

OBSERVER-DRIVEN TEXTURE ANALYSIS IN CT IMAGING

by

Zachary Taylor Garrett

Copyright © Zachary Taylor Garrett 2020

A Thesis Submitted to the Faculty of the
JAMES C. WYANT COLLEGE OF OPTICAL SCIENCES
In Partial Fulfillment of the Requirements
For the Degree of
MASTER OF SCIENCE
In the Graduate College of
THE UNIVERSITY OF ARIZONA

2020

THE UNIVERSITY OF ARIZONA
GRADUATE COLLEGE

As members of the Master's Committee, we certify that we have read the thesis prepared by **Zachary Taylor Garrett**, titled *Observer-Driven Texture Analysis in CT Imaging* and recommend that it be accepted as fulfilling the dissertation requirement for the Master's Degree.



Professor Matthew A. Kupinski, Chair

Date: April 28, 2020



Professor Lars R. Furenlid

Date: April 28, 2020




Professor Eric W. Clarkson

Date: April 28, 2020

Final approval and acceptance of this thesis is contingent upon the candidate's submission of the final copies of the thesis to the Graduate College.

I hereby certify that I have read this thesis prepared under my direction and recommend that it be accepted as fulfilling the Master's requirement.



Professor Matthew A. Kupinski
Master's Thesis Committee Chair
Wyant College of Optical Sciences

Date: April 30, 2020

ARIZONA

ACKNOWLEDGEMENTS

Thank you, first and foremost, to Dr. Matthew Kupinski for taking me under his wing, despite me only wanting to stick around for two years, instead of agreeing to some indefinite time. I would also like to thank Dr. Eric Clarkson for all his help in and out of the classroom, as well as for the lunch breaks to help keep me from pulling my hair out on Wednesdays. Also, thank you to Dr. Lars Furenlid for hearing me out early on, being willing to take me on if Dr. Kupinski hadn't gotten to me first, as well as teaching what is surely the most comprehensive optics course I've ever taken. One more thanks to the aforementioned professors for dealing with me in my undoubtedly most exhausted and stressed state. I would like to also thank Dr. Jim Schwiegerling for keeping me busy outside of class during undergrad and providing me with opportunities to perform and share research (in medical imaging, no less) earlier than I would have expected I would be capable of.

DEDICATION

To my dearest: Courtney Garrett. For being the most honest, loving, and supportive person I've ever met. Thank you for everything – a word I use in its absolute sense because it's all you've ever given to me.

The family that's always been there for me: Mom, Dad, Grandma, and Shennen.

Those that kept me sane and capable of getting through a decade of school: Carlos, Ash, Geoff, Andy, Ed, Nathan, Chris, Sean, Pat, Jilian, and countless others.

*The best companion an emotional train wreck could ever have found: Snickers Zhu-Li Fresnel Stormborn Goddard Carrot, Duchess of Hambridge, Queen of the Andogs and the First Pugs
Lastly, the one who proved himself wrong time and time again: Myself.*

Table of Contents

LIST OF FIGURES	7
LIST OF TABLES	8
ABSTRACT	9
CHAPTER 1 Introduction and Motivations	10
1.1 A Brief History of Computed Tomography (CT)	10
1.2 CT Imaging Process	12
1.2.1 Tomography.....	12
1.2.2 Projections	13
1.2.3 Image Reconstruction	15
1.2.4 A Brief Summary.....	18
1.3 How the Observer Fits In	18
CHAPTER 2 The Model Observer	19
2.1 What is a Model Observer?.....	20
2.2 Statistical Decision Theory	20
2.2.1 Types of Model Observer Tasks and Decisions	21
2.2.2 Classification Tasks	21
2.2.3 Estimation Tasks.....	22
2.3 Classification Task Theory.....	22
2.3.1 Partitioned Data Space.....	22
2.3.2 Binary Classification Decisions.....	23
2.3.3 Sensitivity and Specificity	24
2.3.4 The ROC Curve	26
2.3.5 PDF Plot Visualization	27
2.3.6 AUC.....	28
2.3.7 SNR	29
2.3.8 Detectability Index	30
2.4 The Ideal Observer	30
2.4.1 Likelihood Ratios	31
2.4.2 Likelihood Ratio Tests.....	31
2.4.3 Log-Likelihood Ratio	32

2.4.4	SNR ² and the Ideal Observer.....	33
2.4.5	Karhunen-Loève (KL) Decomposition.....	34
CHAPTER 3	Analysis Method.....	35
3.0	User Defined Variables and Texture Modeling Method.....	35
3.1	Textures.....	36
3.1.1	Governing Mathematics	36
3.1.2	Return of the KL Decomposition	38
3.2	The Channelized Hotelling Observer (CHO).....	38
3.2.1	Signal-to-Noise Ratio (SNR).....	39
3.2.2	Utilizing the CHO.....	40
3.3	Organization is Key.....	41
3.4	Utilizing Useful Textures	42
3.5	Example of Analysis Methods	43
3.6	Code Walkthrough	44
CHAPTER 4	Validations.....	44
4.1	Reconstruction Comparison	44
4.2	Comparison to the Hotelling Observer.....	46
CHAPTER 5	Discussion.....	47
REFERENCES	49

LIST OF FIGURES

Figure 1.1: Example Projection of a Circle	13
Figure 1.2: Projections of a More Complex Subject.....	14
Figure 1.3: Offset Point Source and Detector Setup with Corresponding Sinogram	14
Figure 1.4: Complex Subject and Corresponding Sinogram	15
Figure 1.5: Projection of a Point Source	16
Figure 1.6: Filtered Backprojection Reconstruction of a Point Source	17
Figure 2.1 Example of Partitioned Observation Space	23
Figure 2.2 Example ROC Curve.....	26
Figure 2.3: PDFs of Varying Thresholds.....	27
Figure 2.4: ROC Curves Corresponding to Figure 2.3	28
Figure 2.5: Two Different ROC Curves with Same AUC	29
Figure 3.1(a)-(c): Example Simulation Images and Signal Injected.....	35
Figure 3.2: Example Set of Image Textures	37
Figure 3.3: The Eye Filter	41
Figure 3.4: Example of Top 4 Most Useful Textures	42
Figure 4.1: α Calculations for Different FBP/IR Training/Test Image Set Combinations	45
Figure 4.2: Human observer signal detection performance comparison	46

LIST OF TABLES

Table 2.1 Binary Decision Outcomes	24
Table 2.2 Decision States for Signal Detection	24

ABSTRACT

We have created a faster, less computationally intensive method of estimating human observer performance when tasked with detecting a signal in CT images. This is achieved by using a set of images to train the model observer, extract the “most useful” textures in detecting the input signal, and applying those textures to a new set of images and extracting an SNR^2 -equivalent metric. This is validated by comparing to the Channelized Hotelling Observer (CHO), the field standard for modeling human observer performance, as well as testing on data with known relative performance, which will be elaborated on in the results section. A large merit in this project is that it can be used to help find appropriate radiation dosing per tissue texture to achieve the best differentiation between a signal (such as cancer) and noise or tissue texture, as well as CT image reconstruction algorithm optimization.

CHAPTER 1

Introduction and Motivations

The work presented in this thesis is largely, if not wholly, implemented through software. Due to the nature of 21st century science and engineering frequently being reliant on or driven by software, it can be easy to forget the amount of hands-on experimentation and technological design it took to make it to this point, so this thesis will begin with a brief history of what led to the methods developed in this research, as well as their efficacy in both the present and future. The personal goal of this paper is to convey this information in an order and a language that make it accessible to an audience wider than just those working in the field of image science while maintaining the technical depth to be of use to others continuing research in this field.

1.1 A Brief History of Computed Tomography (CT)

Though observed in experiments over 20 years prior, the field of radiology began with German mechanical engineer and physicist, Wilhelm Röntgen's discovery of x-rays while experimenting with Crookes tubes on the 8th of November 1895.^[1] His initial report on this "new kind of ray" was submitted to the Würzburg Physical-Medical Society journal on the 28th of December the year of his discovery. This being the first published paper about this unknown radiation, Röntgen referred to it as "X". Thus, the term "x-ray" was born, despite some dispute between Röntgen and his colleagues suggesting another name, such as "Röntgen rays". His discovery led to him receiving the first Nobel Prize in Physics.^[2]

Röntgen's continued research in the field of x-rays led to the discovery of their medical imaging efficacy and produced the first x-ray photograph or "shadowgram" of a human subject on the 22nd of December 1895, less than two months after his initial discovery of x-rays. After an uptick in x-ray experimentation, the hazards of x-radiation had been discovered and the methods surrounding their experimentation were refined, leading to the invention of the x-ray microscope in the 1950s, the decade when South African American

physicist, Allan MacLeod Cormack, began theorizing how x-rays can be used for CT imaging.

Cormack was born in Johannesburg, South Africa on the 23rd of February 1924. He received his B.Sc. in physics in 1944, his M.Sc. in crystallography in 1945, and his PhD in physics in 1949. While lecturing at the University of Cape Town, he worked with the Groote Schuur Hospital in 1956 to begin developing the theory of using x-rays for CT scanning, during which time he moved to the U.S. and became a professor at Tufts University. His work resulted in two papers being published to the Journal of Applied Physics; one in 1963 and the other in 1964, two years before becoming a naturalized citizen of the United States.^[3] These papers were used by, English electrical engineer, Sir Godfrey Hounsfield.

Hounsfield was born in Sutton-on-Trent, Nottinghamshire, England on the 28th of August 1919. After his time in the Royal Air Force during WWII, he obtained his Diploma of Faraday House (DFH) from the Faraday House Electrical Engineering College in London. In 1949, he began working on radar and guided weapons systems research at EMI, Ltd. in Hayes, Middlesex. After assisting in the design of the first commercial all-transistor computer, the EMIDEC 1100, in 1958, he began working on the first CT scanner under EMI. Using Cormack's papers on CT imaging theory, Hounsfield and his colleagues developed the first CT scanner in 1971. The first medical success was a scan of a cerebral cyst of a patient at Atkinson Morley Hospital in Wimbledon, London. It wasn't until 1975 that Hounsfield built the first full-body CT scanner.^[4,5]

Hounsfield and Cormack received the Nobel Prize in Physiology or Medicine in 1979. Hounsfield was subsequently knighted in 1981. Since then, CT scanning technology has been used to non-invasively diagnose all manner of internal medical issues; most notably, cancer. It has also improved in many ways since its initial development, such as increased image resolution, faster scan times, improved patient comfort, as well as faster and less noisy image reconstruction. The next section focuses on how CT scanners work to produce 3D reconstructions of patient bodies.

1.2 CT Imaging Process

CT scans, formerly referred to as CAT (computerized axial tomography) scans, are common in diagnosing internal abnormalities, such as cancer, within the body. The result of a CT scan is a 3D rendering of the both the internal and external components of a patient/subject – this could be an entire human body, showcasing superficial skin tissue down to the deep skeletal structure and within. These renderings are comprised of axial slices of the subject. Much like a sliced loaf of bread, when the slices are put in the correct order, the entirety of the original subject can be reconstructed. The next three sections seek to describe the process from both a detailed nontechnical point of view, as well as a mathematical point of view. Heavy influence was drawn and many figures are taken from the general information flow in Zeng's *Medical Image Reconstruction (2010)*.^[6]

1.2.1 Tomography

If one wanted to see inside of a watermelon to see whether it is seedless, the easiest way to check would be to slice it open. Seeing a cross section of an object can yield valuable information. Cutting the watermelon into several even slices can show whether the watermelon is truly seedless throughout. This is the basic principle behind how tomography works.

Tomography is rooted in the Greek word *tomos*, which refers to a slice or section. Therefore, the word *Tomography* is the method of imaging cross sections or slices. In the same manner that one would use a knife to physically slice into the watermelon in the example above, tomography uses radiation to non-invasively take pictures of the cross sections of a subject. In the case of medical imaging, it can be used for diagnosis and finding the location of cancerous masses for treatment. Since images cannot be directly taken of a chosen cross section, each cross-sectional image is reconstructed from several *projections*.

1.2.2 Projections

A *projection* in the case of CT describes the density of an object from a single angle or point of view. X-rays are used to take pictures all around the subject by sending a set dose of x-rays to a detector with the subject directly between the x-ray source and detector. The x-rays penetrate the subject, being scattered and absorbed by different types of tissue, creating an intensity pattern on the detector that acts as a spatial density distribution of the object distinct to the angle from which the picture was taken. *Figure 1.1* shows an example of a projection through a circle. Notice that the projection function is highest in the center, where the x-rays would encounter the most density or travel through the largest amount of the subject. If the object in the figure was a doughnut, there would be a dip in the center of the projection, starting at one edge of the hole and ending at the opposite edge.

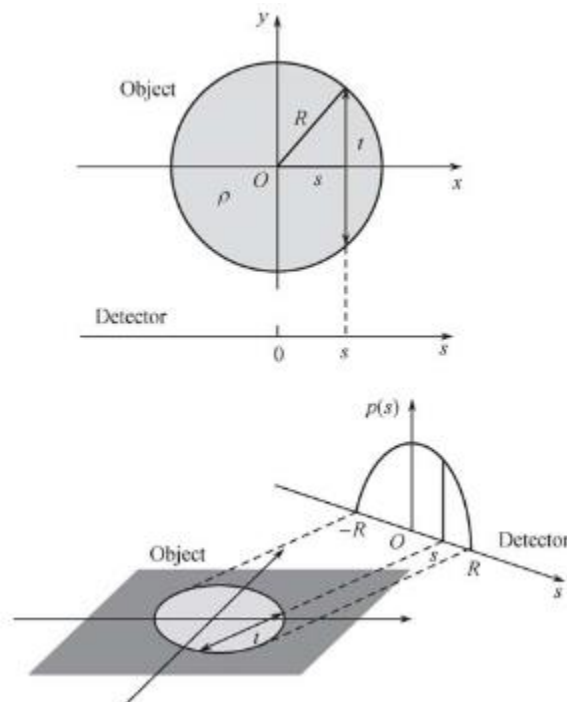
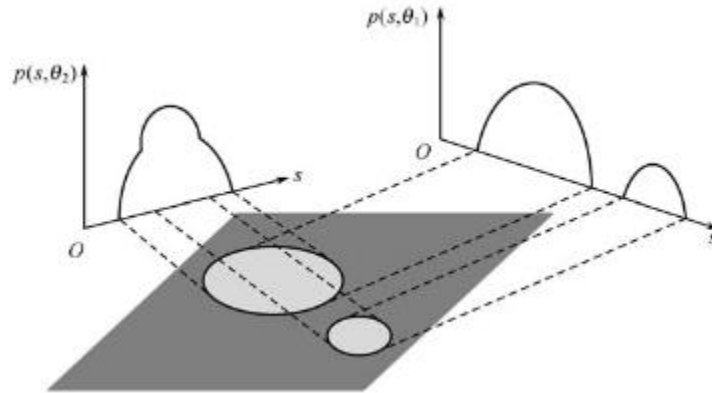


Figure 1.1: Example Projection of a Circle

(Medical Image Reconstruction 2010)

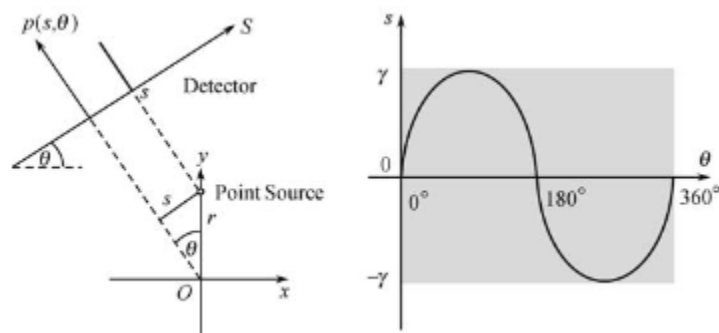
This process is repeated at several angles about the subject, leaving a projection for each chosen angle. Taking images at several different angles of the circle in *Figure 1.1* would yield the same projection every time, proving that the object is indeed a circle, or at least

circularly symmetric. Given a more complex image, like the one seen in *Figure 1.2*, multiple angles are needed to get a full picture of the subject. Two projections are shown, yielding different information and, therefore, projections per angle.



*Figure 1.2: Projections of a More Complex Subject
(Medical Image Reconstruction 2010)*

In *Figure 1.2*, the projection on the left shows a higher density toward the center of the projection, which gives no indication of the number of objects present in the subject. The projection on the right shows that there are at least two objects creating the higher density in the center of the projection on the left. Gathering projections around the entire image can yield information about the number of objects present, as well as their shapes and locations with respect to one another.



*Figure 1.3: Offset Point Source and Detector Setup with Corresponding Sinogram
(Medical Image Reconstruction 2010)*

For computational organization, projections are put into an array, organized by the respective angle each projection was gathered from. These arrays are called *sinograms*. The name comes from the sinusoidal pattern generated when the projections of a point source are organized by observation angle, as seen in *Figure 1.3*.

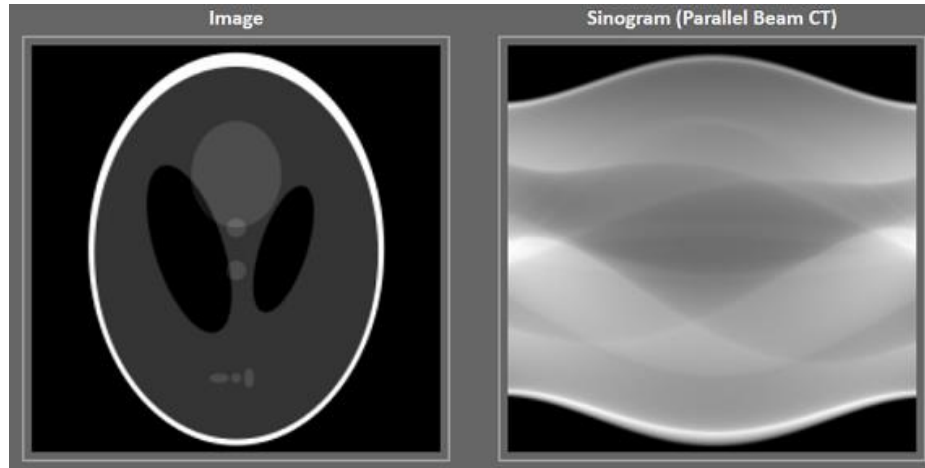


Figure 1.4: Complex Subject (Left) and Corresponding Sinogram (Right)
 (<https://www.codeproject.com/Articles/1113200/Computed-Tomography-Generation-of-the-Sinogram-Ima>)

Figure 1.4 shows a complex subject on the left with its corresponding sinogram on the right. Each vertical line in the sinogram corresponds to a single projection at a specified angle. These projections are rotationally smeared together using different algorithms and techniques to create a reconstruction of a single cross section of the subject.

1.2.3 Image Reconstruction

As described in the previous section, a CT system gathers several 1D projection images of a particular cross section of a subject. However, these projection images themselves do not easily yield a clear cross section of the subject. A reconstruction algorithm is needed to smear these projections together as a function of the angle at which they were taken. Namely, a mathematical process is needed to turn the sinogram of the subject cross section into a 2D image of the actual cross section.

The basic principles of reconstructing CT images rely on the concept of backprojection of the gathered projection images across a map at the respective angles the projection images were taken. For example, consider a point source subject comprised of a 2D plane and a single dot offset from the origin. If a detector was used to take projection images around the entirety of the subject, each projection would have a spike in line with the point source, as seen in *Figure 1.5*. Each projection would be zero everywhere the point source was not in line.

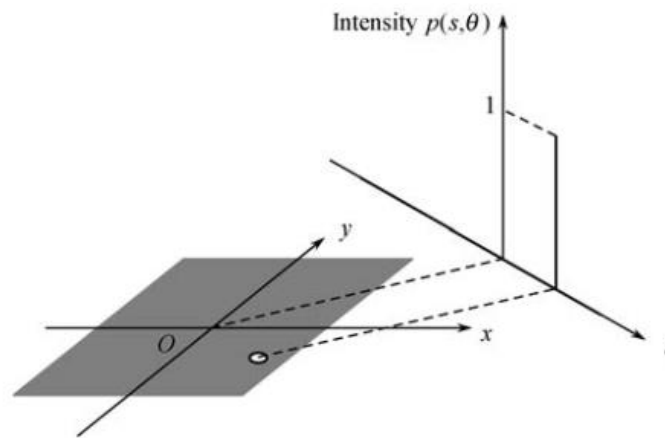
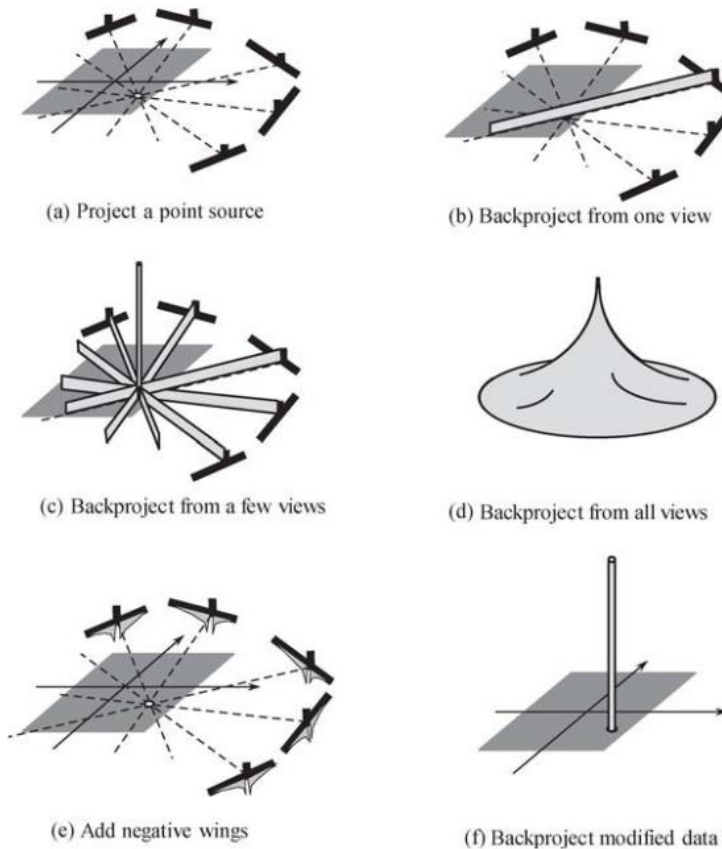


Figure 1.5: Projection of a Point Source
(*Medical Image Reconstruction 2010*)

Backprojections are formed by extending the individual pixel values of the detector across a 2D plane, orthogonal to the detector array. In this example, the s -axis in *Figure 1.5* is the detector, gathering projection images. Imagine the s -axis is segmented into 100 different pixels. Only one of those pixels was in line with the point source, so it is the only pixel with a nonzero value; in this case, that pixel has a value of 1. The backprojection that would be formed in the setup shown in *Figure 1.5* would be a line with a value of 1 extended across the 2D plane directly perpendicular to the s -axis. *Figures 1.6a-c* show how the backprojection process is repeated from all angles at which projection images were acquired.

Using several backprojections at various angles yields the location of the point source in the 2D plane and, therefore, the cross section of the subject. However, as shown in *Figure 1.6d*, this doesn't look much like the original subject. Since each projection image only

contains information from one angle, they cannot yield information regarding how each pixel value is distributed across the subject. Namely, each projection can only show where the point source is located along the s-axis, but not how far away the point source is from the s-axis. This means that, since each backprojection extends a single value per pixel across the entire 2D plane, the superposition of all the backprojections creates a blurred image.



*Figure 1.6: Filtered Backprojection Reconstruction of a Point Source
(Medical Image Reconstruction 2004)*

To eliminate blurring, negative wings are added to the backprojections, as shown in *Figure 1.6e*, creating filtered backprojections. *Figure 1.6e* shows that, with the correct filtering, an image of the original cross section of the subject can be properly reconstructed. However, this is a very simple subject and yields a very simple filtering approach. The cross section of a human being is much more complex and requires much more mathematically intensive ways of creating filtered backprojections and reconstructing cross sectional

images. In a more realistic medical imaging scenario, the collimated beam imaging shown in the previous figures is replaced with cone beam imaging. The geometry involved in having a conical beam is more involved, but a larger volume (multiple slices) of the patient can be imaged at once, resulting in an overall lowered radiation exposure. Given the complexity and variation between human anatomy and physiology, there is no one correct way to filter backprojections. This means that using filtered backprojections is not always the best method for image reconstruction. However, as CT imaging has progressed, new image reconstruction methods have been developed, such as *iterative reconstruction*.

Iterative reconstruction is a method of reconstructing CT images step-by-step, using different algorithms to compare the reconstructed image with an idealized reconstruction based on things such as statistics or machine learning. Each iteration is compared to this ideal reconstruction until some user-defined minimum deviation between the reconstructed image and ideal image is reached. This is a slower and more computationally taxing method of image reconstruction than using single filtered backprojection reconstructions. However, it can also produce images with better signal detectability than filtered backprojections at similar (over even lower) radiation doses.

1.2.4 A Brief Summary

Several x-ray images are taken of a subject at different angles axially about the subject. These images produce projections of a section of the subject, based on the angle from which the image was taken. These projections are smeared together using a selected reconstruction algorithm to create a single cross-sectional slice of the subject. This process is repeated linearly across the length of the subject, creating several slices that, when put together, form a 3D rendering of the subject.

1.3 How the Observer Fits In

An observer, such as a radiologist, takes an image or series of images and performs a given task, such as determining whether a cancerous mass is present in the CT slices. Human

observers can be modeled using software to predict human performance in performing visual tasks, like signal detection. This is helpful when optimizing systems with output intended for human visual use throughout the entirety of the design phase, rather than after a benchtop prototype is built. Model observers eliminate the need for lengthy, high-volume human studies.

In the case of CT imaging, how well a human can detect a particular signal in an image is directly affected by the texture by which the signal is surrounded. Namely, a small signal or cancerous mass may be difficult to resolve amongst a speckled background tissue texture of high contrast. The current field standard observer, the Hotelling Observer, is computationally taxing, meaning that this type of computation would require large amounts of computing power and could still be a lengthy undertaking.

This project seeks to provide a method of estimating human observer performance with comparable accuracy to the Hotelling Observer while alleviating some of the computational weight. The method through which this is performed is by training the observer to focus on the textures that hold the most weight in human signal detection performance and seeking them out in subsequent image sets for use in estimating human observer performance based on their presence in the image sets under test.

CHAPTER 2

The Model Observer

Chapter 1.3 mentions that observers can be modeled and that a field standard model human observer, the Hotelling observer, exists. This chapter seeks to describe what modeling an observer entails, as well as how the mathematical definitions of textures are implemented with the (Channelized) Hotelling observer in this texture analysis and human observer estimation method.

2.1 What is a Model Observer?

The example given in Chapter 1.3 has an observer trying to detect the presence of a signal through a series of objects, where the observer is a radiologist, the signal is a cancerous mass, and the objects are CT images. The observer analyzes the images and is tasked with making a decision as to whether the signal in question is present or absent. In reality, the decisions between human observers will vary, but whether a signal is actually present in an image is a matter of fact. This means that modelling a human observer is largely a matter of statistics and falls under the aptly named subject of statistical decision theory.^[8]

2.2 Statistical Decision Theory

The subject of statistical decision theory with regard to imaging systems, such as a CT scanner, can be described as the theory of making a task-based (not necessarily binary) decision or classification based on the statistics of the image under scrutiny. Where this might sound like an obvious description, given the last couple of sentences in the previous section, note that the statistics are not based on the observer so much as it is in the statistical data from the image itself. Namely, the bigger matter than how one human observer performs compared to another human observer is not as much of a concern as how the image quality affects the ability of any given human observer to detect a signal in said image. What aspects of and how image quality affects model observer performance depend on what type of decision the model observer is tasked with making.

The content of this chapter is taken from the work of Dr. Harrison Barrett and Dr. Kyle Myers. Given the all-encompassing nature of their book, *Foundations of Image Science* (2004)^[9], the theory and information in this chapter is written in an order to mimic the flow of the book with the omission of some extraneous information and theory.

2.2.1 Types of Model Observer Tasks and Decisions

There are two types of decisions model observers can be tasked with – *classification* and *parameter estimation*. In short, when the number of possible outcomes in a task is finite, the decision is referred to as *classification*. Some examples of classification are signal detection, pattern recognition, and hypothesis testing. When the number of hypotheses is infinite, the task is referred to as *estimation*. Namely, any time a particular numerical parameter is extracted from the data, even when the value of the parameter is within a specified range, the task is deemed “estimation”. Both types of tasks are governed by the same image formation equation:

$$\mathbf{g} = \mathcal{H}\mathbf{f} + \mathbf{n} \quad (2.1)$$

where \mathcal{H} is a matrix defining the imaging system, \mathbf{f} is a vector defining the object being imaged, \mathbf{n} is a vector defining the noise present in the system, and \mathbf{g} is a vector defining the resultant image data. In words, this equation shows how the resulting image data, \mathbf{g} , is formed by imaging an object, \mathbf{f} , through a system, \mathcal{H} , and adding the noise present in said system, \mathbf{n} .

In a CT setting, the image data, \mathbf{g} , are the projection images. These images need to be processed by a reconstruction algorithm, θ , to form an approximation of the initial object, $\hat{\mathbf{f}}$.

2.2.2 Classification Tasks

In classification tasks, the observer uses the image, $\hat{\mathbf{f}}$, to classify or categorize the object, \mathbf{f} . As stated before, signal detection is an example of a classification task. The categories the observer can label an image in the case of signal detection are either *signal present* or *signal absent*. However, the hypotheses do not have to be binary. The observer can classify images into one of several categories. In medical imaging it can be something as “simple” as determining whether a tumor is present in the image. It can also be a bit more complicated, such as categorizing whether a tumor is present and also classifying any tumor present as benign or malignant. This is also used in military applications to distinguish tanks

from civilian vehicles in airborne imaging systems or jet exhaust from the sun in daytime images of the sky.

2.2.3 Estimation Tasks

Estimation tasks assume that the image data \mathbf{g} comes from a known probability law, $\text{pr}(\mathbf{g}|\theta)$, and the observer is tasked with estimating some scalar parameter, θ . A medical imaging example of estimation would be using a photon counter to detect the activity in a draining lymph node. As opposed to *scalar parameter estimation* being tasked with estimating a single parameter, θ , *vector parameter estimation* tasks the observer with estimating multiple parameters in the form of a vector, $\boldsymbol{\theta}$. As an example of vector parameter estimation, consider a photodetector detecting light reflected by an object, \mathbf{f} .

2.3 Classification Task Theory

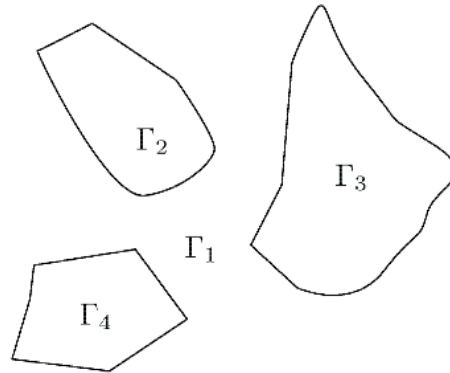
In the case of this work, classification tasks, specifically signal detection, are the focus. This being stated, there is quite a bit to consider when raising either the *signal present* or *signal absent* flag. Thus far, *what* model observers do, as well as *why* they do it has been described, which leaves the important question of *how* they make decisions and estimations, as well as what inaccuracies come along with them.

2.3.1 Partitioned Data Space

Classification tasks require an observer to use the data vector, \mathbf{g} , to classify the source of the data; generally, the object, \mathbf{f} . Namely, the observer uses \mathbf{g} to make a decision, D_1 or D_2 , in favor of a particular hypothesis, H_1 or H_2 . Two stipulations are placed on classification tasks:

1. There can be no randomness in decision ruling.
2. Every observation requires a decision.

The first stipulation ensures that observation in the same data \mathbf{g} results in the same decision ruling, D_i . The second ensures that the observations result in some distinct decision, rather than some combination of decisions or an inconclusive decision. In the case of signal detection, either the data are ruled *signal present* or *signal absent* and, no matter how many times you observe the same data, \mathbf{g} , the decisions will always be the same.



*Figure 2.1 Example of Partitioned Observation Space
(Foundations of Image Science 2004)*

Having these stipulations in place effectively partitions the observation space into nonoverlapping regions or volumes that, when combined, encompass all possible observation points. *Figure 2.1* shows a visualization of a four-region observation. Namely, it shows an observation space with four possible decisions or classifications to categorize observed data into. In this case, Γ_i denotes the partition or region respective to the decision, D_i .

2.3.2 Binary Classification Decisions

In the case of a binary classification task, like signal detection, a scalar test statistic, t , is used to assign data into one of the two decision regions. A discriminant function relates the test statistic to the data by means of:

$$T(\mathbf{g}) = t, \quad (2.2)$$

The observer uses a decision threshold or cutoff, t_c , to compare the test statistic, t , to determine which particular region the data should be assigned to. This means that the edges of the partitions in *Figure 2.1* show the contours of a constant t value.

Binary classification tasks restrict the number of classes or hypotheses in which the object could potentially belong to two, H_1 and H_2 . In signal detection, these would correspond to the *signal absent* and *signal present* classifications, respectively. Since there are two possible decisions and two possible truths, there are four possible outcomes in a binary classification task:

Table 2.1 Binary Decision Outcomes

True Positive (TP):	H_2 is true; observer decides H_2 is true
False Positive (FP):	H_1 is true; observer decides H_2 is true
False Negative (FN):	H_2 is true; observer decides H_1 is true
True Negative (TN):	H_1 is true; observer decides H_1 is true

Just like human observers, model observers can also classify objects incorrectly. As *Table 2.1* shows, two of the possible outcomes detail the observer correctly classifying the data and two of the outcomes detail the observer incorrectly classifying the data. Intuitively, this results in four different decisions states:

Table 2.2 Decision States for Signal Detection

	H₂: Signal Present	H₁: Signal Absent
D₂: Observer Decides H₂	TP, Correct Detection	FP, False Alarm
D₁: Observer Decides H₁	FN, Missed Detection	TN, Good “No” Call

2.3.3 Sensitivity and Specificity

Given that model observers can also make mistakes, it is useful to know the sensitivity of a model before putting it to use. That is, knowing its probability or likelihood of producing each of the decision states listed in *Table 2.2*. An observer with a high sensitivity will produce more TPs, but also FPs, whereas an observer with low sensitivity will produce more

FNs and TNs. Given that both are unfavorable options, the following formulas for calculating these sensitivities are common in characterizing model observers:

$$TPF = Pr(D_2|H_2) = \left\langle \frac{N_{TP}}{N_{TP}+N_{FN}} \right\rangle = \lim_{n \rightarrow \infty} \left[\frac{\text{Number of TP decisions}}{\text{Number of actually positive cases}} \right] \quad (2.3)$$

$$TNF = Pr(D_1|H_1) = \left\langle \frac{N_{TN}}{N_{TN}+N_{FP}} \right\rangle = \lim_{n \rightarrow \infty} \left[\frac{\text{Number of TN decisions}}{\text{Number of actually negative cases}} \right] \quad (2.4)$$

$$FPF = Pr(D_2|H_1) = \left\langle \frac{N_{FP}}{N_{TN}+N_{FP}} \right\rangle = 1 - TNF \quad (2.5)$$

$$FNF = Pr(D_1|H_2) = \left\langle \frac{N_{FN}}{N_{TP}+N_{FN}} \right\rangle = 1 - TPF \quad (2.6)$$

The equations above show how to calculate the *True Positive Fraction* (TPF), *True Negative Fraction* (TNF), *False Positive Fraction* (FPF), and the *False Negative Fraction* (FNF). Given the total number of decisions made by an observer, N , and using the number of TP decisions, N_{TP} , number of TN, N_{TN} , number of FP decisions, N_{FP} , and number of FN decisions, N_{FN} , made by the observer, the above calculations are fairly straightforward. However, the probability density function (PDF) for the scalar test statistic t can also be used to calculate these values:

$$TPF = Pr(t \geq t_c|H_2) = \int_{t_c}^{\infty} dt pr(t|H_2) \quad (2.7)$$

$$FPF = Pr(t \geq t_c|H_1) = \int_{t_c}^{\infty} dt pr(t|H_1) \quad (2.8)$$

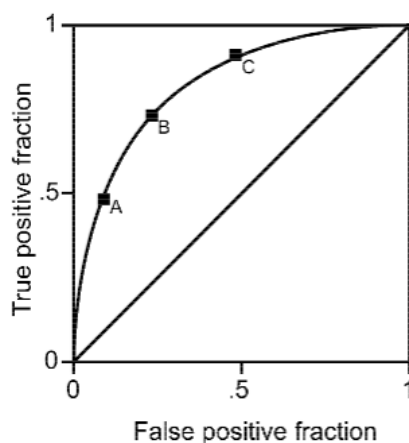
$$TNF = 1 - FPF = \int_{-\infty}^{t_c} dt pr(t|H_1) \quad (2.9)$$

$$FNF = 1 - TPF = \int_{-\infty}^{t_c} dt pr(t|H_2) \quad (2.10)$$

A more immediately tangible way of showing the sensitivity or threshold strictness of an observer is to plot the TPF against the FPF. This plot is referred to as the *receiver operating characteristic* (ROC) curve.

2.3.4 The ROC Curve

The ROC curve provides a detailed summary of several aspects of an observer's performance of a task all in one convenient plot. It provides information on the difficulty of a task, the performance of the observer and how it chooses to make decisions, as well as the quality of the data with respect to allowing the observer to perform a task. The ROC curve is also independent of prevalence, giving its data more merit. It can easily show how the strictness of the threshold, t , affects a specific observer performing a specific task.



*Figure 2.2 Example ROC Curve
(Foundations of Image Science 2004)*

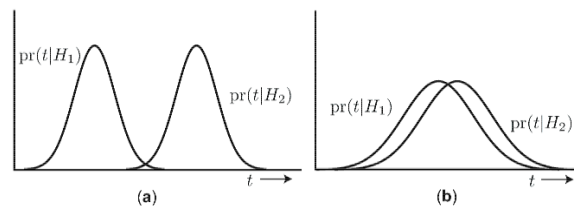
Figure 2.2 shows an example ROC curve with three points, A, B, and C. Point A shows how the observer performs the task with a strict threshold. The observer is less likely to decide on H_2 , resulting in a very low FPF, but also a low TPF. Point C corresponds to a loose threshold, resulting in the observer making more H_2 decisions and, therefore, both a higher TPF and FPF. Point B is a moderate threshold, yielding TPF and FPF results between the two other labeled points. Note that, no matter the task and observer, the ROC curve will always start at origin, (0,0), and end at point (1,1). These two points correspond respectively to an unachievable threshold, resulting in no H_2 decisions at all, and an always achievable threshold, resulting in all decisions being H_2 .

It may seem a little counterintuitive to use a plot like this instead of using absolute accuracy across all decisions as a figure of merit. The problem with using absolute accuracy lies in prevalence. Namely, if patients are being screened for a disease that only affects 0.4%

of the population [$\Pr(H_2)=0.004$], the screener can just decide that none of the patients have the disease while still maintaining 99.6% accuracy, despite misdiagnosing every single patient that did possess the disease. This is the importance of the ROC curve not being subject to prevalence. Beyond that, other useful data can be extracted from the ROC curve, such as the area under the ROC curve (AUC), signal-to-noise ratio (SNR), and detectability index (d_A).

2.3.5 PDF Plot Visualization

Before going directly to the aforementioned metrics, a useful figure to visit is a plot of PDFs of the test statistic, given either hypothesis. Namely, plots of $\text{pr}(t|H_1)$ and $\text{pr}(t|H_2)$. These plots are a good way of determining a threshold for a particular observer. *Figure 2.3* shows two different sets of PDFs with different thresholds and how the threshold affects the observer to separate two different classes (in signal detection, *signal absent* or *signal present*).



*Figure 2.3: PDFs of Varying Thresholds
(Foundations of Image Science 2004)*

Figure 2.4 shows what the ROC curves corresponding to the PDF plots in *Figure 2.3*. The better the class separation, the closer to the left and top borders of the plot area the ROC is. Therefore, with perfect class separation (PDFs exhibit no overlap at all), the ROC curve would show a TPF of one for all given FPFs. This means that *Figure 2.3a* is represented as the curve with a d_A (*detectability index* – more on this later in the chapter) value of 2.0 and *Figure 2.3b* is represented as the curve with a d_A value of 0.5. Other measures of the overlap (class/signal indistinguishability) seen in these plots are the AUC and SNR.

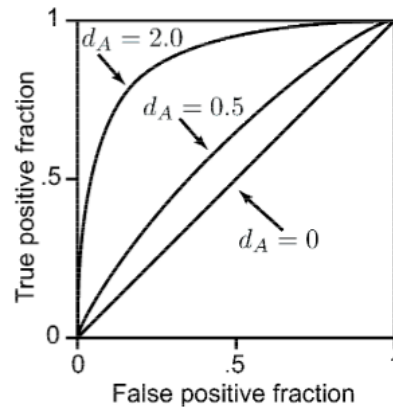


Figure 2.4: ROC Curves Corresponding to Figure 2.3
(Foundations of Image Science 2004)

2.3.6 AUC

For cases where the ROC curve is symmetric about its negative diagonal, such as the ROC curve shown in *Figure 2.2*, the AUC has a direct correlation with TPF per FPF. Namely, the higher the AUC for these cases, the higher the TPF for any given FPF. So, in these cases, the AUC can be a good measurement of model observer performance. Also, given the simple, one-variable nature of the ROC curves, calculating the AUC is given by a simple integral. Given the ROC ranges from zero to one, the AUC will fall in that range, as well.

$$AUC = \int_0^1 dFPF TPF(FPF) \quad (2.11)$$

Of course, not every ROC falls under this special case of symmetry, so AUC isn't always the best performance metric. For example, *Figure 2.5* below shows two ROC curves with the same AUC, but clearly different TPF per given FPF. This being stated, other metrics can be used to determine performance of these other cases.

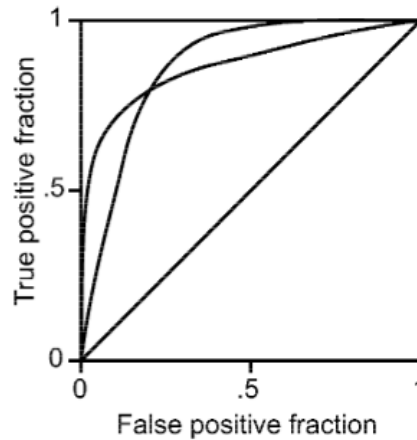


Figure 2.5: Two Different ROC Curves with Same AUC
(Foundations of Image Science 2004)

2.3.7 SNR

The *signal-to-noise ratio* is as intuitive of a signal detectability metric as the name implies. Having a higher SNR means having a smaller chance of the signal being drowned out by the noise floor. Since, in reality, noise is present in every image, this metric is extraordinarily important. In classification tasks, it represents a calculation of the overlap between the two PDFs mentioned in 2.3.6, $\text{pr}(t|H_1)$ and $\text{pr}(t|H_2)$.

$$SNR_t = \frac{\langle t \rangle_2 - \langle t \rangle_1}{\sqrt{\frac{1}{2}\sigma_1^2 + \frac{1}{2}\sigma_2^2}} \quad (2.12)$$

It should be noted that the equation above is meant only for test statistic t that is relatively normally distributed under both hypotheses. Specifically, since this equation is dependent on the variance of the test statistic, σ , test statistic PDFs with a high amount of skewness don't provide a particularly useful measure of the decision variable spread. However, when they are normally distributed under both hypotheses, the AUC can be calculated using the SNR_t calculation using the following equation:

$$AUC = \frac{1}{2} + \frac{1}{2} \text{erf} \left(\frac{SNR_t}{2} \right) \quad (2.13)$$

When the test statistics are not normally distributed, the AUC should be calculated using the integral form mentioned in 2.3.7. However, there are times when the AUC and

SNR_t do not adequately specify an observer/system. In these cases, it helps to be selective and use a partial AUC or compare optimal ROC curve points with respect to the utility of the system. There is no single correct or best figure of merit for all observer models or even classification tasks. The open-ended use of observer models requires circumstantial data manipulation on a task-to-task basis, so it is useful having literature on a number of merit figures.

2.3.8 Detectability Index

The detectability index, d_A , was briefly mentioned 2.3.6. The detectability index provides information along the lines of the SNR. In cases where the AUC has been calculated, the detectability index can be calculated as another figure of merit using the following equation:

$$d_A = 2erf^{-1}[2(AUC) - 1] \quad (2.14)$$

A specific example of when the detectability index is useful is when human observer performance studies directly measure AUC by implementing forced-choice techniques. In these cases, the AUC is estimated directly, but the ROC curve shape is still unknown. *Figure 2.4* shows how the detectability index d_A affects the shape of the ROC curve.

2.4 The Ideal Observer

Thus far, the figures of merit described for classification tasks all have the same goal of optimizing classification (or signal detection) performance. Observers can be given tests based on the likelihood ratios of the two hypotheses, aptly named *likelihood ratio tests*. Observers that properly perform likelihood ratio tests are considered *ideal observers*. Note the plural at the end of the previous sentence – there is no singular ideal observer, but rather there are many different methods of modeling an observer that performs a likelihood ratio test well enough to be deemed an ideal observer. Model observers being *ideal* is based on meeting a particular log-likelihood performance standard. Chapter 3 will detail the particular ideal observer used in the texture analysis methods, the *Hotelling Observer*.

2.4.1 Likelihood Ratios

Likelihood ratios are a simple measure of the likelihood between complementary hypotheses – in this case, H_1 and H_2 . They show to what degree a particular hypothesis is supported with respect to another. Given the data vector, \mathbf{g} , the likelihood ratio for the ideal observer, denoted $\Lambda(\mathbf{g})$, would be a simple ratio of the PDF of a signal being present over the PDF of the signal being absent. A more tangible way of expressing a likelihood ratio is the ratio between the TPF and FPF.

$$\Lambda(\mathbf{g}) = \frac{pr_{\mathbf{g}}(\mathbf{g}|H_2)}{pr_{\mathbf{g}}(\mathbf{g}|H_1)} = \frac{TPF}{FPF} \quad (2.15)$$

2.4.2 Likelihood Ratio Tests

A likelihood ratio test is a decision making method utilizing the likelihood ratio, $\Lambda(\mathbf{g})$, and a threshold, Λ_c , set by the objective of the test. As stated earlier, the ideal observer uses forced-choice techniques. This means that the ideal observer cannot be given data and *not* make a decision as to whether a signal is present or absent. A likelihood ratio test forces the ideal observer to make a decision based on whether the value of the likelihood ratio is higher or lower than that of the set threshold.

$$\text{Choose } H_2 \text{ if } \Lambda(\mathbf{g}) > \Lambda_c \quad (2.16)$$

$$\text{Choose } H_1 \text{ if } \Lambda(\mathbf{g}) < \Lambda_c \quad (2.17)$$

As stated earlier, what makes an observer an ideal observer is the ability to perform this likelihood ratio test properly. Proper performance on likelihood ratio tests requires the observer to have no means of corrupting the input data by means of an internal noise mechanism. It also must apply the same given threshold to each trial to avoid bias. Lastly, it must be able to formulate the PDF $pr(\mathbf{g}|H_j)$ using the information regarding the objects to be classified, the process by which data is measured, as well as noise statistics that come with said process.

Section 2.4.1 showed how the likelihood ratio can be calculated using probabilities. However, for computational purposes used in the theory of this thesis, it is useful to have likelihood tests and ratios in terms of vectors and matrices that can be utilized to compare to a new threshold λ'_c that is set by the *log-likelihood ratio*.

2.4.3 Log-Likelihood Ratio

Before presenting the likelihood ratio and test in terms of vectors and matrices, the covariance matrix needs to be accounted for:

$$\mathbf{K}_j = \langle (\mathbf{g} - \mathbf{s}_j)(\mathbf{g} - \mathbf{s}_j)^t | H_j \rangle = \langle \mathbf{n}\mathbf{n}^t \rangle = \mathbf{K}_n \quad (2.18)$$

The matrix, \mathbf{K}_j denotes the covariance matrix with respect to the j^{th} hypothesis. Since \mathbf{K}_1 and \mathbf{K}_2 are the same, this covariance matrix becomes respective to the noise, \mathbf{n} . The covariance matrix is utilized in redefining the aforementioned probability functions in vector/matrix form:

$$pr(\mathbf{g}|H_j) = [(2\pi)^M \det(\mathbf{K}_n)]^{-\frac{1}{2}} \exp\left[-\frac{1}{2}(\mathbf{g} - \mathbf{s}_j)^t \mathbf{K}_n^{-1}(\mathbf{g} - \mathbf{s}_j)\right] \quad (2.19)$$

This form allows for the reformation of the likelihood ratio test in vector/matrix form:

$$\Lambda(\mathbf{g}) = \frac{\exp\left[-\frac{1}{2}(\mathbf{g} - \mathbf{s}_2)^t \mathbf{K}_n^{-1}(\mathbf{g} - \mathbf{s}_2)\right]}{\exp\left[-\frac{1}{2}(\mathbf{g} - \mathbf{s}_1)^t \mathbf{K}_n^{-1}(\mathbf{g} - \mathbf{s}_1)\right]} \begin{matrix} D_2 \\ > \\ < \\ D_1 \end{matrix} \Lambda_c \quad (2.20)$$

Taking the natural log of the expression above leads to an equivalent test referred to as the *log-likelihood ratio*:

$$\lambda(\mathbf{g}) = \frac{1}{2}(\mathbf{g} - \mathbf{s}_1)^t \mathbf{K}_n^{-1}(\mathbf{g} - \mathbf{s}_1) - \frac{1}{2}(\mathbf{g} - \mathbf{s}_2)^t \mathbf{K}_n^{-1}(\mathbf{g} - \mathbf{s}_2) \quad (2.21)$$

The log-likelihood ratio creates a new threshold test by which the ideal observer makes decisions on which hypothesis the data deems true:

$$\Delta \mathbf{s}^t \mathbf{K}_n^{-1} \mathbf{g} = \lambda'(\mathbf{g}) \begin{matrix} D_2 \\ > \\ < \\ D_1 \end{matrix} \lambda'_c \quad (2.22)$$

The log-likelihood ratio is important in helping to determine the SNR^2 expression for the ideal observer. This expression becomes a primary factor in validating the texture organization method outlined in the next chapter.

2.4.4 SNR^2 and the Ideal Observer

Given that the new threshold variable, $\lambda'(\mathbf{g})$, is a linear transformation of Gaussian random data, it, too, has a Gaussian distribution. This means that the SNR expression in section 2.3.7 still holds true for this test. For this expression, both the mean and variance of $\lambda'(\mathbf{g})$ must be calculated:

$$\langle \lambda(\mathbf{g}) | H_j \rangle = \langle \Delta \mathbf{s}^t \mathbf{K}_n^{-1} \mathbf{g} | H_j \rangle = \Delta \mathbf{s}^t \mathbf{K}_n^{-1} \mathbf{s}_j \quad (2.23)$$

$$\sigma_\lambda^2 = \langle [\lambda(\mathbf{g}) - \langle \lambda(\mathbf{g}) | H_j \rangle]^2 | H_j \rangle = \langle \lambda^2(\mathbf{g}) | H_j \rangle - \langle \lambda(\mathbf{g}) | H_j \rangle^2 \quad (2.24)$$

$$= \langle \Delta \mathbf{s}^t \mathbf{K}_n^{-1} \mathbf{g} \mathbf{g}^t \mathbf{K}_n^{-1} \Delta \mathbf{s} | H_j \rangle - \Delta \mathbf{s} \mathbf{K}_n^{-1} \mathbf{s}_j \mathbf{s}_j^t \mathbf{K}_n^{-1} \Delta \mathbf{s} \quad (2.25)$$

$$= \Delta \mathbf{s}^t \mathbf{K}_n^{-1} \Delta \mathbf{s} \quad (2.26)$$

These equations are plugged into the aforementioned SNR^2 equation to yield a new, more software computation-friendly expression:

$$\text{SNR}_\lambda^2 = \frac{[\Delta \mathbf{s}^t \mathbf{K}_n^{-1} \mathbf{s}_2 - \Delta \mathbf{s}^t \mathbf{K}_n^{-1} \mathbf{s}_1]^2}{\sigma_\lambda^2} = \frac{[\Delta \mathbf{s}^t \mathbf{K}_n^{-1} \Delta \mathbf{s}]^2}{\Delta \mathbf{s}^t \mathbf{K}_n^{-1} \Delta \mathbf{s}} \quad (2.27)$$

$$= \Delta \mathbf{s}^t \mathbf{K}_n^{-1} \Delta \mathbf{s} \quad (2.28)$$

This expression can be written in another software computation-friendly way when the noise is uncorrelated, due to a change in the covariance matrix:

$$[\mathbf{K}_n]_{mm'} = \sigma_m^2 \delta_{mm'} \quad (2.29)$$

$$\text{SNR}_\lambda^2 = \sum_{m=1}^M \frac{[\Delta s_m]^2}{\sigma_m^2} \quad (2.30)$$

Finally, if the noise variance is uniform, the expression simplifies even further:

$$SNR_{\lambda}^2 = \frac{1}{\sigma^2} \sum_{m=1}^M [\Delta s_m]^2 = \frac{\|\Delta s\|^2}{\sigma^2} \quad (2.31)$$

2.4.5 Karhunen-Loève (KL) Decomposition

Another way of simplifying software computation with regard to large vector and matrix products is to expand the data matrix in terms of its KL decomposition:

$$g = \sum_{m=1}^M \beta_m \phi_m = \Phi \beta \quad (2.32)$$

$$\text{where } \beta = \Phi^\dagger g \quad (2.33)$$

In this case, ϕ_m represent the eigenvectors of the data covariance matrix, whose coefficients are uncorrelated random variables. This means that the covariance matrix for the coefficients is a diagonal matrix, $\mathbf{K}_\beta = \mathbf{M}$, with elements μ_m .

Utilizing a similar technique, the mean difference in the data per hypothesis can be expressed:

$$\Delta s = \Delta \bar{g} = \Phi \Delta \bar{\beta} \quad (2.34)$$

These expressions and decompositions can be used to expand the SNR^2 to a vector/matrix linear equation, allowing further simplification to a summation, utilizing the elements of the diagonal matrix, \mathbf{M} :

$$SNR_{\lambda}^2 = \Delta s^t \mathbf{K}_n^{-1} \Delta s = [\Phi \Delta \bar{\beta}]^t [\Phi \mathbf{M}^{-1} \Phi^\dagger] [\Phi \Delta \bar{\beta}] \quad (2.35)$$

$$= \sum_{m=1}^M \frac{[\Delta \bar{\beta}_m]^2}{\mu_m} = \sum_{m=1}^M \frac{[(\phi_m, \Delta s)]^2}{\mu_m} \quad (2.36)$$

The equations and expressions detailed in this chapter are paramount in defining the ideal model observer used in the texture analysis and organization methods detailed in the following chapter and provide the foundation by which these methods are carried out and validated.

CHAPTER 3

Analysis Method

As stated in Chapter 1, modeling observer performance on large image sets can provide useful results in a number of ways, but it is also computationally taxing. Using the Hotelling observer – the ideal observer - to determine human signal detection performance on a set of just a few thousand images requires a fair amount of (relative) time, memory, and processing power. This chapter will explain how the texture analysis seeks to mitigate required time and computing power by finding the most relevant textures in determining human observer performance in signal detection and finding these relevant textures in subsequent image sets to more quickly find an approximation of the human observer signal detection performance.^[10]

3.0 User Defined Variables and Texture Modeling Method

Given that this theory was commissioned for practical use, the program developed from the following methods includes the use of variables that can be defined by the user. Therefore, it would make sense to have an idea from the start as to what kinds of practical numbers have been used in the development of this program, as well as the validations of the program, detailed in *Chapter 4*.

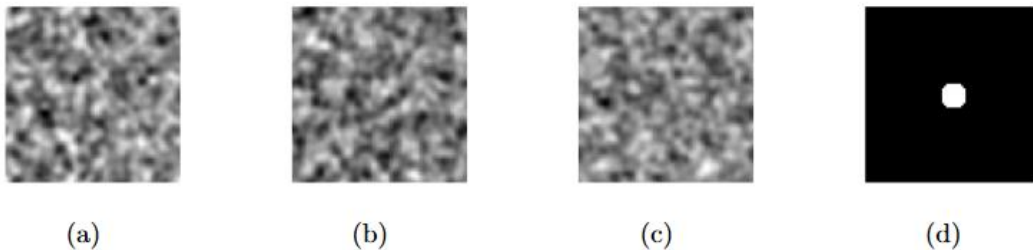


Figure 3.1(a)-(c): Example simulation images used to validate analysis methods. (d) Example signal injected into images for method validation.

The simulations used to formulate and validate this texture analysis are based on an average environment in which an observer, such as a radiologist, is viewing 5000 CT images with a region of interest (ROI) of square 82-pixel dimension around a centered circular signal on a monitor with a pixel pitch of $300\mu\text{m}$ from a viewing distance of 700mm (~ 27.6 in). The signal parameter – a value used to determine the size of the circular signal injected into the CT images (both simulated and physical phantom) – was set to a value of 0.6 and the texture parameter – used to determine the size of the features in the simulated image textures - was set to a value of 0.24. The modeled textures were created by convolving white noise with a gaussian blur function in the Fourier domain. For visual reference, *Figure 3.1* shows an average set of simulated textures next to the simulated signal inserted into the image sets for validation. These values will be repeated where applicable, but this is to give the audience a depiction of the situation being modeled throughout the descriptions of the following methods.

3.1 Textures

The detailed methods used in this analysis and organization rely on finding textures that are most useful in determining human observer performance in signal detection. Given this, it is important to understand what defines a texture, both in the sense of CT images and mathematically. This chapter seeks to define textures in a way that makes tangible, qualitative sense, as well as a way that makes mathematical sense that is useful for the proposed method of estimating human observer signal detection performance.

3.1.1 Governing Mathematics

The term “texture” in the sense of this thesis is the pattern in a 2D image brought about by various tissue microstructures, how x-rays interact with those microstructures, and artifacts brought on by the algorithms used in the image reconstruction stage. The microscopic structures of different types of tissues result in varying degrees of x-rays being absorbed and scattered. The different appearance of the different tissue types contain different

characteristics in the reconstructed images, resulting in the observer being able to qualitatively differentiate between different types of tissues in each reconstructed slice. Namely, bone tissue may appear as dense and bright while muscle tissue may appear more speckled or grainy in texture. Textures in a digital imaging sense are patterns within arrays of pixels with varying brightness. Each pixel contains a brightness value, yielding an array of numbers. This allows textures in digital imaging to be quantitatively characterized mathematically and statistically.

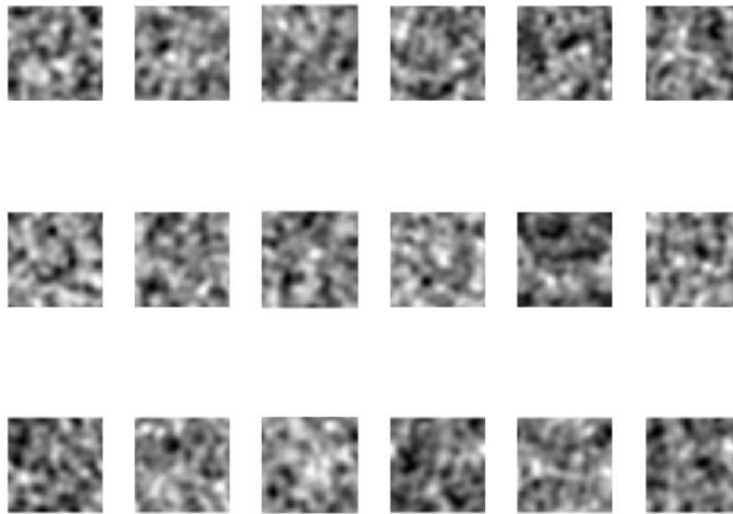


Figure 3.2: Example Set of Image Textures

Mathematically, textures can be described by second-order statistics. Namely, the mean of the pixel values, \mathbf{g} , and covariance matrix, $\mathbf{K}_{\mathbf{g}}$, are quantitative measures of texture characteristics. This makes some intuitive sense knowing that the mean conveys some amount of information about the general range of pixel values and the covariance matrix contains information of how similar each pixel is to the next. Namely, the covariance matrix can speak to the smoothness of a texture and the mean can speak to the general value that smoothness is hovering around.

These definitions are the basis upon which textures can be and have been modeled for testing the proposed analysis method. However, working with $\mathbf{K}_{\mathbf{g}}$ in its complete state can become computationally intensive, as the covariance matrix of an image is the same

dimensions of the original image. The covariance matrix on its own may also not offer immediately physical information of the texture itself at a glance. However, linear algebra can be used to decompose \mathbf{K}_g into the components that define the texture itself, which also provides a less computationally taxing expression of \mathbf{K}_g .

3.1.2 Return of the KL Decomposition

KL or Singular Value Decomposition (SVD) can be used to extract more tangible information about the textures in the following manner:

$$\mathbf{K}_g = \Phi \mathbf{D}(\lambda) \Phi^\dagger \quad (3.1)$$

Where the columns of Φ represent the different orthogonal texture components and $\mathbf{D}(\lambda)$ represents a diagonal matrix containing the eigenvalues, λ , of each corresponding texture component to the overall \mathbf{K}_g matrix. Where, normally, sorting these texture components by their corresponding eigenvalues might seem most intuitive, the next few sections show how these components can be resorted in a manner that shows their relative usefulness with regard to human signal detection performance. This decomposition of textures plays a key role, in part with the *Channelized Hotelling Observer* (CHO), in the analysis and organizing of CT image textures.

3.2 The Channelized Hotelling Observer (CHO)

The Hotelling Observer is the industry standard ideal observer - particularly for modeling human observer performance. However, the eye processes visual information through a series of channels that contain different spatial frequency-dependent response curves. These channels are what allow humans to quickly identify familiar objects and patterns. For instance, one can immediately know the outcome of a die roll without having to count each individual dot. The eye processes the familiar spatial frequencies in the pattern created by the dot positions on each face of a die. This being stated, adding the encoded spatial frequency-dependent channels with the appropriate sensitivities is a way to add more

accuracy in human observer performance estimation. As the name would suggest, utilizing these spatial frequency channels with the theory of the Hotelling Observer gives rise to the *Channelized Hotelling Observer*, which is used in the initial stage of the detailed texture analysis method.

3.2.1 Signal-to-Noise Ratio (SNR)

Being an ideal model observer, the math in section 2.3 holds for the CHO. Therefore, the decomposed texture covariance matrix from *Equation 3.1* can be inserted into the SNR^2 expression shown in *Equation 2.35* to form a nice linear expression for the CHO SNR^2 metric – a useful metric in signal detection for a texture analysis:

$$SNR^2 = s^\dagger K_g^{-1} s = s^\dagger \Phi D(\lambda)^{-1} \Phi^\dagger s \quad (3.2)$$

Where s denotes the signal within the texture that the observer is tasked with detecting. Having decomposed K_g makes the calculation of K_g^{-1} trivial, as this only equates to inverting $D(\lambda)$ and the inverse of a diagonal matrix is a simple inverting of each individual diagonal element. This decomposed expression can be rewritten into a summation for a more informative metric of an overall texture:

$$SNR^2 = \sum_{j=1}^J \frac{(s_j^\Phi)^2}{\lambda_j} \quad (3.3)$$

Where s_j^Φ is the j^{th} component of the signal in the basis of the texture and λ_j denoting the singular value of the j^{th} texture component, again, which is directly proportional to how prominent the component is in the overall texture. This equation yields a scalar valued performance metric for the CHO. This is important for determining a field standard metric for use in comparing against the full texture analysis method used in this paper. However, as stated in *Preface 3.2*, before a reliable metric can be formed, the CHO must be adjusted to reflect the response of the human visual system. This is achieved by using the *eye filter*.

3.2.2 Utilizing the CHO

Given that the CHO is the ideal observer and the human observer is not, the CHO will not outright yield accurate human observer performance. In order to achieve more accurate performance to that of a live human observer, the channels of the CHO must be “tuned” to the channels of the actual human eye. This is achieved by filtering the signal in the Fourier (or spatial frequency) domain. It has been well documented that the use of the following *eye filter* equation can be applied to the signal, in part with the CHO, to yield more accurate human observer performance^[11,12]:

$$E(\rho) = \rho^a \exp(-b\rho^c) \quad (3.3)$$

Where ρ denotes radial spatial frequency and a , b , and c are “shaping factors”, so to speak. Since the eye filter isn’t precisely measured, a , b , and c are used to massage the curve to the shape of the eye filter measured in previous studies. In the case of this project, the values a , b , and c are 1.2, 0.3, and 1, respectively, as these numbers have been used in prior studies. This filter is applied to the signal injected into the image sets being analyzed by this analysis method, resulting in the SNR^2 values obtained by the CHO tuned to the human observer:

$$SNR^2 = \sum_{j=1}^J \frac{(\tilde{s}_j^\phi)^2}{\lambda_j} \quad (3.4)$$

Where \tilde{s}_j^ϕ denotes the filtered signal component in the texture basis.

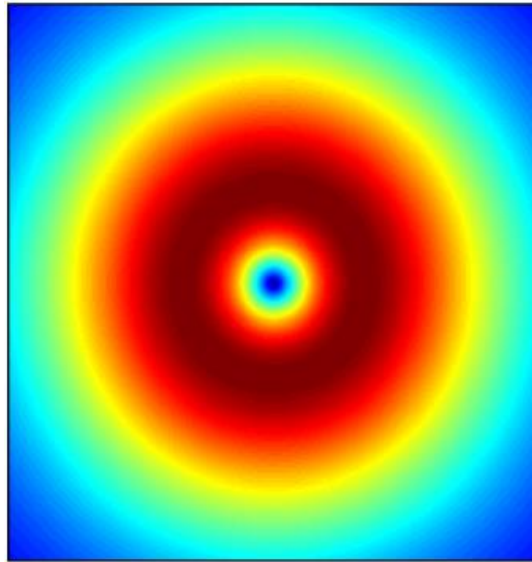


Figure 3.3: The Eye Filter
(Model Observers for Complex Discrimination Tasks: Assessments of Multiple Coronary Stent Placements 2010)

One thing to consider is that the eye filter must be placed over a larger area than that of the ROI. In practice it was found that six times the ROI dimensions in each direction was found to work well. Namely, instead of 82x82 pixels, the eye filter would be placed over a signal created in a 492x492 pixel space. This allows for a finer resolution in the Fourier space where the eye filter is applied.

3.3 Organization is Key

Just as in *Equation 3.2*, the components of *Equation 3.4* still have the same respective meaning regarding the texture components within the overall texture and their corresponding eigenvalues. Namely, just as the j^{th} singular value, λ_j , in *Equation 3.2* directly relates to the j^{th} texture component of the overall texture, it also does in *Equation 3.4*.

These textures are sorted from highest to lowest SNR^2 value calculated in *Equation 3.4*, resulting in an array of most useful to least useful textures regarding signal detection. The proposed method only uses the top L textures to analyze the set of test images because those

top L textures make up the bulk of the filtered covariance matrix. L is a user-defined number of textures. In the case of these simulations, L was chosen to be 50:

$$SNR^2 \approx \sum_{l=1}^L \frac{(\tilde{s}_l^\Phi)^2}{\tilde{\lambda}_l} \quad (3.5)$$

Equation 3.5 defines the SNR^2 approximation, using only the top L most useful textures (regarding human signal detection performance) containing an eye filtered signal. For instance, if 5,000 images were used to train the algorithm with the CHO, maybe only the 50 most contributing textures are needed for this analysis, as any remaining, less prominent texture components.

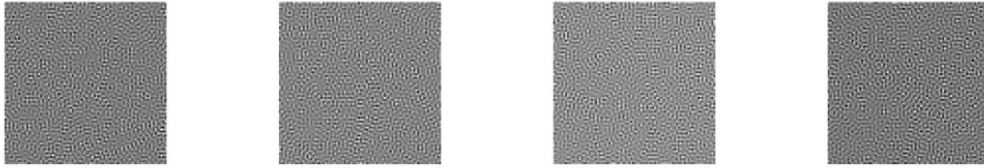


Figure 3.4: Example of Top 4 Most Useful Textures

Figure 3.4 shows some examples of prominent textures within a set of simulated data. They look very similar to each other and all contain high frequency features. This suggests that high frequency textures are generally inobtrusive to human signal detection performance. At this point, the top L textures can be applied to a new set of test images for faster human observer performance evaluation.

3.4 Utilizing Useful Textures

After finding the most useful textures and organizing them from most to least useful, the top L textures can be applied to a new set of images to test by way of multiplication:

$$\mathbf{v} = \tilde{\Phi}^\dagger \mathbf{g} \quad (3.6)$$

Where \mathbf{v} is the resulting images of the filtered texture components, $\tilde{\Phi}^\dagger$, multiplied by the new test images, \mathbf{g} . If any of those L textures exists in any of the images in \mathbf{g} , they will

be accentuated and used to more quickly find the observer performance in a similar method to that used in the evaluation of the SNR^2 metric:

$$\alpha = [\tilde{s}^\Phi]^\dagger K_v^{-1} [\tilde{s}^\Phi]^2 \approx \sum_{l=1}^L \frac{[\tilde{s}^\Phi]^2}{[K_v]_u} \quad (3.7)$$

One other thing to consider is that the SNR^2 value is now scaled differently after applying the top L textures from the original training image set. Therefore, what's being calculated is no longer exactly the SNR^2 , but a surrogate metric, α . In short, this means that α is a relative metric, rather than an absolute metric, like a true SNR^2 value. Despite this, this metric can still be used to determine the usefulness of the texture components regarding signal determination.

3.5 Example of Analysis Methods

As an example, if some set of 5,000 texture images (phantom or simulated), \mathbf{g}' , were to be analyzed for human observer signal detection performance, first, an initial set of training images \mathbf{g} would be injected with a circular signal and decomposed using SVD or KL decomposition. The eye filter would then be applied to the decomposed textures to yield a useful human observer performance estimation via *Equation 3.5*. After, the top 50 most useful texture components can be extracted from the overall texture and applied to a new set of test images \mathbf{g}' containing another eye filtered signal to find the amount of each of those 50 useful texture components contained in the new image set via *Equation 3.6*. The human observer signal detection performance on the new images set can be estimated via *Equation 3.7*, as an SNR^2 surrogate value.

3.6 Code Walkthrough

- ▶ Create or input training images
- ▶ Perform SVD on covariance matrix
- ▶ Eye filtered signal is added to SVD result of input images
- ▶ Extract SNR^2 metric on “useful” textures from training images
- ▶ Input/create new set of test images
- ▶ Add a different eye filtered signal to this set of input images
- ▶ Multiply top L textures to test images
- ▶ Extract SNR^2 metric in the same manner as before
- ▶ This SNR^2 -equivalent represents a human observer performance on the new set of images much faster than the method used on the training images/using the Hotelling observer model

CHAPTER 4

Validations

With proposing a new method of estimating previously validated methods comes the required validation of said new method, including comparing performance between the new and previously validated method. This chapter details the methods used to both qualitatively and quantitatively validate the proposed model observer, including a comparison study between the proposed method and the Hotelling observer performance.

4.1 Reconstruction Comparison

Section 2.2 states how observer task performance is based on image quality.^[13-15] Thus, one validation method used involved using physical phantom data reconstructed using two

different reconstruction methods. Two phantom image sets were reconstructed using filtered back projection (FBP) and another two sets of the same images were reconstructed using iterative reconstruction (IR), for a total of four CT image sets each containing 200 images. These four image sets were used to make a total of 16 α calculations – each calculation with a different training/test set combination:

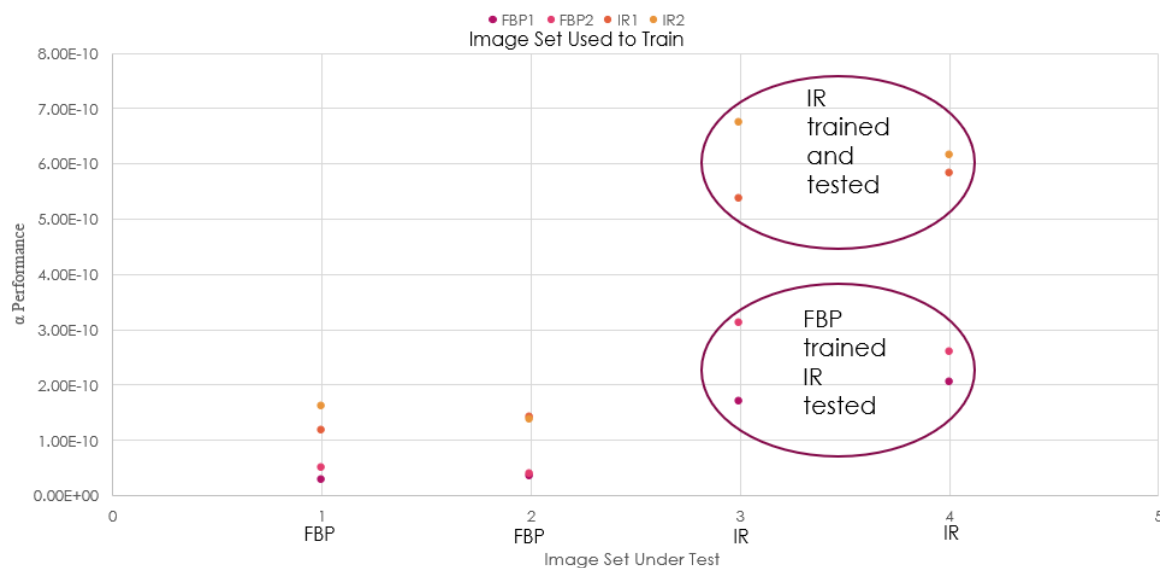


Figure 4.1: α Calculations for Different FBP/IR Training/Test Image Set Combinations

Chapter 1 mentions the quality comparison between IR and FBP reconstruction methods. IR images are less subject to noise and have generally favorable image quality compared to FBP reconstructed images. Figure 4.1 shows the resulting α values for the different training/test combinations of the four FBP/IR image sets. The first two columns show that the FBP evaluated image sets yielded a higher α value when trained with the IR image sets. Likewise, the last two columns also show that the IR evaluated image sets yielded a higher α value when trained with IR image sets.

In short, this test showed that the human observers are more likely to detect signals present in IR images as opposed to FBP reconstructed images. With the quality comparisons mentioned in Chapter 1 in mind, it comes also as no surprise that the IR trained and tested image sets performed much better than any other combinations of training/test image sets

(Note the log scale in *Figure 4.1*). This provides validation that the texture analysis method used is a reliable metric for modeling human observer signal detection performance relative to image quality.

4.2 Comparison to the Hotelling Observer

Since the performance metric of the proposed method is not an absolute SNR^2 metric and the previous validation only provided validation based on relative image quality, comparing direct image-by-image α performance values to the respective d^2 values resulting from the CHO method in an absolute sense are not of any use in validating this method. Namely, comparing the α values from the analysis and estimation methods detailed in this paper have no absolute correspondence to the d^2 values extracted from the CHO of the same image set. However, ensuring that the two methods trend together, per texture parameter (mentioned in *Section 3.0*), is useful.

The following plot shows how the CHO (right), the texture analysis method using only the top 50 textures of a 5,000 image training set (left, red), and the texture analysis method using all 5,000 textures from the training image set (left, green) perform as a function of the aforementioned texture parameter:

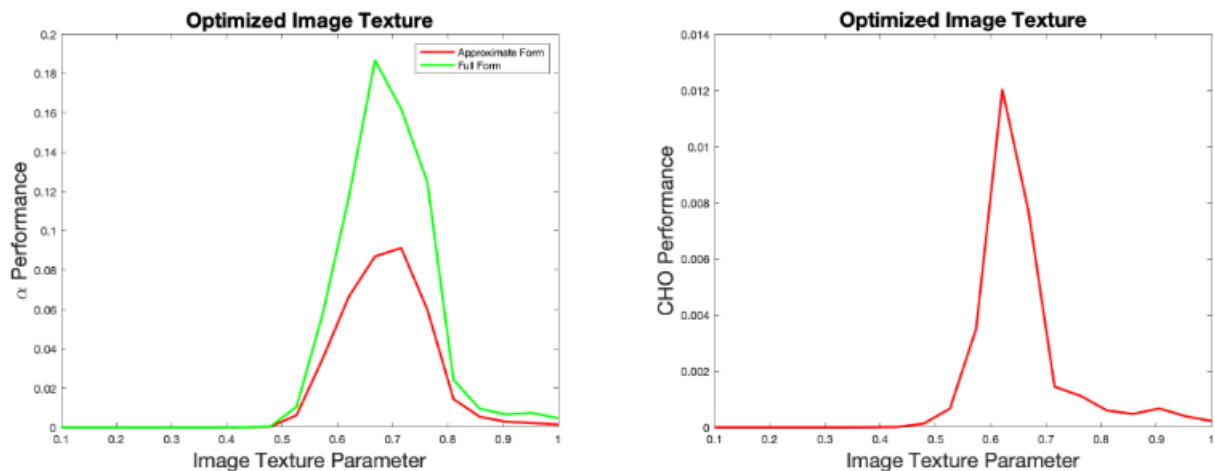


Figure 4.2: Human observer signal detection performance comparison with respect to texture feature size (x-axis) and between texture analysis method in full form (left, green), texture analysis method using only the top 50 most prominent textures (left, red), and the CHO (right)

Figure 4.2 shows the full form (containing all texture components) has a very similar shape to the CHO plot, with only a slightly offset peak. The approximate form, using only the top 50 textures, has a similar shape and peaks around where the full form peaks. Note the different scaling in the y-axes between both plots in *Figure 4.2*. Again, the α and d^2 values are not the same due to the training textures being applied to the data in the α calculation. Despite this, the similar peak and curve shape is a good indication that the method is reliable, given the proper amount of textures, L , are used from the training image set, as the observer performance results of the texture analysis method trend similarly to the field standard method.

CHAPTER 5

Discussion

A method for analyzing textures in CT images and organizing their components based on their contributions to the overall texture regarding human observer signal detection has been presented. The history and theory leading up to the formulation of this method were presented to give context of the usefulness of this analysis method, followed by the theory of the method itself and the methods used to validate the theory. The results presented show that this method trends with the field standard human observer signal detection performance method given by the CHO. This shows potential for this method to be used to quickly produce a surrogate figure of merit for optimizing CT imaging techniques, including radiation dosing, as well as image reconstruction algorithms.

Given that only simulated and phantom images were used in the testing and validation of this analysis method, moving forward, real CT images of actual living test subjects would help to validate the true usefulness of this method using more practical data. Moreover, analyzing more image sets using this method could prove useful in determining whether a simple scaling factor can be placed on the α metric to find a more comparable value to the d^2 metric provided by the CHO or if some other method could be used to

produce a more immediately tangible absolute piece of information from the current α metric. More tested image sets analyzed with this method could also produce a tangible sense of just how much time and money this method could save in the CT imaging optimization process.

REFERENCES

1. Wikipedia (2020): <https://en.wikipedia.org/wiki/X-ray>
2. Wikipedia (2020): https://en.wikipedia.org/wiki/Wilhelm_R%C3%B6ntgen
3. Wikipedia (2020): https://en.wikipedia.org/wiki/Godfrey_Hounsfield
4. Wikipedia (2020): https://en.wikipedia.org/wiki/Allan_MacLeod_Cormack
5. Imaginis (2020): <https://www.imaginis.com/ct-scan/brief-history-of-ct>
6. Zeng, G. (2010), *Medical Imaging Reconstruction*, Higher Education Press, Beijing, and Springer-Verlag Berlin Heidelberg
7. Kupinski, M. (2010), *Implementation of Observer Models in The Handbook of Medical Image Perception and Techniques*, Cambridge University Press, pp. 251-258
8. Barrett, H. (2008), Slides on *Introduction to Image Quality*
9. Barrett, H. and K. Myers (2004). *Foundations of Image Science*, John Wiley and Sons, Inc.
10. Kupinski, M., Z. Garrett, and J. Fan (2020), *Observer-driven Texture Analysis in CT Imaging*, SPIE Medical Imaging Conference Proceeding
11. Zhang, S., C. Abbey, A. Teymorian, X. Da, J. Whiting, and M. Eckstein (2010), *Model Observers for Complex Discrimination Tasks: Assessments of Multiple Coronary Stent Placements*, SPIE Medical Imaging
12. A. E. Burgess, "Statistically defined backgrounds: performance of a modified nonprewhitening observer model," *J. Opt. Soc. Am. A*11, 1237-1242, 1994.
13. Tseng, H.-W., J. Fan, and M. Kupinski (2016), *Design of a Practical Model-Observer-Based Image Quality Assessment Method for X-Ray Computed Tomography Imaging Systems*, SPIE Medical Imaging
14. Tseng, H.-W., J. Fan, M. Kupinski, P. Sainath, and J. Hsieh (2014), *Assessing Image Quality and Dose Reduction of a New X-Ray Computed Tomography Iterative Reconstruction Algorithm Using Model Observers*, *Med. Phys.* 41
15. Barrett, H., K. Myers, C. Hoeschen, M. Kupinski, and M. Little, *Task-Based Measures of Image Quality*, PMB Review Final
16. Code Project: <https://www.codeproject.com/Articles/1113200/Computed-Tomography-Generation-of-the-Sinogram-Ima>