s_T$ that is optimal in some way that best explains the observations.

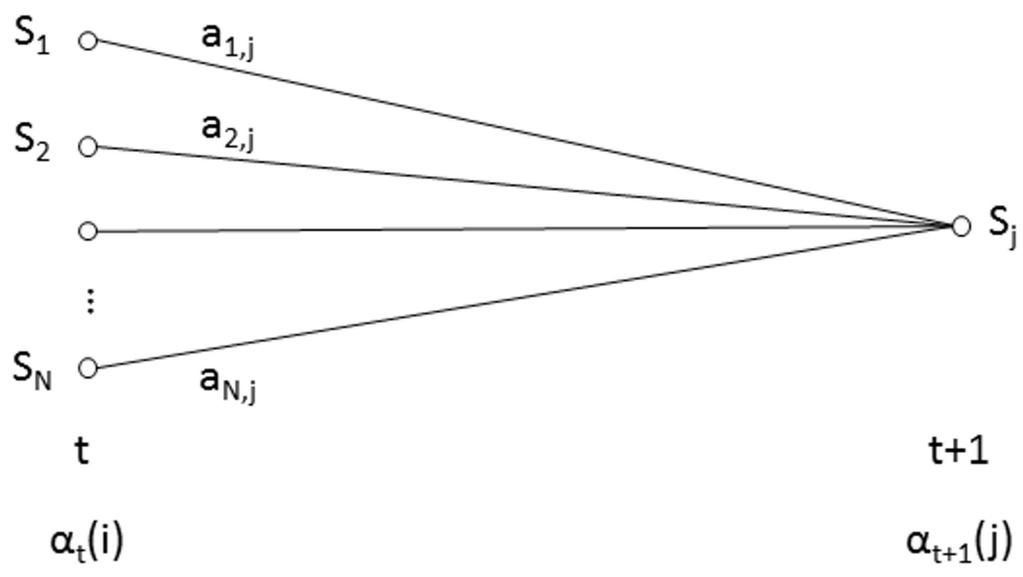
3. Adjust the model parameters $\lambda = (A, B, \pi)$ to maximize $P(\mathcal{O}|\lambda)$, the probability of observing the sequence \mathcal{O} given the model λ .

The three problems all have computationally simple computation methods. Problem 1 conceptually is somewhat trivial to solve (See Stamp [76] for a discussion of this and dynamic programming approaches). The obvious approach of summing over all possible state sequences involves on the $O(2TN^T)$ calculations but the standard HMM approach is N^2T computations [78]. In problem 2, we want to uncover the hidden part of the Hidden Markov Model [76]. This requires knowledge of the model and a set of observations. Problem 3 is classically called “the training problem” which is not a concern since we expect an existing model from another method such as EMAN or from the biology. Fortunately, all three problems have tractable solutions that are interrelated. We follow the terminology provided by Stamp [76] for solutions to the first two problems but with state indexing as $1 \leq i \leq N$ and observation indexing as $1 \leq t \leq T$ as shown in the diagrams from [78]. See [78, 76] for details on the solution method for problem three.

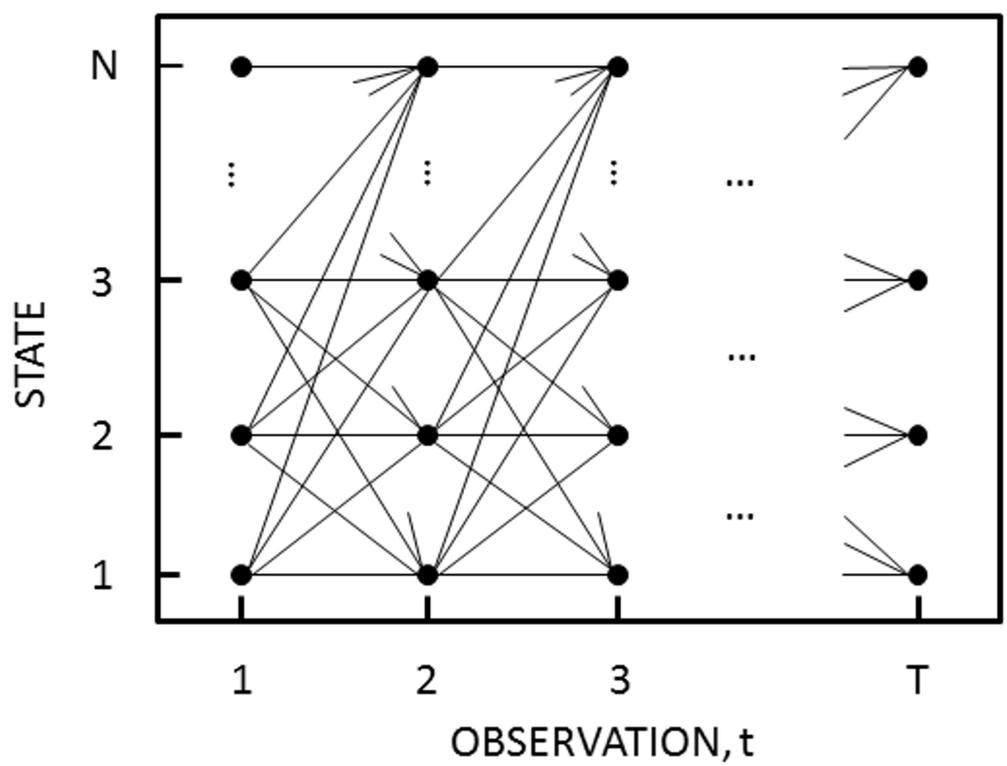
2.5.2.1 HMM Problem 1

The common solution to problem one is called the forward procedure. The algorithm is based on a lattice mesh [78] diagram shown in Figure 8 that helps visual the computations.

The rows of the mesh are based on the possible states and the columns of the mesh are the observations. Part (a) shows the probabilities of transitioning from each state, s_i , at time t to a new state s_j at time $t + 1$. Each cell in a column in the mesh represents a state at an observation time. Part (b) shows how all the states are interconnected across all observations. Each state has a probability of transitioning to each other state, including itself, so there is a graph edge from



(a)



(b)

Figure 8. Forward Algorithm Lattice Mesh (after [78]) (a) all *from* states to just one *next* state and (b) all *from* states to all *next* states.

each node in the given column in the mesh to each node in the next column in the mesh. This lattice allows for a computationally effective algorithm to compute the probability of the observation sequence \mathcal{O} given the model λ .

The procedure solves for $P(\mathcal{O}|\lambda)$ with a 3-phase process. This procedure defines the probability of a given partial sequence of observations as

$$\alpha_t(i) = P(\mathcal{O}_1, \mathcal{O}_2, \dots, \mathcal{O}_t, s_t = S_i | \lambda). \quad (2.5.4)$$

This represents the probability of the given partial sequence of length t ending in the defined state S_i with the HMM model λ . The initialization step in equation 2.5.5 sets the initial column of nodes in the mesh at observation time $t = 1$ in the mesh.

$$\alpha_1(i) = \pi_i b_i(\mathcal{O}_1), \quad 1 \leq i \leq N. \quad (2.5.5)$$

Thus, the partial probability is initialized for each of N possible states S_i as the probability of the initial state π_i being S_i times the probability of first observation being \mathcal{O}_1 given the first state is S_i . The middle induction step computes $\alpha_{t+1}(j)$ for each possible state. For each portion of the lattice, as shown in part (a) of Figure 8, equation 2.5.6 computes the joint probability that the prior states (at time t) reach the given state at time $t + 1$. The probability that the hidden process partial sequence takes on state S_i at time $t + 1$, given \mathcal{O}_{t+1} is

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^N \alpha_t(i) a_{ij} \right] b_j(\mathcal{O}_{t+1}), \quad 1 \leq t \leq T - 1, \quad 1 \leq j \leq N, \quad (2.5.6)$$

where a is indexed by i, j since the summation is over all possible previous states i for the next state j . The final probability of an observation sequence \mathcal{O} is then given by

$$P(\mathcal{O}|\lambda) = \sum_{i=1}^N \alpha_T(i), \quad (2.5.7)$$

which is the summation of the final joint probabilities for each of the N states at time T .

The forward procedure is a very computable algorithm. The forward procedure requires on the order of $T \cdot N^2$ calculations as opposed to the $2T \cdot N^T$ required by direct calculation. For $N = 5$ and $T = 100$, the forward method needs about 2500 computations instead of the 10^{72} needed for direct computation.

2.5.2.2 HMM Problem 2

The solution to problem 2 is not a simple closed form calculation as was the problem 1 solution. The “optimal” state sequence associated with the given observation sequence \mathcal{O} and the model λ can be defined in more ways than just a single closed form solution.

Some solutions to the second HMM problem (estimation of most probable states) require a helper variable for the backwards version of α . We define $\beta_t(i)$ as

$$\beta_t(i) = P(\mathcal{O}_{t+1}, \mathcal{O}_{t+2}, \dots, \mathcal{O}_T | s_t = S_i, \lambda). \quad (2.5.8)$$

Following a similar approach as was used to compute α , we perform the initialization step as

$$\beta_T(i) = 1, \quad 1 \leq i \leq N. \quad (2.5.9)$$

The induction step is given by

$$\beta_t(i) = \sum_{j=1}^N a_{ij} b_j(\mathcal{O}_{t+1}) \beta_{t+1}(j), \quad T-1 \geq t \geq 1, \quad 1 \leq i \leq N, \quad (2.5.10)$$

where it is noted that a indexed by (i, j) since the summation is over all possible current states j to the previous states i .

If the best estimate is defined by choosing the states that are individually most likely (at each observation time t), then the variable

$$\gamma_t(i) = P(s_t = S_i | \mathcal{O}, \lambda) \quad (2.5.11)$$

denotes the probability of being in state S_i at time t . Equation 2.5.11 can be

expressed in terms of the forward and backwards variables as

$$\gamma_t(i) = \frac{\alpha_t(i)\beta_t(i)}{P(\mathcal{O}|\lambda)} = \frac{\alpha_t(i)\beta_t(i)}{\sum_{k=1}^N \alpha_t(k)\beta_t(k)} \quad (2.5.12)$$

where $\alpha_t(k)$ accounts for the partial observations up to and including time t and $\beta_t(k)$ accounts for the partial observations after time t . Division by $P(\mathcal{O}|\lambda)$ makes $\gamma_t(i)$ a probability measure by normalizing over the possible set of states at time t . As such, we can solve for the most likely state s_t at time t by taking the maximum value of $\gamma_t(i)$ for all states at time t . The resulting set of states is therefore the more probable set of states from a purely statistical definition. But, there could be problems with the computed states since some states could have a zero probability of being selected from the previous state. Other methods of defining the “optimum” state sequence are needed.

Some possible definitions of the most likely sequence could be based on subsequences. For example, a sequence that maximizes the number of correct pairs or triples could be used. This option, however, could lead to gaps if the states are not mapped in some overlapped way. Or, if the states are overlapped, other discontinuities in the solution could arise. The best way is to maximize the entire sequence of states. Luckily, the Viterbi algorithm [80, 81] provides a well documented solution based on dynamic programming techniques. Rabiner [78] provides an overview of the Viterbi algorithm using the same notation as above; his derivation is presented below.

To find the single best state sequence, $S = \{s_1, s_2, \dots, s_T\}$, for the given observation sequence $\mathcal{O} = \{\mathcal{O}_1, \mathcal{O}_2, \dots, \mathcal{O}_T\}$, we need to define the quantity

$$\delta_t(i) = \max_{s_1, s_2, \dots, s_{t-1}} P[s_1 s_2 \dots s_t = i, \mathcal{O}_1 \mathcal{O}_2 \dots \mathcal{O}_t | \lambda] \quad (2.5.13)$$

i.e. $\delta_t(i)$ is the best score (has the highest probability) along a single path, at time t , which accounts for the first t observations and ends in state S_i . By induction

we have

$$\delta_{t+1}(j) = \left[\mathbf{max}_i \delta_t(i)a_{ij} \right] \cdot b_j(\mathcal{O}_{t+1}). \quad (2.5.14)$$

To actually retrieve the state sequence, we need to keep track of the argument which maximized (2.5.14), for each t and j . We do this via the array $\psi_t(j)$. The complete procedure for finding the best state sequence can now be stated as follows.

The initialization steps are:

$$\delta_1(i) = \pi_i b_i(\mathcal{O}_1), \quad 1 \leq i \leq N \quad (2.5.15)$$

and

$$\psi_1(i) = 0. \quad (2.5.16)$$

The recursion steps are

$$\delta_t(j) = \mathbf{max}_{1 \leq i \leq N} [\delta_{t-1}(i)a_{ij}] b_j(\mathcal{O}_t), \quad 2 \leq t \leq T, \quad 1 \leq j \leq N \quad (2.5.17)$$

and

$$\psi_t(j) = \mathbf{argmax}_{1 \leq i \leq N} [\delta_{t-1}(i)a_{ij}], \quad 2 \leq t \leq T, \quad 1 \leq j \leq N. \quad (2.5.18)$$

The termination steps are:

$$p^* = \mathbf{max}_{1 \leq i \leq N} [\delta_T(i)] \quad (2.5.19)$$

and

$$q_T^* = \mathbf{argmax}_{1 \leq i \leq N} [\delta_T(i)]. \quad (2.5.20)$$

The state sequence path backtracking step is:

$$s_t^* = \psi_{t+1}(s_{t+1}^*), \quad T-1 \geq t \geq 1. \quad (2.5.21)$$

It should be noted that the Viterbi algorithm is similar (except for the backtracking step) in implementation to the forward calculation of (2.5.5)-(2.5.7). The major difference is the maximization in (2.5.17) over previous states which is used in place of the summing procedure in (2.5.6). It also should be clear that the trellis structure efficiently implements the computation of the Viterbi procedure.

2.5.2.3 HMM Problem 3

While there are fairly straight-forward approaches to solving the first two problems, the “training” problem solution is not of closed form. See Rabiner [78] for more details on the solution to problem 3. In [78], the Baum-Welch method is presented as one possible technique. Other possible approaches are also discussed. And since we expect to have a model based on biological or other reconstruction techniques, training is not expected to be an issue in the reconstruction of transparent objects from a single image. In fact, “training” may yet prove to be a useful tool if many images are considered but this is beyond the scope of this research.

2.5.2.4 Other Types of HMM

The above discussion is in regards to a special (but commonly used) case of a HMM. The above HMM models a discrete first order Markov chain where each state only depends on the prior state and the observations take on discrete values. There are other, more complex models available, with correspondingly more complicated solutions.

A common constraint to the model is to limit the states that can be reached from a given state as a function of the state or time. A one-step ergodic (or fully connected) model has the property that each state can be reached from every other state. The simplest constraint is to reduce the number of possible state transitions from each state. Thus, some of the A matrix coefficients will be zero. Models such as the Bakis or left-right model [82, 83] only have state indexes that increase with observation time. These models have an A matrix that is upper triangular, which leads to additional constraints to the most probable state sequence.

Another extension of HMMs is to allow for multiple observations. This seems to be a natural extension to the problem at hand given the desire to use multiple images. However, much research in this area, see for example [84, 85, 86, 87], was

directed towards training of the HMM engine (problem 3). As noted above, the training is beyond the scope of this research.

Other research investigated use of state sub-sequence length to aid in the modeling. Zaki et al. [88] present a variable order HMM that includes duration periods for states. This extension of prior states leading to the next state means that the simple first order HMM techniques do not hold. In this work, very long sequences such as protein amino acid chains were modeled in order to perform protein classification. While Figure 5 shows some repeating structure, the voxel sequence lengths are not expected to be long enough to provide suitable repeats for the VOGUE algorithm. Thede [89] provides a second order HMM for speech use. This HMM uses statistics for the two prior states to determine probabilities for speech recognition. In this work, the standard HMM A matrix is extended to a third dimension to account for the additional prior state. However, the A elements are weighted based on the number of occurrences of triples. The voxel sizes of the 3-D reconstruction problem would need to be carefully selected so that the weighting factors would make biological sense. du Preez [90] provides a method of reformulating any higher order HMM to a first order model. The ORED algorithm makes unnecessary the extending of specific HMM algorithms to specific higher orders. This will certainly have value if a higher order HMM can be formulated that solves the current problem. Each of the preceding ideas has issues that prevent its direct use to the individual particle reconstruction problem, they all have nuggets that may be exploited. Lee [91] used Higher Order HMMs to recognize Mandarin Digits. While in the acoustic space, Lee constrains the beginning and end states to be non-utterances, which is similar to our expected constraint of sequences beginning and ending in suspension buffer material rather than target matter. This paper derives recursive formulas for the log-likelihood of

partial paths. Dewar et al. [92] extend the concept of HMMs to include not only a probabilistic state value but also a duration variable. This method was used to estimate distributions from Monte Carlo simulations but the use seems promising given the repeating structure in Figure 5.

CHAPTER 3

Methodology

The purpose of this research was to seek a method of performing a 3-D reconstruction of an individual transparent particle, preferably from a single *cryo-tem* 2-D image. Having only a single image leads to an obvious question as to whether *any* meaningful data can be gleaned from the single image. Support vector machines (SVM) were applied [75] in an attempt to create a mathematical model that could be used in the reconstruction processing. This SVM processing provided valuable insights into the *cryo-tem* imagery and processing. However, the attempted SVM approaches resulted in classifying pixels associated with either the particles or the background based on the mean of the pixel data. No obvious support was found for determining the 3-D structure in small angle rotation images. Additional processing performed in this research [93, 94] looked at the ability to use standard correlation processing to separate images of simulated targets when rotated at small angles. The preliminary research concluded by establishing achievable accuracies of computing the particle height and diameter from real data [95]. While searching via correlation processing requires an inordinate number of basis models, these early results showed that it may be possible to perform reconstruction of individual transparent particles from a single projection image if enough a priori information can be deduced. That is, the information to perform a 3-D reconstruction seems to be there, how to glean it is the question. As is shown below, higher order HMM (HOHMM) processing was successfully used as the basis for reconstruction of a transparent object from a single image.

A software framework was developed to generate simulated projection images and to test various reconstruction methods. The software suite is called ARTEMIS

(**A** priori **R**econstruction from **T**ransmission **E**lectron **M**icroscope **I**mage **S**ets). This software is outlined in Section A.1 while the details of the processing are provided in the remainder of Appendix A. With this software, a reconstruction technique that utilized a single projection image was developed. The remainder of this chapter describes the processing that was performed to provide reconstruction of a transparent object from a single image. Underlying concepts of HMM processing for reconstruction are discussed in Section 3.1. The extended HMM processing that is the major product of this research is detailed in Section 3.2.

3.1 HMM Reconstruction Processing Concepts

The reconstruction of density voxels in 3-D space can be considered to be a HMM problem. HMM processing is well suited to finding the most probable chain of hidden states from an associated set of observations. The reconstruction of a transparent object problem becomes one of determining the most probable 3-D voxel space such that the chains of states (where the voxel densities represent the unobservable states) can be deduced from the observables that are the pixels in the *cryo-tem* imagery.

The geometry for this reconstruction problem is depicted in Figure 9. The 3-D voxel space represents a 3-D matrix of density values for the target of interest (blue discoid object) and the background buffer solution (empty voxels) in which the target rests. The red arrow follows the contributions of the voxels that contribute to the center pixel in the projection image. All of the projection pixels are generated in a similar fashion. The image is assumed to have been generated from parallel rays (as is the case in *cryo-tem*) so the rays that create the projection image pixels follow projection paths that are orthographic to the image plane.

The remainder of this section provides background information regarding the use of HMM processing to reconstruct transparent objects from a single projection

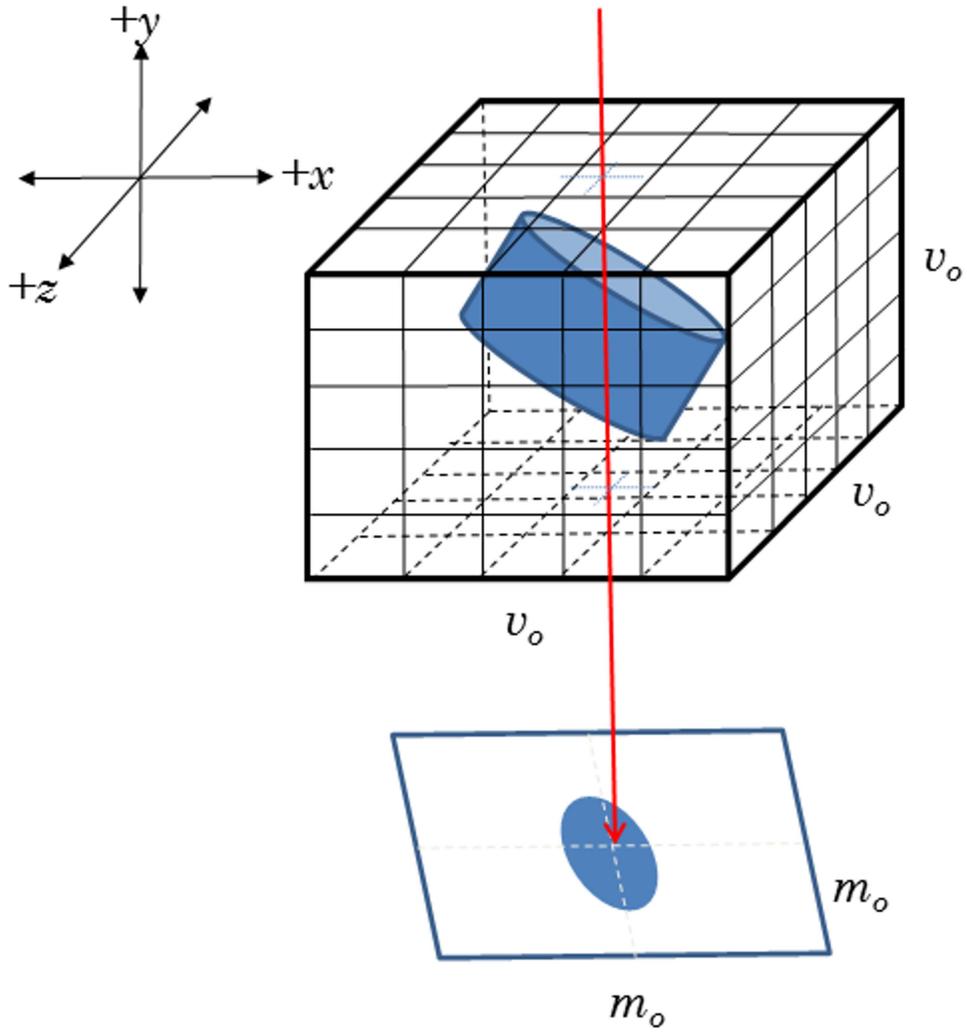


Figure 9. Single Image Reconstruction Geometry

image. Section 3.1.1 discusses basic FOHMM reconstruction processing. Simple HOHMM processing is described in Section 3.1.2. Section 3.1.3 details the training of the HMM processing model with respect to noise. The complete HOHMM processing steps are detailed in Section 3.2.

3.1.1 Basic FOHMM Reconstruction Processing

Determination of the most probable states in a hidden Markov chain involves traversing the states and applying the a priori state-to-state statistics as well as the a priori state-to-observation statistics in order to determine the chain of most probable state (density) values. The 3-D view of the problem statement in Figure 9 can be considered in two dimensions to understand the construction needed for HMM processing. Figure 10 shows a 2-D slice (in the $\{x, y\}$ plane) of the 3-D voxel space and associated observation (pixel) values. The three colors {red, green, blue} represent different density levels of the target and white represents the suspension buffer material. The density is considered to be consistent throughout each voxel so each of the yellow dots along the black lines represents a state in the given Markov chains. The voxel size can be set arbitrarily small to ensure this consistency constraint with only a linear computational penalty. The gray-levels at the bottom represent the observed pixel values associated with this slice of the reconstruction space. Pixels created from just the lower density biological buffer material have larger gray-level values since more of the electron beam particles pass through this sample area. Likewise, regions of the more dense target have varying but generally lower gray-level projection pixel values.

The HMM model probabilities can easily be deduced from the input model. The A matrix is computed by traversing the base model states and tracking the *from* and *to* state transitions. For example, there are 3 white-to-white, 1 white-to-blue, 1 blue-to-green, 8 green-to-green, 1 green-to-blue, 1 blue-to-white, and 3

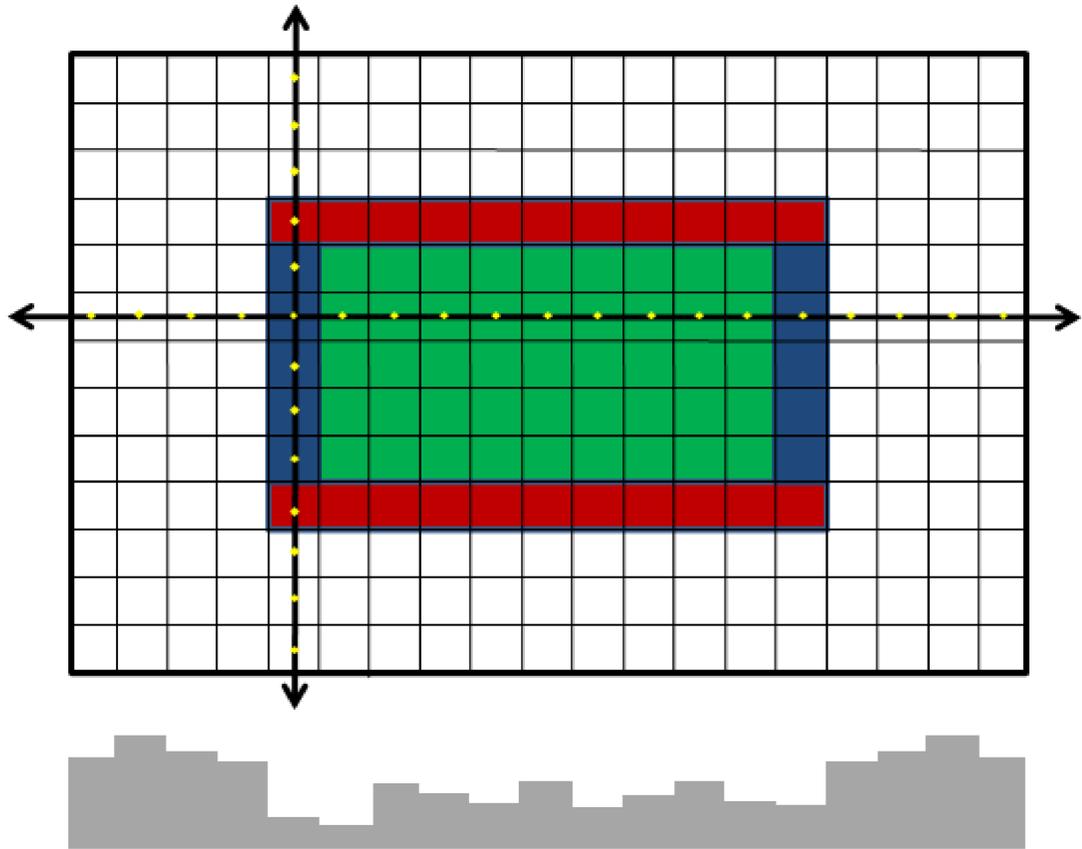


Figure 10. First Order HMM Statistics Generation

additional white-to-white transitions along the black horizontal line. All of these transitions are totaled for each direction (e.g. left-to-right) or for some combination of directions and then are normalized to be row-stochastic (the sum of probabilities from each *individual* state to *all* states equals 1). The B matrix is computed in a similar fashion by tracking the observation that is seen for each of the available states. For an 8-bit gray-level image this means that each state can be mapped to one of 256 possible observations (pixel values). Again, the B matrix is normalized so that the sum of all probabilities for each *individual* state to *all* possible observations equals 1. The B matrix construction is not direction dependent. The π vector of initial probabilities is deduced from the first element of each stack in the voxel space. An assumption is made that the Markov chains are initially in the buffer state since we do not want target particles touching the edge of the reconstruction space so π always forces the initial state of each chain to be the buffer {white} density state.

The basic HMM processing occurs as noted in section 2.5.2 for each Markov chain (stack through the reconstruction space). The black lines with yellow dots represent two orthogonal stacks (or chains) With a single model, $\lambda = \{\pi, A, B\}$ constructed as shown above, each new set of observations (projection image pixels) is processed to determine the most probable chain of states (densities). The trellis computations are made based on the λ model statistics and new observation values.

There are two basic approaches for computing the most probable state chain. One approach is to generate the statistics in a single direction; that is from a single face of the voxel cube. The other approach is to generate the combined statistics from all 6 faces of the voxel cube. The former gives six separate sets of statistics against which the target can be processed while the latter gives a single set that includes directional dependencies. Both methods assume a single projection image.

The processing for this research is performed based on six directional sets of statistics that creates six separate reconstructions that have to be merged in order to have a final reconstruction result. For each voxel, the state that wins a simple majority vote is selected. For votes that have a tie, the state with the highest probability as computed by the trellis is selected. The results of this simple FOHMM processing are shown in Section 4.1.1. This approach is obviously not viable so other variations of HMM processing were explored.

3.1.2 Simple HOHMM Processing

As noted in Section 2.5.2.4, there are several variants of HMM processing. HOHMM processing generally relies on additional state to state information while FOHMM processing only uses the prior state and the observation statistics.

The HMM problem 2 processing was modified to include the *previous* and *from* state information as well as the *next* state. The *from* and *previous* data was tracked so that the appropriate statistics were applied as a part of the A matrix part of the α -pass. The β and γ passes were not altered. The determination of the most probable sequence (equation 2.5.17) used the extended HMM model information from the A matrix results in the most probable state determination step. As was shown in Section 4.1.2, the additional state information provided suitable results for only the vertical direction, but it showed that additional state information could provide a suitable reconstruction method.

3.1.3 Training the HMM

The basic concepts in this research include the training of the HMM model. This training is applicable for all variants of the HMM reconstruction use. The “training” aspect of a HMM as defined here is *not* the same as determining the underlying HMM as defined in Problem 3 of [78]. In our use, training means show-

ing the HMM one or more images in order to glean an accurate understanding of the state-to-observation statistics. That is, enough images must be examined to ensure that the B-matrix properly represents the actual state-to-observation statistics. Therefore, our training is performed by building up the state-to-observation statistics across a necessary set of images then normalizing to make the B-matrix row-stochastic as required for the HMM processing.

From the above discussion, it should be obvious that with no noise added and a perfectly separable target with N states, the state-to-observation probabilities results in N non-zero regions across the gray level(s) representing each state. This case (for $N = 4$) is shown in Figure 11. The four state-to-observation probabilities

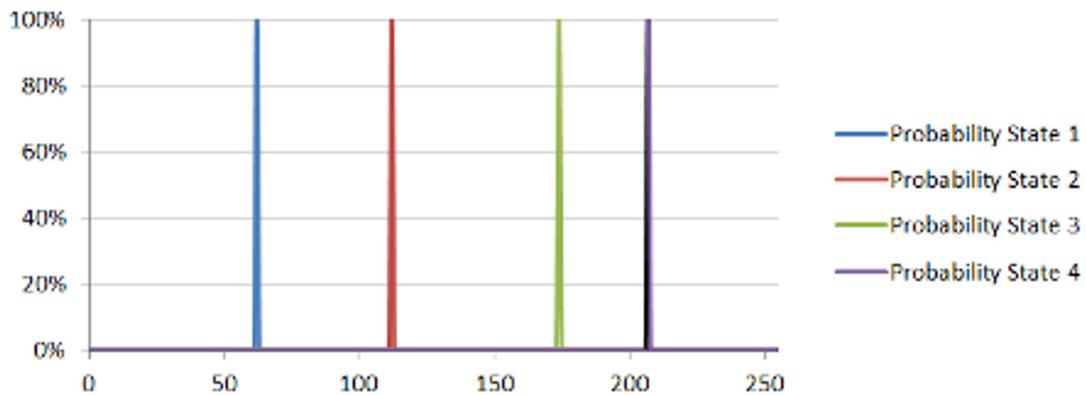


Figure 11. Example State-to-Observation Statistics

for the separable states are shown in blue, red, green, and black. Of course this is the resulting set of statistics since no matter how many images are viewed, the states all pile up in the shown relative bins and are normalized to 100%.

Training therefore means showing sufficient images so that the probabilities are filled in around the delta functions for each state. In effect, a histogram of the noise is built around each center value. With background noise following a Gaussian distribution, the delta functions normalize to normal distributions if sufficient images are examined.

Since the noise of the images (whether simulated or real) is known, the HMM state-to-observation statistics (B-Matrix) can be computed as well. The unnormalized version of the occurrence histogram shown in Figure 11 is simply the count of voxels in each state. If the adjacent zero values are incremented by the normal distribution value of the offset from the center value times the center value sum then the resulting distribution for each state represents the perfect state-to-observation statistics. This is repeated for all such non-zero values for each state, then the B-matrix is made row-stochastic. The resulting B-matrix represents the complete state-to-observation probabilities while only needing a single noise-free image simulated from the model target states.

3.2 Extended HMM Reconstruction Processing

This section provides a formalism that allows the HOHMM processing to extract the structure of a transparent object from a single projection image based on the HOHMM model of the target. This research uses a HOHMM schema based on both multiple prior states and duration in each of the prior states. A HOHMM that includes two prior states and duration in each of the prior states is effective in reconstructing the simple discoid target. The schema is easily extended to more than two prior states for more complex structures. The following sections detail the extension of HOHMM processing with L multiple levels and durations in each prior level.

3.2.1 Initial State Statistics

The probability of the initial state is extended by including L levels before the first actual state in the reconstruction zone. The initial probability matrix, π_l , is defined as a set of $(L+1)$ -tuples based on an imaginary set of states $\{\psi_i\}$ that exist outside of the 3-D reconstruction voxel space. These imaginary states

are not part of the reconstructed space but are used to help the extended HMM processing get started from its initial state. Given that the initial state is assumed to be the buffer state, the π_2 vector is simply $P\{\psi_1, 1, \psi_0, 1, W\} = 1$.

3.2.2 State to State Statistics

In this extended reconstruction schema, the elements of the extended A matrix follow the convention of the FOHMM A matrix which defines $A(f, n) = P\{s_f \rightarrow s_n\}$. Here the states are defined as *from* = f and *next* = n is used instead of the traditional *to* = t in order to prevent confusion with the observation step t . For the extended schema, the state-to-state statistics are defined as

$$A'_{(s_{f'}, s_n)} = P\{s_p, d_p, s_f, d_f \rightarrow s_n\}, \quad (3.2.1)$$

where $s_{f'}$ is the *from* state node that is defined as the 4-tuple of s_p , the *previous* state, d_p , the duration for which the chain was in the *previous* state, s_f , the most recent *from* state, d_f , the duration for which the chain was in the *from* state, and s_n , which is the *next* state.

The new $a(s_{f'} = \{s_p, d_p, s_f, d_f\}, s_n)$ values can be easily computed from the model voxel space as shown in Figure 12. The circled regions denote states and durations. Not shown are two “pseudo” states $\{\psi_0, \psi_1\}$ at the beginning of each chain that are used to start the process. The extended states denoted by the horizontal black line in Figure 12 are summarized in Table 2. Note that the transition from a given density to itself is still considered a true state transition because the *from* state is defined as density and the duration in that density. Thus, a transition from green to green counts as a new transition because the new *next* state-tuple has a duration of one more than the *from* state-tuple.

As with the original A matrix, the elements of A' are normalized so that the probability of transitions *from* each prior-state tuple state is row-stochastic. In the horizontal example shown in Figure 12, the tuple $\{\psi_1(1), W(4)\}$ transitions 6 times

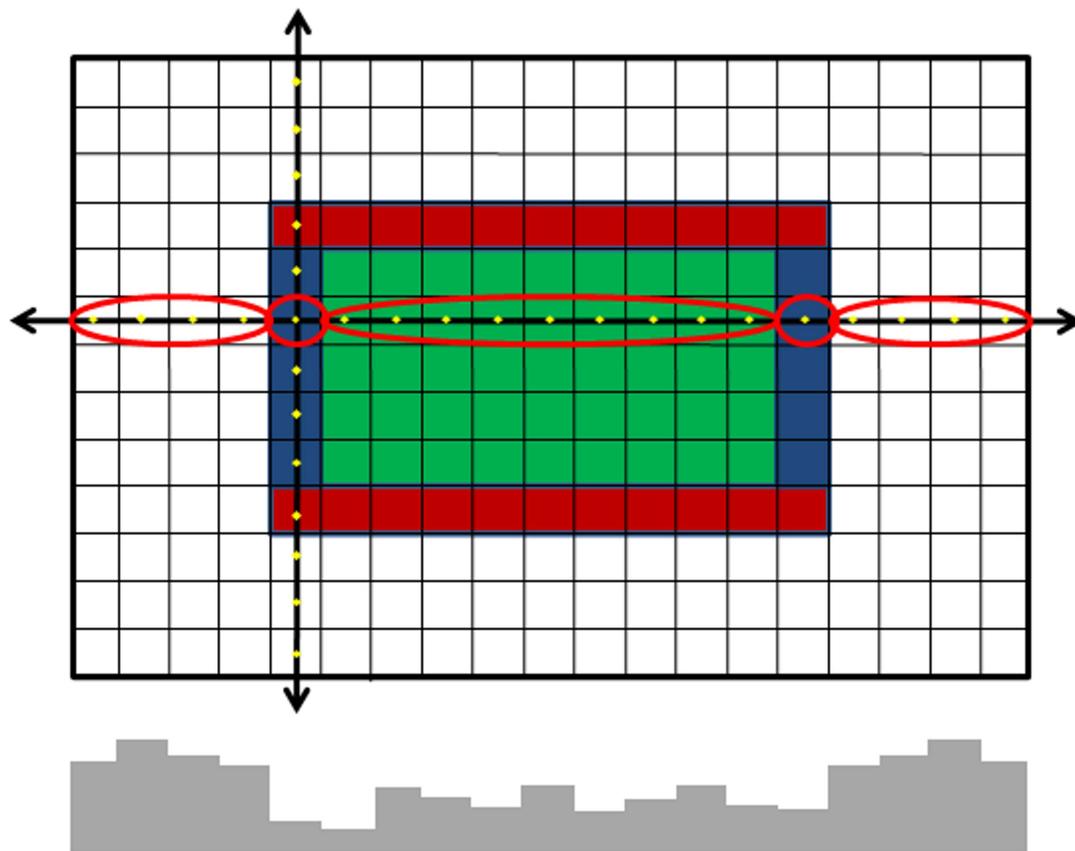


Figure 12. Higher Order HMM Statistics Generation

<i>previous</i>	D_p	<i>from</i>	D_f	<i>next</i>
ψ_0	(1)	ψ_1	(1)	W
ψ_1	(1)	W	(1)	W
ψ_1	(1)	W	(2)	W
ψ_1	(1)	W	(3)	W
ψ_1	(1)	W	(4)	B
W	(4)	B	(1)	G
B	(1)	G	(1)	G
B	(1)	G	(2)	G
B	(1)	G	(3)	G
B	(1)	G	(4)	G
B	(1)	G	(5)	G
B	(1)	G	(6)	G
B	(1)	G	(7)	G
B	(1)	G	(8)	G
B	(1)	G	(9)	B
G	(9)	B	(1)	W
B	(1)	W	(1)	W
B	(1)	W	(2)	W
B	(1)	W	(3)	W

Table 2. Extended HMM state example where $\{W,R,G,B\}$ represent the normal density states and $\{\psi_0,\psi_1\}$ represents the “pseudo” states. The number in parenthesis following each state represents the duration the chain has been in the associated state. No number is required for the *next* state since its duration is always 1.

to the W state, twice to the R state and 5 times to the B state. When normalized, $P\{\psi_1, 1, W, 4 \rightarrow W\} = 6/13$, $P\{\psi_1, 1, W, 4 \rightarrow R\} = 2/13$, and $P\{\psi_1, 1, W, 4 \rightarrow B\} = 5/13$.

This expanded trellis node structure is shown in Figure 13. The left side of the single trellis fan shows example *previous*, *from* nodes. Note that some 4-tuples cannot coexist at each step in the overall trellis. For example, the first state-tuple shown cannot occur concurrently with the second state-tuple since S_2 cannot be of length 1 and 2 at the same time step. However, the A probability may exist since in one area of the voxel space the $(S_1, 1, S_2, 1) \rightarrow (S_1, 1)$ transition may occur and in

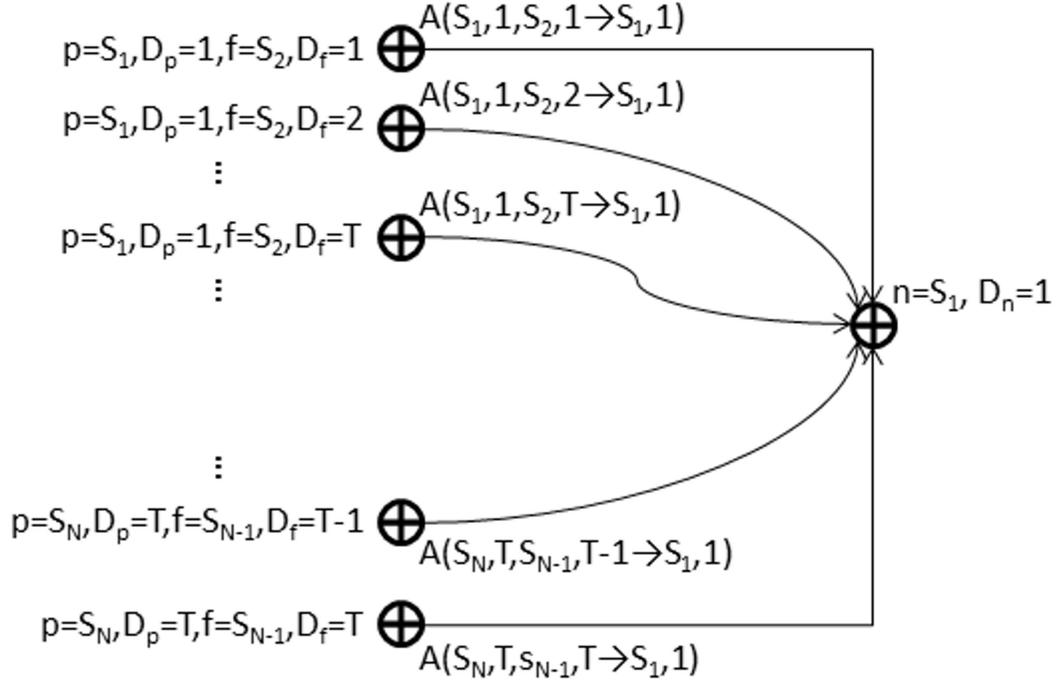


Figure 13. Expanded HOHMM Trellis Structure

another area the $(S_1, 1, S_2, 2) \rightarrow (S_1, 1)$ transition may happen. This information can be utilized to minimize memory and processing requirements.

Of note in this scheme is the fact that a given state symbol can be in both the *from* and *next* tuples. Specifically, the *next* state is always of duration 1 since it has just been reached. But, the construction allows for the chain to transition from a state symbol s' with a duration d' to the same state symbol s' . While the overall duration of the symbol s' at this locale in the chain is now $d' + 1$, we only care about the *from* duration d' since the *next* duration by necessity is $d' + 1$. In general, the states can be expressed as tuples in the form $\langle S_{f_{L-1}}; \dots; S_{f_0}; S_n \rangle$, where S_{f_i} represent a series of states and durations and the S_n is either the same as the last from state S_{f_0} or is a new state. This feature not only handles state to state statistics of $\langle S_{f_i}; d_i \rangle$ to $\langle s_n; 1 \rangle$ but it also allows for treatment of

$\langle S_{f_i}; d' \rangle$ to $\langle S_{f_i}; d' + 1 \rangle$. This provides for a more full-bodied set of statistics that can better model the reconstruction space. From Figure 13 it also can be seen that all of the *next* states for each of the sub-trellises become the *from* states for the next step in the overall chain processing.

3.2.3 Flattening the HOHMM

The HOHMM structure presented above can have a very large state to state probability structure. For the simple FOHMM case, the A matrix is dimensioned as $N \times N$, where N is the number of states. For our HOHMM structure, each “state” is now defined by the list of *from* tuples of state and duration as well as the *next* state. With L levels of history in the *from* state and T possible duration values, there can be $O((TN)^L)$ possible *from* states. And in the general case, there are the same number of *next* states. For our example case where $N = 4$, $T = 65$, and $L = 2$, this equates to an A matrix that is approximately $2^{16} \times 2^{16}$ elements in size. Since the processing for the HMM runs as N^2T and the number of states is large, the processing may become untenable.

A first step to increase the usability of this extended HOHMM is to flatten it into a FOHMM. This compression occurs easily by traversal of the original model in order to build the list of the possible states-duration to state transitions. Even when flattened, a completely random voxel space would result in nearly the same $N(TN)^L$ possible *from* state-duration to *next* state combinations. But, any reasonably expected state pattern will result in significantly less elements in the list; the number of combinations for the simple model presented above is only $O(TN^2)$.

The flattened extended FOHMM provides a way to optimize the processing for computational efficiency. If the modeled object does not completely fill the voxel space, there are at most two paths through the voxel space: one that misses the

object and one that intersects the object. The state chain that misses the object obviously has T tuples. Each of the chains that does intersect the object will add less than T elements since the first elements before the intersection will just add to the statistics of the non-intersecting chain. For a reasonable small number of states, and an object height of H , the total number of list elements will be approximately $T + N((T - H)/2 + H)$. For our example case ($T = 65, N = 4, H = 20$) we expect about 320 elements. Given the simple model, there are only 128 state elements so the computational and memory savings are significant.

The state chains that miss the object start with the initial tuple that includes the external imaginary states and ends with the $T - 1$ tuples for each step along the non-target chain. For two levels, $L = 2$, a non-object state denoted as X , the *from* tuples denoted with single brackets $\langle state; duration \rangle$, and the *next* tuple(s) denoted with double brackets, the chain that does not intersect the object contains the state tuples

- $\{\langle N + 1; 0 \rangle, \langle N; 0 \rangle, \langle\langle X; 0 \rangle\rangle\}$,
- $\{\langle N; 0 \rangle, \langle X; 0 \rangle, \langle\langle X; 0 \rangle\rangle\}$,
- $\{\langle N; 0 \rangle, \langle X; 1 \rangle, \langle\langle X; 0 \rangle\rangle\}$,
- ...
- $\{\langle N; 0 \rangle, \langle X; T - 3 \rangle, \langle\langle X; 0 \rangle\rangle\}$, *and*
- $\{\langle N; 0 \rangle, \langle X; T - 2 \rangle, \langle\langle X; 0 \rangle\rangle\}$,

which has T members. Note that a duration of 1 is denoted as 0 for coding convenience.

When the state chain does intersect the object, elements numbering less than T are added to the state list. If the object is of height H with a single state Y ,

then there are only H additional states added to the overall list where the state chains do intersect the object. In this case, additional states such as

- $\{ \langle N; 0 \rangle, \langle X; (T - H)/2 \rangle, \langle \langle Y; 0 \rangle \rangle \}$, // first intersecting voxel
- $\{ \langle X; (T - H)/2 \rangle, \langle Y; 0 \rangle, \langle \langle Y; 0 \rangle \rangle \}$, // second intersecting voxel
- ...
- $\{ \langle X; (T - H)/2 \rangle, \langle Y, H - 2 \rangle, \langle \langle Y; 0 \rangle \rangle \}$, and // last intersecting voxel
- $\{ \langle X; (T - H)/2 \rangle, \langle Y, H - 1 \rangle, \langle \langle X; 0 \rangle \rangle \}$ // first voxel back into non object

are added to the overall list of transition elements. Once the chain leaves the object, the object state becomes the oldest *from* tuple and the remaining states are added in the form $\{ \langle Y, H - 1 \rangle, \langle X; n \rangle, \langle \langle X; 0 \rangle \rangle \}$ until the voxel space is traversed. It should be obvious that with a small number of expected states, the growth in list elements will be limited as more *next* states gets added to existing elements rather than adding new elements.

These new *next* states just change the stochastic results without adding to the computational burden. Since each list item consists of *from* tuples and the probability of the *from* tuples transitioning to each of the internal states, the state tuple $\{ \langle N; 0 \rangle, \langle X; ((T - H)/2) - 1 \rangle, \langle \langle Y; 0 \rangle \rangle, \langle \langle X, 0 \rangle \rangle \}$ does not add an additional item to the overall count. The list of *to* states in that element just grows to handle the second possible destination state from the given *from* tuples. And, since the *next* states are always duration = 1, the storage for the *next* tuples can be reduced to a simple array of *next* states.

This construct allows for a simple data structure to describe the state to state statistics for the extended HMM. The list elements can be described as an array of L tuples describing the *from* states and associated durations and an array

of N states that represent the probabilities of the *from* tuple transitioning to each of those states. The requirement that A be row-stochastic is satisfied by normalization across all of the *next* states within each list element. The total number of list elements obviously depends on the structure of the object, but the total number of state tuples provides a much smaller basis set of “states” for all subsequent processing.

The state to state statistics can be generated by walking through the model definition voxels. This algorithm follows the discussion in Section 3.2.2. This process can be performed from any of the 6 voxel space cube faces in order to generate 6 different sets of statistics.

3.2.4 State to Observation Statistics

The state to observation statistics are generated by traversal through the model voxel space in the same general fashion as the state to state statistics. The state for each voxel was recorded for each direction during the state to state processing so it can be used for the observation to state processing. Remember that the simulated projection images are created by tracing from the pixel centers through the voxel space (see section A.2.1). In general, the voxel observations are not the same as the pixel values and thus cannot be used for the observation values. Thus, the observation for each voxel is determined by back-projecting from the voxel center to the image surface. The state to observation statistics are then tabulated and normalized so that the probability of a given state tuple gives rise to an observation.

Figure 10 shows the ray used for the observations for vertical stack. The density is consistent throughout each voxel so each of the yellow dots along the black line represents a state in the given Markov chain. Since a single observation is used for all of the states, the independence condition of the HMM is violated.

The independence problem can be solved with a single rotation of the reconstruction voxel space. As is shown in Figure 14, rotation of the space by 45° allows for 6 independent state chains to be reconstructed. These are shown by the bi-direction arrows and the head and tail symbols into and out of the front and back faces. The yellow spots represent the state centers and the dashed blue lines are

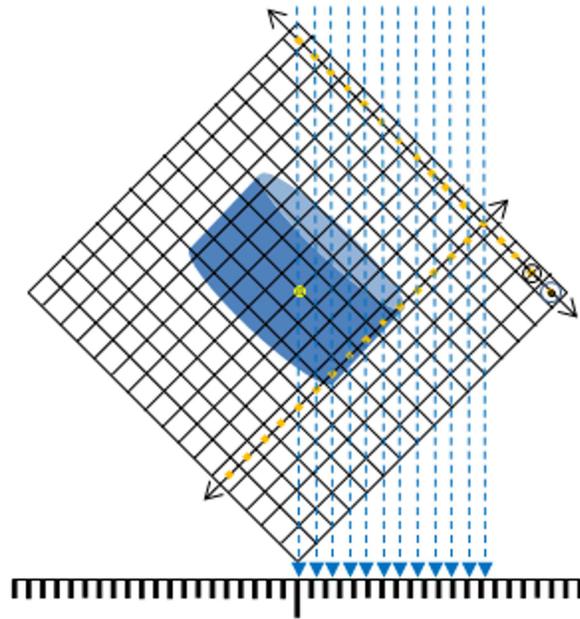


Figure 14. Extended HOHMM Processing Directions

the projection rays through the reconstruction space resulting in the projection image of the transparent object. As can be seen, the 45° rotation of the reconstruction space causes the rays denoting the observations for each state in a chain to not intersect the voxel space of adjacent states. This condition holds as long as the pixel sizes do not grow too large so that the same individual pixel value is used as the observation value for two state neighbors. For the case where the pixel size and the voxel face size are the same, the state centers are offset by $\sqrt{(2)}/2$ while the pixel centers are offset by 1. While the imaging generates a physical 50% overlap between adjacent voxels at 45° , the variations between the pixel center offsets

and the state center offsets allows for enough variation between the observations to allow the HMM processing to work as expected. Smaller pixels can be used to increase the independence. The constraint on pixel size versus voxel size can be relaxed further by using multiple pixel values to generate the observation value.

Each observation is determined by the intersection with the projection image of the orthographic ray from a state to the image. When the ray from each state center intersects the image surface, the intersection point lies in a subset of pixels in the image. Since the voxel centers generally do not align with the projection pixel centers, bilateral interpolation is used to determine the observation value.

Figure 15 shows the intersection point in a field F of 4 arbitrary pixels ($F = \{(r, c), (r, c + 1), (r + 1, c), (r + 1, c + 1)\}$). The large black dot represents

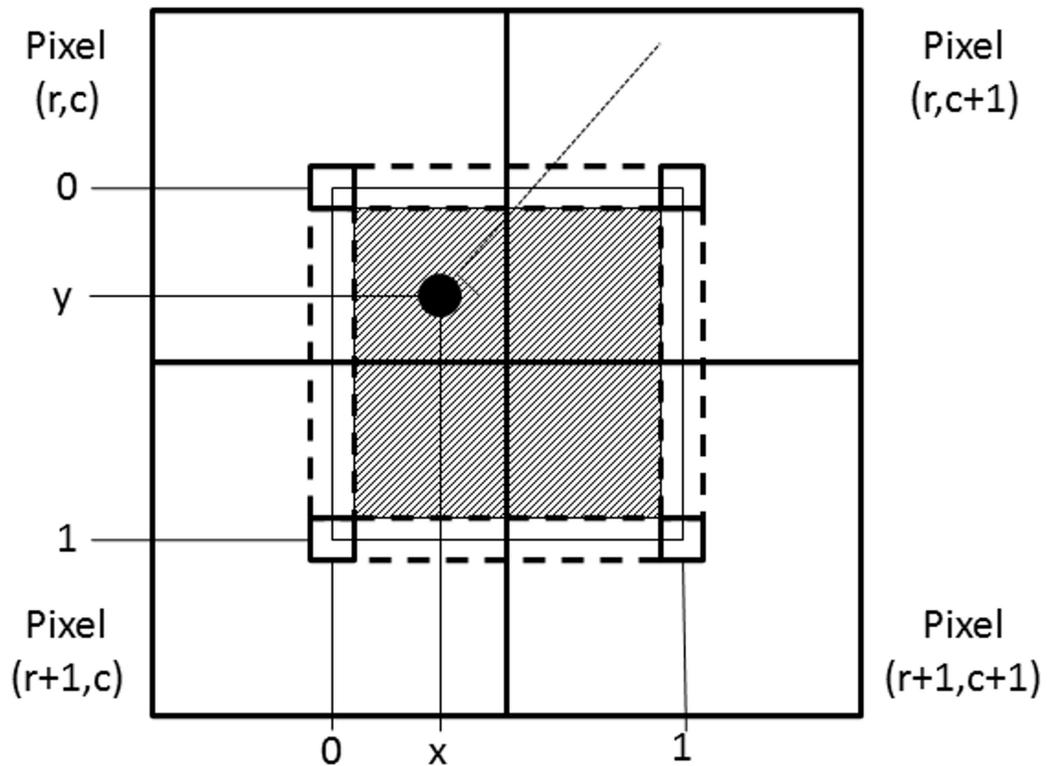


Figure 15. Bilateral Interpolation for HMM Observation Generation

the orthographic intersection of the projection line from the voxel centroid to the

image plane. The four smaller boxes represent regions where the current pixel value will be used as the observation value. These boxes are defined to prevent underflow for intersection points near the true pixel centers. Likewise, the dashed boxes represent areas in which a linear interpolation between just two pixel values will be used. All points that fall within the inner hashed area are computed according to Equation 3.2.2 [96].

$$\mathcal{O}(x, y, z, \theta, \psi) = [1 - x \ x] \begin{bmatrix} Pixel(r, c) & Pixel(r, c + 1) \\ Pixel(r + 1, c) & Pixel(r + 1, c + 1) \end{bmatrix} \begin{bmatrix} 1 - y \\ y \end{bmatrix} \quad (3.2.2)$$

All other intersection points are handled through the corresponding set of adjacent image pixels in a similar fashion. If the intersection is outside of the shown constraints, then an adjacent set of pixels are used. This leads to the need for an outer ring of extra pixels to support the observation generation. Thus, M_0 must account for this extra two rows and columns of pixels in the reconstruction geometry.

Deriving most probable sequences from the six cube faces leads to limits in the reconstruction of shapes that vary from the model target. If the target is held “square” to the voxel grid, then all state sequences from the $+Y$ or $-Y$ direction will all have the same statistics of length and state. This limits the shapes that can be reconstructed since the only height available is the one that matches the training target. If the target is placed at an angle in the voxel space grid, then a range of heights are included in the statistics. However, these statistics still have constraints. The reason for this limitation can be seen in Figure 16. If the target is rotated at an angle (around an axis) then the sloped top creates a range of state durations. However, there are only a few at the shortest length as the first wave hit the target top. As shown, the discoid has an increasing number of stacks at each duration value. The black line across the top represents the stacks that have the given length. As the line walks down the slope, more and more, then less and less voxel stacks end at the specified duration length.

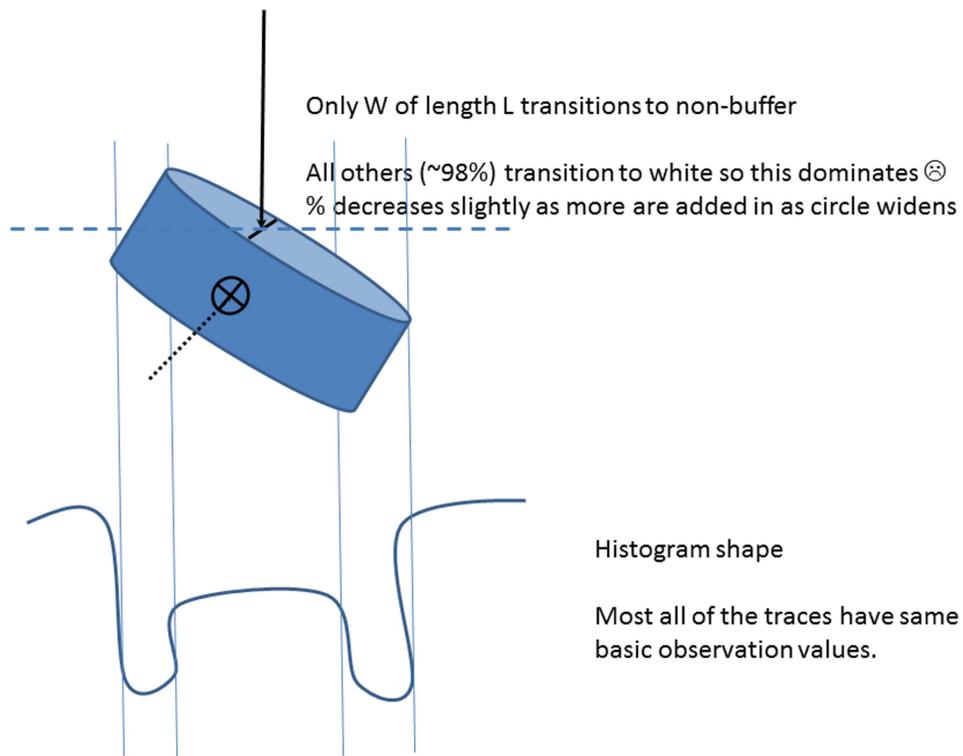


Figure 16. Model State Limitations

These state-to-state statistics are biased by the excessive longer durations. For each length that does intercept the target, there are W stacks that end at the target tangent line but many more advance to the next length. While there are W stacks that transition to the target, almost all other stacks (except those to the left that do intercept the target) transition to the buffer state. This drives the statistics to encourage a most probable state of buffer for those lengths. The statistics average out to about 98% for buffer and the rest for the target top layer. If the target is a larger percentage of the entire voxel space, this will decrease, but when the HMM problem two processing is performed, the buffer state will almost always be the winner.

This paradigm also has an effect when the target is held square to the grid space and the grid space is rotated to allow for independence of the observations (as is shown in Figure 14). When the most probable state direction is along the rotation axis, an effect similar to the one noted above occurs. Generally, only a few of the stacks register non-buffer states; all the others register buffer.

These effects can be handled by rotating the reconstruction space in the other direction (around the y -axis) so that the effective target tilt does not lead to this arrangement along the same physical axis of the target for each rotation. As is shown in Figure 17b, the original reconstruction cube is rotated around the z -axis that comes out of the screen. Four useful and two unusable reconstructions are created in this orientation. By rotating the reconstruction space around the y -axis three more times as shown, an additional 18 total and 12 useful reconstructions can be generated. The resulting reconstructed voxel spaces are then rotated to align it with the original space for the first four passes. This rotation schema is easy to implement by simply rotating the single projection image 90° around its center point for each processing set.

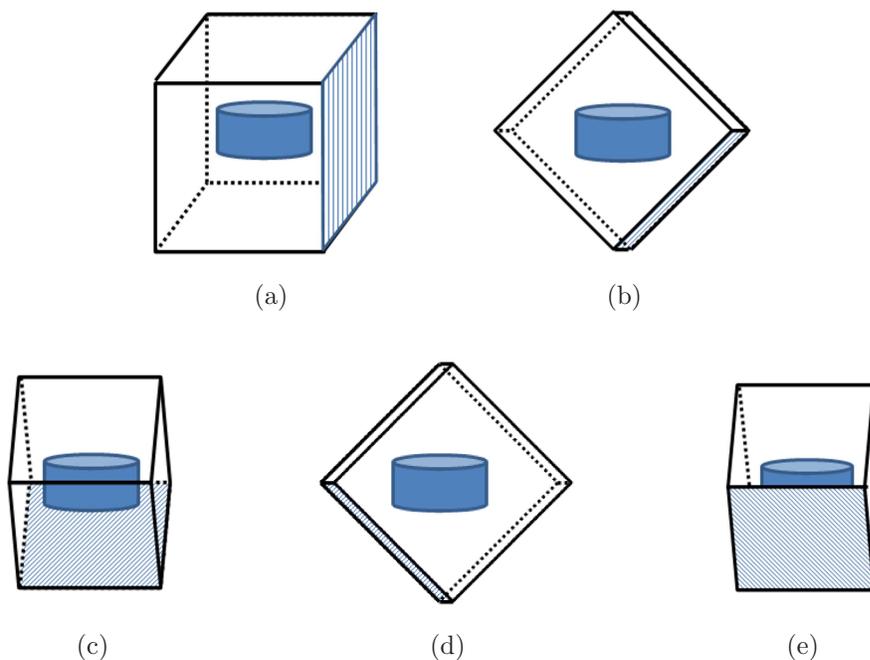


Figure 17. The original (a) unrotated reconstruction space and the rotated reconstruction spaces for the four cases of rotation around (b) Z-axis at 45° , (c) Y-axis at 90° , (d) Y-axis at 180° , and (e) Y-axis at 270° .

3.2.5 Alpha Pass

Recalling that the α -pass describes the partial probability that the state-chain is at a given value for each state-chain step in the forward direction, the extended HMM processing can be described in terms of the list of possible states. Figure 18 shows a single fan of the overall α -pass trellis between state chain steps t and $t + 1$. The prime notation for the states and the *hat* notation for the state-to-state probabilities represents the list nature of the possible transitions. There are N elements in the list of possible states so $\alpha_{t+1}(n')$ is created in the forward direction from all of those states and the corresponding transition probabilities. Using the state list identifiers of $next_state = n'$, the probability of a given partial sequence of observations at step $t + 1$ is

$$\alpha_{t+1}(n') = P(\mathcal{O}_0, \mathcal{O}_1, \dots, \mathcal{O}_t, s_{t+1} = S_{n'} | \lambda). \quad (3.2.3)$$

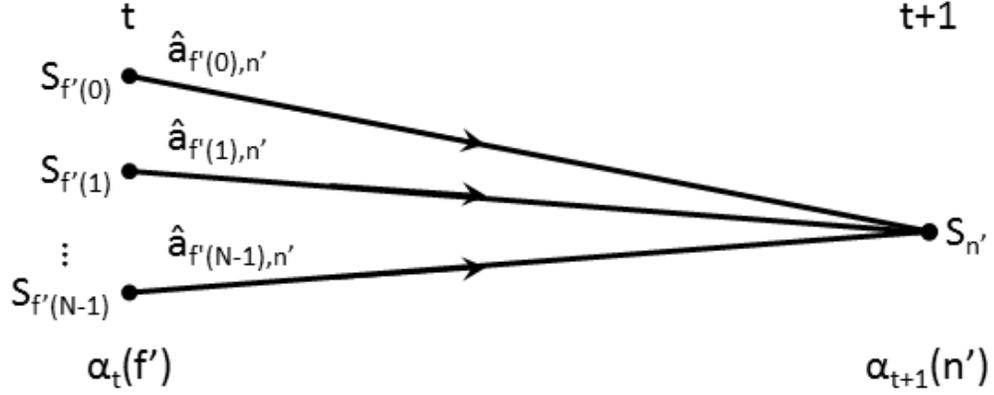


Figure 18. Single Fan of Forward α -Pass Trellis

The α -pass processing can be generalized for the list of transitions. The induction phase is given by

$$\alpha_0(f') = \pi(f')b_{f'}(\mathcal{O}_0), \quad (3.2.4)$$

where $f' \in \{\text{list_of_states}\}$. The general iterative expression for the *alpha*-pass from equation 2.5.6 can be rewritten as

$$\alpha_{t+1}(n') = b_{n'}(\mathcal{O}_{t+1}) \left[\sum_{f'=0}^{N-1} \alpha_t(f')a(f', n') \right], \quad (3.2.5)$$

for $0 \leq t \leq T - 2$ and $0 \leq n' \leq N - 1$. The summation part of the equation processes all of the edges of the (n') th trellis sub-structure and the iteration over the set of n' states computes all of the fans in the trellis to the *next* step. Stepping over t computes the α values for all steps along the state chain, using the $t = 0$ results from the induction step. Thus, the α term for each possible state at the *next* step in the chain ($\alpha_{t+1}(n')$) is given by the probability that the possible state at the *next* step in the chain gives rise to the observation at the next step in the chain ($b_{n'}(\mathcal{O}_{t+1})$) times the sum of all the α terms *from* the prior step in the chain multiplied by the probability that the chain can transition *from* state $S_{f'}$ to the possible state at the *next* step in the chain, $S_{n'}$.

With the state-duration list generated above, a much simpler description can be provided. Note from the sample code provided by Stamp [76], the α -pass is just an outer loop over all *next* states and an inner loop over all *from* states. From our state-to-state construction, the α -pass is computed as shown below.

```

 $c_0 = 0$ 
 $\forall S_{n'} \in \{\text{state-duration-elements list}\}$ 
   $\alpha_0(S_{n'}) = \pi[S_{n'}] * B[S_{n'}][\mathcal{O}_0]$ 
   $c_{0+} = \alpha_0(S_{n'})$ 
end loop over  $S_{n'}$ 
 $\forall S_{n'} \in \{\text{state-duration-elements list}\}$ 
   $\alpha_0(S_{n'}) / = c_0$ 
end loop over  $S_{n'}$ 
 $\forall t \in \text{rest}(\{\text{state-chain-steps list}\})$ 
   $c_t = 0$ 
   $\forall S_{n'} \in \{\text{state-duration-elements list}\}'$ 
     $\alpha_t(S_{n'}) = 0$ 
     $\forall S_{f'} \in \{\text{state-duration-elements list}\}$ 
       $\alpha_t[S_{n'}] + = \alpha_{t-1}[S_{f'}] * A[S_{f'}][S_{n'}]$ 
    end loop over  $S_{f'}$ 
     $\alpha_t[S_{n'}] * = B[S_{n'}][\mathcal{O}_t]$ 
     $c_{t+} = \alpha_t[S_{n'}]$ 
  end loop over  $S_{n'}$ 
   $\forall S_{n'} \in \{\text{state-duration-elements list}\}$ 
     $\alpha_t[S_{n'}] / = c_t$ 
  end loop over  $S_{n'}$ 
end loop over  $t$ 

```

$S_{f'}$ and $S_{n'}$ represent the elements in the list of state-durations, $A[\][\]$ represents the probabilities from the state-duration list, and $B[\]$ represents the state to observation probabilities. The $\{\text{list}\}'$ notation used for the *to_state* sets represents the fact that there is a limited set of possible *to_state* values based on the duration constraint. Note that while matrix notation is used for α , A and B , the

data are maintained in lists for storage efficiency. Also noted is that while the loop over the *to_states* only needs to cover the N modeled states, the results must be placed in the correct α element based on the appropriate *from_state* duration. This treatment allows for normalization as shown by Stamp [76].

3.2.6 Beta Pass

Similarly to the α -pass processing, the β pass can be rewritten in terms of the list of states with duration. Figure 19 shows a single fan of the overall β -pass trellis between state chain steps $t + 1$ and t . The same notations used in Figure 18

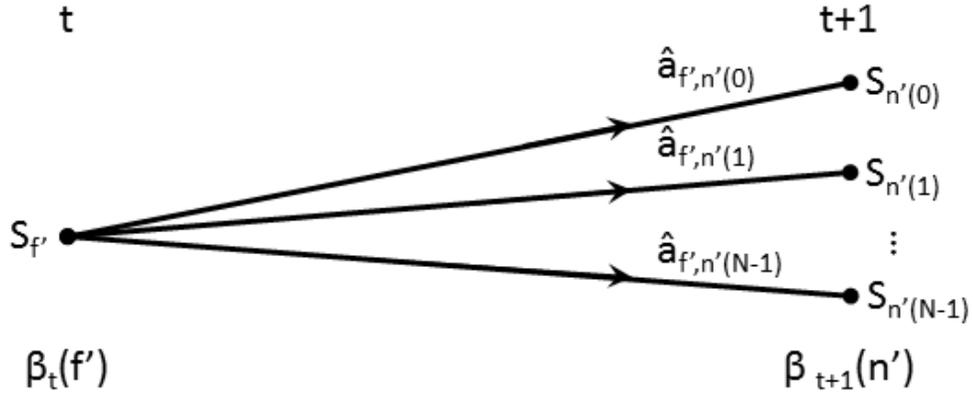


Figure 19. Single Fan of Backwards β -Pass Trellis

are reused for the states and probabilities. The same N elements in the list of possible states are used here so $\beta_t(f')$ is created in the backwards direction from all of those states and the corresponding transition probabilities. Using the state list identifiers of *from_state* :: $f' = i$, the probability of a given partial sequence of observations at step t is

$$\beta_t(f') = P(\mathcal{O}_{t+1}, \mathcal{O}_{t+2}, \dots, \mathcal{O}_{T-1} | s_t = S_{f'}, \lambda). \quad (3.2.6)$$

The β -pass processing also can be generalized for the list of transitions. The induction phase is given by

$$\beta_{T-1}(n') = 1, \quad (3.2.7)$$

where $n' \in \{list_of_states\}$. The general iterative expression for the β -pass from equation 2.5.8 can be rewritten as

$$\beta_t(f') = \sum_{n'=0}^{N-1} a(f', n') b_{n'}(\mathcal{O}_{t+1}) \beta_{t+1}(n'), \quad (3.2.8)$$

for $T - 2 \geq t \geq 0$ and $0 \leq f' \leq N - 1$. As expected from the trellis structure, this computes the partial probability that each of the possible f' states at step t gives rise to each of the possible n' states at step $t + 1$.

From our given state-to-state list construction, the β -pass is computed as shown below.

```

∀ Sn' ∈ {state-duration-elements list}
  βT-1(Sn') = 1/cT-1
end loop over Sn'
∀ t = T - 2 to 0 by - 1
  ∀ Sf' ∈ {state-duration-elements list}'
    βt(Sf') = 0
    ∀ Sn' ∈ {state-duration-elements list}
      βt[Sf']+ = A[Sf'][Sn']B[Sn'][Ot+1]βt+1[Sn']
    end loop over Sn'
    βt[Sf']/ = ct
  end loop over Sf'
end loop over t

```

3.2.7 Gamma Pass

With the above formulations for α and β , a list of allowable states, and again following the structure provided by [76] , the γ pass then becomes

```

 $\forall t = 0 \text{ to } T - 2$ 
   $denom = 0$ 
   $\forall S_{f'} \in \{\text{state-duration-elements list}\}'$ 
     $\forall S_{n'} \in \{\text{state-duration-elements list}\}$ 
       $denom+ = \alpha_t(S_{f'})A[S_{f'}][S_{n'}]B[S_{n'}][\mathcal{O}_{t+1}]\beta_{t+1}[S_{n'}]$ 
    end loop over  $S_{n'}$ 
  end loop over  $S_{f'}$ 
   $\forall S_{f'} \in \{\text{state-duration-elements list}\}'$ 
     $\gamma_t(S_{f'}) = 0$ 
     $\forall S_{n'} \in \{\text{state-duration-elements list}\}$ 
       $\gamma_t[S_{f'}][S_{n'}] = (\alpha_t(S_{f'})A[S_{f'}][S_{n'}]B[S_{n'}][\mathcal{O}_{t+1}]\beta_{t+1}[S_{n'}])/denom$ 
       $\gamma_t(S_{f'})+ = \gamma_t[S_{f'}][S_{n'}]$ 
    end loop over  $S_{n'}$ 
  end loop over  $S_{f'}$ 
end loop over  $t$ 

```

Note, however, that the loop over t only extends to $T - 2$ so there are only $T - 1$ di- γ terms. This is due to the β_{t+1} term. The effect is to shift the γ terms to the left with respect to the indexing t subscript. Thus, γ_0 represents the connection between step 0 and step 1, γ_1 connects $t = 1$ and $t = 2$, and so forth. Since the π terms are used to determine the first most probable state, the only affect of this scheme is in the indexing for the rest of the steps in the most probable state chain.

3.2.8 Most Probable State Sequence

The most probable state sequence can be deduced from the γ values computed above, the initial state probabilities, π , and the probabilities of the initial observations, $B[S_0][\mathcal{O}_0]$, Recall that each γ value is computed between steps in the chain. The α term provides the value to the left of the step. A gives the transition probability and B gives the probability that the next state gives rise to the observation. Finally, the β term provides the probability for the right side of the chain. Thus,

$\gamma_t(S_{n'})$ is really the step from t to $t + 1$ so these values give the final $T - 1$ state probabilities. The initial state probabilities and observation probabilities give the initial chain states. The processing to determine the most probable state sequence is provided below.

```

MostProb(0) = -1
∀ Sn' ∈ {state-duration-elements list}
  if(π(Sn')B[Sn'][O0] > MostProb(0))
    MostProb(0) = Sn'
  endif
end loop over Sn'
∀ t = 0 to T - 2
  MostProb(t) = -1
  ∀ Sn' ∈ {state-duration-elements list}'
    if(γt(Sn') > MostProb(t))
      MostProb(t) = Sn'
    endif
  end loop over Sn'
end loop over t

```

Note that the comparisons offer a chance for multiple equal options. The probabilities are saved to help with the merger processing and to help determine the overall confidence of a state in the chain.

3.2.9 Merging State Sequences

Given the extended HMM processing can be performed from 6 sides of the cube, but only 4 are “useable” and the cube can be rotated 4 times, the 16 resulting reconstructions must be merged in order to deduce the best fit for the reconstruction space. Given the directional nature of the generation of the most probable state sequence, the generation direction can be considered a separable process. Thus, the multiple state sequences can be merged in order to produce

a single reconstruction space. The directional reconstructions are combined as shown below.

```
foreach voxel index
  select the state representing the majority of amongst all voxels
  if a majority tie exists
    select the state representing the majority
    with the highest probability amongst all copies of that state
  end if a majority tie exists
end foreach voxel index
```

```
foreach Voxel index
  check selected state and replace if a FOHMM error exists
end foreach voxel index
```

The final step of FOHMM cleanup involves testing each voxel against its neighbors to ensure that the first order state-to-state statistics are not violated. In general, there should be few voxels that need cleanup since the extended state-to-state statistics are derived from the actual voxel density values, but the merger process can cause unallowable state-to-state transitions to be created. By checking each voxel against the allowable transitions that are derived from the underlying physical or biological model, the final reconstruction space is guaranteed to accurately represent the allowable state-to-state transitions of the original model. The processing for this final check is shown below.

```
foreach Voxel in the reconstruction space
  check selected state against all neighbors
  if a FOHMM error exists
    replace the voxel with the highest probability allowable state
    decide ties with majority of next nearest neighbors
  end if a FOHMM error exists
end foreach voxel
```

CHAPTER 4

Findings

This chapter describes the findings of the studies conducted for this research. Section 4.1 discusses the basic hidden Markov modeling efforts. Section 4.2 provides results from the use of the higher order HMM model with various noise models. Section 4.3 details the results when various geometric shapes and sizes of training and test targets are considered.

4.1 Hidden Markov Modeling

HMM processing was investigated because of the inherent match between the requirements of the HMM processing and the desired goals of the individual particle reconstruction. These models are based on an underlying state to state relationship and the nature of biological particles implies some sort of order. For the example target particle, the well-known “states” of phosphate head group {red}, lipid {green}, protein {blue}, and buffer {white} were chosen as a first set of states. The shape was deduced from the original images in [7] and is shown in Figure A.43. Assuming the sample is not imaged to the point where damage occurs, the frozen nature of the imaged samples ensures that the states do not change. Due to the structure of HMM processing, state and observation independence is needed. The construction of the problem space can guarantee the necessary constraints while the efficient nature of HMM processing leads to a computationally tractable solution for reconstructing individual particles.

This section discusses the processing with a FOHMM and effects of state chain direction as motivation for the ensuing HOHMM results. This is followed by results of basic extended HOHMM processing and includes the effects of merging the directional results.

4.1.1 First Order HMM

FOHMM results can be skewed by the direction in which the statistics are gleaned and processed. For example, a reconstruction in the vertical direction from the basic model provided in Figure 10 leads to an extended reconstruction result. The reconstruction from a vertical direction looks like a tall cylinder as shown in Figure 20. This is simply a result of the model statistics and the reuse

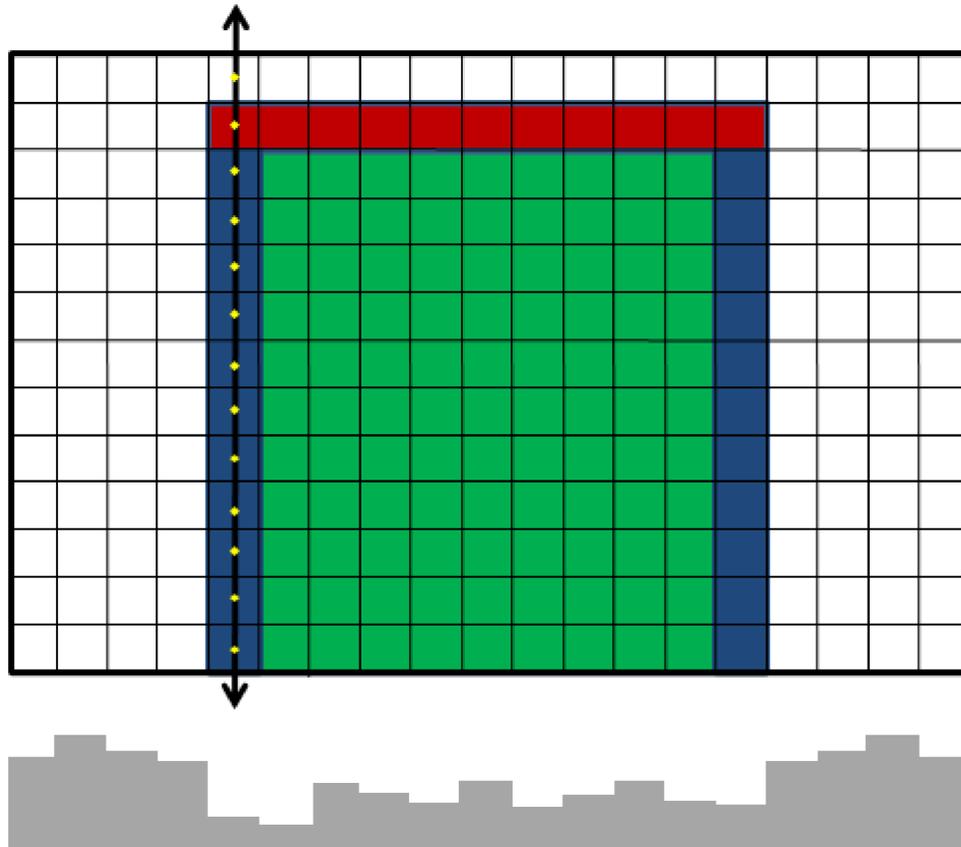


Figure 20. First Order Vertical HMM Reconstruction

of observations. For the vertical stack (black line with yellow dots representing the state locations), the initial state is destined to be buffer (white) due to the π statistic. Immediately, however, the chain wants to change to red since the A

statistics only allow a white-to-red transition and the B statistics desire a target {red, green, or blue} voxel for the lower gray level observations associated with the voxel state subtending the target region of the reconstruction space. The chain then selects blue since there are more blue than red voxels in the model upon which the statistics were generated, so for the given observation, blue is more probable. As there is no driving force to change states, the stack stays in the blue state until the end of the chain. Likewise, the middle stacks stay in the green state since this is most probable for the given observation. The reconstruction from a horizontal direction looks like a tall cylinder as shown in Figure 21. The more probable blue

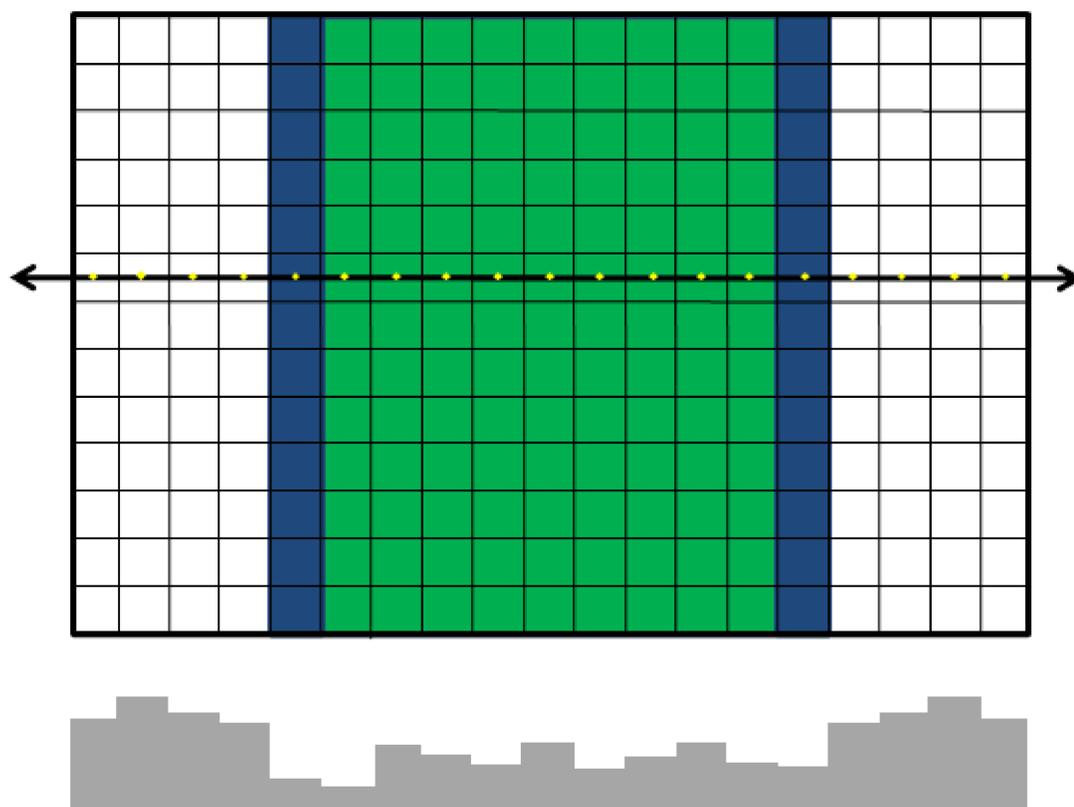


Figure 21. First Order Horizontal HMM Reconstruction

state is selected (over the less probable red) as soon as the observations reach the target levels. The allowed blue-to-green transition happens when the first green observation is reached. Likewise, the green-to-blue transition happens at the edge of the target when the blue observation is reached.

Several tests of the FOHMM were performed; however, none of these methods worked well. All gave reconstruction results for the target particle that spanned the entire vertical space of the reconstruction area. Figure 22a shows the results (sliced open to reveal inner densities) when an angle of 0° was used as the rotation amount. This is as expected since the probabilities only have the flat projection

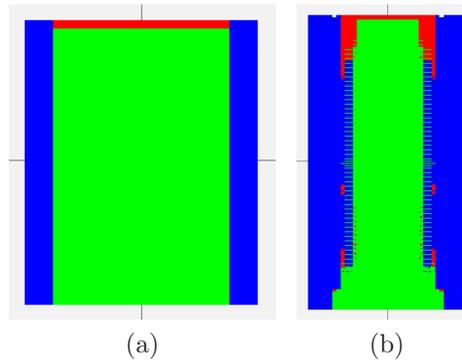


Figure 22. Simple HMM reconstruction results when viewed at a) 0° and b) 45° .

image (Figure A.43c) from which observations can be gleaned. In regions where the observations were from all buffer (white) regions, the HMM processing gave rise to the buffer as the only possible solution since the probabilities of the white pixel observations were only possible when all the voxels from which the pixel was derived were all buffer density. Likewise, the blue outer ring and the middle red and green voxels mostly gave rise to single states. The resulting shape was a cylinder for all voxels except the top slice of the voxel space, which was forced to be the buffer state by the π statistics. The cylinder itself was a blue edge region and a green center area except for the top layer which was red as dictated by the white-to-red statistics (the white to green transition was prohibited by the

A-matrix statistics). The expected red bottom layer was not enforced since there was nothing in the model to cause the chains to revert back to buffer (white), and thus through the red state to prevent green touching white. The cylinder simply extends to the bottom of the reconstruction space. When a non-zero rotation was applied, the cylinder was smeared but with the same basic results, as is shown in Figure 22b.

While the FOHMM processing results were as expected, the results are not what is desired in that the basic shape of the particle is not reproduced. However, there are a few things to note about the results of this experiment. First, the 0° case and the 45° case gave basically the same results. The case with no rotation was not expected to give sensible results since the observations were *not* independent. However, the base state-to-observation statistics ignored the lack of observational independence; all states in those columns expected the same observation so they were by default not independent. Second, the directionality of the HMM processing did matter. The reconstruction result from the x and z directions did not include the red top layer as expected but still resulted in a cylinder that spanned the non-buffer-only pixels in the image. Since the Markov chain was horizontal (in the $x - z$ plane) the π constraint of first voxels being buffer were met but the observation probability drove the voxels to stay white until the darker bands of the target were reached. Similarly, the middle of the cylinder was all green, again as expected by the fact that there were more green than red states for the middle gray region observations so the green state was most probable. The directionality in y did cause the red top layer to alternate between the top and the bottom as expected.

From these results, it can be concluded that basic first order Markov sequences are not useful for this 3-D reconstruction. The single pass with merged statistics

approach does not seem to make sense as a processing schema because while the directionality of the most probable state sequence does provide additional information, the first order Markov chain is overly biased by the state-to-projection statistics while seemingly ignoring the observational dependencies as well as the state-to-state probabilities. The greedy statistical nature of a FOHMM means that, in general, the voxel state that fills the majority of voxels in a stack will most likely dominate for a given observation even if the projection image is at an angle to the voxel stack. As can be seen in Figure 5, a current model of the LDL particle consists of several layers of alternating fat and cholesterol encased in the phospholipid head group shell. Which ever density prevails in a voxel stack will dominate the simple first order statistics processing. This results in a reconstruction that cannot determine the height of the particle. The reconstruction is simply a tower (in this case a cylinder) where the top slice is *red* then the remaining slices of the middle region are *green* since the *green* statistics dominate. It is this internal structure, however, that should help make use of HOHMM processing feasible.

4.1.2 Simple Extended HOHMM Processing

A test of extended processing shown in Section 3.1.2 was performed that reconstructed the example target (with no noise) exactly when viewed directly from the top (down the Y axis). Figure 23 shows the original model (rotated at 45° and zoomed in to show the internal structure) in the left panel, the simulated projection image (not rotated) in the center panel, and the reconstructed particle (also rotated and zoomed) in the right panel.

The HMM statistics for matrix A modeled the fact that the white buffer region (see figure A.43a) was of length T , T_{top} , or T_{bottom} states. These statistics also captured the effect that the sequence of unknown states followed by a buffer (*white*) region of length T_{top} gave rise to a transition to the *red* state. Likewise, a

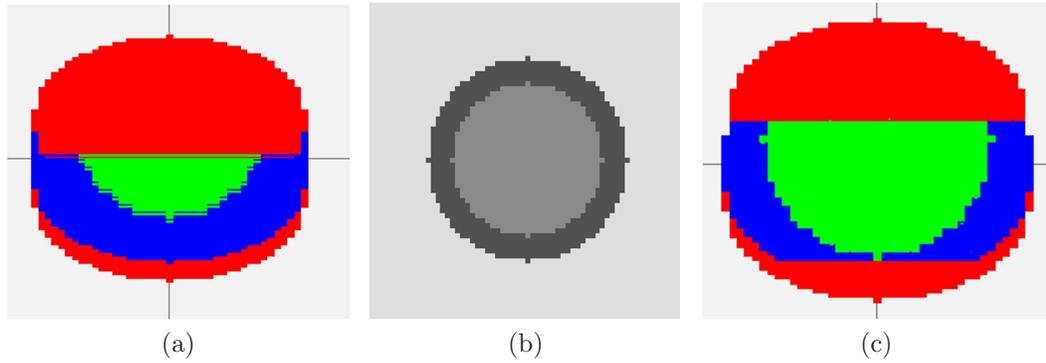


Figure 23. a) Original discooid model, rotated to 45° and sliced to show internal structure, b) Resulting projection image taken along the Y axis, and c) HMM reconstruction results, also rotated to 45° and sliced.

sequence of *white* to *red* with appropriate lengths predicted a transition to either *blue* or *green*, where the selected state depended upon the associated observation value. Similarly the *blue* or *green* regions transitioned back to *red* then *white* based on the modeled sequence lengths.

This simple processing, however, only works well for the vertical case because only the state-to-state statistics are used, not the complete statistical basis. With the flat top of the model, *all* of the transitions were at the same length paradigm. When processing from an angled approach was tried, the results did not work nearly as well since the statistics varied for the chains hitting the rounded edges of the target. Figure 24 shows the original model rotated to 45° with the same perspectives as shown in Figure 23 above. As can be seen, while the basic shape is recovered, the contents are not correct since the statistics did not include the angled approach details. The fuzzy regions to the top and bottom are all fused to a single state in the state to observation statistics so the reconstructed object is smeared accordingly. To account for this, the π and B statistics must also utilize extended state information and the HMM processing in section 2.5.2 must function so the α , β , and γ passes include the new information.

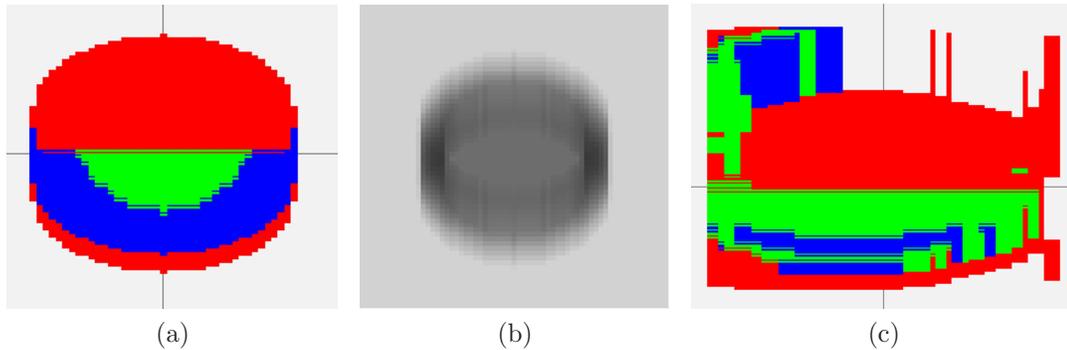


Figure 24. a) Original Discoid model sliced to show internal structure, b) Resulting projection image taken along the Y axis, and c) HMM reconstruction results.

4.1.3 Extended HOHMM Processing

The complete extended HOHMM processing detailed in Section 3.2 provided extremely good reconstruction results. A test of this processing was performed that reconstructed the example target exactly. In this example, the simple target shown in figure A.42 was changed so that the entire top and bottom sections were *red* and the *blue* protein band was sandwiched between the *red* top and bottom layers. Figure 25 shows the original model (rotated at 45° and zoomed in to show the internal structure) in the left panel, the simulated projection image (not rotated) in the center panel, and the reconstructed particle (also rotated and zoomed) in the right panel. The HMM statistics for matrix A modeled the fact that the *white* buffer region was of length T , T_{top} , or T_{bottom} states. These statistics also captured the effect that the sequence of unknown of length 1 followed by a *white* buffer region of length T_{top} gave rise to a transition to the *red* state. Likewise, a sequence of *white* to *red* with appropriate lengths predicted a transition to either *blue* or *green*, where the selected state depended upon the associated observation value. Similarly the *blue* or *green* regions transitioned back to *red* then *white* based on the modeled sequence lengths.

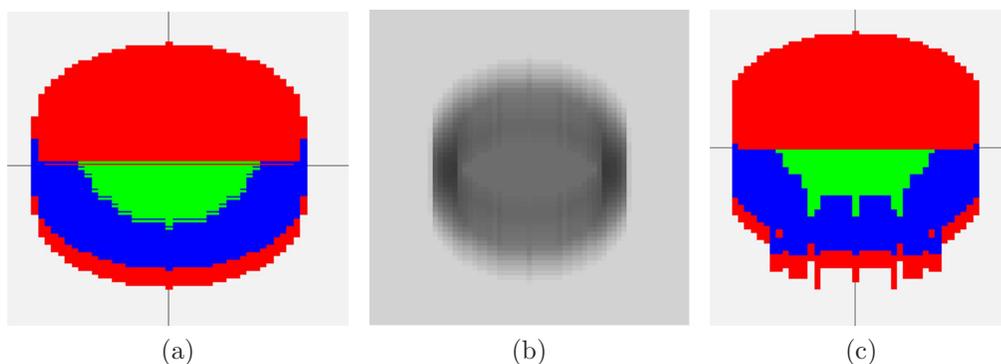


Figure 25. a) Original Discoid model, rotated to 45° and sliced to show internal structure, b) Resulting projection image taken at -45° along the X axis, and c) HMM reconstruction results, also sliced open to show internal structure.

This test shows that the basic processing is functioning but it does not provide any significant insights into the overall reconstruction issue for arbitrary angles. This single-pass reconstruction relies on the fact that the target was square to the reconstruction voxel space. The statistics match exactly “from the top” and there is no ambiguity of observations and states.

4.1.4 Sequence Direction

The above discussion makes it clear that the direction of the most probable state sequence generation plays a role in the model results. While the Viterbi algorithm [80, 81] draws its efficiencies from computing in both the forward and backwards directions, ultimately, the resulting state sequence is directional in nature. Since the model of the target of interest may have features that reflect this directionality, an understanding of the directional nature is needed.

One example of a directional constraint can be seen from the simple model shown in Figure A.42. In this example model, a simple model of phospholipid head groups, fat, and protein (red, green, and blue respectively) for the LDL particle are used to represent the major electron density regions of the macro-assembly. Notice that if the statistics are generated from the top (or bottom) then the probability

of a transition from the surrounding buffer region (white) to either the red or blue regions is possible. Of course the hydrophobic fat (green) is not allowed to touch the water-based buffer (white) so there is zero probability that white voxels can transition to green or vice-versus. But, if the statistics are generated from any of the four sides of the voxel space then a white-to-red and red-to-white transitions are also forbidden.

Another directional effect arises from the nature of the higher order HMM model that is used. Additional reconstruction results were obtained by performing the HMM processing in four orthogonal directions as depicted in Figure 14. In effect, the reconstruction space was rotated around the z axis. The reconstruction voxel space was constructed at $\pm 45^\circ$ to the orientation of the modeled target in order to produce independent observations for each step in the chains (yellow boxes). The directions were chosen so as to keep the reconstruction space aligned to the rows and columns of the projection images. This artificial constraint was chosen to help with analysis; a rotation around the y axis could have also been performed. The same HMM model parameters, which were deduced from the $+y$ to $-y$ direction for all 4 images, were used for the 4 reconstruction passes. Thus, the HMM state chains followed the directions of the black arrows.

The results of these 4 reconstructions are shown in Figure 26. As is shown, the basic particle shape is also reconstructed but with a “tail” in the trailing direction of the HMM chain. There is no “tail” on the leading edge since the statistics of the “white space” say that there can be no particle states until at least the minimum distance from the edge has been traversed by the state chain. In addition, there are many cases where longer stretches of white buffer are traversed before hitting the target so these drive the leading edge to stay as buffer for as long as possible. On the trailing edge, however, the state to state probabilities *can* result in particle

states with longer durations, up to the maximum supported length across the target, thus the tails appear when the observations support them. As with the

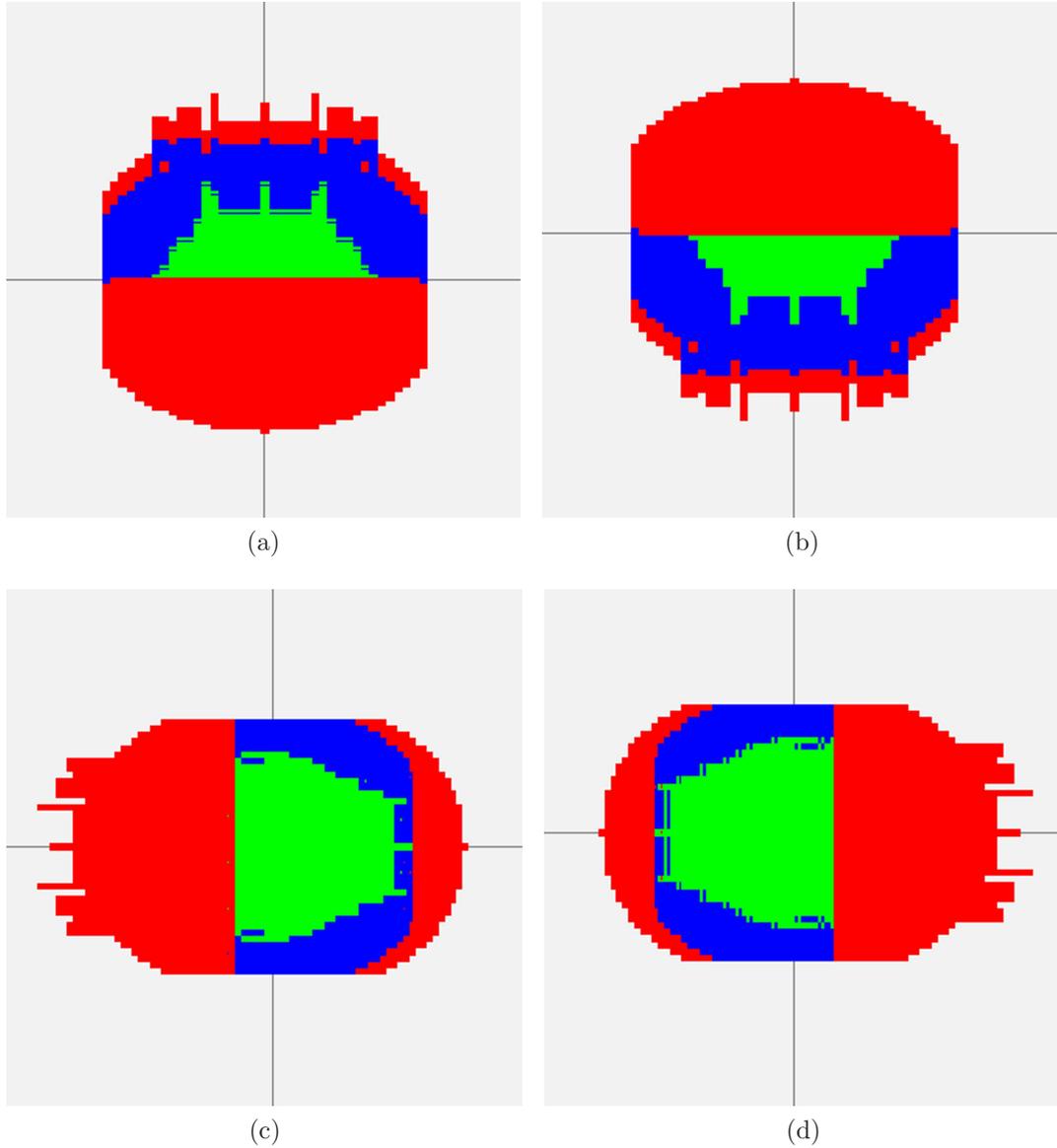


Figure 26. Reconstruction results for the four rotated cases of a) $X +45^\circ$, b) $X -45^\circ$, c) $Z +45^\circ$, and d) $Z -45^\circ$ as viewed from the $+y$ axis.

external tails, the internals are also subject to distortion since the view is at an angle to the direction of the state-to-state probabilities. These can be seen where the reconstructions have been sliced open to reveal the internal structure.

The detailed voxel to voxel comparison is provided in Table 3. As is shown, the reconstruction space matches the original space to 98.76%. This calculation utilizes

Original / Recon	Blue	Green	Red	White	By Color
Blue	7770	420	0	30	94.53%
Green	150	10365	0	0	98.57%
Red	0	0	12470	20	99.84%
White	1665	0	1110	240625	98.86%

Table 3. 45° Reconstruction Voxel Accuracy

the entire voxel reconstruction space so it may appear that the overall accuracy is skewed by the abundance of buffer (white) voxels. Computing the reconstruction accuracy by color results in nearly similar results except for the blue voxels. For simplicity, the overall metric will be used to compare reconstruction accuracy. But an eye will be kept open for distortions since this may indicate useful information regarding the overall processing.

The directional HMM processing did not produce any useful results at the 0° case. As expected, the four reconstructions from the x and z directions resulted in the same answer while the y directions provided the same basic answer as did the x and z cases except with the red top and bottom regions as before. Several combination methods were tried but in general, the x and z reconstructions won since a majority was biased by the 4 vs. 1 vs. 1 voting. Given the symmetric sample target, and the overriding observations, this was expected.

4.1.5 State Sequence Merging

Since the directional reconstructions share the same aligned voxel space, the separate results can be easily combined without the need for state interpolation or other schemes. Given the orthogonal method by which the reconstructions were generated, a simple majority scheme can be employed to deduce the final reconstruction. Since, for the most part, the tails extend along the trailing edge,

each voxel is defined by the images that are not being affected by the trailing effect. Figure 27 shows an example projection image with noise. Figure 28 shows

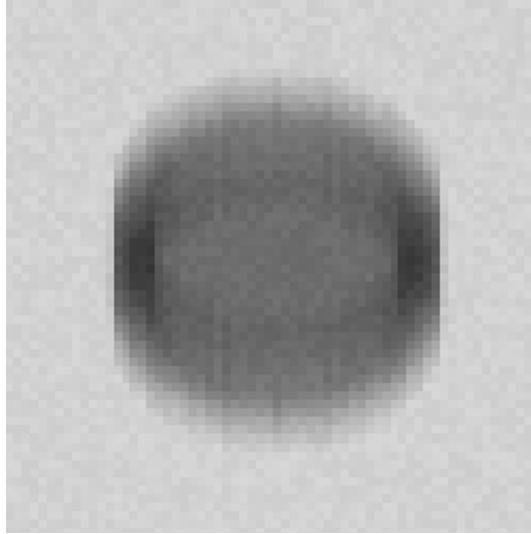


Figure 27. Projection Image at 45° with Noise Added .

the case of 4 directional reconstructions merged with a simple majority scheme. This reconstruction also matches the original model better than 98% but the tail

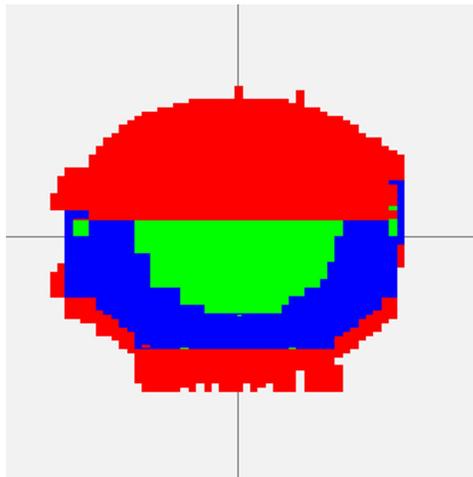


Figure 28. Merged reconstruction results shown at 45° and sliced open to show the internals.

artifacts have been reduced.

4.2 Noise

This section describes the effect of noise on the HMM processing. The results range from processing with no noise in the observation image to multiple runs to build up the effects of the noise on the state-to-observation statistics. A final result describes the results of using a model of the noise during training and compares this to the other cases.

4.2.1 No Noise

Runs with no noise consisted of training the HMM statistics against images with no noise added in. That is, a single simulated image without noise was generated for use as both the training image and the reconstruction image. As was shown above in Figure 23, the processing works very well but with the tails. Good results are expected because the training image was also used as the processing image. Therefore, there were no variations between the training state-to-observation statistics and those from the processing image.

4.2.2 Single Training Run

Single training runs with noise consisted of training the HMM statistics against images with noise added in. That is, a single simulated image with noise was generated for use as both the training image and the reconstruction image. The addition of noise simply changed the gray-level values of the statistics and not the distribution since the same image was used for both training and processing. As expected, the processing was good due to the lack of variation between the training and processing statistics. As with the no noise case, the use of a single image with noise added has very good results of 98% as discussed in Section 4.1.5.

4.2.3 Multiple Training Runs

Performing several training runs to build up the state-to-observation histograms actually performs very poorly, unless a significant number of runs are taken. This is seen from the prior discussion in Section 3.1.3. The state-to-observations statistics are simply a reflection of the noise statistics applied to the δ -functions shown in Figure 11. If a single training run is used to set these statistics, then there is an almost 0% chance that the processing run will match the statistics. This means that the B term will be zero so no transition will be selected. As more and more training runs, are added in, the results get better but there still may be holes in the reconstruction. Thus, for images with larger sigma values, more and more training runs must be performed to more fully represent the true state-to-observation statistics.

4.2.4 Modeled Training

While the very good results without the use of reducing noise by averaging images may seem counter-intuitive, a look at the state-to-observation statistics gives insight into why HMM processing works. Figure 29 shows the state-to-observation probabilities. The left side shows the base statistics for no rotation of the voxel space. The peak at the right is for the pure buffer case; all buffer-only observations are at high levels in the gray level space. The lower two peaks are for the combined projection pixels. Thus, only a handful of observation values are allowed. The spread denotes the noise level added. The right side of the figure shows the effects of rotating the voxel space. The buffer region stays clearly separated but the two peaks for the particle voxels start to merge, which follows the smearing shown in Figure 27. The extended HMM processing, however, uses *both* the state-to-state and state-to-observation statistics so the effect of noise is minimized. As long as the contrast level of the image is sufficient to prevent

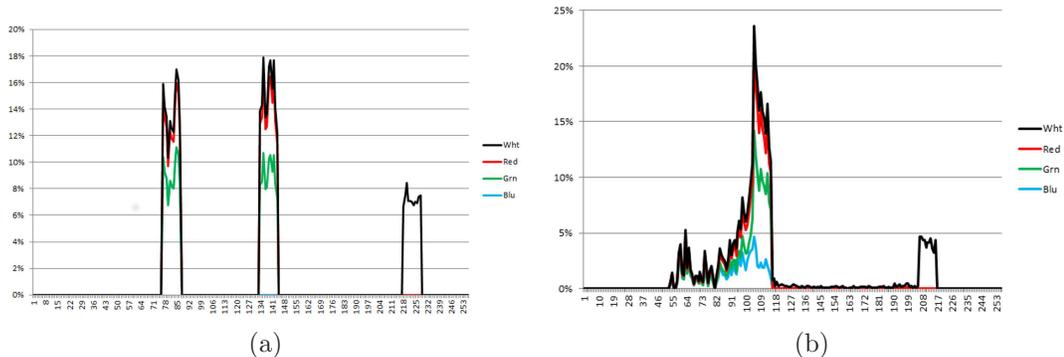


Figure 29. State (color) to observation probabilities for a) aligned voxel space and b) rotated voxel space.

destructive overlap of the observations, the state-to-state portion of the HMM processing reasonably is able to determine the correct next state in each sequence.

The above discussion leads to the understanding that if there was no noise, the allowable regions would be pseudo- δ functions at the resulting projection gray levels. This is shown in Figure 30. This data shows an occurrence histogram of 131 state tuples for each of the 255 observation symbols. There is a preponderance of occurrences near gray level value 100 and a majority of values near 200. The spikes near gray level 100 represent the smeared values of the target; the larger values are for the buffer-only regions where there is no interfering target. When noise is

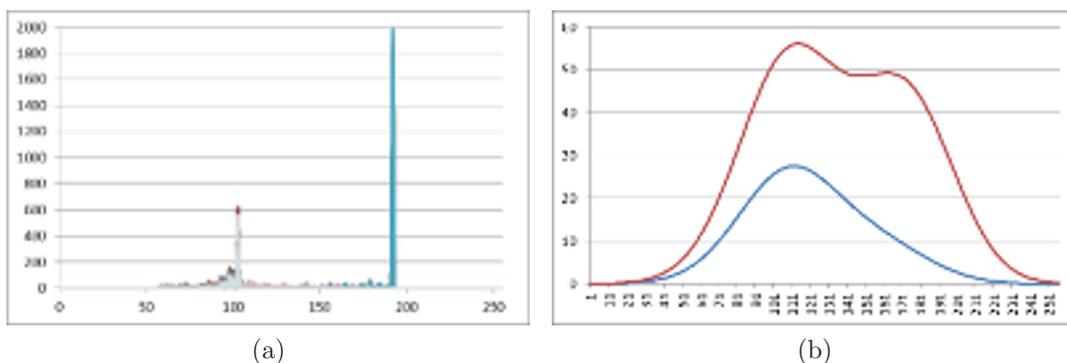


Figure 30. Probability of a) 131 state-tuples with no noise and b) Sum of Gaussian distributions for all target observations (blue) and sum of all observations (brown), including the majority of buffer only observations near gray-level 200.

included in the training image each histogram entry will be shifted by the noise at that observation location. If a sufficiently large number of images is used to train the HMM, then it should be obvious that the pseudo- δ functions will grow into representations of the noise statistics. With proper normalization, the histogram will follow the underlying noise statistics, for example as shown in Figure A.44b. Figure 30b shows the distributions for two cases for a single state tuple. The first (blue) graph shows the sum of Gaussian distributions ($\sigma = 25$) centered at each pseudo- δ function location near 100 gray level which include the target region. The second (brown) graph shows the sum of Gaussian distributions for all pseudo- δ function locations, including the majority of buffer-only tuples. This brown graph dominates the ultimate probability distribution, as is expected, since most of the observations are for buffer-only. Figure 31 shows the row-stochastic probability distributions for all 131 state-tuples. The single peak graphs (lighter gray centered

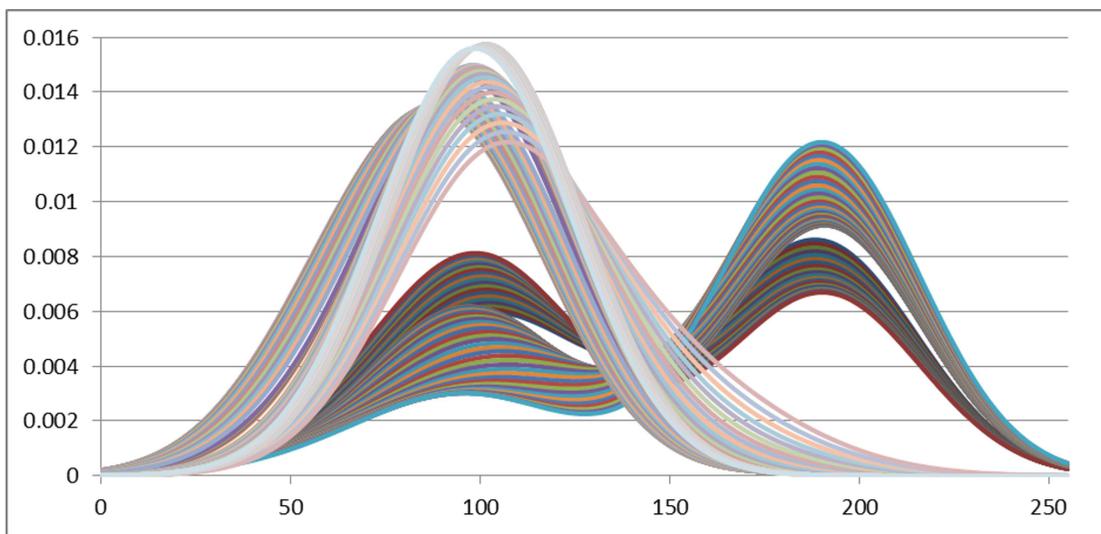


Figure 31. Simulated HMM Observation Probabilities

near 100) represent those tuples that are entirely involved in the target area. The double peak graphs (darker colors) show various state-tuples that do include the effects of the large number of buffer-only tuples.

Three features regarding image acquisition and preparation can be gleaned from this understanding of the state-tuple to observation probability distributions. First, the noise variance can impact the smearing of the probabilities. Care should be taken during the sample preparation and imaging to help minimize this variance. Second, the ratio of target and non-target observations can affect the probabilities. As is seen in Figure 30a, the preponderance of buffer-only pixels drives the state to observation distributions, and thus, the overall probability separations. The extremely high ratio of buffer to target observations causes the simulated probabilities to spread since the height of the central pseudo- δ function drives the total energy under the Gaussian distribution. Finally, the overall contrast level of the image has an effect on the separability of the state-tuple to observation probabilities.

As seen in Figure 30a reductions in the confluence of these factors causes ambiguity of the probabilities for many state-tuples. Obviously the variance could be reduced by a factor of $1/\sqrt{N}$ if N images were averaged, but this defeats the purpose of using a single image from the *cryo-tem* system. Thus, increasing the overall contrast ratio of the image and cropping the image to increase the ratio of target to non-target observations can serve to enhance the state-to-observation probability resolution. An example of the same target but with an increased ratio of target to buffer-only observations and contrast ratio is provided in Figure 32. As is seen, there is much better separation between the two distributions.

Since the training images are deduced from simulated images, there is no need to generate multiple images to accumulate the state tuple to observation probabilities. This approach has an obvious benefit of reduced setup time since the observations are only based on the target model and the observed or expected noise statistics rather than having to perform multiple runs in order to determine

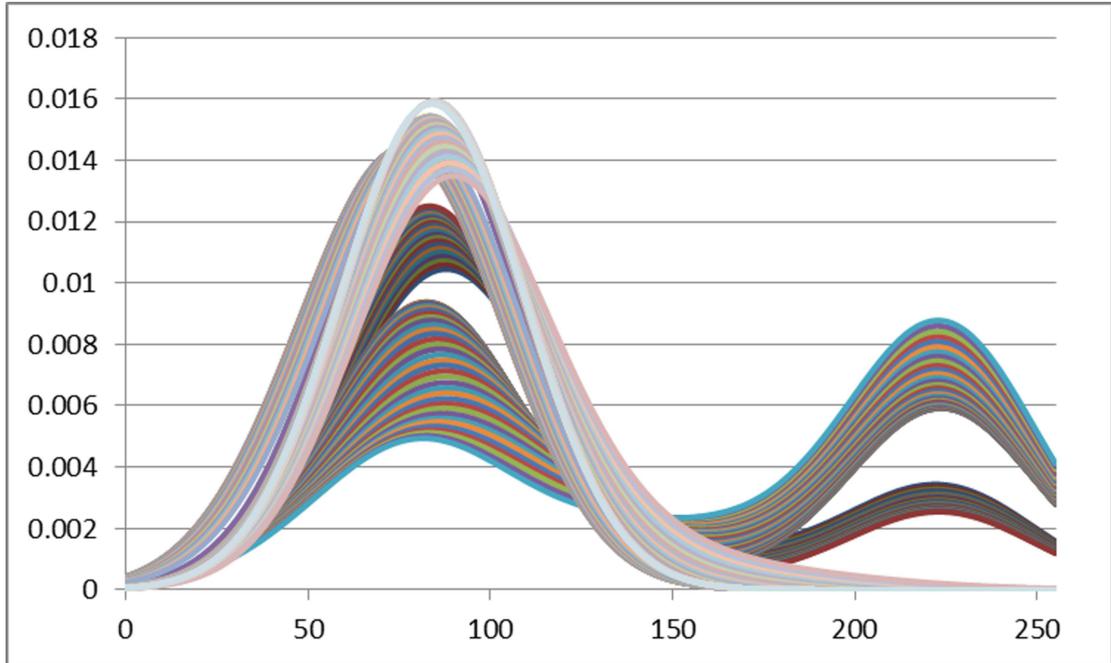


Figure 32. Enhanced Simulated HMM Observation Probabilities

the state-to-observation statistics.

4.3 Geometry Parameters

This section describes results obtained by varying the geometric parameters of the target. This testing altered geometric parameters of the actual target versus the training parameters.

4.3.1 Size Changes

This section describes the effects of changing the size of the modeled LDL tri-layer discoid. To perform this testing, the processing was trained with a given set of size parameters. Modeled noise (as shown in Section 4.2.4) was used to reduce the effect of noise on the understanding of the effect of size changes.

A test of size changes was performed by deducing the HMM model at a fixed radius and then processing simulated targets over a range of radius values. Figure 33 shows the voxel to voxel match percentage for training and processing radii

ranging from 15 to 28 voxels. As can be seen, the basic match percentage is around 90% regardless of the radius mismatch.

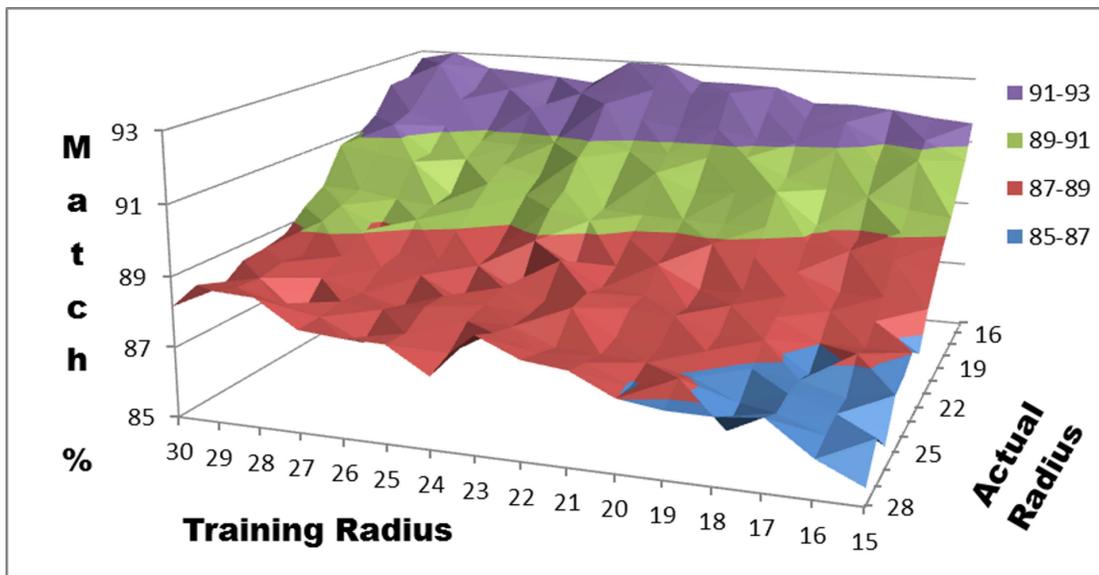


Figure 33. Different Training vs. Processing Radii Matching

The higher match percentage with decreasing actual radius is a result of having a fixed voxel space. As the actual target gets smaller, more and more voxels are correctly decided to be the “white” buffer since there is no target in those voxel stacks. For example, when the target radius is 30 voxels, there were 185,785 “buffer” voxels but when the radius drops to 15, there are 238,700 “buffer” voxels. If we assume that all errors are in the target voxels, the 10% error rate means there are approximately 8,884 and 3,392 incorrect pixels. With $65^3 = 274,625$ voxels in all, this leaves 88,840 and 33,925 target voxels respectively. Since these are almost always correctly identified due to the obviously brighter observations, the error drops accordingly.

The general trend is that for a given actual target radius, the matching error remains fairly constant regardless of the training radius. The radius, of course, affects the internal structure. Since the outer blue “protein” band was held con-

stant, the inner “fat” region becomes smaller with a decrease in radius. Thus, the extended HMM reconstruction shows very good immunity to training particle size versus actual particle size.

4.3.2 Density Changes

Since the reconstruction is in state space and not actual density space, the effects of slight differences in actual voxel densities are not pronounced. Changes in density of the target simply slide the peak of the observation histograms shown in Figure 31. As long as the distributions do not overlap too badly, and the means do not shift too dramatically, the IPR algorithm can still reconstruct the targets.

4.3.3 Shape Changes

A final geometry issue that can affect the IPR is the overall shape. In general, model-based reconstruction techniques have separate models for different shapes. For example, squares are not used to represent circles. Even the RBC method decomposes the shapes into geons for the sub-parts. The superquadric method can morph the single equation into various shapes, but boxes and rounded shapes have fundamental variations of the superquadric parameters based on the shape. Thus, understanding how the IPR handles shapes that are different than the model helps compare it to other methods.

Understanding how the IPR method handles shapes can be found by attempting to reconstruct an extremely distorted target. Such a shape is shown in Figure 34. This is the standard target but with one side “bent” down at a 45° angle. When viewed from the $+X$ axis, towards the $-X$ axis, the target is seen as bending down along the $+Z$ axis. A view of the target rotated 45° around the Z axis is shown in Figure 35a.

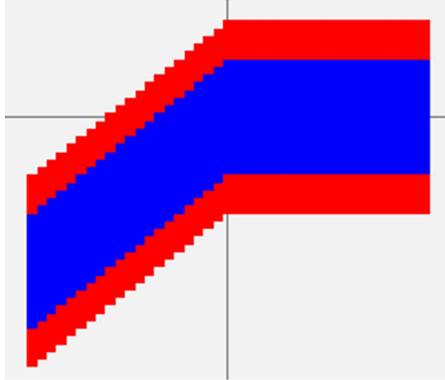


Figure 34. Warped Disc Model

Training images for each of the six directions with no noise added were used so that the simulated statistics could be generated for training. The HMM model parameters are built from the flat discoid that lies along the $X - Z$ plane. In addition, the HMM parameters were built in voxel stacks that are either parallel or orthogonal to the edges of the discoid target. The training image is shown in Figure 35b and an example projection image is shown in Figure 35c.

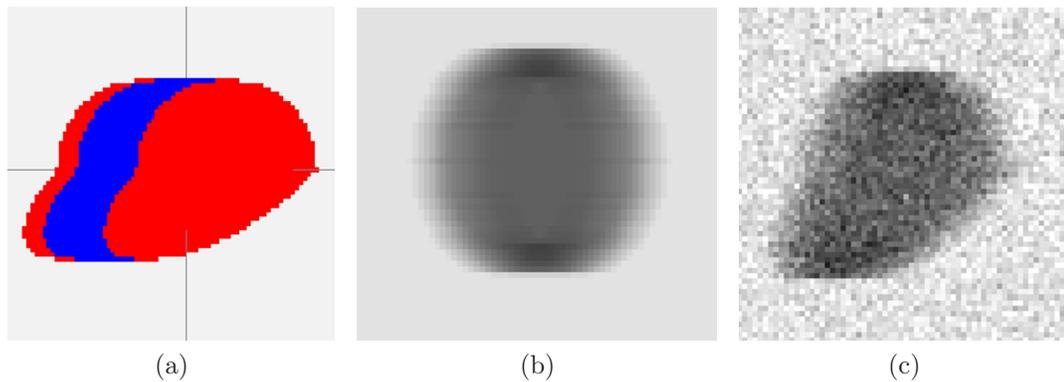


Figure 35. a) Modeled warped target, rotated 45° , b) Training image with no noise, and c) Processing image with noise.

The reconstructions with this setup provides insights into the capabilities of the IPR to find targets of shapes that differ from the model target. The approach for geons and superquadrics allows for decomposition of objects with “detectable” surfaces to be decomposed into smaller geometric ions. Identification of deformed

geons can be handled with extra processing to handle additional minimization constraints. Likewise, it appears that the IPR will need additional states (and thus more processing time) to capture variations of modeled shapes. Further research into constraining this problem to keep it out of the NP-complete regime are needed.

CHAPTER 5

Conclusion

This chapter provides a concluding discussion about the technology presented in this dissertation, directions for continuing this project, and its impact. Besides this dissertation, some of the results of this research have been published in [94, 95, 97]. Additional information regarding this research may be found in technical reports [93, 75].

5.1 Technology Remarks

This dissertation introduces a new approach to the reconstruction of transparent objects. The basis imaging modality is *cryo-tem*. Existing SPR methods of reconstructing biological nanoparticles from *cryo-tem* images require significant numbers of images, averaged together to reduce noise, and significant computational processing to produce a reconstruction that represents the ensemble average particle. The approach presented in this dissertation requires only a single image and a reasonable computational engine to provide a true individual particle reconstruction (IPR). While the SPR methods produce a “best fit” of the voxels from a mathematical perspective, the IPR uses an a priori model from biological and/or physical constraints to deduce the voxel to voxel statistics, and thus, to determine the most probable voxel states. A further promise of this technique is that the internal structure contains information that can lead to a better understanding of the model. The IPR approach offers all of these enhancements with a much simpler approach to solving the transparent object reconstruction problem than required by existing SPR methods.

The IPR method presented in this dissertation provides its effectiveness due to its innovative construction. The presented approach offers a simple reconstruction

method that requires significantly less computational processing than the current state-of-practice methods. By utilizing the HMM problem-two paradigm, a non-iterative solution is found in $O(N^2T)$ deterministic steps instead of the iterative approach of SPR. The use of a rotated reconstruction space allows for 16 different HMM solutions to be found from a single image in a co-aligned reconstruction space. Selection based on majority and most probable states allows merging of the HMM solutions; the first order state-to-state statistics are used to decide ties. The inherent state-to-state and state-to-observation statistics and the rotated reconstruction space also allow for scalability based on the desired reconstruction resolution with only a linear processing cost. The IPR processing shown requires on the order of a minute of processing on a modest laptop computer versus the hours of computations with heavy duty workstations for existing SPR methods. The construction also allows for massive parallelization where each chain, or small groups of chains, can be computed concurrently.

The method described in this dissertation can provide reconstruction results that match the original target better than 98% on a voxel to voxel basis. The IPR shown here also has an ability to work regardless of the noise found in non-averaged images as long as the noise statistics are known. With care in setting the contrast range of the images, it is expected that even better performance can be achieved in even more general use cases. The merging process helps increase the usability of the solutions. Eliminating errors arising from the HMM directional processing allows the core target structure to be reconstructed more accurately. Taken together, these features of the IPR approach provide for a very robust methodology for reconstructing transparent objects imaged via *cryo-tem*.

5.2 Future Efforts

Despite the encouraging results shown in this research so far, there are additional areas in which this technology can be studied for further advancements. These areas, however, are mainly directed towards specific targets. The basic parameters of the IPR method are defined by the process so understanding how these parameters are affected by specific targets would be the goal of these new studies. For example, one model of LDL particles shows multiple internal layers. Should this model be found to be biologically correct, then additional levels of extension of the HMM state levels may be needed. The algorithm as presented easily accepts more levels than the two presently used. More spherical objects are expected to follow the same regime as the sides of discoids but other shapes may require additional study. Since a physical model is needed, simulations can be run for any new shape that needs to be processed before actual data is studied.

The resolution of differences between the actual and modeled shape are also an area ripe for future study. While the IPR in its current representation can provide useful information for “stable” shapes, the ability to recognize outliers in the shape schema would certainly be useful in studies such as LDL; finding rogue individual LDL particles would certainly be useful. The data provided in the internal statistical results is expected to play an important role in this area. Having assurance levels based on the statistical model can allow for a better understanding of where the present model of a particle has higher and lower confidence metrics.

Once actual *cryo-tem* data is available for a well characterized object, one area that could offer more insights is to use the real image to try solving the HMM problem-three. This approach is an iterative solution and as such, a solution is not assured, but it can possibly offer insights into how to set up the HMM states and levels.

5.3 Impact

The IPR approach presented in this dissertation has a significant impact from a technology perspective. The presented method has an obvious significance as a new algorithm for solving the transparent object reconstruction problem. The results from this study, however, also have an impact on the image acquisition parameters. Maximizing the contrast ratio of an image is generally important, but this research shows how widening the gray level range of particles and the surrounding non-target region can have a positive impact on the IPR processing. The desire to minimize the variation in IPR images leads to the need to understand how the *cryo-tem* electron voltage affects pixel to pixel deviation. Resolving power increases with electron voltage so studies will be needed to see how various buffer solutions and other image acquisition parameters affect sigma. While defocus is one of the important factors in SPR processing, the IPR images may need a different set of concerns to be addressed.

There are many classes of problems that need individual particle reconstruction; the IPR offers a way to advance the state of the art in these areas. Example particles based on LDL were used as a basis problem but drug quality assurance, general proteomic studies, and other *cryo-tem* imaging of transparent particles can use IPR. Any problem where a histogram of geometry and density parameters are needed can use the IPR. The approach also supports problems where damage results from excessive exposure to the imaging beam. One possible use would be for analysis of dental x-rays. The general structure of teeth is known so a single x-ray could possibly be processed to look for cavities, root issues, or other defects. General X-ray inspection for manufacturing is also a candidate use for the IPR technology. Here the overall tact time, and thus minimizing the number of images, is probably of more importance than exposure damage. As there had been no

similar methodology until now, there are many other opportunities for use of the IPR to solve reconstruction problems for transparent objects.

LIST OF REFERENCES

- [1] University of Rhode Island. “A Guide to Producing Your Thesis with LaTeX.” Last visited on 28 Dec 2013. June 2006. [Online]. Available: <http://www.ele.uri.edu/info/thesis/guide>
- [2] R. Henderson, “The Potential and Limitations of Neutrons, Electrons, and X-rays for Atomic Resolution Microscopy of Unstained Biological Macromolecules,” in *Quarterly Review of Biophysics*, May 1995, pp. 171–193.
- [3] Bridget Carragher and Nick Kisseberth and David Kriegman and Ronald A. Milligan and Clinton S. Potter and James Pulokas and Amy Reilein, “Legion: An Automated System for Acquisition of Images from Vitreous Ice Specimens,” in *Journal of Structural Biology*, Oct. 2000, pp. 33–45.
- [4] Douglas Trumbull, “Immersive Cinema Technology,” *SID Symposium Awards Luncheon*, 2011.
- [5] Gang Ren and Gabby Rudenko and Steven J. Ludtke and Johann Deisenhofer and Wah Chiu and Henery J. Pownall, “Model of Human Low-Density Lipoprotein and Bound Receptor Based on CryoEM,” in *PNAS*, Jan. 2010, pp. 1059–1064.
- [6] M. van Heel and E. V. Orlova and G. Harauz and H. Stark and P. Dube and F. Zemlin and M. Schatz, “Angular Reconstitution in Three-Dimensional Electron Microscopy: Historical and Theoretical Aspects,” in *Scanning Microscopy Vol. 11*, 1997, pp. 195–210.
- [7] Rik van Antwerpen and Michael LaBelle and Edita Navratilova and Ronald M. Krauss and Cesar Luna-Chavez and Trudy M. Forte and John C. Gilkey, “Structural Heterogeneity of apoB-Containing Lipoproteins Visualized using Cryo-Electron Microscopy,” in *Journal of Lipid Research*, Oct. 1999, pp. 1827–1836.
- [8] NHLBI, “Heart Diseases,” National Institutes of Health, Tech. Rep., Mar. 2013, <http://www.nhlbi.nih.gov/health/health-topics/topics/cad/> Last visited on 31 Dec 2013.
- [9] Elias J. Jeyarajah and William C. Cromwell and James D. Otvos, “Lipoprotein Particle Analysis by Nuclear Magnetic Resonance Spectroscopy,” in *Clinics in Laboratory Medicine*, Dec. 2006, pp. 847–870.
- [10] Juha Vakkilainen and George Steiner and Jean-Claude Ansquer and Francois Aubin and Stephanie Rattier and Christelle Foucher and Anders Hamsten

- and Marja-Riitta Taskinen, “Relationships Between Low-Density Lipoprotein Particle Size, Plasma Lipoproteins, and Progression of Coronary Artery Disease: The Diabetes Atherosclerosis Intervention Study (DAIS),” in *Circulation* (www.circulationaha.org), Mar. 2003, pp. 1733–1737, Last visited on 31 Dec 2013.
- [11] James Otvos and R. Marabini and I.M. Masegosa and M.C. San Martn and S. Marco and J. Fernndez and L. G. de la Fraga and C. Vaquerizo and J.M. Carazo, “Radio Signals Give New Spectrum For Cholesterol Lipoprotein Readings,” in *Arteriosclerosis, Thrombosis and Vascular Biology: Journal of the American Heart Association*, July 1998, pp. 2223–2231.
- [12] Jeremy S. De Bonet and Paul Viola, “Poxels: Probabilistic Voxelized Volume Reconstruction,” in *Proceedings of ICCV*, Sept. 1999, pp. 391–398.
- [13] Ivo Ihrke and Kiriakos N. Kutulakos and Hendrik P. A. Lensch and Marcus Magnor and Wolfgang Heidrich, “State of the Art in Transparent and Specular Object Reconstruction,” *EUROGRAPHICS 2008*, 2008.
- [14] Ralph D. Freeman, “Stereoscopic Vision: Which Parts of the Brain are Involved?” *Current Biology*, vol. 9, pp. R610–R613, 1999.
- [15] Pengcheng Liu and Andrew Willis and Yunfeng Sui, “Stereoscopic 3D Reconstruction Using Motorized Zoom Lenses Within an Embedded System,” *Image Processing: Machine Vision Applications II*, edited by Kurt S. Niel and David Fofi, *Proceedings of SPIE-IS&T Electronic Imaging*, vol. 7251, pp. 72 510W1–72 510W12, 2009.
- [16] Petra A. Arndt and Hanspeter A. Mallot and Heinrich H. Bülthoff, “Human Stereovision Without Localized Image Features,” *Biological Cybernetics*, vol. 72, pp. 279–293, 1995.
- [17] M. Z. Brown and D. Burschka and G. D. Hager, “Advances in Computational Stereo,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 8, pp. 993–1008, 2003.
- [18] D. Scharstein and R. Szeliski, “A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms,” *International Journal of Computer Vision*, vol. 47, pp. 7–42, 2002.
- [19] Neus Sabater and Andrés Almansa and Jean-Michel Morel, “Meaningful Matches in Stereovision,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 5, pp. 930–942, 2012.
- [20] M. Tarr and S. Pinker, “Mental Rotation and Orientation – Dependence in Shape Recognition,” *Cognitive Psychology*, vol. 21, pp. 233–282, 1989.

- [21] S. Edelman and Heinrich H. Bülthoff and D. Weinshall, “Stimulus Familiarity Determines Recognition Strategy for Novel 3D Objects,” Massachusetts Institute of Technology, Boston, MA, USA, Tech. Rep., 1989.
- [22] S. Seitz and B. Curless and J. Diebel and D. Scharstein and R. Szeliski, “A Comparison and Evaluation of Multi-View Stereo Reconstruction Algorithms,” *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 519–526, 2006.
- [23] Patrick Labatut and Jean-Philippe Pons and Renaud Keriven, “Efficient Multi-View Reconstruction of Large-Scale Scenes Using Interest Points, Delaunay Triangulation and Graph Cuts,” in *ICCV*, 2007, pp. 1–8.
- [24] Kong-man (German) Cheung and Simon Baker and Takeo Kanade, “Shape-From-Silhouette Across Time Part I: Theory and Algorithms,” *International Journal of Computer Vision*, vol. 62, no. 3, pp. 221–247, Mar. 2005.
- [25] Laurentini, A., “The Visual Hull Concept for Silhouette-Based Image Understanding,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 2, pp. 150–162, Aug. 1994.
- [26] B.G. Baumgart, “Geometric Modeling for Computer Vision,” Ph.D. dissertation, Stanford, 1997.
- [27] D. Marr and T. Poggio, “Cooperative Computation of Stereo Disparity,” *Science*, vol. 194, no. 4262, pp. 283–287, Oct. 1976.
- [28] H. C. Longuet-Higgins, “A Computer Algorithm for Reconstructing a Scene from Two Projections,” *Nature*, vol. 293, pp. 133–135, Sept. 1981.
- [29] Y. Dong and G. R. Hillman, “Three-Dimensional Reconstruction of Irregular Shapes Based on a Fitted Mesh of Contours,” *Image and Vision Computing*, vol. 19, pp. 165–176, 2001.
- [30] Giblin, Peter and Weiss, Richard, “Reconstruction of Surfaces from Profiles,” University of Massachusetts, Amherst, MA, USA, Tech. Rep., 1987.
- [31] Kiriakos N. Kutulakos and Steven M. Seitz, “A Theory of Shape by Space Carving,” *International Journal of Computer Vision*, vol. 38, no. 3, pp. 199–218, 2000.
- [32] Annie Yao and Andrew Calway, “Interpolating Novel Views from Image Sequences by Probabilistic Depth Carving,” in *ECCV 2004*, May 2004, pp. 379–390.
- [33] A Broadhurst and T.W Drummond and R Cipolla, “A Probabilistic Framework for Space Carving,” in *Proc Int Conf on Computer Vision*, 2001.

- [34] P. A. Midgley and M. Weyland, “3D Electron Microscopy in the Physical Sciences: The Development of Z-contrast and EFTEM Tomography,” *Ultra-microscopy*, vol. 96, pp. 413–431, 2002.
- [35] Borislav Trifonov and Derek Bradley and Wolfgang Heidrich, “Tomographic Reconstruction of Transparent Objects,” in *Proceedings of Eurographics Symposium on Rendering*, June 2006, pp. 51–60.
- [36] Joachim Frank, “Single-Particle Reconstruction of Biological Macromolecules in Electron Microscopy - 30 years,” in *Q Rev Biophys*, Aug. 2009, pp. 139–158.
- [37] Marin van Heel and Brent Gowen and Rishi Matadeen and Elena V. Orlova and Robert Finn and Tillmann Pape and Dana Cohen and Holger Stark and Ralf Schmidt and Michael Schatz and Ardan Patwardhan, “Single-Particle Electron Cryo-Microscopy: Towards Atomic Resolution,” in *Quarterly Reviews of Biophysics*, Apr. 2000, pp. 307–369.
- [38] J. Fessler, *Chapter 3. Analytical Tomographic Image Reconstruction Methods*. web.eecs.umich.edu/~fessler/course/516/1/c-tomo.pdf, 2009, Last visited on 31 Dec 2013.
- [39] Harvard University. “The Central Section Theorem.” Last visited on 31 Dec 2013. Feb. 2010. [Online]. Available: https://software.rc.fas.harvard.edu/leschziner_public/index.php/Central_Section_Theorem
- [40] Harold P. Erickson and A. Klug, “The Fourier Transform of an Electron Micrograph: Effects of Defocusing and Aberrations, and Implications for the use of Underfocus Contrast Enhancement,” in *Berichte der Bunsengesellschaft für physikalische Chemie*, Nov. 1970, pp. 1129–1137.
- [41] K. J. Hanszen, “The Optical Transfer Theory of the Electron Microscope: Fundamental Principles and Applications,” in *Advances in Optical and Electron Microscopy*. Academic Press, New York, 1971, pp. 1–84.
- [42] P.A. Penczek and J. Zhu and R. Schröder and J. Frank, “Three Dimensional Reconstruction with Contrast Transfer Compensation from Defocus Series,” in *Scanning Microscopy*. Scanning Microscopy International, 1997, pp. 147–154.
- [43] www.wadsworth.org. “Contrast Transfer Function (CTF) Correction.” Last visited on 31 Dec 2013. Aug. 2012. [Online]. Available: http://www.wadsworth.org/spider_doc/spider/docs/techs/ctf/ctf.html
- [44] Steven J. Ludtke and Philip R. Baldwin and Wah Chiu, “EMAN: Semiautomated Software for High-Resolution Single-Particle Reconstructions,” in *Journal of Structural Biology*, Dec. 1999, pp. 82–97.

- [45] J.J. Fernández and J.I. Agulleiro and J.R. Bilbao-Castro and A. Martínez and I. García and F.J. Chichón and J. Martín-Benito and J.L. Carrascosa, “Image Processing in Electron Tomography,” in *Microscopy: Science, Technology, Applications and Education: A. Mendez-Vilas and J. Diaz (Eds.)*, 2010, pp. 19–28.
- [46] M. Radermacher and T. Wagenknecht and A. Verschoor and J. Frank, “Three-dimensional Reconstruction from a Single-Exposure, Random Conical Tilt Series Applied to the 50S Ribosomal Subunit of *Escherichia coli*,” in *Journal of Microscopy*, May 1987, pp. 113–136.
- [47] University of California San Diego. “Cryo-electron Microscopy at UCSD.” Last visited on 31 Dec 2013. June 2010. [Online]. Available: <http://cryoem.ucsd.edu/>
- [48] University of California San Diego. “The Electron Microscopy Outreach Program.” Last visited on 31 Dec 2013. June 2010. [Online]. Available: <http://em-outreach.ucsd.edu/index.html>
- [49] Wikipedia. “Fourier shell correlation.” Last visited on 31 Dec 2013. Aug. 2012. [Online]. Available: http://en.wikipedia.org/wiki/Fourier_shell_correlation
- [50] www.imagescience.de. “Image Science: Advanced Image Processing & Image Analysis.” Last visited on 31 Dec 2013. Aug. 2012. [Online]. Available: <http://www.imagescience.de/imagic.html>
- [51] Rik van Antwerpen and John C. Gilkey, “Cryo-Electron Microscopy Reveals Human Low Density Lipoprotein Substructure,” in *Journal of Lipid Research*, Dec. 1994, pp. 2223–2231.
- [52] Rik van Antwerpen and G. Chi Chen and Clive R. Pullinger and John P. Kane and Michael LaBelle and Ronald M. Krauss and Cesar Luna-Chavez and Trudy M. Forte and John C. Gilkey, “Cryo-Electron Microscopy of Low Density Lipoprotein and Reconstituted Discoidal High Density Lipoprotein: Imaging of the apoLipoprotein Moiety,” in *Journal of Lipid Research*, Apr. 1997, pp. 659–669.
- [53] Elena V. Orlova and Michael B. Sherman and Wah Chiu and Hiro Mowri and Louis C. Smith and Antonio M. Grotto Jr., “Three-dimensional Structure of Low Density Lipoproteins by Electron Cryomicroscopy,” in *Proceedings of the National Academy of Science USA*, July 1999, pp. 8420–8425.
- [54] Tom Teerlink and Peter G. Scheffer and Stephan J. L. Bakker and Robert J. Heine, “Combined Data from LDL Composition and Size Measurement are Compatible With a Discoid Particle Shape,” in *Journal of Lipid Research*, May 2004, pp. 954–966.

- [55] Larry N. Thibos, “Image Processing by the Human Eye,” in *Visual Communications and Image Processing IV*, William A. Pearlman (ed), *Proc SPIE 1199*, 1989, pp. 1148–1153.
- [56] S.J. Thorpe and A. Delorme and R. Van Rullen and W. Paquier, “Reverse Engineering of the Visual System Using Networks of Spiking Neurons,” in *IEEE International Symposium on Circuits and Systems*, 2000, pp. 405–408.
- [57] Prince, S.J.D., *Computer Vision: Models Learning and Inference*. Cambridge University Press, 2012.
- [58] Haim Schweitzer and Sanjeev R. Kulkarni, “Computational Limitations of Model-Based Recognition,” *International Journal of Intelligent Systems*, vol. 9, pp. 431–443, 1998.
- [59] Irving Biederman, “Recognition-by-Components: A Theory of Human Image Understanding,” *Psychological Review*, vol. 94, no. 2, pp. 115–147, 1987.
- [60] G. Humphrey, “The Psychology of the Gestalt,” *Journal of Educational Psychology*, vol. 15, no. 7, pp. 401–412, 1924.
- [61] A. H. Barr, “Superquadrics and Angle-Preserving Transformations,” *IEEE Computer Graphics and Applications*, vol. 1, no. 1, pp. 11–23, 1981.
- [62] A. P. Pentland, “Perceptual Organization and the Representation of Natural Form,” *Artificial Intelligence*, vol. 28, no. 2, pp. 293–331, 1986.
- [63] Franc Solina and Ruzena Bajcsy, “Recovery of Parametric Models from Range Images: The Case for Superquadrics with Global Deformations,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 2, pp. 131–147, 1990.
- [64] Ales Leonardis and Ales Jaklic and Franc Solina, “Superquadrics for Segmenting and Modeling Range Data,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 11, pp. 1289–1295, 1997.
- [65] Irving Biederman, “Recognizing Depth-Rotated Objects: A Review of Recent Research and Theory,” *Spatial Vision*, vol. 13, no. 3, pp. 241–253, 2001.
- [66] David J. Kriegman and Jean Ponce, “On Recognizing and Positioning Curved 3-D Objects from Image Contours,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 12, pp. 1127–1137, 1990.
- [67] Demetri Terzopoulos and Dimitri Metaxas, “Dynamic 3D Models with Local and Global Deformations: Deformable Superquadrics,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 7, pp. 703–714, 1991.

- [68] Sven J. Dickinson and Alex P. Pentland and Azriel Rosenfeld, “3-D Shape Recovery Using Distributed Aspect Matching,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 174–198, 1992.
- [69] Daphna Weinshall and Michael Werman, “On View Likelihood and Stability,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 2, pp. 97–106, 1997.
- [70] Sven J. Dickinson and Dimitri Metaxas and Alex P. Pentland, “The Role of Model-Based Segmentation in the Recovery of Volumetric Parts From Range Data,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 3, pp. 259–267, 1997.
- [71] Yan Zhang and Andreas Koschan and Mongi Abidi, “Superquadrics Based 3D Object Representation of Automotive Parts Utilizing Part Decomposition,” in *Proceedings of SPIE 6th International Conference on Quality Control by Artificial Vision*, May 2003, pp. 241–251.
- [72] Jiehua Zhu and Shiyong Zhao and Yangbo Ye and Ge Wang, “Computed Tomography Simulation with Superquadrics,” *Medical Physics*, vol. 32, no. 10, pp. 3136–3143, Oct. 2005.
- [73] Samuel W. Hasinoff and Kiriakos N. Kutulakos, “Photo-Consistent Reconstruction of Semi-Transparent Scenes by Density Sheet Decomposition,” in *PAMI*, May 2007, pp. 870–885.
- [74] Navdeep Jaitly and Marcus A. Brubaker and John L. Rubinstein and Ryan H. Lilien, “A Bayesian Method for 3D Macromolecular Structure Inference using Class Average Images from Single Particle Electron Microscopy,” *Bioinformatics*, vol. 26, no. 19, pp. 2406–2415, Oct. 2010, Last visited on 31 Dec 2013. [Online]. Available: <http://dx.doi.org/10.1093/bioinformatics/btq456>
- [75] W. Lewis Collier, “SVM Analysis of Cryogenic Transmission Electron Microscopy Images,” in *URI CSC-581, Topics in AI: Support Vector Machines, Spring 2007, Final Project*, May 2007, pp. 1–22.
- [76] Mark Stamp, “A Revealing Introduction to Hidden Markov Models,” San Jose State University, San Jose, CA, USA, Tech. Rep., Sept. 2012, www.cs.sjsu.edu/~stamp/RUA/HMM.pdf Last visited on 31 Dec 2013.
- [77] Michael Sipser, *Introduction to the Theory of Computation, 2nd. edn.* Boston, Massachusetts, United States of America: Thomson Course Technology, 2006.
- [78] Lawrence R. Rabiner, “A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition,” in *Proceedings of the IEEE*, 1989, pp. 257–286.

- [79] R. L. Cave and L. P. Neuwirth, “Hidden Markov Models for English,” *Hidden Markov Models for Speech*, Oct. 1980.
- [80] A. J. Viterbi, “Error Bounds for Convolutional Codes and an Asymptotically Optimal Decoding Algorithm,” in *IEEE Transactions on Information Theory*, 1967, pp. 260–269.
- [81] G. D. Forney, “The Viterbi Algorithm,” in *Proceedings of the IEEE*, 1973, pp. 268–278.
- [82] F. Jelinek, “Continuous Speech Recognition by Statistical Methods,” in *Proceedings of the IEEE*, 1976, pp. 532–536.
- [83] R. Bakis, “Continuous Speech Word Recognition via Centi-Second Acoustic States,” in *Proceedings of ASA Meeting (Washington, DC)*, 1976.
- [84] James Zhao, “Hidden Markov Models with Multiple Observation Processes,” *Honours Thesis*, Nov. 1980.
- [85] Xiaolin Li and Marc Parizeau and Rejean Plamondon, “Training Hidden Markov Models with Multiple Observations – A Combinatorial Method,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Apr. 2000.
- [86] Davis, R.I.A. and Lovell, B.C. and Caelli, T., “Improved Estimation of Hidden Markov Model Parameters from Multiple Observation Sequences,” in *Proceedings, 16th International Conference on Pattern Recognition*, vol. 2, 2002, pp. 168–171.
- [87] Shun-Zheng Yu and Hasashi Kobayashi, “A Hidden Semi-Markov Model with Missing Data and Multiple Observation Sequences for Mobility Tracking,” *Signal Processing*, pp. 235–250, Dec. 2001.
- [88] Mohammed J. Zaki and Christopher D. Carothers and Boleslaw K Szymanski, “VOGUE: A Variable Order Hidden Markov Model with Duration Based on Frequent Sequence Mining,” *ACM Transactions on Knowledge Discovery From Data*, vol. 4, Jan. 2010.
- [89] Thede, Scott M. and Harper, Mary P., “A Second-Order Hidden Markov Model for Part-of-Speech Tagging,” in *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, ser. ACL ’99. Stroudsburg, PA, USA: Association for Computational Linguistics, 1999, pp. 175–182. [Online]. Available: <http://dx.doi.org/10.3115/1034678.1034712>
- [90] Johan A. du Preez, “Efficient High-Order Hidden Markov Modelling,” *Doctorial Dissertation*, Nov. 1997.

- [91] Lee-Min Lee, “High-Order Hidden Markov Model and Application to Continuous Mandarin Digit Recognition,” *Journal of Information Science and Engineering*, vol. 27, pp. 1919–1930, 2011.
- [92] Dewar, M. and Wiggins, C. and Wood, F., “Inference in Hidden Markov Models with Explicit State Duration Distributions,” *Signal Processing Letters, IEEE*, vol. 19, no. 4, pp. 235–238, 2012.
- [93] W. Lewis Collier, “Acquisition and Analysis of Cryogenic Transmission Electron Microscope Biological Images,” in *URI CSC-592*, May 2005, pp. 1–22.
- [94] W. Lewis Collier and Lenore M. Martin and Rik van Antwerpen, “Accuracy Limits for Efficiently Determining Shape and Size of Low-Density Lipoprotein Macromolecules from Cryogenic TEM Images,” in *SPIE Medical Imaging Conference*, vol. 6511-97, Feb. 2007, pp. 1–9.
- [95] W. Lewis Collier and Lenore M. Martin and Rik van Antwerpen, “Computer Vision Determination of 3-D Geometric Parameters of LDL Particles via Cryogenic Transmission Electron Microscopy,” in *Journal of the Society for Information Display*, Aug. 2007, pp. 391–398.
- [96] Wikipedia : The Free Encyclopedia. “Bilinear interpolation.” Last visited on 31 Dec 2013. Sept. 2012. [Online]. Available: http://en.wikipedia.org/wiki/Bilinear_interpolation
- [97] W. Lewis Collier and Jean-Yves Hervé and Lenore M. Martin, “Towards Individual Particle Reconstruction from Cryogenic Transmission Electron Microscopy,” in *2013 ACM Conference on Bioinformatics, Computational Biology and Biomedical Informatics*, ser. ACM-SIGBio-BIB-2013. 2 Penn Plaza, Suite 701, New York New York 10121-0701, USA: Association for Computing Machinery, Sept. 2013, pp. 526–535.
- [98] M Beer and J Frank and K J Hanszen and E Kellenberger and RC Williams, “The Possibilities and Prospects of Obtaining High-Resolution (below 30 Å) on Biological Material Using the Electron Microscope. Some Comments and Reports Inspired by an EMBO Workshop Held at Gais, Switzerland, October 1973,” in *Quarterly Reviews of Biophysics*, vol. 7, Aug. 1974, pp. 211–238.

APPENDIX A

Processing Details

This appendix provides details of the overall processing software. Section A.1 outlines the basic ARTEMIS framework. Details of the simulated projection images are provided in section A.2. A linear representation of the reconstruction space is provided in section A.3.

A.1 ARTEMIS Software Framework

The ARTEMIS processing structure is shown in Figure A.36. Each of the processes is summarized here with more details in the appropriate section of Chapter 3.

`genMRC` creates a 3-D voxel space programmatically and stores the results in a `.MRC` file.

`cvt2MRC` allows for generation of `.MRC` file from various other program file formats (e.g. PDB).

`genProjections` creates a set of projection images and associated transmission ray information and statistics from a voxel space based on a model of the *transmission electron microscope* transmission through the voxel space.

`linSolver` generates a 3-D voxel space from the model statistics and observations via linear algebra processing.

`genObservations` generates observations for use in the hidden Markov model from the images created by simulation or from *Cryo-tem*.

`hmmSolver` generates a 3-D voxel space from the model statistics and observations via hidden Markov Model processing.

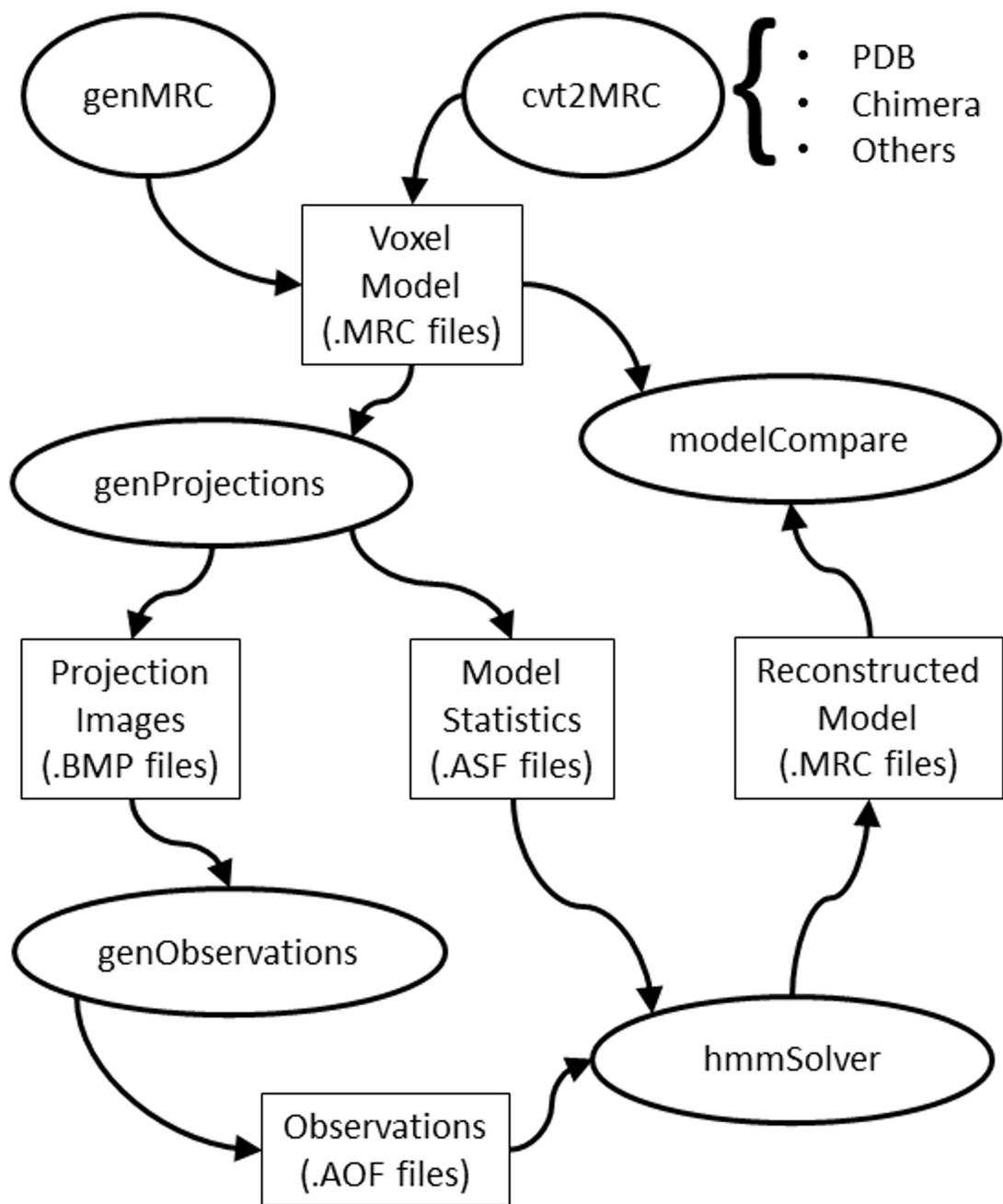


Figure A.36. Artemis Processing Flow

`modelCompare` compares the models in order to determine convergence of various reconstruction methods.

A.1.1 `genMRC`

The `genMRC` program allows for creation and visualization of a transparent voxel space programmatically and parametrically. The simulated target is created in a 3-D space that is $v_0 \times v_0 \times v_0$ voxels in size, where v_0 is usually odd so that the center pixel is at location $\{0, 0, 0\}$. Each voxel is given a unique density based on the target to be modeled. The density is treated as a transmission coefficient where a value of 1 means full transmission and 0 means no transmission (opaque). The voxel parameters are defined in a subroutine so that exact definition of the 3-D volume may be defined by the programmer. This utility is useful for generating simple density shapes for testing and algorithm validation.

A.1.2 `cvt2MRC`

The `cvt2MRC` program follows the `genMRC` functionality except this is a conversion utility that is programmed to read in various file formats so that a `.MRC` file can be created for further processing. As with the `genMRC` utility, the `cvt2MRC` utility allows for programmatic conversion from various formats to the `.MRC` file.

A.1.3 `genProjections`

The `genProjections` utility reads a 3-D voxel space from the `.MRC` file to create projection images and some of the associated statistics that are used in ensuing voxel space reconstruction processing. The 2-D projection images are created by treating the density specifications in the file as transmission coefficients. The voxel densities also used used to tabulate statistics about the density relationships between adjacent voxels and between voxels the pixels in the projections images.

The `genProjections` utility also tabulates statistics of voxel to voxel transi-

tions along each intersecting ray. This provides data for a model of the target and surrounding buffer region based on the transitions from density to density. In the HMM paradigm, this corresponds to the A matrix of probabilities of state to state transitions.

PI, A, and B are created from the voxel intersections used to create the projection images.

```

1 pi==0;a==0;b==0 // define these again
2 for each intersection i=0 to N-1
3 {
4   if (i==0)
5   {
6     pi[density of this voxel] += intersection length L.i;
7   }
8   else
9   {
10    A[last_intersection][this_density] += intersection length L.i;
11  }
12
13 B[this_density][pixel_value] += intersection length L.i;
14
15 last_density = this_density;
16 }
17
18 normalize pi, A, and B to be row stochastic.

```

A.1.4 genObservations

The `genObservations` utility reads in projection images and reference data created by the `genProjections` program and generates the sequence of observations required for reconstruction of the 3-D voxel space via the HMM approach. By convention, vertical stacks of the voxel space are used as the state sequences

for the HMM. The voxel space can be rotated such that any one of the 6 cube faces points downward and the target can be oriented at any angle within the voxel space so this does not create a restriction on possible model estimates. Observation generation follows the same basic idea as projection image creation but the rays are based on the voxel centers rather than the pixel centers. Once the intersection with the image is determined, the observation is created from the pixel values neighboring the intersection point. This operation can be repeated for any number of projection images to support multiple observation regimes. A synopsis of the software processing for this utility is provided in Table A.4.

Inputs	<ul style="list-style-type: none"> • voxel base spacing • image angle • sub-voxel information
Processing	<ul style="list-style-type: none"> • generate observations
Outputs	<ul style="list-style-type: none"> • header • observations

Table A.4. `genObservations` Software Specification Synopsis

Sub-voxelization may be used in the observation generation processing to ensure observation independence. Since most *cryo-tem* images are taken in the -45° to $+45^\circ$ range, and the problem can be symmetric beyond the 45° boundary, only the -45° to $+45^\circ$ case for sub-voxels is considered. For angles smaller than 45° , each voxel can be divided into an integral number of sub-voxels. For larger angles, each voxel is divided into 8 ($2 \times 2 \times 2$) sub-voxels. As the angle gets smaller, the voxels are divided into ($3 \times 3 \times 3$) sub-voxels, and so on. If every sub-voxel in a stack is used, the number of sub-voxels needed to preserve pure independence of the observations can be seen from Figure A.37. The number of sub-voxels required for pure independence is give by Equation A.1.1.

$$N_{sv} = (\text{ceiling}(1/\tan(\theta)))^3. \quad (\text{A.1.1})$$

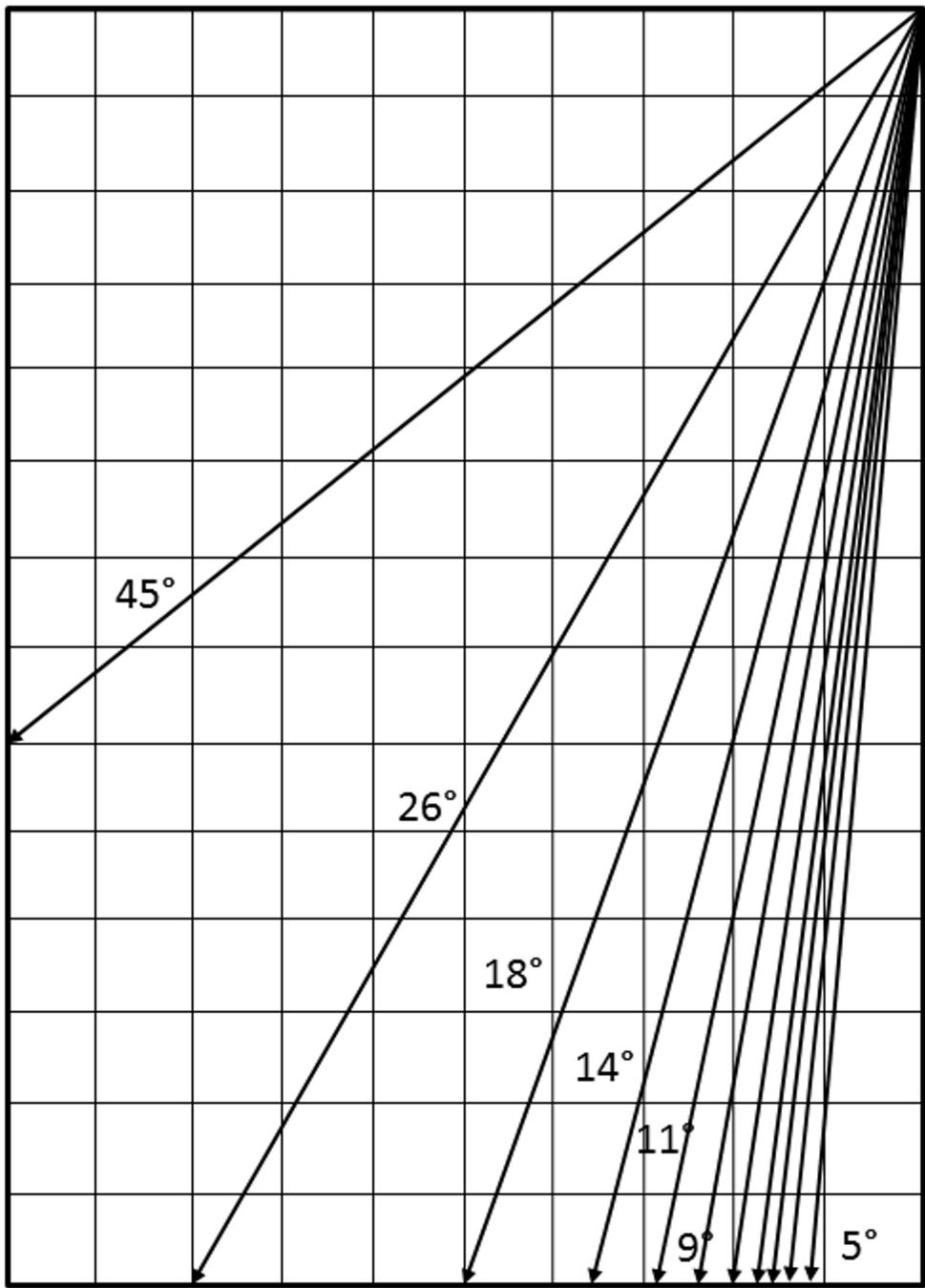


Figure A.37. Sub-voxels Needed for Observation Independence

Note that as the projection angle gets small, the number of sub-voxels can grow to ∞ since all traces are within the same voxel stack. With cubic sub-voxelization, this constraint places a lower bound on the rotation angle that can be used for the HMM reconstruction.

This number of sub-voxels coupled with the expectation of small angles may be too much of a constraint for tractable modeling. A rotation of approximately 5° would need $(12 \times 12 \times 12)$ sub-voxels, an increase of 1728 over the base number of voxels, which means a corresponding increase in observations since each voxel (or sub-voxel) gives rise to a single observation. The processing is linear with the number of observations, this is not an insurmountable computation increase. But with an expected basis voxel size of $(303 \times 303 \times 101)$ the memory requirements of 1.6×10^{10} sub-voxels make this doable but unattractive for implementation.

The sub-voxels have some degrees of flexibility in order to accommodate the tractability issues. Hidden Markov models only need independence of observations and a series of states so the sub-voxels do not have to be uniform in all dimensions. And since the basis voxels are divided into the sub-voxels, some sort of recombination of sub-voxels to voxels will need to be performed so this can be used to help with the independence constraint. Figure A.38 shows a possible sub-voxel partitioning for an expected rotation of 5.2° . Figure A.38a shows (in 2-D) voxels divided into $(12 \times 12 \times 1)$ sub-voxels so that the observations are completely independent. Figure A.38b shows (in 2-D) voxels divided into $(12 \times 6 \times 1)$ sub-voxels so that the observations are almost independent. As can be seen, simple geometric calculations can be performed to determine the expected independence. Testing with actual models and data can be performed to ascertain the effect of such voxel partitioning. These adaptations reduce the total observation count by over an order of magnitude so the problem becomes much less resource intensive.

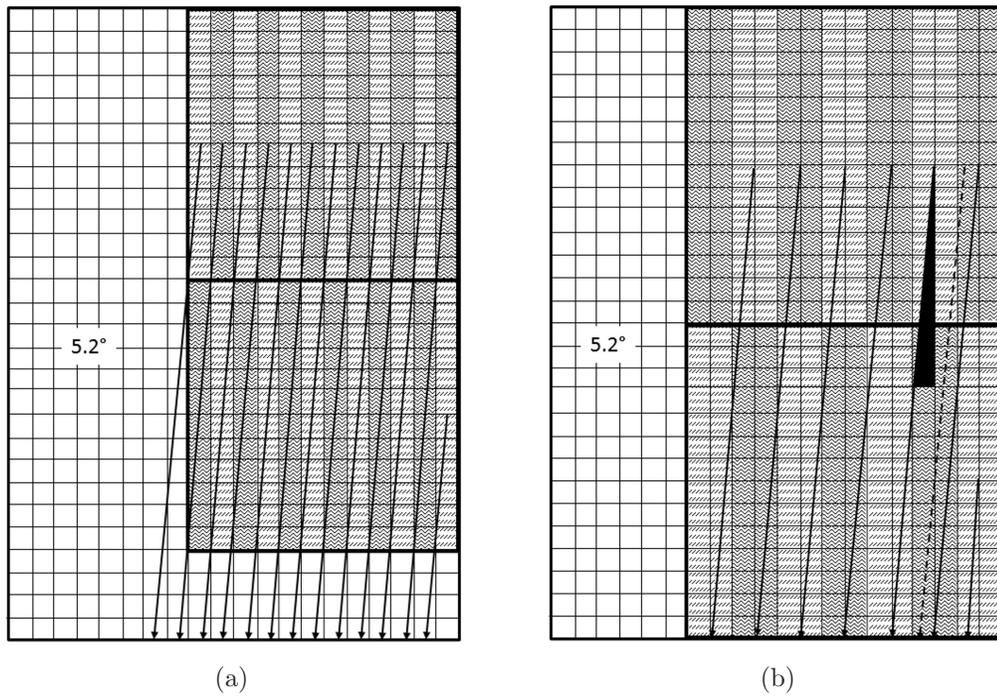


Figure A.38. Observation independence with thin voxel stacks a) 5° with single sub-voxels and b) 5° with double wide but taller stacks.

The small rotation angle also gives rise to a limited spatial extent of the state columns. For example, the 5.2° case shown only spans 101/12 columns in the $\{x, z\}$ directions. When computing the projections, only those voxels contributing to the pixel need be considered. And since the rotation of the *cryo-tem* is generally along a single axis, only a slice of the voxel space affects the reconstruction process. Thus, the memory space required can be significantly reduced as only the 9×101 voxels need be considered for generation of each observation. With the same voxel value used for each of the 144 sub-voxels, only the sub-voxel centers need be accessed for the ray processing.

A.1.5 hmmSolver

The `hmmSolver` utility reads in data created by the `genProjections` and `genObservations` programs and reconstructs a 3-D voxel space that approximates the original data via a HMM approach. The processing updates an estimate (Vhat or \widehat{V}) for each voxel in a Markov chain (gray stack) by applying the beta pass as described in Section 2.5.2.2. A list of voxel estimates is kept based on the image used for observations and any prior estimates. When the observations from all the images have been processed, the voxel estimates are combined to determine the final voxel estimate. The following pseudo code shows the basic processing for the 45° case where sub-voxels are not needed.

```

1 Loop i over images
2 {
3   Compute Vhat(x,y,z,i) in each stack in Y with BetaPass
4   if 1st picture
5   {
6     Voxel(x,y,z) = Vhat(x,y,z,0)
7   }
8   else

```

```

9  {
10  if (all Vhat are the same)
11  {
12    V(x,y,z) = Vhat(x,y,z,0)
13  }
14  else if there is a majority in the set of Vhat(x,y,z,i)
15  {
16    V(x,y,z) = majority( Vhat(x,y,z,i) )
17  }
18  else
19  {
20    V(x,y,z) = state (element of Vhat(x,y,z,i)) that minimizes
21    | pixel(x,z) - computed projection value of current stack |
22  }
23 }
24 loop over x,z
25 {
26   compute log(P(0|lambda)) for this stack
27   // maybe do some overall stats as well to look for convergence
28   ?
29 }

```

When sub-voxels are needed, the above processing is still used, but with slight changes. As expected, the loops are over the sub-voxels rather than the whole voxels. The estimated values \hat{S} are computed for the small thin sub-voxel regions rather than the whole voxel. All of the \hat{S} s can be used to against the vertical pixel value voxel to compute the \hat{V} but this leads to an explosion of possibilities so the \hat{S} s need to be combined. A simple voting process can be used to generate \hat{V} from the \hat{S} 's. The \hat{V} 's can then be combined across images as above.

A.2 Simulation of Projection Images

The ARTEMIS framework can be used to generate simulated projection images in order to facilitate reconstruction algorithm studies. An example simulated target and associated geometry is depicted in Figure A.39. A simple disc, which emulates the expected LDL shape, is shown in the figure but this simulated shape can be set to any configuration programmatically. The 3-D space is filled with the expected

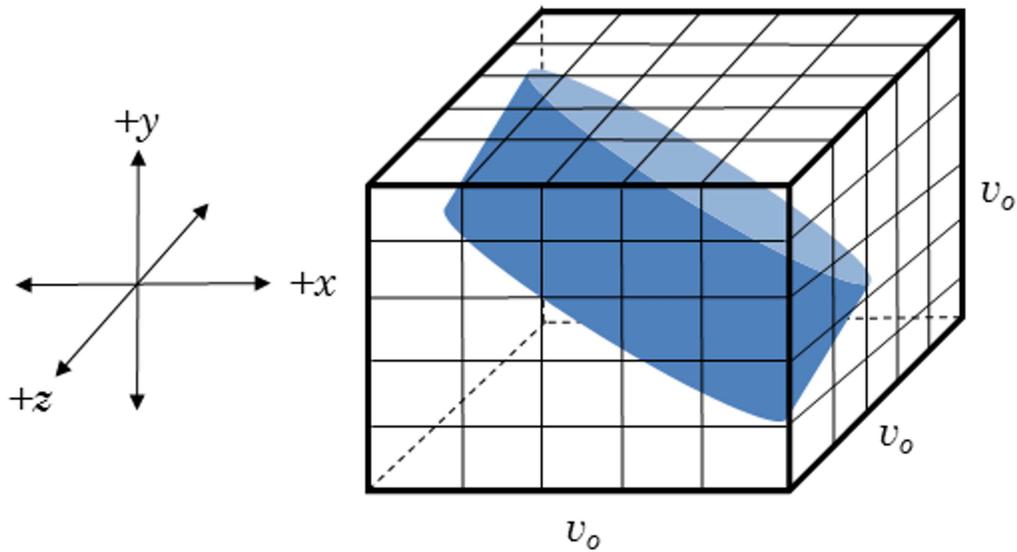


Figure A.39. Simulated Target, Voxel Space, and Geometry Orientation

density (transmission coefficient) for each voxel based on the simulated target.

Orthographic projections through the target space can be used to generate a base image of $m_0 \times m_0$ pixels in size. The number of pixels (m_0) is usually set to match the number of voxels (v_0) but these parameters are not constrained to be equal. The physical overall size of the $m_0 \times m_0$ image, however, does represent the same physical extent as the $v_0 \times v_0$ face of the voxel cube. Since the image resolution can vary to match the needed reconstruction dimensions, the mapping of the v_0 to m_0 defines the reconstruction resolution.

For reconstruction purposes, however, projection images are expected to be

needed at more than the normal angle. The 3-D voxel space must be extended (to $V_0 \times V_0 \times v_0$) so that all transmission rays oriented at an angle to the y axis will pass through the same vertical dimension. Figure A.40 shows the same example target from Figure A.39 but with the voxel space extended in the x and z axes. The dark outlined cube is the original v_0 dimension voxel space, the larger parallelepiped denotes the extended voxel space. The new x and z dimension V_0 is determined by the maximum rotation angle of 45° around either the x (left-right) or z (front-back) axes. The 45° case gives rise to $V_0 = 3v_0$ from the geometric reconstruction shown in Figure A.40. These extreme cases mean that a 3×3 grid of $v_0 \times v_0 \times v_0$ cubes is needed in order to allow the intersecting rays to traverse the same vertical dimension of voxels. For computational simplicity, v_0 is generally odd, resulting in there being $(v_0 - 1)/2$ voxels on each side of the center voxel, so $V_0 = 3v_0$ is also odd.

2-D projections generated from the $V_0 \times V_0 \times v_0$ voxel space require more pixels than do the orthographic projections of size $m_0 \times m_0$. The dimension of each image is defined as $M_o \times M_o$ to follow the convention of square images from the *cryo-tem* camera. The mapping of v_0 to m_0 still defines the reconstruction resolution but for real data, M_0 is set to match the camera image output size so the allowable reconstruction resolutions may be limited by the number of pixels in the images. Likewise, additional pixels may be included in the images in order to specify the reconstruction resolution. For simulated data M_0 is also defined by the maximum rotation at 45° in the front-back and left-right directions. The ARTEMIS software utilizes 6 compute regions corresponding to the voxel space cube faces so the 45° limit holds.

An image that is larger than the normal $m_0 \times m_0$ size is needed to provide total coverage for computation of the HMM observations. A construction using

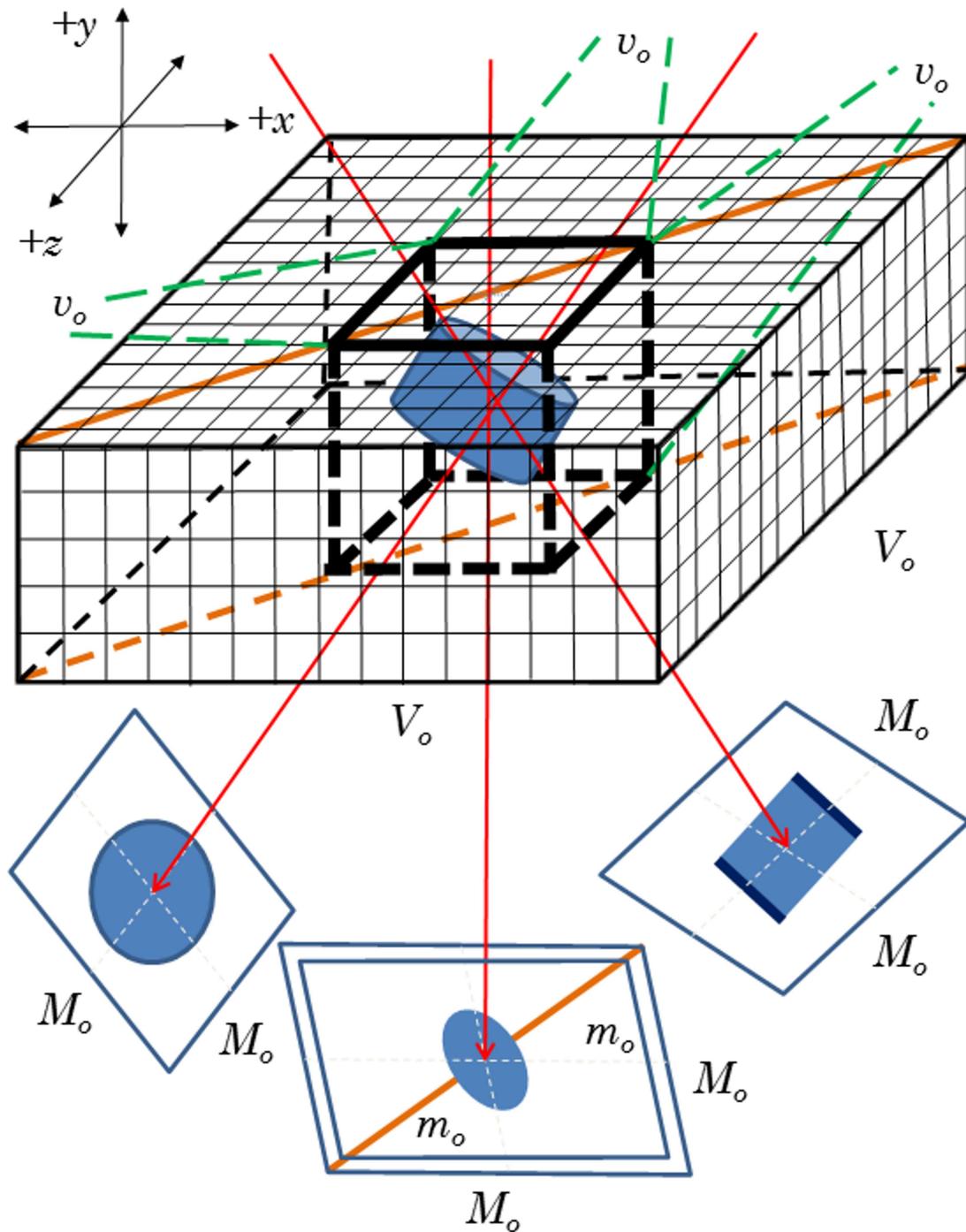


Figure A.40. Generation of Projection Images from a Simulated Target

Figure A.41 defines the extensions needed to the projection images beyond what may be generated from the normal image. The black dots represent the extreme

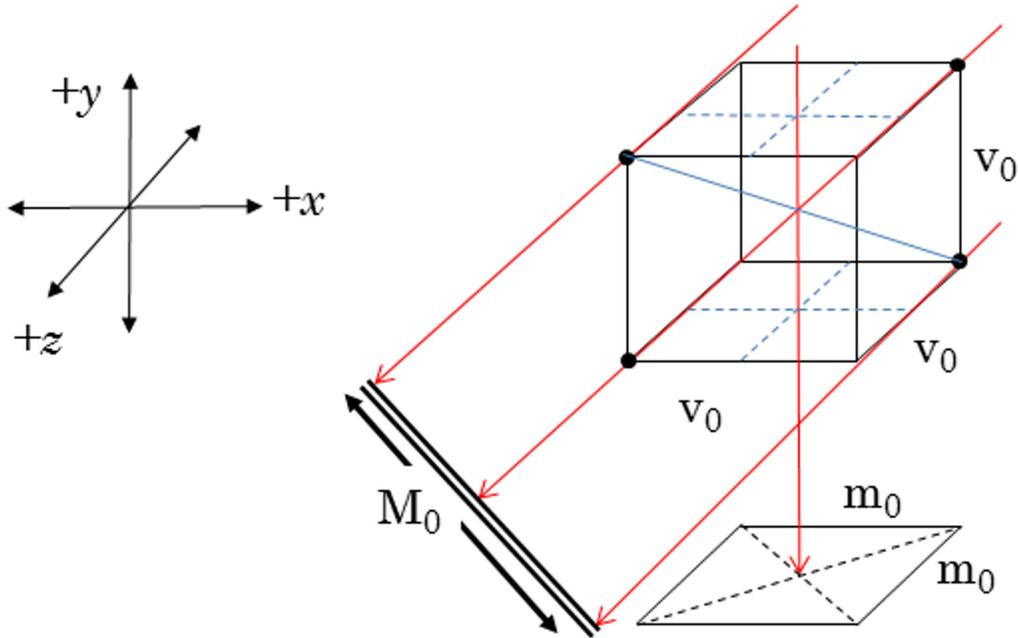


Figure A.41. Extension of m_0 to M_0

intersections so the longest intersecting ray at a -45° angle in the original voxel space goes from the back-upper-right corner to the front-lower-left corner. The construction gives $|M_0| = |m_0|\sqrt{3}$. Since the camera images are generally square, the largest image size needs to be $M_0 \times M_0$ regardless of the actual image rotation angle. Care is taken in that the linear extent of the m_0 pixels is equal to the linear extent of v_0 but since the number of pixels m_0 and number of voxels v_0 do not have to be equivalent, the lengths are matched, not the pixel and voxel counts. M_0 may also need adjustment to round up to an integer since fractional pixels are nonsensical. And M_0 must be large enough so that the bilateral interpolation used in the observation processing has an extra row and column along all four edges.

This construction implies some constraints on the physical setup but these are handled easily. The extent of the voxel space implies that the vertical thickness of

the voxel space is flat over the region of interest. This technically is not true but the vertical variation due to the meniscus effect of the voxel space containing each individual particle is expected to be very small relative to the overall vertical size of the sample space. And, since the RCT approach discussed in section 2.2.2.3 is not being followed for these images, the overall voxel space vertical extent is assumed to be much thicker than the target of interest so that any particle orientation can be obtained.

A.2.1 Orthographic Projections

The 2-D projections of the simulated 3-D targets are generated as orthographic projections of the voxel space onto the images. The center of the voxel space is mapped to the center of the projection image as is shown by the red rays in Figure A.40. Similarly, each pixel center location is projected through the target voxel space. The projection ray for each pixel is used to compute intersection lengths for each voxel that contributes to the given pixel. This process is repeated for each desired pixel in the projection image.

The algorithm for this processing is based on the location of the projection image pixels relative to the voxel space. The voxel space is held fixed and the projection plane is rotated. This rotation is used to locate each pixel in the projection image relative to the target center based on the desired rotation of the target. The simulated value for each pixel in the projection image is generated with the following steps:

1. Compute the location (rotation and translation) of pixel(i,j) in 3-D space based on the rotation angle and the projection center.
2. Compute the endpoints for the segment for use in the voxel intersection processing at a fixed distance above and below the voxel space to conserve

accuracy across the image.

3. Find the intersection of the line with the top ($+y$) of the voxel space.
4. Determine the voxel that contains the top intersection.
5. While the intersected voxel is contained in the voxel space:
 - (a) Find the intersection of the line and the remaining sides of the current voxel.
 - (b) Determine and record the length of the line segment that passes through the current voxel.
 - (c) Determine the next voxel to be processed.
6. Compute the pixel value by computing the product.

A.2.2 Projection Image Pixel Values

The passage of the electron beam through the target leads to orthographic projection images where each pixel represents the energy that passes through the voxel space along a ray. This makes an assumption that the energy of the electrons used in the *cryo-tem* process is high enough so that the resolving distance of the electron beam is small compared to the voxel size. Typical *cryo-tem* processing meets this criterion since the electron beam energy is set such that a resolution on the order of angstroms is achieved; reconstruction voxels are on the order of nanometers. The density of the target causes scattering of the electron beam along each ray such that electrons are either transmitted through the voxel or are deflected away and do not contribute to the electron flux through the remaining voxels. This results in a multiplicative loss, where the value at the exit point from each voxel of the ray may be described as the entry flux times a transmission

coefficient. Totally opaque voxels have a transmission coefficient of zero, completely transparent voxels have a transmission coefficient of one.

The computation of the simulated pixels accounts for the voxel intersection lengths, e-beam energy / camera gain, and camera DC offset. The value at the j th pixel, p_j , of each image may be described as shown in Equation A.2.1.

$$p_j = E_j \prod_{i=0}^I (T_i^{(L_i/L_{max})}) + C_j, \quad (\text{A.2.1})$$

where I represents the number of voxel intersections for p_j . The effect of each voxel must account for the length L_i that the ray travels through voxel i where the length is known from the construction. The length L_i is normalized such that each voxel's contribution is weighted by the maximum path length L_{max} through a single voxel. The density (or transmission coefficient) T_i is given by the reconstruction model (e.g. as supplied by the model shown in Figure 5). This approach follows the physics of *cryo-tem* where an intersection length that approaches 0 through a region of constant density results in a transmission coefficient approaching 1 so that there is very little loss. And when the intersection length nears the maximum intersection length, a value nearer to the base transmission coefficient of the voxel is used. The constants E_j and C_j provide the camera scale and offset adjustments. These constants are expected to be the same for all pixels in a projection image but are provided on a per-pixel basis in case camera distortions are to be included in the projection images. Given the transmission coefficients are all less than or equal to 1, the initial energy coefficient, E_j , also helps prevent underflow from the product of many small numbers.

A reference example of the simulated target is shown in Figure A.42 at 90°, 45°, and 0° rotations. This target has three different constant density regions, which are shown in red, green, and blue. The overall shape is a stack of three discs (red and green layers shown in Figure A.42a) that is surrounded by a wrapper

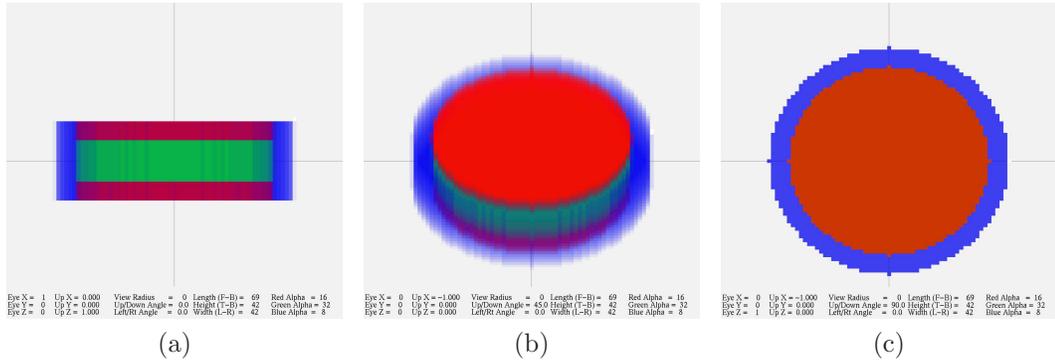


Figure A.42. Symmetric discoidal density targets at (a) 90° (Looking to the -Z direction) (b) 45° and (c) 0° (Looking to the -Y direction) rotations.

region (blue band shown in Figure A.42c). The target rotated to 45° relative to the other positions is shown in Figure A.42b. Each unit-sized cube represents a voxel with a density that is set to match the expected target; in this example, all similarly colored voxels represent the same density value.

The above processing leads to high-fidelity simulated projection images. These images take into account the orientation as well voxel densities and noise. The simulated projection images for the 90°, 45°, and 0° rotation cases with no noise are provided in Figure A.43. It has been shown [75] that the background variation

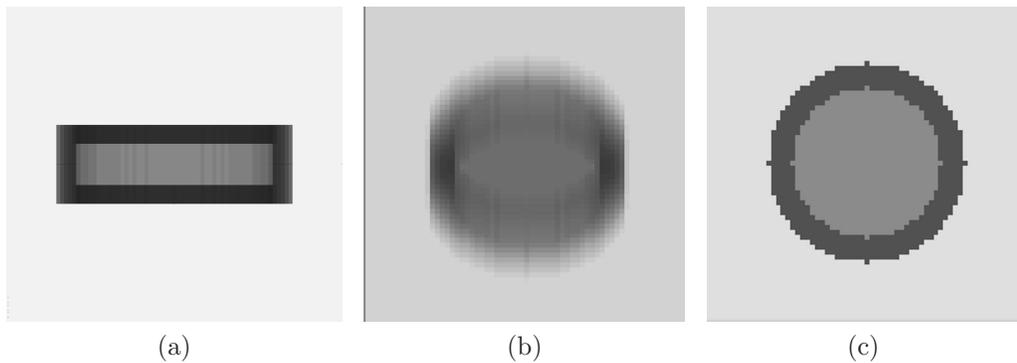


Figure A.43. Simulated projection images of symmetric discoidal density targets at (a) 90° (Looking to the -Z direction) (b) 45° and (c) 0° (Looking to the -Y direction) rotations.

does not depend on the presence or absence of particles. Thus, a background

variation that matches the camera settings and other image parameters can be added to the simulated images to produce realistic projection images. Figure A.44 shows an example image and the resulting background statistics that match a Gaussian distribution. Figure A.44c shows an example simulated projection image with Gaussian noise added.

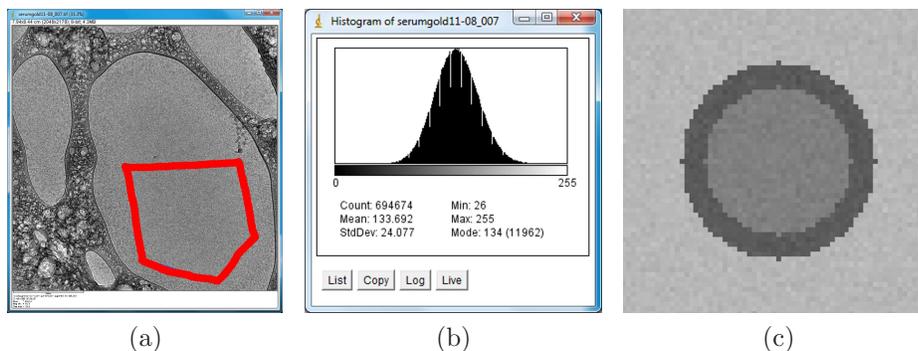


Figure A.44. (a) Example *Cryo-tem* image with artifact-free region highlighted (b) Statistics of the highlighted region and (c) 0° simulated projection image (Looking to the -Y direction) with Gaussian noise added.

The above examples provide for “perfect” simulated image but real-life imagery is never perfect. In fact, it is well known that each image acquisition degrades the sample. The generation of multiple simulated images must provide a means for degrading the images in a similar way. Beer [98] provides a model to simulating beam damage should it be needed.

A.3 Linear Reconstruction Representation

Given the construction of the simulated projection images, a straightforward approach is to build linear equations from the reconstruction space to solve for the individual voxel density values. Intuition tells us that this will be a sparse matrix and intuition is correct. However, mapping the problem leads to insights that are helpful when visualizing the other solutions so it is presented.

Linear equations for each pixel in terms of the unknown transmission coef-

ficients can be formed by swapping sides and taking logarithms of both sides of Equation A.2.1 as shown in Equation A.3.1.

$$L_0 \ln(T_0) + L_1 \ln(T_1) + \dots + L_I \ln(T_I) = L_{max}(\ln(p_j - C_j) - \ln(E_j)). \quad (\text{A.3.1})$$

Thus, the ray traces through the voxel space generate a set of linear equations as shown in Equation A.3.2.

$$A_{(M,N)}x_{(N,1)} = b_{(M,1)}. \quad (\text{A.3.2})$$

Matrix $A_{(M,N)}$ represents the voxel traversal length contributions L_i to each pixel p_j in a set of P images. Vector $x_{(N,1)}$ represents the voxel transmittance values ($\ln(T_i)$ from Equation A.3.1) to be solved. The right hand side of Equation A.3.2, $b_{(M,1)}$, represents the values of the pixel and camera constants from all of the P images.

The dimensions of the reconstruction problem come from the constraints of the input data. Matrix A is of dimension $M \times N$. $M = Pm^2$ where P is the number of images and m is the number of pixels along each edge of the image region that is used in the reconstruction. $N = v_0^3$ where v_0 is the dimension of the voxel space (shown as 5 in Figure A.40).

$$A_{m,n} = \begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \cdots & a_{m,n} \end{pmatrix} \quad (\text{A.3.3})$$

A is a highly sparse matrix. The slots defined by Equation A.3.1 are filled with non-zero values only when a voxel contributes to a pixel. Each pixel ray trace can transverse at most $\sqrt{2v^2 + v^2} = \sqrt{3}v$ voxels so the fill factor is at most $\sqrt{3}v/v^3 = \sqrt{3}/v^2$. The voxel dimension, v , is expected to be at least 2^5 so for this case, the fill factor has an upper bound of $\sqrt{3}/2^{10} \approx 2^{-8}$ or 0.4%.

The dimensions of A , and thus its solution potential, are strongly influenced by the image acquisition constraints. The goal is to achieve the maximum resolution

in the reconstruction. For *cryo-tem* imagery, the number of pixels in the images and the number of images both lead to sample degradation. Pixel count equates to magnification, which means more energy per unit area in the acquisition for each image. The number of images leads to more total exposure for the sample region. Both issues lead to sample degradation so a balance must be struck between the physical limitations of the *cryo-tem* technique and the desired reconstruction resolution.

The goal of maximizing reconstruction resolution leads to the desire of having a square system of linear equations to solve. Many of the sparse matrix solvers require a square system and efficiencies can also be found with square systems. Other solvers place constraints on the dimensions of A (e.g. for singular value decomposition (SVD), $M \geq N$). Thus, the size of the image m and the voxel space v can be selected to help optimize the solution computational load. A set of sizes is shown in Table A.5. It is impractical to achieve more than 5 or 6 *cryo-tem* images of a single target and the later rows in Table A.5 show that the achievable reconstruction resolution does not increase with more images. Thus, the reconstruction resolution for 4 images represents an achievable trade-off between magnification and images taken.

P	m	$M = Pm^2$	v	$N = v^3$	Methods
2	2^7	2^{15}	2^5	2^{15}	Square
3	2^7	$\frac{3}{2}2^{15}$	2^5	2^{15}	$M > N$
4	2^8	2^{18}	2^6	2^{18}	Square
5	2^7	$\frac{5}{4}2^{16}$	2^5	2^{15}	$M > N$
6	2^7	$\frac{6}{4}2^{16}$	2^5	2^{15}	$M > N$
7	2^7	$\frac{7}{4}2^{16}$	2^5	2^{15}	$M > N$
8	2^7	2^{19}	2^6	2^{18}	$M > N$

Table A.5. Matrix Solution Parameters

As was noted above, preliminary research indicates that the information *is* available, how to glean the information is the question. Stability with such sparse

matrices implies that this is not the way. While it may be more tractable than NP-complete correlation searching, this method is not expected to provide any meaningful results. The idea that does arise from this discussion, however, is that a way of “walking” through the voxels should provide the means by which the voxel densities could be found. Hidden Markov modeling is such a way to traverse the voxels and determine the densities.

APPENDIX B

Constructions

This appendix provides details of applicable constructions and derivations.

B.1 Image Rotation Limits

The construction diagram that shows the projection image rotation limits is shown in Figure B.45. This construction follows the generation of projection images as shown in Figure A.40. The dashed orange line represents the diagonal

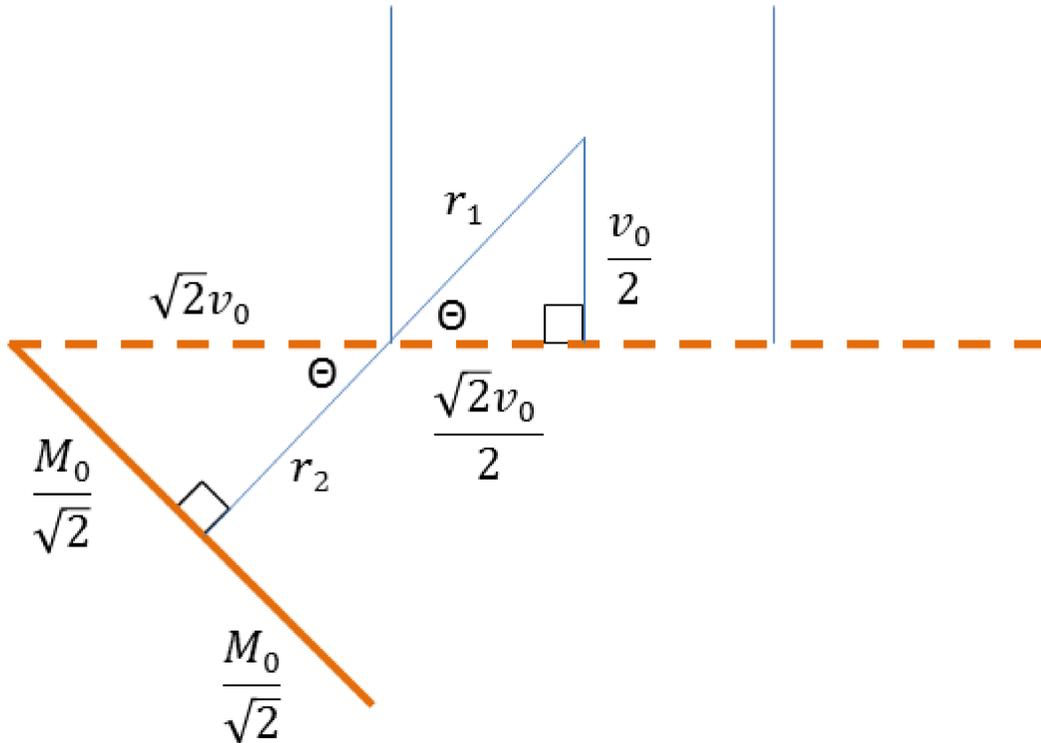


Figure B.45. Image Rotation Limits

along the bottom of the extended voxel space from Figure A.40 and the solid orange line represents the diagonal from the $M_0 \times M_0$ extended projection image. Each segment of the dashed line is $\sqrt{2}v_0$ long so the first part r_1 of the rotation

radius from the center of the projection image to the center of the voxel space is given by B.1.1 as

$$r_1 = \sqrt{\left(\frac{v_0}{2}\right)^2 + \left(\frac{\sqrt{2}v_0}{2}\right)^2} = \frac{\sqrt{3}}{2}v_0. \quad (\text{B.1.1})$$

The common angle, Θ is $\arctan\left(\frac{v_0/\sqrt{2}}{v_0/2}\right) = \arctan\left(\frac{1}{\sqrt{2}}\right) \approx 35^\circ$, which, of course, is not 45° since the sides of the triangle of the center voxel region are not congruent.

The lower part of the radius is given by B.1.2 as

$$r_2 = \sqrt{(\sqrt{2}v_0)^2 - \left(\frac{M_0}{\sqrt{2}}\right)^2} = \frac{v_0}{\sqrt{2}}, \quad (\text{B.1.2})$$

where M_0 is replaced by $\sqrt{3}v_0$ which also equals $\sqrt{3}m_0$ since $|m_0| \equiv |v_0|$. The minimum allowed radial distance, R_m , that the center of the projection image can be from the center of the voxel space so that the edge pixels do not interfere with the extended voxel space, there are given by B.1.3 as

$$R_m = r_1 + r_2 = \frac{\sqrt{3}}{2}v_0 + \frac{v_0}{\sqrt{2}} = v_0 \left(\frac{\sqrt{3}}{2} + \frac{1}{\sqrt{2}} \right). \quad (\text{B.1.3})$$

But R_m is a minimum value which is $\approx v_0 1.573$ so any value larger than this will push the projection image far enough away so that the edge pixels do not interfere with the edge voxels.

B.2 Tuple To Tuple Probabilities

The tuples in the lists become the states in the HOHMM processing. The transition probabilities need to be determined in order for the extended HMM processing to follow the standard $\alpha, \beta, \text{ and } \gamma$ operations. When the tuples are determined, the probabilities were calculated and normalized. But, since the tuple order does not determine tuple to tuple state changes, they must be determined separately based on how the tuples were determined, in order to provide the overall A matrix for the extended HMM processing. An example set of states and tuples is provided in Table B.6. The left side of the table shows a continuous stream of

ID	Tuple	State	State	Tuple
1	-	5	5	-
2	-	4	4	-
3	5,0 - 4,0 : 3	3	3	5,0 - 4,0 : 3
4	4,0 - 3,0 : 3	3	3	4,0 - 3,0 : 3
5	4,0 - 3,1 : 3	3	3	4,0 - 3,1 : 3
6	4,0 - 3,2 : 3	3	3	4,0 - 3,2 : 3
7	4,0 - 3,3 : 3	3	3	4,0 - 3,3 : 3
8	4,0 - 3,4 : 3	3	2	4,0 - 3,4 : 2
9	4,0 - 3,5 : 3	3	2	3,4 - 2,0 : 2
10	4,0 - 3,6 : 3	3	2	3,4 - 2,1 : 2
11	4,0 - 3,7 : 3	3	1	3,4 - 2,2 : 1
12	4,0 - 3,8 : 3	3	1	2,2 - 1,0 : 1
13	4,0 - 3,9 : 3	3	2	2,2 - 1,1 : 2
14	4,0 - 3,10 : 3	3	2	1,1 - 2,0 : 2
15	4,0 - 3,11 : 3	3	2	1,1 - 2,1 : 2
16	4,0 - 3,12 : 3	3	3	1,1 - 2,2 : 3
17	4,0 - 3,13 : 3	3	3	2,2 - 3,0 : 3
18	4,0 - 3,14 : 3	3	3	2,2 - 3,1 : 3
19	4,0 - 3,15 : 3	3	3	2,2 - 3,2 : 3
20	4,0 - 3,16 : 3	3	3	2,2 - 3,3 : 3

Table B.6. Example Tuple to Tuple Chains

state = 3 tuples while the right side shows a chain with some changing states. The first two rows show the *imaginary* states outside of the reconstruction space. This example shows two levels of history. The tuples are denoted by state-length tuples and the current state of the form {state,length - state,length : state}.

For the A matrix generation, only states that can occur from the model are allowed. For example, tuple 5 cannot follow tuple 3 even though they are the same state. The from tuples do not align so the transition cannot occur. Similarly, tuple 9 cannot follow tuple 7 for the same reason. Tuple 8, however, *can* follow tuple 7 in either of the state chains. In the left chain, the state remains the same but the length-state tuples are consistent. In the right chain, the tuples are also consistent even though the state changes. By checking the allowable tuple transitions, and by using the computed probabilities, the A matrix that describes the tuple to tuple transition probabilities can be created.

The probabilities computed for the tuples give rise to the tuple to tuple probabilities. The allowable transitions between tuples determines the probabilities for the A matrix for the extended processing. This algorithm is shown below.

```

dReturnProbability = 0.0;
if (nFromTupleIndex == nToTupleIndex)
{
    dReturnProbability = 0.0;
}
else if ( From.state == To.State)
{
    if (To.Level[0].Length>0)
    {
        if ( (From.Level[0].State == To.Level[0].State) &&
            (((From.Level[0...L-2] == To.Level[1...L-1]) &&
              (To.Level[0].State == To.State)) ||
            ((From.Level[1...L-1] == To.Level[1...L-1]) &&
            (From.Level[0].Length+1 == To.Level[0].Length) )))
        {
            dReturnProbability = To.dProbability
        }
    }
}

```

```

    }
    else // To.Level[0].Length==0
    {
        if ( ((From.Level[0...L-2] == To.Level[1...L-1]) &&
            (To.Level[0].State == To.State)) ||
            ((From.Level[1...L-1] == To.Level[1...L-1]) &&
            (From.Level[0].Length+1 == To.Level[0].Length) ))
        {
            dReturnProbability = To.dProbability
        }
    }
}
else // state change
{
    if ( (From.Level[1...L-1] == To.Level[1...L-1]) &&
        (To.Level.[0].State == From.Level[0].State) &&
        (To.Level[0].Length + 1 == From.Level[0].Length) )
    {
        dReturnProbability = To.dProbability
    }
}
return dReturnProbability;

```

This function returns a previously computed and stochastic probability value or zero if the tuple to tuple transition is not possible.

Index

- CTEM, ii–iv, 1, 7, 9–12, 15, 17, 18, 21–23, 25–27, 29–31, 33, 37, 38, 41, 43, 58, 59, 105, 111–114, 125, 129, 133, 136, 140, 141, 143, 145
- recognition-by-components, 35, 37–39, 108
- signal-to-noise, 26, 28
- singular value decomposition, 145
- first order HMM, 46, 61, 64, 67, 71, 87–89, 91–93
- SPR, 1, 22, 23, 25, 28, 42, 43, 111, 112, 114
- superquadric, 34, 36, 37, 39
- SVM, 58
- Fourier shell correlation, 29
- geon, 34–38
- HDL, 8, 9
- TEM, 7, 21–24, 29, 125
- higher order HMM, 58, 61, 64, 66, 70, 71, 75, 88, 93, 95, 148
- HMM, iv, 45, 46, 49, 58, 59, 61–69, 71, 73, 74, 76, 79, 80, 86, 88, 91–95, 97, 99, 101–104, 106, 108, 109, 112, 113, 136, 148
- IPR, ii–iv, 1, 108–115
- LDL, iv, v, 8, 9, 15, 27, 29, 31, 41, 93, 106, 113, 114, 135
- lipoprotein, 8–10
- Markov chain, 61, 63, 92
- random conical tilt, 26, 139

BIBLIOGRAPHY

- A Broadhurst and T.W Drummond and R Cipolla, “A Probabilistic Framework for Space Carving,” in *Proc Int Conf on Computer Vision*, 2001.
- A. H. Barr, “Superquadrics and Angle-Preserving Transformations,” *IEEE Computer Graphics and Applications*, vol. 1, no. 1, pp. 11–23, 1981.
- A. J. Viterbi, “Error Bounds for Convolutional Codes and an Asymptotically Optimal Decoding Algorithm,” in *IEEE Transactions on Information Theory*, 1967, pp. 260–269.
- A. P. Pentland, “Perceptual Organization and the Representation of Natural Form,” *Artificial Intelligence*, vol. 28, no. 2, pp. 293–331, 1986.
- Ales Leonardis and Ales Jaklic and Franc Solina, “Superquadrics for Segmenting and Modeling Range Data,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 11, pp. 1289–1295, 1997.
- Annie Yao and Andrew Calway, “Interpolating Novel Views from Image Sequences by Probabilistic Depth Carving,” in *ECCV 2004*, May 2004, pp. 379–390.
- B.G. Baumgart, “Geometric Modeling for Computer Vision,” Ph.D. dissertation, Stanford, 1997.
- Borislav Trifonov and Derek Bradley and Wolfgang Heidrich, “Tomographic Reconstruction of Transparent Objects,” in *Proceedings of Eurographics Symposium on Rendering*, June 2006, pp. 51–60.
- Bridget Carragher and Nick Kisseberth and David Kriegman and Ronald A. Milligan and Clinton S. Potter and James Pulokas and Amy Reilein, “Legion: An Automated System for Acquisition of Images from Vitreous Ice Specimens,” in *Journal of Structural Biology*, Oct. 2000, pp. 33–45.
- D. Marr and T. Poggio, “Cooperative Computation of Stereo Disparity,” *Science*, vol. 194, no. 4262, pp. 283–287, Oct. 1976.
- D. Scharstein and R. Szeliski, “A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms,” *International Journal of Computer Vision*, vol. 47, pp. 7–42, 2002.
- Daphna Weinshall and Michael Werman, “On View Likelihood and Stability,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 2, pp. 97–106, 1997.

- David J. Kriegman and Jean Ponce, “On Recognizing and Positioning Curved 3-D Objects from Image Contours,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 12, pp. 1127–1137, 1990.
- Davis, R.I.A. and Lovell, B.C. and Caelli, T., “Improved Estimation of Hidden Markov Model Parameters from Multiple Observation Sequences,” in *Proceedings, 16th International Conference on Pattern Recognition*, vol. 2, 2002, pp. 168–171.
- Demetri Terzopoulos and Dimitri Metaxas, “Dynamic 3D Models with Local and Global Deformations: Deformable Superquadrics,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 7, pp. 703–714, 1991.
- Dewar, M. and Wiggins, C. and Wood, F., “Inference in Hidden Markov Models with Explicit State Duration Distributions,” *Signal Processing Letters, IEEE*, vol. 19, no. 4, pp. 235–238, 2012.
- Douglas Trumbull, “Immersive Cinema Technology,” *SID Symposium Awards Luncheon*, 2011.
- Elena V. Orlova and Michael B. Sherman and Wah Chiu and Hiro Mowri and Louis C. Smith and Antonio M. Grotto Jr., “Three-dimensional Structure of Low Density Lipoproteins by Electron Cryomicroscopy,” in *Proceedings of the National Academy of Science USA*, July 1999, pp. 8420–8425.
- Elias J. Jeyarajah and William C. Cromwell and James D. Otvos, “Lipoprotein Particle Analysis by Nuclear Magnetic Resonance Spectroscopy,” in *Clinics in Laboratory Medicine*, Dec. 2006, pp. 847–870.
- F. Jelinek, “Continuous Speech Recognition by Statistical Methods,” in *Proceedings of the IEEE*, 1976, pp. 532–536.
- Franc Solina and Ruzena Bajcsy, “Recovery of Parametric Models from Range Images: The Case for Superquadrics with Global Deformations,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 2, pp. 131–147, 1990.
- G. D. Forney, “The Viterbi Algorithm,” in *Proceedings of the IEEE*, 1973, pp. 268–278.
- G. Humphrey, “The Psychology of the Gestalt,” *Journal of Educational Psychology*, vol. 15, no. 7, pp. 401–412, 1924.
- Gang Ren and Gabby Rudenko and Steven J. Ludtke and Johann Deisenhofer and Wah Chiu and Henery J. Pownall, “Model of Human Low-Density Lipoprotein and Bound Receptor Based on CryoEM,” in *PNAS*, Jan. 2010, pp. 1059–1064.

- Giblin, Peter and Weiss, Richard, “Reconstruction of Surfaces from Profiles,” University of Massachusetts, Amherst, MA, USA, Tech. Rep., 1987.
- Greenshields, Ian R., “Extended Superquadric Models of Elastic Hollow Viscera,” in *Proceedings of the 12th IEEE Symposium on Computer-Based Medical Systems*, ser. CBMS '99. Washington, DC, USA: IEEE Computer Society, 1999, pp. 98–102. [Online]. Available: <http://dl.acm.org/citation.cfm?id=791226.792105>
- H. C. Longuet-Higgins, “A Computer Algorithm for Reconstructing a Scene from Two Projections,” *Nature*, vol. 293, pp. 133–135, Sept. 1981.
- Haim Schweitzer and Sanjeev R. Kulkarni, “Computational Limitations of Model-Based Recognition,” *International Journal of Intelligent Systems*, vol. 9, pp. 431–443, 1998.
- Harold P. Erickson and A. Klug, “The Fourier Transform of an Electron Micrograph: Effects of Defocusing and Aberrations, and Implications for the use of Underfocus Contrast Enhancement,” in *Berichte der Bunsengesellschaft für physikalische Chemie*, Nov. 1970, pp. 1129–1137.
- Harvard University. “The Central Section Theorem.” Last visited on 31 Dec 2013. Feb. 2010. [Online]. Available: https://software.rc.fas.harvard.edu/leschziner_public/index.php/Central_Section_Theorem
- Irving Biederman, “Recognition-by-Components: A Theory of Human Image Understanding,” *Psychological Review*, vol. 94, no. 2, pp. 115–147, 1987.
- Irving Biederman, “Recognizing Depth-Rotated Objects: A Review of Recent Research and Theory,” *Spatial Vision*, vol. 13, no. 3, pp. 241–253, 2001.
- Ivo Ihrke and Kiriakos N. Kutulakos and Hendrik P. A. Lensch and Marcus Magnor and Wolfgang Heidrich, “State of the Art in Transparent and Specular Object Reconstruction,” *EUROGRAPHICS 2008*, 2008.
- J. Fessler, *Chapter 3. Analytical Tomographic Image Reconstruction Methods*. web.eecs.umich.edu/~fessler/course/516/l/c-tomo.pdf, 2009, Last visited on 31 Dec 2013.
- James Otvos and R. Marabini and I.M. Masegosa and M.C. San Martn and S. Marco and J. Fernandez and L. G. de la Fraga and C. Vaquerizo and J.M. Carazo, “Radio Signals Give New Spectrum For Cholesterol Lipoprotein Readings,” in *Arteriosclerosis, Thrombosis and Vascular Biology: Journal of the American Heart Association*, July 1998, pp. 2223–2231.
- James Zhao, “Hidden Markov Models with Multiple Observation Processes,” *Honours Thesis*, Nov. 1980.

- Jeremy S. De Bonet and Paul Viola, “Poxels: Probabilistic Voxalized Volume Reconstruction,” in *Proceedings of ICCV*, Sept. 1999, pp. 391–398.
- Jiehua Zhu and Shiyong Zhao and Yangbo Ye and Ge Wang, “Computed Tomography Simulation with Superquadrics,” *Medical Physics*, vol. 32, no. 10, pp. 3136–3143, Oct. 2005.
- J.J. Fernández and J.I. Agulleiro and J.R. Bilbao-Castro and A. Martínez and I. Garca and F.J. Chichón and J. Martín-Benito and J.L. Carrascosa, “Image Processing in Electron Tomography,” in *Microscopy: Science, Technology, Applications and Education: A. Mendez-Vilas and J. Diaz (Eds.)*, 2010, pp. 19–28.
- Joachim Frank, “Single-Particle Reconstruction of Biological Macromolecules in Electron Microscopy - 30 years,” in *Q Rev Biophys*, Aug. 2009, pp. 139–158.
- Johan A. du Preez, “Efficient High-Order Hidden Markov Modelling,” *Doctorial Dissertation*, Nov. 1997.
- Juha Vakkilainen and George Steiner and Jean-Claude Ansquer and Francois Aubin and Stephanie Rattier and Christelle Foucher and Anders Hamsten and Marja-Riitta Taskinen, “Relationships Between Low-Density Lipoprotein Particle Size, Plasma Lipoproteins, and Progression of Coronary Artery Disease: The Diabetes Atherosclerosis Intervention Study (DAIS),” in *Circulation (www.circulationaha.org)*, Mar. 2003, pp. 1733–1737, Last visited on 31 Dec 2013.
- K. J. Hanszen, “The Optical Transfer Theory of the Electron Microscope: Fundamental Principles and Applications,” in *Advances in Optical and Electron Microscopy*. Academic Press, New York, 1971, pp. 1–84.
- K. L. Turabian, *A Manual for Writers of Term Papers, Theses, and Dissertations, 6th. edn.* Chicago, Illinois, United States of America: University of Chicago Press, 1987.
- Kiriakos N. Kutulakos and Steven M. Seitz, “A Theory of Shape by Space Carving,” *International Journal of Computer Vision*, vol. 38, no. 3, pp. 199–218, 2000.
- Kong-man (German) Cheung and Simon Baker and Takeo Kanade, “Shape-From-Silhouette Across Time Part I: Theory and Algorithms,” *International Journal of Computer Vision*, vol. 62, no. 3, pp. 221–247, Mar. 2005.
- Larry N. Thibos, “Image Processing by the Human Eye,” in *Visual Communications and Image Processing IV, William A. Pearlman (ed), Proc SPIE 1199*, 1989, pp. 1148–1153.

- Laurentini, A., “The Visual Hull Concept for Silhouette-Based Image Understanding,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 2, pp. 150–162, Aug. 1994.
- Lawrence R. Rabiner, “A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition,” in *Proceedings of the IEEE*, 1989, pp. 257–286.
- Lee-Min Lee, “High-Order Hidden Markov Model and Application to Continuous Mandarin Digit Recognition,” *Journal of Information Science and Engineering*, vol. 27, pp. 1919–1930, 2011.
- M. Agrawal and L.S. Davis, “A Probabilistic Framework for Surface Reconstruction From Multiple Images,” in *Proc Conf on Computer Vision and Pattern Recognition*, 2001, pp. II470–II476.
- M Beer and J Frank and K J Hanszen and E Kellenberger and RC Williams, “The Possibilities and Prospects of Obtaining High-Resolution (below 30 Å) on Biological Material Using the Electron Microscope. Some Comments and Reports Inspired by an EMBO Workshop Held at Gais, Switzerland, October 1973,” in *Quarterly Reviews of Biophysics*, vol. 7, Aug. 1974, pp. 211–238.
- M. Radermacher and T. Wagenknecht and A. Verschoor and J. Frank, “Three-dimensional Reconstruction from a Single-Exposure, Random Conical Tilt Series Applied to the 50S Ribosomal Subunit of Escherichia coli,” in *Journal of Microscopy*, May 1987, pp. 113–136.
- M. Tarr and S. Pinker, “Mental Rotation and Orientation – Dependence in Shape Recognition,” *Cognitive Psychology*, vol. 21, pp. 233–282, 1989.
- M. van Heel and E. V. Orlova and G. Harauz and H. Stark and P. Dube and F. Zemlin and M. Schatz, “Angular Reconstitution in Three-Dimensional Electron Microscopy: Historical and Theoretical Aspects,” in *Scanning Microscopy Vol. 11*, 1997, pp. 195–210.
- M. Z. Brown and D. Burschka and G. D. Hager, “Advances in Computational Stereo,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 8, pp. 993–1008, 2003.
- Marin van Heel and Brent Gowen and Rishi Matadeen and Elena V. Orlova and Robert Finn and Tillmann Pape and Dana Cohen and Holger Stark and Ralf Schmidt and Michael Schatz and Ardan Patwardhan, “Single-Particle Electron Cryo-Microscopy: Towards Atomic Resolution,” in *Quarterly Reviews of Biophysics*, Apr. 2000, pp. 307–369.
- Mark Stamp, “A Revealing Introduction to Hidden Markov Models,” San Jose State University, San Jose, CA, USA, Tech. Rep., Sept. 2012, www.cs.sjsu.edu/~stamp/RUA/HMM.pdf Last visited on 31 Dec 2013.

- Michael Sipser, *Introduction to the Theory of Computation, 2nd. edn.* Boston, Massachusetts, United States of America: Thomson Course Technology, 2006.
- Mohammed J. Zaki and Christopher D. Carothers and Boleslaw K Szymanski, “VOGUE: A Variable Order Hidden Markov Model with Duration Based on Frequent Sequence Mining,” *ACM Transactions on Knowledge Discovery From Data*, vol. 4, Jan. 2010.
- Navdeep Jaitly and Marcus A. Brubaker and John L. Rubinstein and Ryan H. Lilien, “A Bayesian Method for 3D Macromolecular Structure Inference using Class Average Images from Single Particle Electron Microscopy,” *Bioinformatics*, vol. 26, no. 19, pp. 2406–2415, Oct. 2010, Last visited on 31 Dec 2013. [Online]. Available: <http://dx.doi.org/10.1093/bioinformatics/btq456>
- Neus Sabater and Andrés Almansa and Jean-Michel Morel, “Meaningful Matches in Stereovision,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 5, pp. 930–942, 2012.
- NHLBI, “Heart Diseases,” National Institutes of Health, Tech. Rep., Mar. 2013, <http://www.nhlbi.nih.gov/health/health-topics/topics/cad/> Last visited on 31 Dec 2013.
- P. A. Midgley and M. Weyland, “3D Electron Microscopy in the Physical Sciences: The Development of Z-contrast and EFTEM Tomography,” *Ultramicroscopy*, vol. 96, pp. 413–431, 2002.
- P.A. Penczek and J. Zhu and R. Schröder and J. Frank, “Three Dimensional Reconstruction with Contrast Transfer Compensation from Defocus Series,” in *Scanning Microscopy*. Scanning Microscopy International, 1997, pp. 147–154.
- Patrick Labatut and Jean-Philippe Pons and Renaud Keriven, “Efficient Multi-View Reconstruction of Large-Scale Scenes Using Interest Points, Delaunay Triangulation and Graph Cuts,” in *ICCV*, 2007, pp. 1–8.
- Pengcheng Liu and Andrew Willis and Yunfeng Sui, “Stereoscopic 3D Reconstruction Using Motorized Zoom Lenses Within an Embedded System,” *Image Processing: Machine Vision Applications II*, edited by Kurt S. Niel and David Fofi, *Proceedings of SPIE-IS&T Electronic Imaging*, vol. 7251, pp. 72 510W1–72 510W12, 2009.
- Petra A. Arndt and Hanspeter A. Mallot and Heinrich H. Bülthoff, “Human Stereovision Without Localized Image Features,” *Biological Cybernetics*, vol. 72, pp. 279–293, 1995.
- Prince, S.J.D., *Computer Vision: Models Learning and Inference*. Cambridge University Press, 2012.

- R. Bakis, "Continuous Speech Word Recognition via Centi-Second Acoustic States," in *Proceedings of ASA Meeting (Washington, DC)*, 1976.
- R. Henderson, "The Potential and Limitations of Neutrons, Electrons, and X-rays for Atomic Resolution Microscopy of Unstained Biological Macromolecules," in *Quarterly Review of Biophysics*, May 1995, pp. 171–193.
- R. L. Cave and L. P. Neuwirth, "Hidden Markov Models for English," *Hidden Markov Models for Speech*, Oct. 1980.
- Ralph D. Freeman, "Stereoscopic Vision: Which Parts of the Brain are Involved?" *Current Biology*, vol. 9, pp. R610–R613, 1999.
- Rik van Antwerpen and G. Chi Chen and Clive R. Pullinger and John P. Kane and Michael LaBelle and Ronald M. Krauss and Cesar Luna-Chavez and Trudy M. Forte and John C. Gilkey, "Cryo-Electron Microscopy of Low Density Lipoprotein and Reconstituted Discoidal High Density Lipoprotein: Imaging of the apoLipoprotein Moiety," in *Journal of Lipid Research*, Apr. 1997, pp. 659–669.
- Rik van Antwerpen and John C. Gilkey, "Cryo-Electron Microscopy Reveals Human Low Density Lipoprotein Substructure," in *Journal of Lipid Research*, Dec. 1994, pp. 2223–2231.
- Rik van Antwerpen and Michael LaBelle and Edita Navratilova and Ronald M. Krauss and Cesar Luna-Chavez and Trudy M. Forte and John C. Gilkey, "Structural Heterogeneity of apoB-Containing Lipoproteins Visualized using Cryo-Electron Microscopy," in *Journal of Lipid Research*, Oct. 1999, pp. 1827–1836.
- S. Edelman and Heinrich H. Bülhoff and D. Weinshall, "Stimulus Familiarity Determines Recognition Strategy for Novel 3D Objects," Massachusetts Institute of Technology, Boston, MA, USA, Tech. Rep., 1989.
- S. Seitz and B. Curless and J. Diebel and D. Scharstein and R. Szeliski, "A Comparison and Evaluation of Multi-View Stereo Reconstruction Algorithms," *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 519–526, 2006.
- Samuel W. Hasinoff and Kiriakos N. Kutulakos, "Photo-Consistent Reconstruction of Semi-Transparent Scenes by Density Sheet Decomposition," in *PAMI*, May 2007, pp. 870–885.
- Shun-Zheng Yu and Hasashi Kobayashi, "A Hidden Semi-Markov Model with Missing Data and Multiple Observation Sequences for Mobility Tracking," *Signal Processing*, pp. 235–250, Dec. 2001.

- S.J. Thorpe and A. Delorme and R. Van Rullen and W. Paquier, “Reverse Engineering of the Visual System Using Networks of Spiking Neurons,” in *IEEE International Symposium on Circuits and Systems*, 2000, pp. 405–408.
- S.M. Seitz and C.R. Dyer, “Photorealistic Scene Reconstruction by Voxel Coloring,” in *CVPR97*, 1997, pp. 1067–1073.
- Steven J. Ludtke and Philip R. Baldwin and Wah Chiu, “EMAN: Semiautomated Software for High-Resolution Single-Particle Reconstructions,” in *Journal of Structural Biology*, Dec. 1999, pp. 82–97.
- Sven J. Dickinson and Alex P. Pentland and Azriel Rosenfeld, “3-D Shape Recovery Using Distributed Aspect Matching,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 174–198, 1992.
- Sven J. Dickinson and Dimitri Metaxas and Alex P. Pentland, “The Role of Model-Based Segmentation in the Recovery of Volumetric Parts From Range Data,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 3, pp. 259–267, 1997.
- Sven J. Dickinson and Robert Bergevin and Irving Biederman and Jan-Olof Eklundh and Roger Munck-Fairwood and Anil K. Jain and Alex Pentland, “The Potential of Geons for Generic 3-D Object Recognition,” *IMAGE AND VISION COMPUTING*, vol. 15, pp. 277–292, 1997.
- Svetlana Lazebnik and Yasutaka Furukawa and Jean Ponce, “Projective Visual Hulls,” in *International Journal of Computer Vision*, Aug. 2007, pp. 137–165.
- Thede, Scott M. and Harper, Mary P., “A Second-Order Hidden Markov Model for Part-of-Speech Tagging,” in *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, ser. ACL ’99. Stroudsburg, PA, USA: Association for Computational Linguistics, 1999, pp. 175–182. [Online]. Available: <http://dx.doi.org/10.3115/1034678.1034712>
- Tom Teerlink and Peter G. Scheffer and Stephan J. L. Bakker and Robert J. Heine, “Combined Data from LDL Composition and Size Measurement are Compatible With a Discoid Particle Shape,” in *Journal of Lipid Research*, May 2004, pp. 954–966.
- University of California San Diego. “Cryoelectron Microscopy at USCD.” Last visited on 31 Dec 2013. June 2010. [Online]. Available: <http://cryoem.ucsd.edu/>
- University of California San Diego. “The Electron Microscopy Outreach Program.” Last visited on 31 Dec 2013. June 2010. [Online]. Available: <http://em-outreach.ucsd.edu/index.html>

- University of Rhode Island. “A Guide to Producing Your Thesis with LaTeX.” Last visited on 28 Dec 2013. June 2006. [Online]. Available: <http://www.ele.uri.edu/info/thesis/guide>
- W. Lewis Collier, “Acquisition and Analysis of Cryogenic Transmission Electron Microscope Biological Images,” in *URI CSC-592*, May 2005, pp. 1–22.
- W. Lewis Collier, “SVM Analysis of Cryogenic Transmission Electron Microscopy Images,” in *URI CSC-581, Topics in AI: Support Vector Machines, Spring 2007, Final Project*, May 2007, pp. 1–22.
- W. Lewis Collier and Jean-Yves Hervé and Lenore M. Martin, “Towards Individual Particle Reconstruction from Cryogenic Transmission Electron Microscopy,” in *2013 ACM Conference on Bioinformatics, Computational Biology and Biomedical Informatics*, ser. ACM-SIGBio-BIB-2013. 2 Penn Plaza, Suite 701, New York New York 10121-0701, USA: Association for Computing Machinery, Sept. 2013, pp. 526–535.
- W. Lewis Collier and Lenore M. Martin and Rik van Antwerpen, “Accuracy Limits for Efficiently Determining Shape and Size of Low-Density Lipoprotein Macromolecules from Cryogenic TEM Images,” in *SPIE Medical Imaging Conference*, vol. 6511-97, Feb. 2007, pp. 1–9.
- W. Lewis Collier and Lenore M. Martin and Rik van Antwerpen, “Computer Vision Determination of 3-D Geometric Parameters of LDL Particles via Cryogenic Transmission Electron Microscopy,” in *Journal of the Society for Information Display*, Aug. 2007, pp. 391–398.
- Wikipedia. “Fourier shell correlation.” Last visited on 31 Dec 2013. Aug. 2012. [Online]. Available: http://en.wikipedia.org/wiki/Fourier_shell_correlation
- Wikipedia : The Free Encyclopedia. “Bilinear interpolation.” Last visited on 31 Dec 2013. Sept. 2012. [Online]. Available: http://en.wikipedia.org/wiki/Bilinear_interpolation
- www.imagescience.de. “Image Science: Advanced Image Processing & Image Analysis.” Last visited on 31 Dec 2013. Aug. 2012. [Online]. Available: <http://www.imagescience.de/imagic.html>
- www.wadsworth.org. “Contrast Transfer Function (CTF) Correction.” Last visited on 31 Dec 2013. Aug. 2012. [Online]. Available: http://www.wadsworth.org/spider_doc/spider/docs/techs/ctf/ctf.html
- Xiaolin Li and Marc Parizeau and Rejean Plamondon, “Training Hidden Markov Models with Multiple Observations – A Combinatorial Method,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Apr. 2000.

- Y. Dong and G. R. Hillman, “Three-Dimensional Reconstruction of Irregular Shapes Based on a Fitted Mesh of Contours,” *Image and Vision Computing*, vol. 19, pp. 165–176, 2001.
- Yan Zhang and Andreas Koschan and Mongi Abidi, “Superquadrics Based 3D Object Representation of Automotive Parts Utilizing Part Decomposition,” in *Proceedings of SPIE 6th International Conference on Quality Control by Artificial Vision*, May 2003, pp. 241–251.