STEGANALYSIS OF ADDITIVE NOISE MODELABLE INFORMATION HIDING

By

Jeremiah Joseph Harmsen

A Thesis Submitted to the Graduate

Faculty of Rensselaer Polytechnic Institute

in Partial Fulfillment of the

Requirements for the Degree of

MASTER OF SCIENCE

Approved:

William A. Pearlman
Thesis Adviser

Rensselaer Polytechnic Institute
Troy, New York

April 2003
(For Graduation May 2003)

© Copyright 2003
by
Jeremiah Joseph Harmsen
SETEC Astronomy
All Rights Reserved

CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGMENTS

Ann and Allen Harmsen: Hi low move.

Joshua Harmsen: Befit my nerds.

Prof. Pearlman: I owe key dials.

Katie Toohey: A spy we meet.

Peter Greenauer and Andrew Burdick: They're Zen poise.

David Watt and Mark Unrath at Electro Scientific Industries: Visit a spy web above.

ABSTRACT

This thesis presents a steganalysis of additive noise modelable information hiding[1]. The process of information hiding is modeled in the context of additive noise. Under an independence assumption, the histogram of the stegotext is a convolution of the noise probability mass function (PMF) and the original histogram. In the frequency domain this convolution is viewed as a multiplication of the histogram characteristic function (HCF) and the noise characteristic function. Least significant bit, spread spectrum, and DCT hiding methods for images are analyzed in this framework. It is shown that these embedding methods are equivalent to a lowpass filtering of histograms that is quantified by a decrease in the HCF center of mass (COM).

Table 1: Notation

α	fraction of pixels used for embedding
$h_\alpha[n]$	stegonoise histogram with embedding rate α
$h_s[n]$	stegonoise histogram
$h_c[n]$	coverimage histogram
$f_\Delta[n]$	stegonoise probability mass function
$f_\Delta(x)$	stegonoise probability density function
$H_s[k]$	stegoimage histogram characteristic function
$H_c[k]$	stegoimage histogram characteristic function
$F_\Delta[k]$	stegoimage histogram characteristic function
$\mathcal{C}(\cdot)$	center of mass
$\mathcal{C}_k(\cdot)$	center of mass along kth axis
$DFT(\cdot)$	Discrete Fourier Transform
$Pr\{\cdot\}$	probibility of event $\cdot$
$\mathcal{N}(\mu, \sigma^2)$	normal distribution with mean μ and variance σ^2
$\lvert\cdot\rvert$	magnitude of $\cdot$

A man can hide all things, excepting twain–
That he is drunk, and that he is in love.
- **Antiphanes of Macedonia**, Fragmenta

CHAPTER 1
Introduction

1.1 Why Steganography?

Since the beginnings of human communication, the desire to communicate in secrecy has existed. Whether planning a surprise birthday party or overthrowing a government, exchanging data in secret is essential. There have been many solutions to this problem, the most widely used and investigated being cryptography [2][3][4].

Historically, sensitive information has been protected using encryption. Encryption uses powerful mathematics to map plaintext into an unreadable cyphertext that is sent over a channel to the recipient. When a message is encrypted it is done so using a secret key. To decrypt a message, the secret key is used to reverse the process. For an easedropper to defeat the system he or she must acquire the secret key. Typically it is assumed that this must be done by searching over the entire keyspace; a so called "brute-force" attack. As this is a very time consuming endeavor, the encrypted message is considered safe.

A second method of communication, called steganography offers data protection in a somewhat different manner. Steganography offers security similar to cryptography in that, if an adversary does not know information is being transfered, he cannot intercept and read it. While keeping the contents of a message secret is desirable in many cases, steganography has a much more powerful use: steganography hides the very fact that a communication is taking place.

The distinction between cryptography and steganography is an important one, and is summarized by the following:

> Encryption prevents an unauthorized party from discovering the *contents* of a communication. Steganography prevents discovery of the very *existence* of a communication.

A simple example shows where the use of a covert channel is applicable.

1.1.1 Alice and Bob

Consider our old friends Alice and Bob. Alice and Bob have been placed in a jail guarded by Warren the Warden. Alice and Bob are planning an escape, with Alice digging out a tunnel under the fence. If Alice sends Bob an encrypted message about the progress of the escape plan:

$$IM\ AHEAD \rightarrow ORRETBBQ$$

Warren will easily see the message "`ORRETBBQ`" but, it may be extremely hard interpreting nonsensical digits.

The oddity of Alice and Bob sending seemingly random letters back and forth may be enough to make Warren suspect something is amiss. If Alice and Bob had been using steganography, they would have concealed the existence of a communication. For example, in the phrase: "it may be extremely hard interpreting nonsensical digits," if Bob takes the first letter of each word, he receives Alice's message "`IM BEHIND`".

1.1.2 History

Steganography literally means covered writing. Herodotus provides the first records of steganography in Greece [5]. To communicate Greeks would etch the message they wished to send into the wax coating of a wooden tablet. The tablet would then be transported to the recipient who would read the message, then remelt the wax to etch their reply. In order to communicate in secret, the army would remove the wax completely, carve the secret message into the wood, and re-coat the tablet with wax. The apparently unused tablet would then be sent to the recipient who would remove the wax to view the message.

In another primitive example of steganography from Persia, a messenger's head would be shaved and a message tattooed onto the scalp. Once the messengers hair had re-grown, he or she was able to pass by unsuspecting sentries to deliver the message.

1.2 Modern Steganography

Modern uses of steganography fall into one of two broad categories, watermarking and communication. While watermarking has enjoyed a great deal of use in digital rights management, steganographic communication systems are only now receiving academic attention.

1.2.1 Watermarking

There has been a large amount of work in embedding data to maintain ownership of digital media[6][7][8]. In this case the watermark should be as resistant to as many distortions as possible. These distortions include both intentional and unintentional alterations of the media. For example, in a collusion attack, numerous parties with different copies of the same watermarked media, may perform an averaging operation to try and estimate the original media.

Another type of watermark is used to determine if a file has been altered[9]. These watermarks are different in the respect that they are very fragile; any changes in the media should cause them to break. These fragile watermarks can be used to detect malicious tampering.

1.2.2 Communication

Using steganography as a viable form of communication has been propelled largely by the growth of the Internet. The Internet offers an opportunity to exchange large amounts of digital information over great distances. The prevalence of media such as audio, video, and images on the Internet provides an ideal channel for steganographic communication.

1.3 Steganalysis

The broad goal of steganalysis is to understand the effects of hiding data into a medium. This knowledge is typically used to either strengthen the hiding system or detect the use of data hiding.

In order to develop a hiding scheme which is difficult to detect, it is necessary to analyze the results of prospective methods. This is typically done by comparing

statistical changes introduced when embedding data. If a method causes distinct predictable changes it will be fairly easy to detect and should be modified.

The detection of steganographic communication is a very important application of steganalysis. The loss of sensitive data, by both civilian and military entities, is very undesirable. As steganography provides a means for covertly transferring information, it is especially well suited to an insider sneaking information out.

1.3.1 Steganographic Stealth

The concept of the performance of a hiding scheme is a difficult concept. There have been numerous theories on defining steganographic performance [10][11][12][13].

In cryptography, the encryption strength of a method is usually cited as conveying this information. For example, a system that is computationally difficult to break is considered to be a "strong" system. Analogously, we introduce the term steganographic stealth to represent the relative performance of a steganographic system. A scheme which is difficult to detect is denoted as a "high stealth" or "stealthy" scheme. Of course these terms are defined in relation to the detection methodologies. For example, under visual detection, Least Significant Bit embedding has a high stealth, whereas under statistical analysis it is very low[14].

1.3.2 Current Detection Schemes

Many current detection methods rely on detecting the "thumbprint" of a particular embedding method [15][16][17][14][18][19]. Generally speaking, specific hiding methods are analyzed for the changes they make to the image or anomalies in a resulting file. These methods suffer from a very fundamental flaw, that being they are always one step behind. In order to implement a detection scheme, the hiding method must be known and well understood. When considering a steganalysis scheme applicable in the real world, one needs to consider the possibility that an adversary will use an unpublished algorithm. While the casual steganographer may be content to use off the shelf tools, one would certainly expect that an agent for a government with greater resources would take advantage of the additional security in an unpublished scheme.

This creates the potential for an adversary to use an unknown hiding scheme to avoid detection. For these reasons a different model should be used in steganalysis. This thesis investigates using statistics found naturally in images to form a model of a "natural" image. Once we have this model, we can assume that modifying an image will noticeably alter these statistics. If this is the case, we can flag an image as being abnormal. Once this is done, expanded efforts can be focused toward extracting the message.

1.3.3 Contributions of this thesis

This thesis begins by taking the position that steganalysis should be separated from the hiding algorithms. With this goal in mind, a class of data hiding algorithms called additive noise modelable is defined. These hiding algorithms are characterized by an equivalence to the independent addition of noise to a coverimage. This broad definition allows for the generalization of the analysis to a number of hiding algorithms.

The first order statistical changes caused by these algorithms are derived using probability theory. Descriptions of the histogram of a stegoimage in relation to the coverimage and additive noise are presented. In addition a center of mass metric is shown to be sensitive to additive noise embedding, and bounds on this metric are proved for certain classes of additive noise.

Finally the theoretical results are verified experimentally by creating two detection schemes. The first detection scheme allows for detection when the embedding method is known, while a second method is presented using only an estimate of the coverimage properties.

CHAPTER 2
Data Hiding as Additive Noise

The motivation to model the steganographic process as the addition of noise arises from a number of factors. In the process of sampling and transmitting signals there are numerous sources of noise such as quantization[20], sensor[21], and channel[22]. A number of steganographic hiding schemes have used these noise sources as a foundation for noise based data hiding. The goal is to disguise the message as a naturally present noise and add it to the coverimage. To this extent a generalized additive noise scheme has been developed in [23] that is able to embed data with any given distribution.

While the additive noise framework is especially well suited to schemes which rely on noise based embedding, it may be easily generalized to any method that results in the addition of independent and identically distributed noise. Sampled signals have a large amount of correlation present- both from the natural statistics of the original signal and the sampling device. If data is hidden without regard to this correlation, it can be considered as an foreign disturbance which corrupts the image. This formulation allows us to model many hiding methodologies which do not directly rely on additive noise.

2.1 Modeling

A model of a general steganographic system is shown in Figure 2.1. The embedding process begins with the selection of a *coverimage*, in which the message

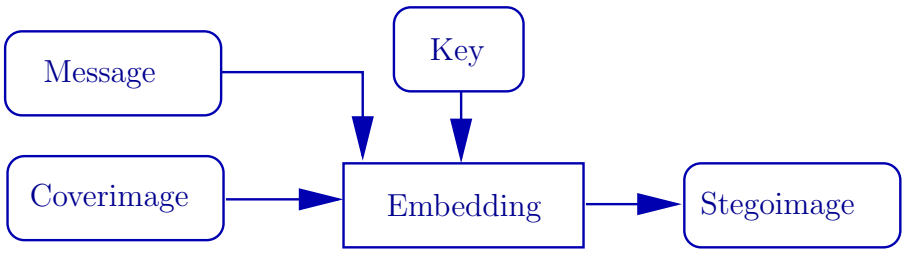

Figure 2.1: Overview of the embedding process

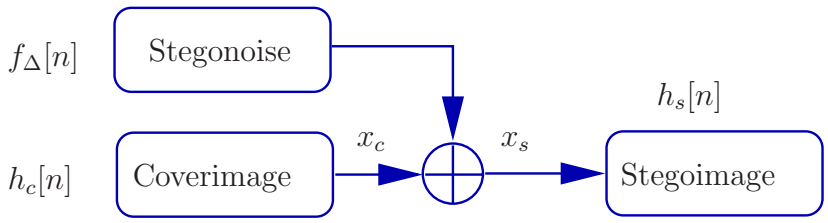

Figure 2.2: NM Steganography Model

will be hidden. The coverimage can be any type of image and is chosen to resemble a typically transmitted image, so to avoid raising suspicion. The information to be transfered is called the *message* and assumed to be a general stream of binary values. The message is placed in the coverimage through the process of *embedding*. During embedding a keying variable, or *key* may be used. This key is available to both the sender and the recipient and is used to synchronize the hiding and recovery process. The result of embedding the message in the coverimage is called the *stegoimage*.

The additive noise model seeks to represent this general system as one shown in Figure 2.2. Again we begin with the coverimage, having histogram $h_c[n]$. The *stegonoise* is a representation of the message as a psuedo-random sequence that is constructed to resemble noise. In the additive noise model the stegonoise is i.i.d. and is completely characterized by it's probability mass function (PMF), $f_\Delta[n]$. The embedding process in the additive noise model is simply the addition of the stegonoise to the coverimage. The resulting stegoimage has a histogram $h_s[n]$.

The justification for the additive noise model will be presented in Chapter 4, where a number of hiding algorithms are placed in the context of this model.

2.1.1 Stegonoise Probability Mass Function

The stegonoise probability mass function is the distribution of the additive noise defined as,

$$f_\Delta[n] \triangleq p(x_s - x_c = n), \quad n = 0, \pm 1, \pm 2, \ldots \tag{2.1}$$

Where x_s is the pixel value after embedding, and x_c is the pixel value prior to embedding. Generally speaking, $f_\Delta[n]$ is the probability that a pixel will be altered

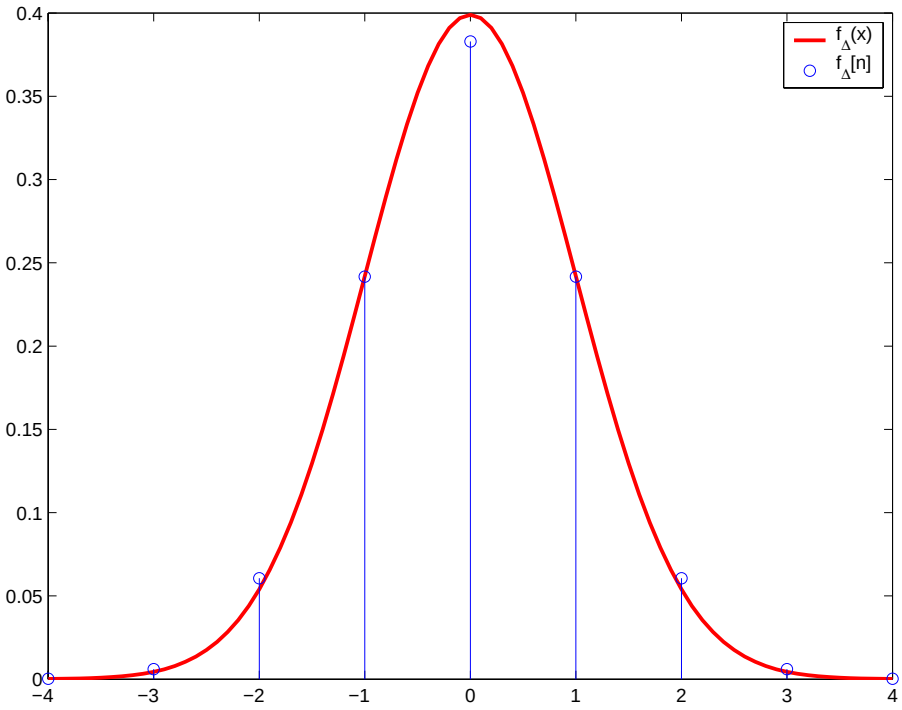

Figure 2.3: $f_\Delta(x)$ **and** $f_\Delta[n]$ **for** $\mathcal{N}(0,1)$

by n. In this model it is assumed that the noise acts independently on each pixel. So $f_\Delta[0]$ is the probability that, after embedding, a pixel is unchanged. Whereas $f_\Delta[-1]$ is the probability that the pixel is decreased by one.

2.1.2 Discretization

Many times it is more convenient to work with a continuous probability density function, $f_\Delta(x)$, rather than the discrete probability mass function. Of course, when digital media is stored, the values must be quantized to a finite number of bits. If this quantization occurs, we assume that a rounding operation is used to minimize the quantization error. When this is the case, we can consider transforming the *pdf* into a PMF using,

$$f_\Delta[n] = \int_{n-0.5}^{n+0.5} f_\Delta(x)\,dx. \tag{2.2}$$

The continuous $f_\Delta(x)$ and discretized $f_\Delta[n]$ for $\mathcal{N}(0,1)$ are shown in Figure 2.3.

2.2 Effects Of Additive Noise

We are interested in the effect which additive noise has on the statistics of a signal. We are primarily interested in these changes and how they can be used to flag suspicious images. The histogram, $h[n]$, of an image is the frequency count of the pixel intensities present in an image, defined as,

$$h[n] = \sum_{n_1,n_2} I\left(n, x\left(n_1, n_2\right)\right), \tag{2.3}$$

where

$$I\left(n, x\left(n_1, n_2\right)\right) = \begin{cases} 1, & n = x\left(n_1, n_2\right) \\ 0, & else. \end{cases}$$

We use $h[n]$ as an estimate for of the PMF that generated the pixel intensities in an image as the histogram is simply the PMF multiplied by the number of pixels in the image.

When the stegonoise is added to the image it is assumed that it alters each pixel. The amount by which each pixel is modified is a discrete random variable with PMF $f_\Delta[n]$. Using the stegonoise PMF $f_\Delta[n]$, and the cover histogram $h_c[n]$ we can form an estimate of the stegoimage histogram, $h_s[n]$. Theorem 2.2.1 gives this relation.

Theorem 2.2.1. *In a hiding system where the additive noise is i.i.d. and independent of the coverimage, the histogram of the stegoimage is equal to the convolution of the stegonoise PMF and the coverimage histogram,*

$$h_s[n] = h_c[n] * f_\Delta[n]. \tag{2.4}$$

Proof. Consider the histogram as a probability mass function multiplied by a constant. From probability theory [24], we know the addition of two independent random variables results in a convolution of their probability mass functions. □

From Theorem 2.2.1 we see that the effect of the additive noise on the image histogram is equivalent to a convolution of the stegonoise PMF and the cover histogram. Figure 2.4 shows $h_c[n]$ (original histogram), $h_s^{est}[n]$ (estimated histogram),

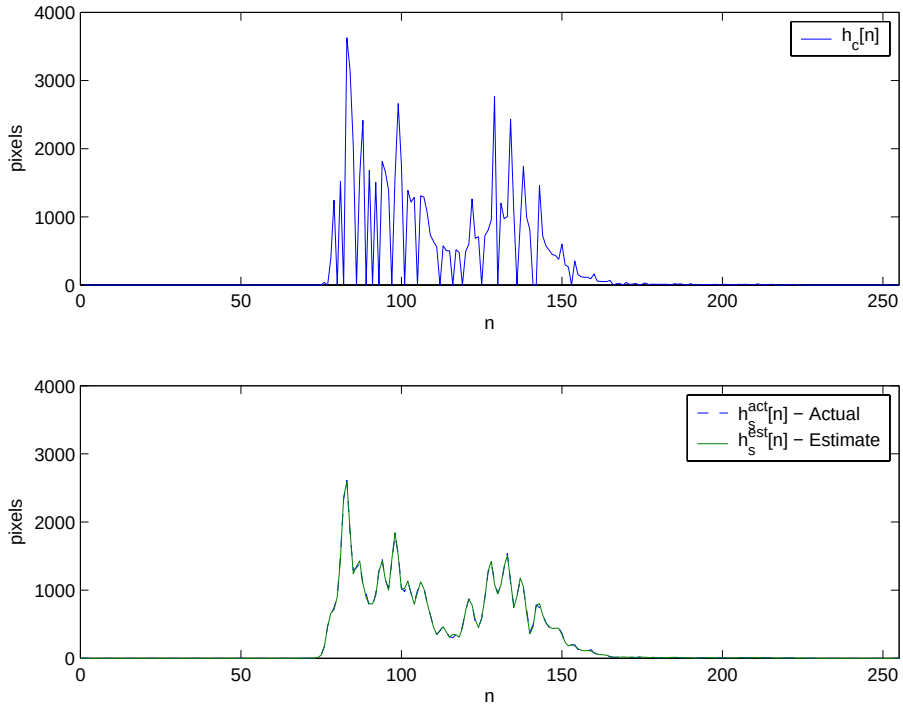

Figure 2.4: Original Histogram (top), Estimate Noisy Histogram and Actual (bottom)

and $h_s^{act}[n]$ (actual histogram) with additive noise, $\mathcal{N}(0,1)$ for the image shown in Figure 2.5.

Using the Discrete Fourier Transform[25], given in (2.5), we can gain insight into the frequency components of a histogram.

$$X[k] = DFT(x[n]) = \sum_{n=0}^{N-1} x[n] e^{-\frac{2\pi jnk}{N}}. \tag{2.5}$$

Here, N equals the largest value possible in the intensity of the image. For example, in an 8 bit grayscale image N would be 2^8 or 256.

By taking the DFT of the PMFs involved, we have the characteristic functions

Figure 2.5: Pout.tif

defined as,

$$F_\Delta[k] \triangleq DFT\left(f_\Delta[n]\right), \tag{2.6a}$$

$$H_c[k] \triangleq DFT\left(h[n]\right), \tag{2.6b}$$

$$H_s[k] \triangleq DFT\left(h_s[n]\right). \tag{2.6c}$$

It should be noted that these are only approximations of the characteristic functions, just as the histogram is an approximation of the PMF.

These characteristic functions will be central in additive noise steganalysis. In particular the DFT of a histogram will be referred to as the histogram characteristic function, or $\mathcal{HCF}$.

We can rewrite (2.4) in the frequency domain as the following corollary,

Corollary 2.2.1 ($\mathcal{HCF}$ Multiplication). *In a hiding system where the additive noise is i.i.d. and independent of the coverimage, the $\mathcal{HCF}$ of the stegoimage is equal to the multiplication of the stegonoise characteristic function and the coverimage $\mathcal{HCF}$,*

$$H_s[k] = F_\Delta[k]H_c[k]. \tag{2.7}$$

Proof. Taking the DFT of (2.4) the convolution becomes a multiplication in the frequency domain. □

This formulation gives us an insight into how embedding a message alters the $\mathcal{HCF}$ of an image. This will be particularly useful in the steganalysis of images explored in Section 4.

Thus far it has been assumed that the additive noise has operated on each pixel in the image. In practice the embedding rate may be reduced for a number of reasons. The most likely is to increase the stealth of a hiding method. The following assumes that when only a fraction, α, of the pixels are used for embedding, these pixels are randomly chosen from the entire image. This prevent spatial-statistical attacks such as those discussed in [26].

Theorem 2.2.2 (α-Bitrate Embedding). *In a system where α is the fraction of pixels chosen at random for embedding and the stegonoise is i.i.d. and independent of the coverimage. The stegoimage histogram is given by,*

$$h_\alpha[n] = \alpha\,(h_s[n]) + (1-\alpha)\,h_c[n]. \tag{2.8}$$

Proof. Letting C be the number of pixels in the image, $Pr\{x_\alpha = n\}$ be the probability that a pixel in the α rate stegoimage is valued n, and denoting an embedding pixel as "*e.p.*" and an unchanged pixel as "*u.p.*", we have,

$$\begin{aligned}
h_\alpha[n] &= CPr\{x_\alpha = n\} && \text{(2.9a)}\\
&= C\,\{Pr\{x_\alpha = n|e.p.\}Pr\{e.p.\} + Pr\{x_\alpha = n|u.p.\}Pr\{u.p.\}\} && \text{(2.9b)}\\
&= C\,\{\alpha Pr\{x_\alpha = n|e.p.\} + (1-\alpha)Pr\{x_\alpha = n|u.p.\}\} && \text{(2.9c)}\\
&= \alpha\,(f_\Delta[n] * h_c[n]) + (1-\alpha)h_c[n] && \text{(2.9d)}\\
&= \alpha h_s[n] + (1-\alpha)h_c[n]. && \text{(2.9e)}
\end{aligned}$$

□

An illustration of this linearity is shown in Figure 2.6. In this figure we observe the contents of three histogram bins, (115, 125, and 135), as the embedded pixel rate, α, is varied from 0.0 to 1.0. The embedding method used is SSIS described in

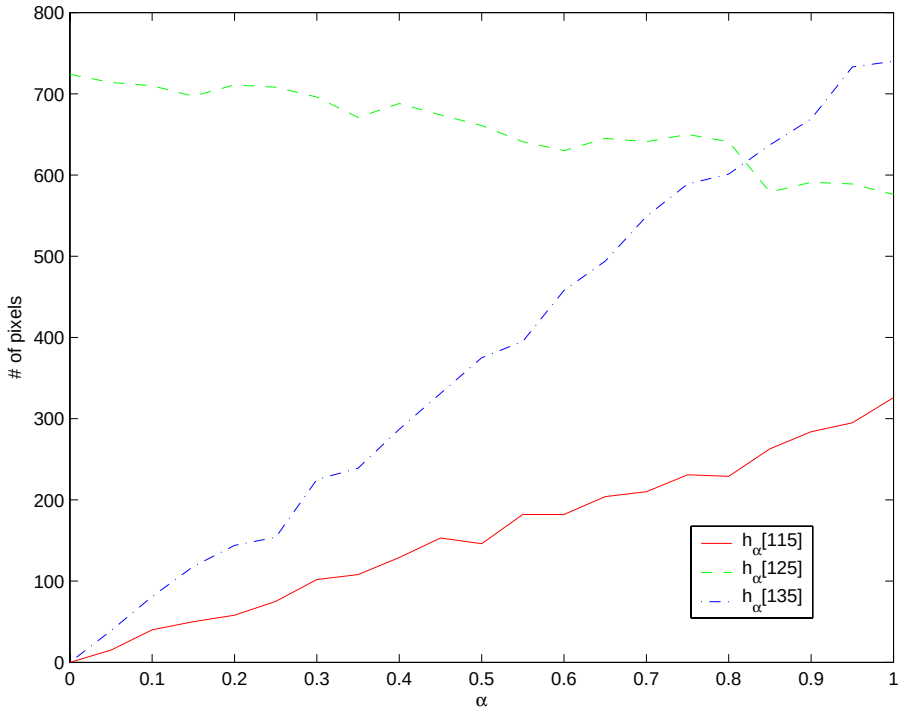

Figure 2.6: Various values of $h_\alpha[n]$ as embedding rate α changes.

Section 4.2. Here we see that the alterations of the histogram are roughly linear as predicted by Thm. (2.2.2).

Equation 2.8 is easily extended to the frequency domain in the following.

Corollary 2.2.2 ($\alpha - \mathcal{HCF}$ Multiplication). *In a system where α is the fraction of pixels chosen at random for embedding and the stegonoise is i.i.d. and independent of the coverimage, the stegoimage $\mathcal{HCF}$ is given by,*

$$H_\alpha[k] = \alpha H_c[k] F_\Delta[k] + (1-\alpha)\, H_c[k] \tag{2.10}$$

Proof. Taking the DFT of (2.8) the convolution becomes a multiplication in the frequency domain. □

To represent the addition of noise at any bitrate, as a single convolution we use the following Theorem.

Theorem 2.2.3 (Unified α-Bitrate Embedding). *In a system where α is the fraction of pixels chosen at random for embedding and the stegonoise is i.i.d. and*

independent of the coverimage, the stegoimage histogram is given by,

$$h_\alpha[n] = f_\Delta^\alpha[n] * h_c[n], \tag{2.11}$$

with,

$$f_\Delta^\alpha[n] \triangleq \alpha f_\Delta[n] + (1-\alpha)\delta[n].$$

Proof.

$$\begin{aligned} h_\alpha[n] &= \alpha\left(f_\Delta[n] * h_c[n]\right) + (1-\alpha)\, h_c[n] && \text{(2.12a)} \\ &= \alpha\left(f_\Delta[n] * h_c[n]\right) + (1-\alpha)\left(\delta[n] * h_c[n]\right) && \text{(2.12b)} \\ &= \left(\alpha f_\Delta[n] + (1-\alpha)\,\delta[n]\right) * h_c[n] && \text{(2.12c)} \\ &= f_\Delta^\alpha[n] * h_c[n]. && \text{(2.12d)} \end{aligned}$$

□

With (2.4) and more generally (2.8) as well as an assumption about the coverimage, we are able to analyze image histograms for evidence of processing by $f_\Delta[n]$.

CHAPTER 3
The Histogram Characteristic Function

This section deals with the histogram characteristic function ($\mathcal{HCF}$). The $\mathcal{HCF}$ is a representation of the image histogram in the frequency domain. Much of the natural correlation as well as that introduced by the capturing device is apparent in the frequency domain. The histogram characteristic function center of mass (COM) is introduced as a measure of the energy distribution in an $\mathcal{HCF}$.

3.1 HCF Center of Mass

The $\mathcal{HCF}$ COM a simple metric that will be used in the steganalysis of images. We would like to use a metric which will show evidence of processing by $f_\Delta[n]$ or equivalently $F_\Delta[k]$. From this we choose to look at the center of mass of the $\mathcal{HCF}$,

$$\mathcal{C}\left(H[k]\right) \triangleq \frac{\sum\limits_{k\in\mathcal{K}} k\left|H[k]\right|}{\sum\limits_{i\in\mathcal{K}}\left|H[i]\right|}. \tag{3.1}$$

Where $\mathcal{K} = \{0,\ldots,\frac{N}{2}-1\}$ and N is the DFT length. The $\mathcal{HCF}$ COM gives a general information about the energy distribution in the histogram characteristic function. The following provides a useful result for a class of additive noise modelable steganographic schemes.

Theorem 3.1.1. *For an embedding scheme with a nonincreasing $|F_\Delta[k]|$ for $k = (0,\ldots,\frac{N}{2}-1)$, the $\mathcal{HCF}$ COM decreases or remains the same after embedding,*

$$\mathcal{C}\left(H_s[k]\right) \leq \mathcal{C}\left(H_c[k]\right), \tag{3.2}$$

with equality if and only if $|F_\Delta[k]| = 1,\ \forall\ k = 0,\ldots,\frac{N}{2}-1$.

Proof. By the discrete Čebyšev inequality [27], for a nondecreasing sequence, $a = (a_0,\ldots,a_n)$, a nonincreasing sequence, $b = (b_0,\ldots,b_n)$, and a non-negative sequence,

$p = (p_0, \ldots, p_n)$,

$$\sum_{k=0}^{n} p_k \sum_{k=0}^{n} p_k a_k b_k \leq \sum_{k=0}^{n} p_k a_k \sum_{k=0}^{n} p_k b_k. \tag{3.3}$$

Letting $a_k = k$, $b_k = |F_\Delta[k]|$, $p_k = |H_c[k]|$ and $\mathcal{K} = \{0, \ldots, \frac{N}{2} - 1\}$ we have,

$$\sum_{k\in\mathcal{K}} |H_c[k]| \sum_{k\in\mathcal{K}} k\,|F_\Delta[k]|\,|H_c[k]| \leq \sum_{k\in\mathcal{K}} k\,|H_c[k]| \sum_{k\in\mathcal{K}} |F_\Delta[k]|\,|H_c[k]|\,, \tag{3.4}$$

or,

$$\frac{\sum\limits_{k\in\mathcal{K}} k\,|F_\Delta[k]|\,|H_c[k]|}{\sum\limits_{k\in\mathcal{K}} |F_\Delta[k]|\,|H_c[k]|} \leq \frac{\sum\limits_{k\in\mathcal{K}} k\,|H_c[k]|}{\sum\limits_{k\in\mathcal{K}} |H_c[k]|}. \tag{3.5}$$

Note that (3.4) holds with equality if and only if $|F_\Delta[k]| = 1,\ \forall\ k \in \mathcal{K}$. In the spatial domain, the equality condition is satisfied if $f_\Delta[n] = \delta[n]$. □

There exists a number of distributions having monotonically decreasing characteristic function magnitudes, these include the Gaussian and Laplacian.

3.2 HCF of Color Images

The above arguments can easily be extended for use with RGB color images as follows. We consider a pixel, $\mathbf{x}(n_1, n_2)$, as a vector of RGB intensities,

$$\mathbf{x}(n_1, n_2) = [x_r(n_1, n_2)\ \ x_g(n_1, n_2)\ \ x_b(n_1, n_2)].$$

We define an RGB histogram as,

$$h[\mathbf{n}] = \sum_{n_1, n_2} \mathcal{I}(\mathbf{k}, \mathbf{x}(n_1, n_2)) \tag{3.6}$$

with

$$\mathcal{I}(\mathbf{n}, x(n_1, n_2)) = \begin{cases} 1, & \mathbf{n} = \mathbf{x}(n_1, n_2) \\ 0, & else \end{cases}$$

where $\mathbf{n}$ is a vector of the RGB intensities, and the value of the histogram evaluated at $\mathbf{n}$ is the number of pixels with that RGB triplet. Taking the 3 dimensional discrete Fourier transform of $h[\mathbf{n}]$ we define the histogram characteristic

function, $\mathcal{HCF}$ for an RGB image as

$$H[\mathbf{k}] \triangleq DFT_3 h[\mathbf{n}] \tag{3.7}$$

Since the length N DFT is of real data its magnitude is symmetric about $\frac{N}{2}$ such that we only need to observe $[0, \frac{N}{2} - 1]^3$ of the $[0, N-1]^3$ DFT coefficients.

We now consider the centers of mass for $H[\mathbf{k}]$ along each of it's three axes,

$$\mathcal{C}_{k_1}(H[\mathbf{k}]) \triangleq \frac{1}{\beta} \sum_{\mathbf{k} \in \mathcal{K}} k_1 |H[\mathbf{k}]|, \tag{3.8a}$$

$$\mathcal{C}_{k_2}(H[\mathbf{k}]) \triangleq \frac{1}{\beta} \sum_{\mathbf{k} \in \mathcal{K}} k_2 |H[\mathbf{k}]|, \tag{3.8b}$$

$$\mathcal{C}_{k_3}(H[\mathbf{k}]) \triangleq \frac{1}{\beta} \sum_{\mathbf{k} \in \mathcal{K}} k_3 |H[\mathbf{k}]|. \tag{3.8c}$$

Where $\mathcal{K}$ is the set of first octant indices, i.e. $\mathbf{k} \in [0, \frac{N}{2} - 1]^3$ and the normalization constant,

$$\beta = \left(\sum_{\mathbf{k}} |H[\mathbf{k}]| \right).$$

Combining the values of each of (3.8) we can define a point in 3 dimensional space to be a "center of mass" for the RGB $\mathcal{HCF}$.

3.3 Moments

To gain further insight into the structure of the $\mathcal{HCF}$ we consider observing higher order central moments. These moments represent fundamental properties of the $\mathcal{HCF}$. For example, the first moment was used in the previous section as the mean of the $\mathcal{HCF}$, while the higher order moments capture the variance, kurtosis, skewness, etc...

These moments are defined as,

$$m_{r_1 r_2 r_3} = \frac{1}{\beta} \sum_{\mathbf{k} \in \mathcal{K}} \prod_{n=1}^{3} \left(k_n - \mathcal{C}_{k_n}(H[\mathbf{k}]) \right)^{r_n} |H[\mathbf{k}]| \tag{3.9}$$

Again, $\beta = \left(\sum_{\mathbf{k}} |H[\mathbf{k}]| \right)$, a normalization constant and $\mathcal{K}$ is the set of first octant indices, i.e. $\mathbf{k} \in [0, \frac{N}{2} - 1]^3$. These higher order moments will be used in the detection algorithms of Sections 5.5 and 5.6 to improve accuracy.

CHAPTER 4
Modeling Systems

In this section a number of information hiding methodologies are analyzed. The goal in each analysis is to derive the probability mass function of the stegonoise. Once we have this expression we use Theorem 2.2.1 to estimate the stego image histogram.

4.1 LSB

Least significant bit (LSB) steganography is the most simplistic form of steganography. It hides information by replacing the least significant bit of a pixels intensity with a message bit[28]. This system can be approximated as an additive noise scheme. First we consider the message bits (mb) to be i.i.d. with $Pr\{mb = 0\} = Pr\{mb = 1\} = \frac{1}{2}$. Likewise we assume that the LSBs of the cover image (x_c^{LSB}) are i.i.d. with $Pr\{x_c^{LSB} = 0\} = Pr\{x_c^{LSB} = 1\} = \frac{1}{2}$. It is then easily shown,

$$f_\Delta[-1] = Pr\{mb = 0\}Pr\{x_c^{LSB} = 1\} = 0.25, \tag{4.1a}$$

$$f_\Delta[0] = Pr\{mb = 0\}Pr\{x_c^{LSB} = 0\} + Pr\{mb = 1\}Pr\{x_c^{LSB} = 1\} = 0.5, \tag{4.1b}$$

$$f_\Delta[1] = Pr\{mb = 1\}Pr\{x_c^{LSB} = 0\} = 0.25. \tag{4.1c}$$

The LSB $|F_\Delta[k]|$ and $f_\Delta[n]$ for a DFT length $N = 256$ are shown in Figure 4.1.

Notice that this scheme acts as a lowpass filter on the histogram of the image. This filtering causes the histogram bins to "bleed" together, resulting in more unique intensities, as well as more close intensity pairs. These results are exploited in [14] to detect the presence of LSB steganography. In addition to being lowpass, $|F_\Delta[k]|$ is monotonically decreasing, which allows us to use Theorem (3.1.1).

In this analysis, $f_\Delta[n]$ approximates the alterations caused by LSB embedding as an additive noise. The actual embedding is not independent of the coverimage,

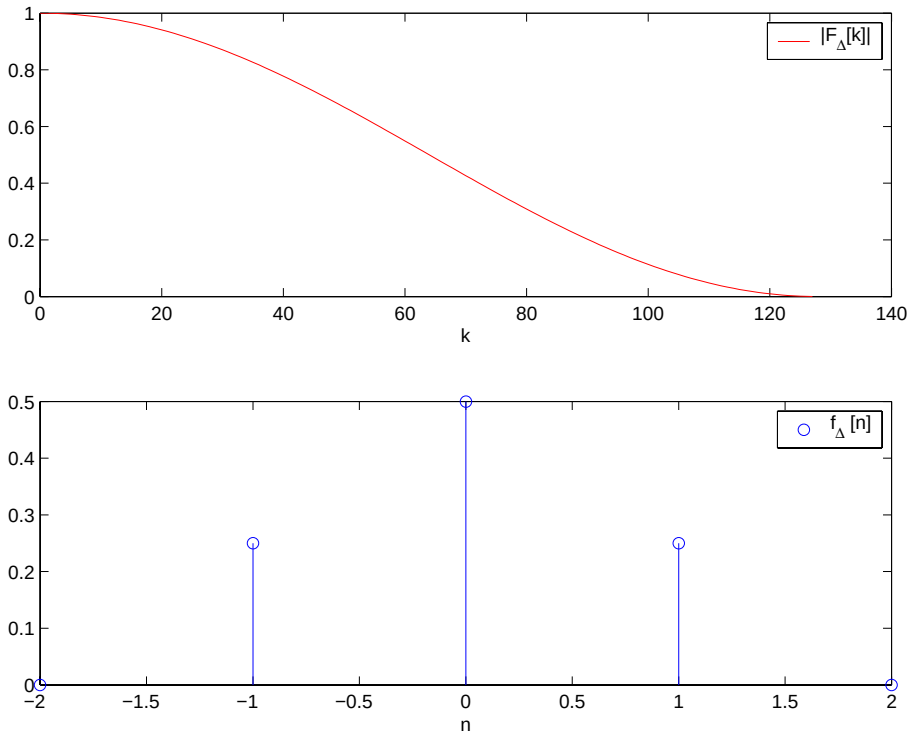

Figure 4.1: $|F_\Delta[k]|$ and $f_\Delta[n]$ for LSB

for example,

$$f_\Delta[n=-1] \neq f_\Delta[n=-1|x_c^{LSB}=0] = p(x_s - x_c = -1|x_c^{LSB}=0) = 0,$$

because when $x_c^{LSB} = 0$, only the addition of 0 or 1 can result.

4.2 Spread Spectrum Image Steganography

In this discussion we analyze spread spectrum image steganography (SSIS)[29]. The SSIS scheme hides data in a Gaussian stegonoise that is added to the cover image. This additive noise signal is equivalent to a direct-sequence spread spectrum system [30] wherein the PN-code is distributed as $\mathcal{N}(\mu, \sigma^2)$ with a chip period of every pixel. The details of the SSIS algorithm may be found in Appendix A.

The use of Gaussian noise in this scheme is motivated by the assumption that AWGN is a common distortion in images.

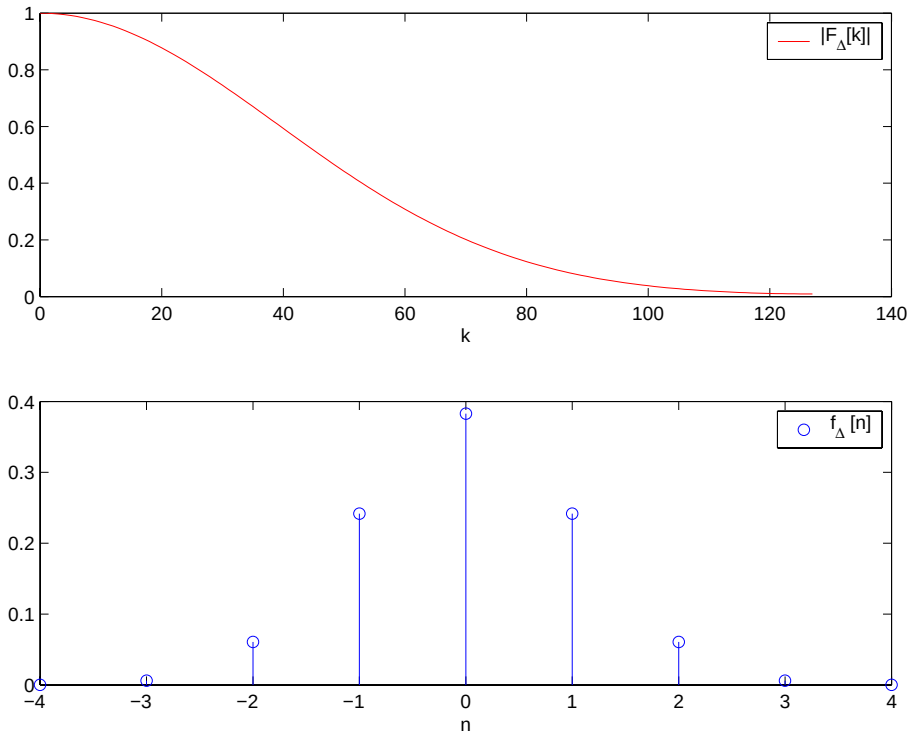

Figure 4.2: $|F_\Delta[k]|$ and $f_\Delta[n]$ for WGN

The distribution function of the pseudo-noise is defined as,

$$f_\Delta(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2\sigma^2}\cdot(x-\mu)^2} \tag{4.2}$$

For this discussion we will assume $\mu = 0$ and $\sigma^2 = 1$. To determine the affect this additive noise will have on the histogram of the coverimage we use (2.2) to find $f_\Delta[n]$. This yields the coefficients plotted in Figure 4.2 along with their corresponding frequency response for a DFT length $N = 256$.

Notice that the effect of the independent additive noise is a monotonically decreasing lowpass filter on the histogram. This is illustrated in the histogram in Figure 4.3 as well as the $\mathcal{HCF}$ magnitude in Figure 4.4.

To reduce error rate the stegonoise may be multiplied by a scale-factor, β, to adjust the power. From stochastic theory the variance of a scaled random variable

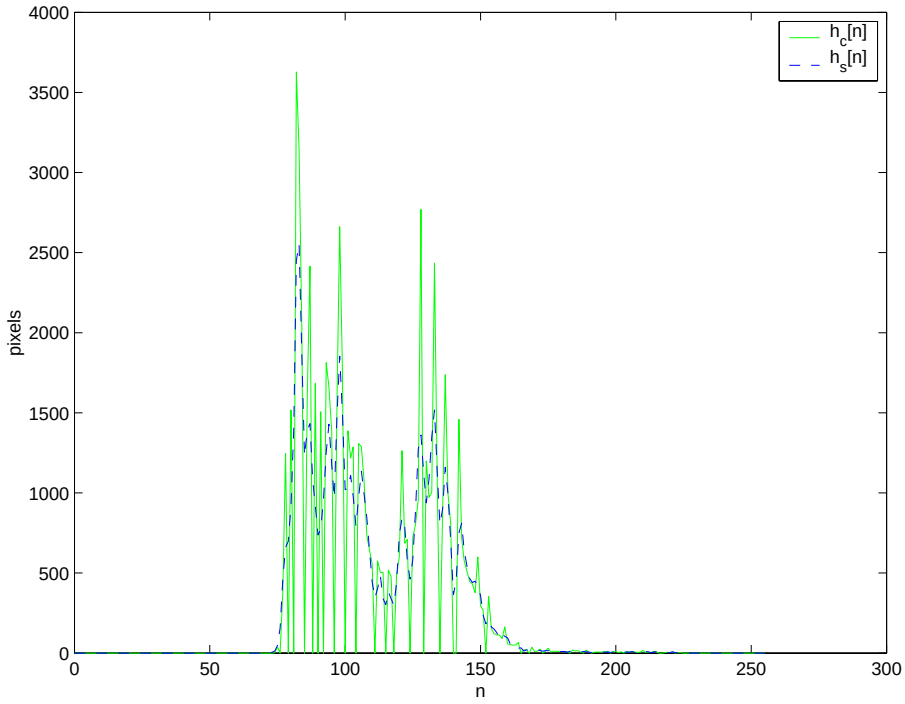

Figure 4.3: $h_c[n]$ and $h_s[n]$ for pout.tif

behaves as,

$$\begin{aligned} \sigma^2_{scale} &= E[\beta\,(X-\mu)\,\beta\,(X-\mu)] \\ &= \beta^2 E[(X-\mu)^2] \\ &= \beta^2\sigma^2 \end{aligned} \tag{4.3}$$

As the variance of the additive noise increases by β^2, the stegonoise PMF will spread out. This spreading of $f_\Delta[n]$ yields a lower cutoff point in $|F_\Delta[k]|$. This effect is plotted in Figure 4.5 for $\beta = \{1, 2, 3, 4, 5\}$ and $\sigma^2 = 1$. The alteration of $h_c[n]$ becomes increasingly pronounced as β increases.

4.3 Discrete Cosine Transform Steganography

To improve robustness and stealth, many steganographic schemes utilize projections to embed data in an alternate space. In this section we consider the effects

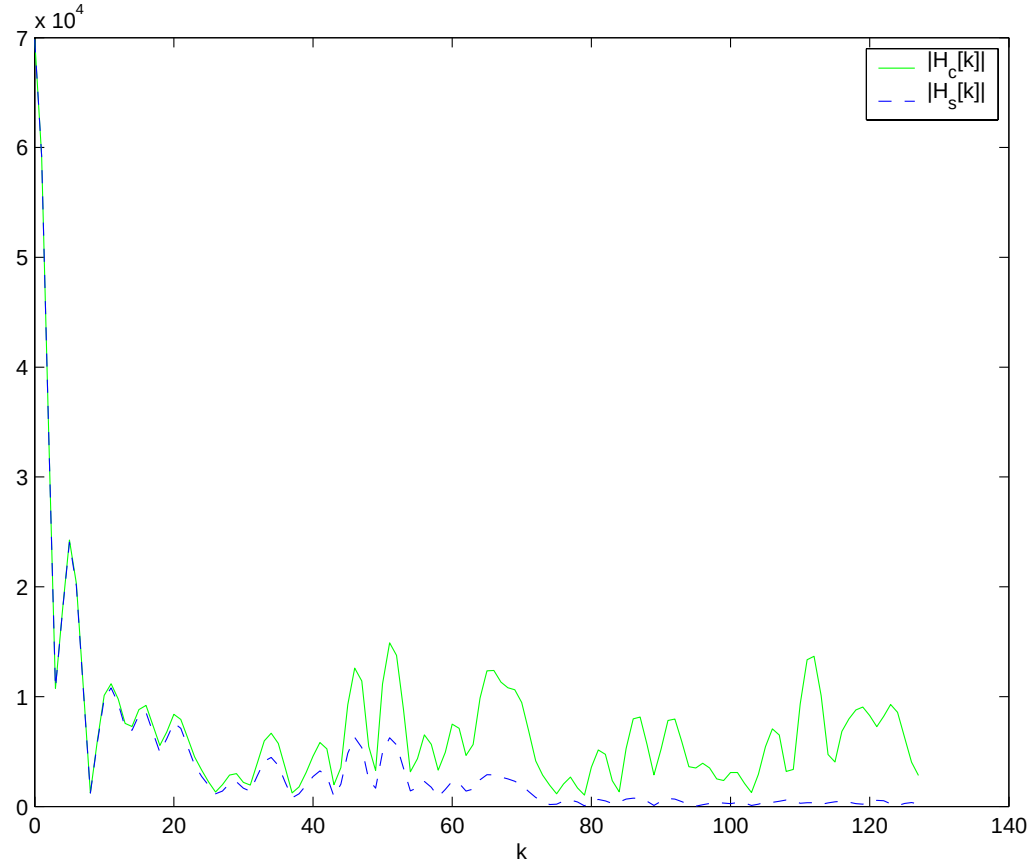

Figure 4.4: $|H_c[k]|$ and $|H_s[k]|$ for pout.tif

of hiding information as an additive noise in discrete cosine transform (DCT) coefficients. We choose the DCT as it is a common transform in image processing. The process we discuss is generally similar to the DCT hiding of [31], with the exception our model will hide data as an additive noise rather than a quantization.

The actual embedding process begins by decorrelating the image by reordering the pixels based on a keying variable. Next, the mean of the pixels is subtracted and an $L \times L$ block DCT [32] is taken over the image. The decorrelation of the pixels serves to whiten the image and increase the energy in the high frequency DCT coefficients, making them more useful in hiding data. Once in the frequency domain, an i.i.d. stegonoise is added to each coefficient [1]. An $L \times L$ block IDCT is

[1]In [31] the DCT coefficients are quantized to hide information. The error introduced in this process is a deterministic function of the coefficients. As this error would be considered the stegonoise in our framework, the heavy dependence between the cover-coefficients and stegonoise

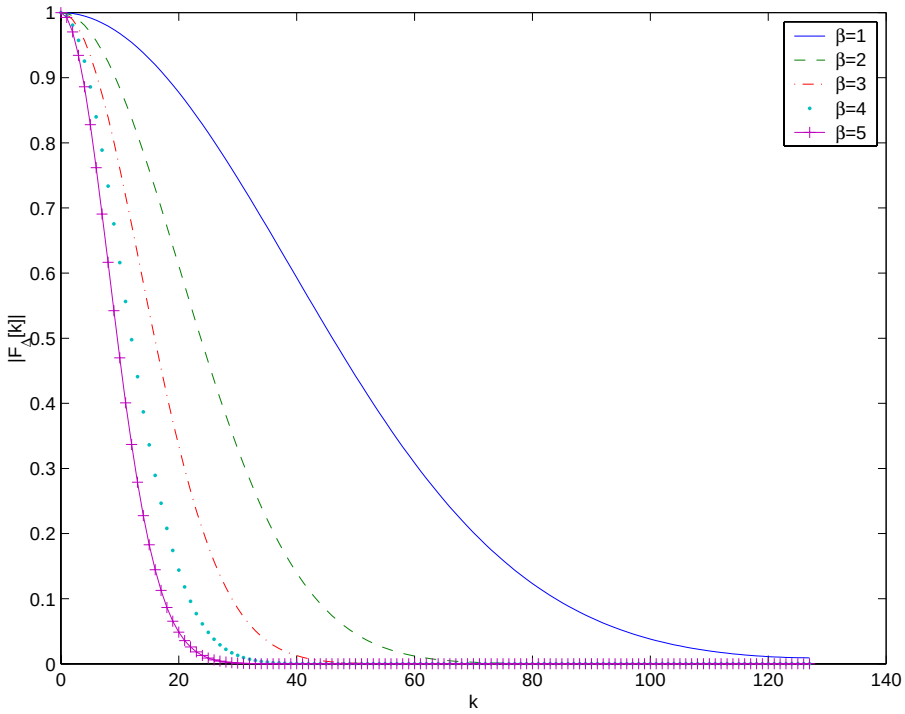

Figure 4.5: Effect of scaling factor β on $|F_\Delta[k]|$

performed and the previously subtracted mean is added to each pixel. Finally, the pixels are rounded to integers and returned to their original order using the keying variable.

Considering the signals involved we have,

$$\mathcal{X}_c = DCT\{x_c\},$$

$$\mathcal{X}_s = \mathcal{X}_c + stegonoise,$$

$$x_s = IDCT\{\mathcal{X}_c + stegonoise\} = x_c + IDCT\{stegonoise\}.$$

The additive noise embedding in the frequency domain is modeled as the addition of spatial stegonoise, $IDCT\{stegonoise\}$.

We now present an informal argument that the spatial stegonoise is i.i.d Gaussian using statistical properties of the DFT. The DCT inherits these same properties.

does not allow for a direct additive noise analysis.

In [33] it is shown that for a stationary sequence with finite second-order moments and mixing, the DFT elements are asymptotically independent. In [34]2 it is shown that for sequences obeying the Lindeberg condition, the DFT elements asymptotically approach normal distributions. With this we can consider the spatial stegonoise to be roughly equivalent to i.i.d. Gaussian. This allows us to consider the addition of an i.i.d. stegonoise in the frequency domain, as approximately i.i.d. Gaussian stegonoise in the spatial domain. With these assumptions the effect of additive noise in the frequency domain is modeled as in Section 4.2, in particular the monotonically decreasing $|F_{\Delta}[k]|$.

CHAPTER 5
Detection Schemes

5.1 Overview

This chapter uses the ideas previously developed to build classifiers that are able to differentiate altered images from original. The method presented in Section 5.2 builds a classifier trained on both the coverimages as well as stegoimages. A second method is presented in Section 5.3 which uses no information about the hiding method. Section 5.4 presents enhancements in classification using higher order $\mathcal{HCF}$ moments in classification.

5.2 Known Scheme Detection I

In known scheme detection the method of hiding is assumed to be available in classifier construction. This provides a significant advantage in detection as a concrete notion of the effects of embedding can be developed.

Using results from 4.2 we create a simple classification scheme. This scheme will be built specifically to detect the addition of SSIS information into an image. The classification of a test image will be into one of two categories: containing SSIS data or unaltered.

Recalling that the addition of noise affects the $\mathcal{HCF}$ as a multiplication by the lowpass filter shown in Figure 4.4 as well as the bound given in Thm. 3.1.1, we expect $\mathcal{C}\left(H_s[k]\right)$ to be lower as the higher frequencies are attenuated. Indeed this is the case as we see in Figure 5.1.

This reduction results in the shifting of the center of mass closer to the origin. If we extend this result to 3 dimensions we would expect that the center of mass would move toward the origin.

To verify these results 24 images from the Kodak PhotoCD PCD0992 [35] were used. These images are 24-bit, 768x512 pixel, truecolor images stored in the PNG format.

For each image the three dimensional RGB $\mathcal{HCF}$ COM was computed for the

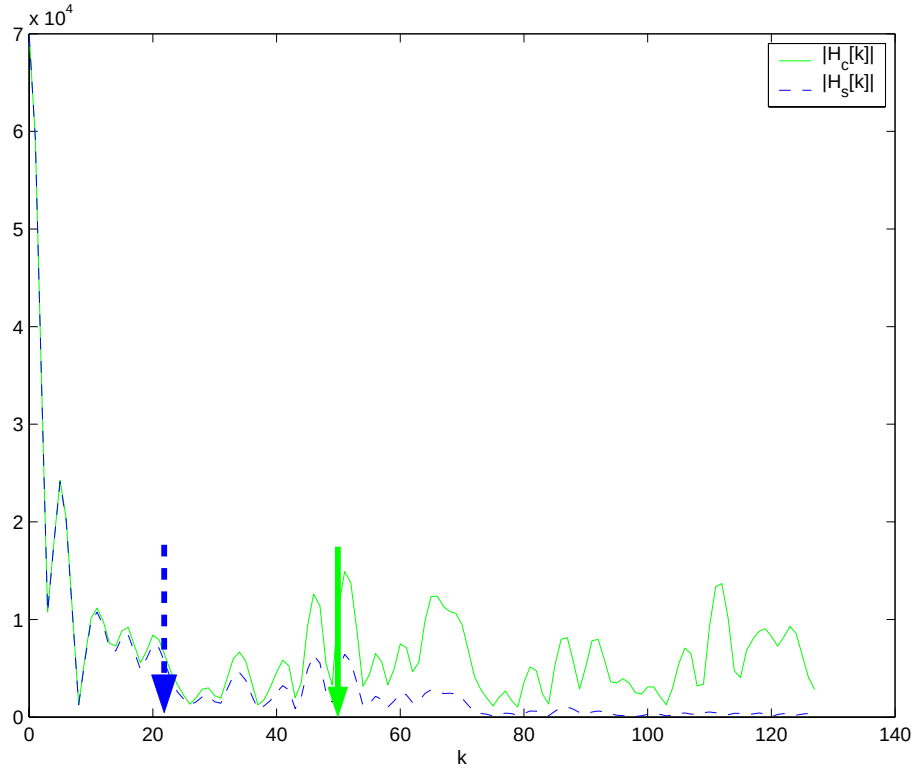

Figure 5.1: Center of mass for $|\mathcal{HCF}|$

original image as well as a stegoimage with $\mathcal{N}(0,1)$. A 3-D scatter plot of these points is shown in Figure 5.2. As expected the centers of mass for the stegoimages are considerably lower than those of the originals.

To create the classifier, we first assume the distribution of COMs is Gaussian to make use of the maximum likelihood multivariate classifier [36], detailed in Appendix B. The maximum likelihood multivariate classifier requires that the mean vectors, $\boldsymbol{\mu}$, and covariance matrices $\boldsymbol{\Sigma}$, of the source distributions be known or estimated. For our application we estimate these values using the maximum likelihood estimators,

$$\boldsymbol{\mu}_i = \frac{1}{S}\sum_{j=1}^{S} \mathbf{x}_i^{(j)} \tag{5.1}$$

$$\boldsymbol{\Sigma}_i = \frac{1}{S}\sum_{j=1}^{S} \left(\mathbf{x}_i^{(j)} - \boldsymbol{\mu}_i\right)\left(\mathbf{x}_i^{(j)} - \boldsymbol{\mu}_i\right)^T \tag{5.2}$$

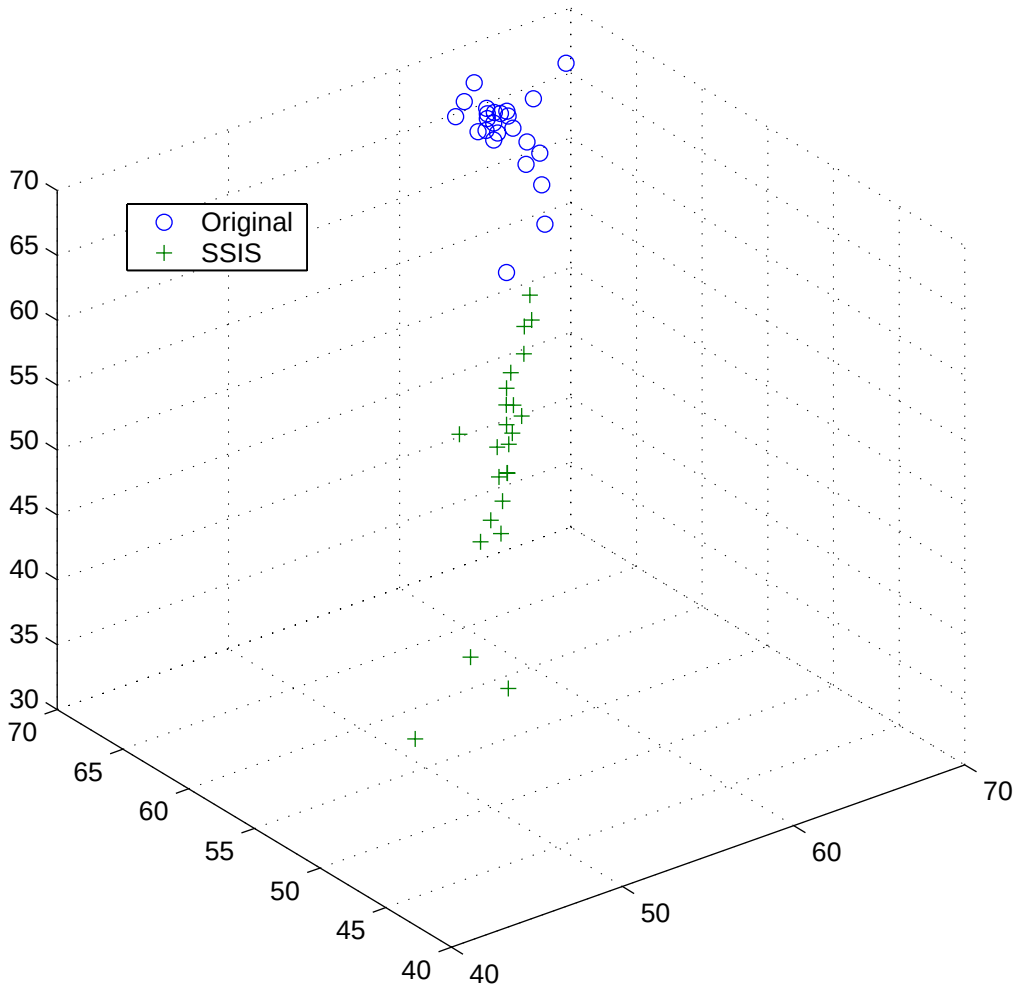

Figure 5.2: Center of Mass for Test Images

where $\mathbf{x}_i^{(j)},\ j = 1 \ldots S$ is the training set of RGB $\mathcal{HCF}$ COMs for the ith multivariate.

The general multivariate discriminant functions are then,

$$g_i(\mathbf{x}) = \mathbf{x}^T \mathbf{W}_i \mathbf{x} + \mathbf{w}_i^T \mathbf{x} + w_i, \tag{5.3}$$

with

$$\mathbf{W}_i = -\frac{1}{2}\boldsymbol{\Sigma}_i^{-1}, \tag{5.4a}$$

$$\mathbf{w}_i = \boldsymbol{\Sigma}_i^{-1}\boldsymbol{\mu}_i, \tag{5.4b}$$

$$w_i = -\frac{1}{2}\boldsymbol{\mu}_i^T\boldsymbol{\Sigma}_i^{-1}\boldsymbol{\mu}_i - \frac{1}{2}\ln|\boldsymbol{\Sigma}_i|. \tag{5.4c}$$

To classify an unknown sample vector $\mathbf{x}$, each discriminant function is evaluated at $\mathbf{x}$. If $g_1(\mathbf{x}) > g_2(\mathbf{x})$ the pattern is assigned to ω_1, else it is assigned as ω_2.

To evaluate the classifier, the 24 Kodak images are divided into four groups. Each group is created by randomly selecting (without replacement) the appropriate number of images. The groups are as follows,

1. 10 Unaltered image COMs used to find $\boldsymbol{\mu}_1$ and $\boldsymbol{\Sigma}_1$ for ω_1.
2. 10 SSIS $\mathcal{N}(0,1)$ image COMs embedding used to find $\boldsymbol{\mu}_2$ and $\boldsymbol{\Sigma}_2$ for ω_2.
3. 2 Unaltered image COMs classified.
4. 2 SSIS image COMs classified.

Where $\boldsymbol{\mu}_1$ and $\boldsymbol{\Sigma}_1$ are the estimated mean and covariance matrices of the original $\mathcal{HCF}$ COM class, ω_1. Likewise, $\boldsymbol{\mu}_2$ and $\boldsymbol{\Sigma}_2$ are the estimated mean and covariance matrices of the SSIS stegoimage $\mathcal{HCF}$ COM class ω_2. Using these distributions, the remaining 4 images are classified by evaluating the discriminant functions of each class at the test COMs.

The above process was repeated 1000 times giving an average classification rate of 94.4% correct. As detailed in Table 5.1 this equates to 223 errors in classification. Of these, 209 were Type I (false alarms), while only 14 of the 223 errors were Type II (missed signals).

5.3 Unknown Scheme Detection I

In practice it is desirable to detect the presence of a message regardless of the embedding method. The foremost reason for this is that the algorithm used in

Table 5.1: Known Scheme Classification Performance

Tests	4000	
Errors	223	
Correct	94.4%	
	Original	Stegoimage
Tests	2000	2000
Errors	209	14
Correct	89.55%	99.3%

embedding may not be known. With this in mind we now describe an unknown scheme detection.

In contrast to the previous section where we made use of statistics from both original and modified images, we now only consider the availability of original images. It is worth emphasizing that *we assume no explicit knowledge of the hiding method* in the classifier construction. We only have what we consider to be "normal" images available to train on, and knowledge of Theorem (3.1.1).

Again we focus on the $\mathcal{HCF}$ COM as our feature in the detection scheme. As we would like to measure how similar (or dissimilar) a COM in question is to our trained statistic, we consider the Mahalanobis distance defined as,

$$d^2 = (\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}) . \tag{5.5}$$

Where $\boldsymbol{\Sigma}$ and $\boldsymbol{\mu}$ are the covariance and mean estimates defined in (5.1) and (5.2), using measurements gathered from a training set.

The Mahalanobis distance essentially gives a statistical measure of how far a given point is from the estimated mean, with consideration toward the variance of each variable. Generally speaking, the greater the Mahalanobis distance, the less likely the test point is of the same distribution as the training set. The surface defined by $d^2 = 1$ is a surface where each point is one standard deviation away from the mean.

To evaluate the classifier, the 24 Kodak images are divided into five groups. Each group is created by randomly selecting (without replacement) the appropriate

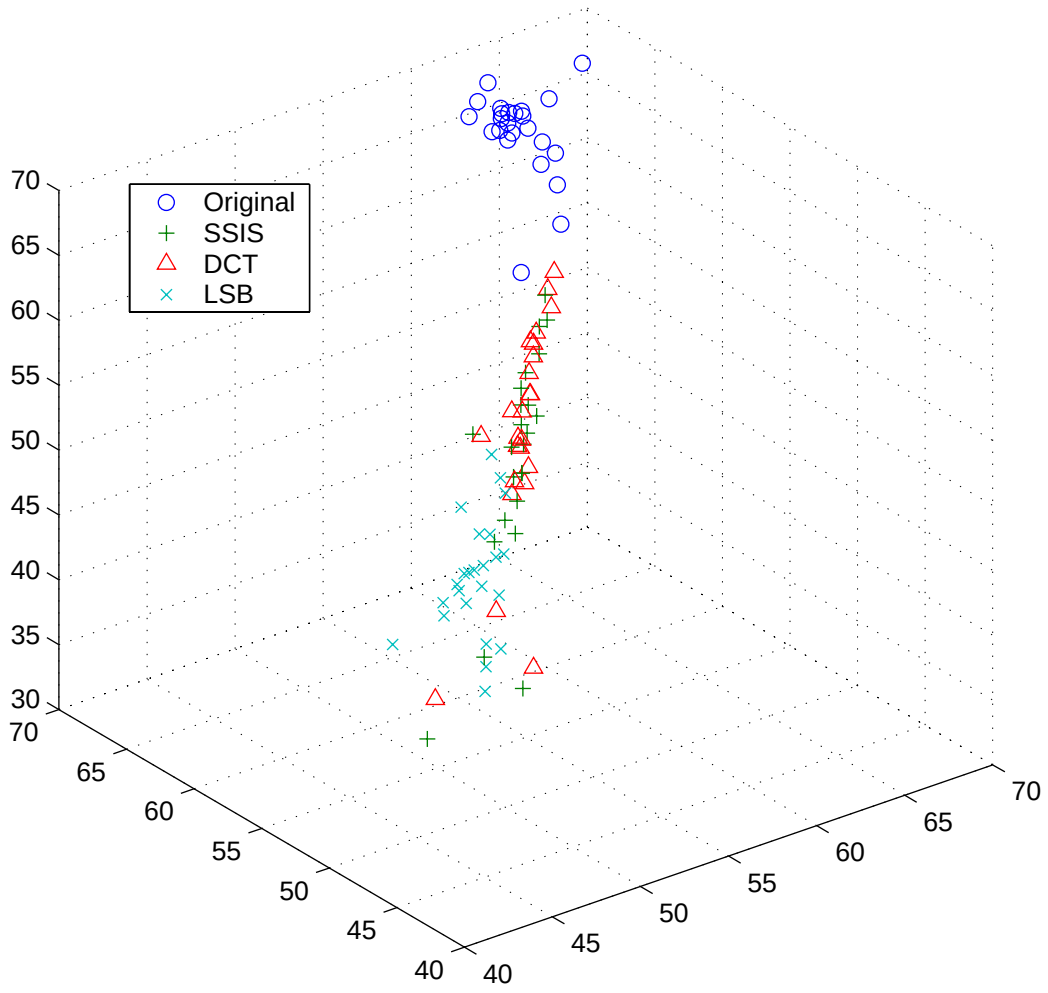

Figure 5.3: Centers of mass

number of images. The groups are as follows,

1. 20 original image $\mathcal{HCF}$ COMs used to estimate $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$
2. 1 Unaltered image COM classified
3. 1 SSIS image COM classified
4. 1 DCT image COM classified
5. 1 LSB image COM classified

The 20 unaltered COMs are used to form an estimate of the mean vector and covariance matrix. The multivariate described by these is considered to be a natural

Table 5.2: Unknown Scheme Classification Performance

Tests	4000			
Errors	24			
Correct	96.05%			
	Original	SSIS	DCT	LSB
Tests	1000	1000	1000	1000
Errors	19	59	75	0
Correct	94.1%	97.6%	92.5%	100%

$\mathcal{HCF}$ COM distribution, and any images which differ significantly will be classified as containing steganographic data. The first test image is the unaltered image in its original form without any modifications. The SSIS image has a message embedded in it using the method described in Section 4.2. The stegonoise has a variance of 1 and is equivalent to adding i.i.d. $\mathcal{N}(0,1)$. The DCT images are created using the general method in Section 4.3. A DCT block size of 4×4, was used to hide one bit in each coefficient ($m=1$) with a quantization step size of 2 ($\Delta=2$). This embedding is modeled by adding a uniformly distributed noise over $[-2,2]$ to each coefficient in the frequency domain. The LSB image is formed as described in Section 4.1, by replacing the least significant bit of each pixel with the message bit. Figure 5.3 shows a plot of the $\mathcal{HCF}$ COMs for all 24 images with the three types of embedding and the original images.

A Mahalanobis cutoff of approximately 40 was chosen to yield a Type I, (false alarm), rate of approximately 5%. As can be seen the classifier performs very well, with a correct classification rate of approximately 95%.

5.4 Extensions Using Moments

In Section 3.3 the higher order moments of the $\mathcal{HCF}$ were defined. In this section the use of these moments for improving classification is explored.

5.4.1 Choosing the Optimal Moment

In this section we will consider features of the form $\mathbf{m}_i = [m_{i00}\ m_{0i0}\ m_{00i}]$, where m_{ijk} is the ith moment of the HCF defined by (3.9). Our goal is to find the

feature set that results in the best classification performance. To accomplish this we make use of the Chernoff error bound described in Appendix C.

The Chernoff bound estimates the probability of error in a multivariate maximum likelihood classification scheme, such as that described in Section 5.2. To make use of the Chernoff bound we must first create an estimate of the distribution of moments for each order. To do this we repeat the following steps for moments $i = 1 \dots 10$:

1. Calculate $\mathbf{m}_i^{(k)}$, the ith HCF moments for $k = 1 \dots 24$, the 24 original images using (3.9).
2. Calculate $\boldsymbol{\mu}_i$, the mean of the ith moments using $\mathbf{m}_i^{(k)},\ k = 1 \dots 24$ and (5.1).
3. Calculate $\boldsymbol{\Sigma}_i$, the covariance of the ith moments using $\mathbf{m}_i^{(k)},\ k = 1 \dots 24$ and (5.2).

This allows formulatation of the class conditional distribution for the unaltered image HCF moments: $p(\mathbf{m}_i|\omega_1)$. Next we repeat the procedure using SSIS images created with $\mathcal{N}(0, 1)$:

1. Calculate $\mathbf{m}_i^{(k)}$, the ith HCF moments for $k = 1 \dots 24$, the 24 SSIS images using (3.9).
2. Calculate $\boldsymbol{\mu}_i$, the mean of the ith moments using $\mathbf{m}_i^{(k)},\ k = 1 \dots 24$ and (5.1).
3. Calculate $\boldsymbol{\Sigma}_i$, the covariance of the ith moments using $\mathbf{m}_i^{(k)},\ k = 1 \dots 24$ and (5.2).

This gives the class conditional distribution for the stegoimage HCF moments: $p(\mathbf{m}_i|\omega_2)$.

Once we have $p(\mathbf{m}_i|\omega_1)$ and $p(\mathbf{m}_i|\omega_2)$, we make use of the Chernoff bound. Specifically we seek,

$$i^* = \arg\min_i P(error|\mathbf{m}^i), \tag{5.6}$$

where $P(error|\mathbf{m}^i)$ is the estimated error of a classifier based on $\mathbf{m}^i$.

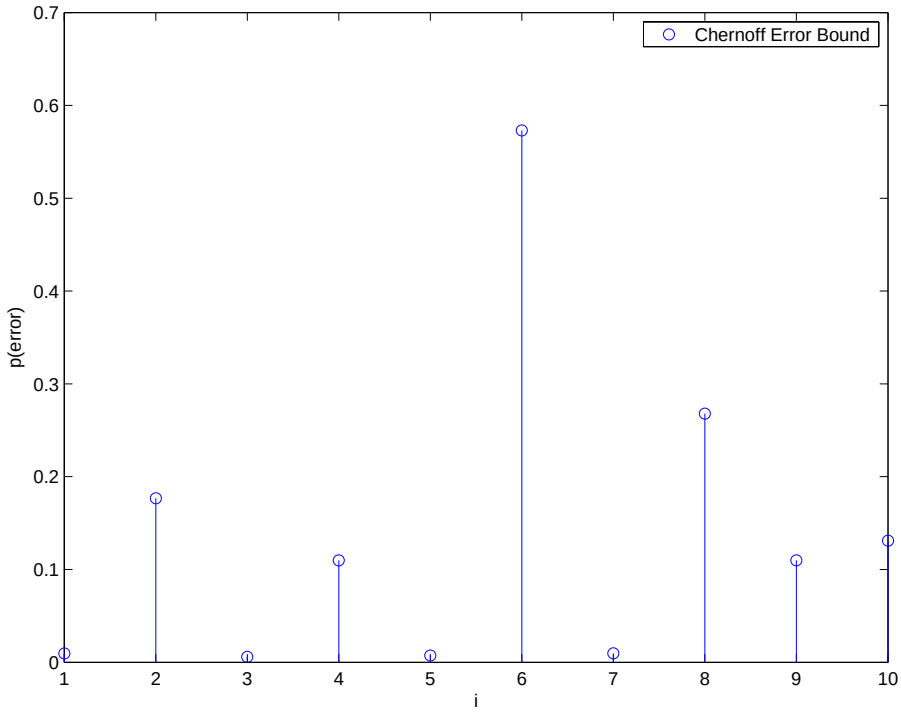

Figure 5.4: Center of Mass for Test Images

The Chernoff Bound (C.5) was used to to estimate the error of each classifier. Figure 5.4 shows a plot of $P(error|\mathbf{m}^i)$ for $i = 1 \ldots 10$. The results for each moments are listed in Table 5.3. Note the odd moments convey much more information regarding the image, with the third moment, kurtosis, representing the lowest probability of error.

5.5 Known Scheme Detection II

We now repeat the experiment from Section 5.2 using the 3^{rd} central moment (kurtosis) found in Section 5.4.1. Note that these results have been obtained by running the classification in parallel with the experiments in Section 5.2, so the results are directly comparable. Here the 3rd moment system performs approximately 2.4% better than that of the 1st moment.

Table 5.3: Chernoff Error Bounds for Moments

Moment	P(error) Bound
1	0.0096
2	0.1767
3	0.0059
4	0.1099
5	0.0073
6	0.573
7	0.0097
8	0.268
9	0.110
10	0.131

Table 5.4: Known Scheme Classification Performance Moment 3

Tests	4000	
Errors	128	
Correct	96.8%	
	Original	Stegoimage
Tests	2000	2000
Errors	121	7
Correct	93.95%	99.65%

5.6 Unknown Scheme Detection II

We now repeat the experiment from Section 5.3 using the 3^{rd} central moment found in Section 5.4.1. Note that these results have been obtained by running the classification in parallel with the experiments in Section 5.3, so the results are directly comparable. Here the 3rd moment system performs approximately 1.2% better than that of the 1st moment.

Table 5.5: Unknown Scheme Classification Performance Moment 3

Tests	4000			
Errors	111			
Correct	97.22%			
	Original	SSIS	DCT	LSB
Tests	1000	1000	1000	1000
Errors	57	10	44	0
Correct	94.3%	99%	95.6%	100%

CHAPTER 6
Discussion

To conclude this thesis, a number of "rules" for steganography and steganalysis are discussed. The relation of this work to trends in steganography and steganalysis is explored. Finally a commentary on the use of steganalysis for detecting covert communication is presented.

6.1 Jeremiah's Rules of Steganography

1. Strength is not stealth
2. Assume the hiding method is known by the adversary
3. Never divulge side information
4. Don't add noise to a cover image where it doesn't belong

The first item seems to be a common misinterpretation of the goal of steganography. With steganography, the purpose is to communicate without revealing the *existence* of that communication. Thus, if any data hiding scheme raises a suspicion that extra information is being transfered, it is a poor data hiding scheme, *irregardless of whether an attacker can read the message.*

The second item is essential in creating a steganographic system with a high stealth and is known as Kerckhkoff's principle in cryptography. When hiding one should always assume an adversary knows the method with which you are hiding information. That is, if there are any tell-tale statistical changes made by an embedding scheme, assume the adversary knows what they are and can test for them. This is a difficult problem and central to data hiding, as it is difficult to know *a priori* what statistics an attacker will concentrate on.

The third item is minimize available side information. If an attacker knows any additional information about how an image has been captured or what processes it has undergone, he will be able to exploit that knowledge in analyzing images. For

example, if the type of camera used in capturing an image is known, a statistical model of that camera could be used to improve the analysis.

In an extreme case of side information, the attacker would have access to the coverimage itself, destroying the stealth of the system[13]. This side information rule can be extended to using images which are very similar. If a steganographer uses many images of the same scene, the statistics of the original scene or capturing device could be estimated and the discrepancies caused by embedding detected.

The final item is in regard to additive noise steganography. It states never to add noise where it doesn't belong. This echoes the ideas of [12],

> "In steganography, the use of noise may make things worse, not better. One can use the inherent noise in a cover image, but adding additional noise may cause the steganography to be discovered."

This thesis substantiates this concept in terms of the bounds of Thm. (3.1.1), as well as results in Section 5. As the quality of the average consumer grade digital camera increases, the amount of noise present in the system will be reduced even further. The assumption that adding small amounts of arbitrary noise will not disturb the statistics of the image is increasingly a poor one.

6.2 Jeremiah's Rules of Steganalysis

1. Never assume you know the hiding method

2. Never assume order in the stegonoise

The first rule states that one should never assume that the hiding method is known. This is very important in real world steganalysis. While security through obscurity is known not to be dependable, in a properly designed and tested scheme the secrecy can only tip the game toward the hider.

An ideal analysis scheme should be free from the concept of a stegoimage. By building a model of natural images, the divergence of a test image from these statistics should be the only needed indication that the image is suspect.

The second rule is specifically for additive noise information hiding. It is to never assume that there is an order to the stegonoise. By encrypting a message,

almost all correlation between the input text and cypher-text is destroyed. Furthermore, with a proper noise generation algorithm it is is computationally difficult to estimate or extract a spreading sequence.

This rule has an interesting consequence: adhering to it means that we must flag any image with a large amount of noise. This concept is explored in the next section.

6.3 The Additive Noise Arms Race

Steganography is a game between the hider and the seeker. The hider wants to hide as much information as they can without being discovered. This maximum amount is defined by the tools a seeker is able to analyze data with. As shown in this thesis it is possible to detect high bitrate embeddings that make use of simple noise models. This gives a hider using an additive noise algorithm two choices: to use pseudo-noise models such as Gaussian noise at a lower bitrate or explore alternatives. As the hider would like to maximize capacity, the second choice is the most likely option. In this respect a logical step for the hider is to seek systems containing a large amount of inherent noise for hiding. A system with inherent noise is characterized by the presence of a persistent, probabilistic distortion. In such a system, the hider would attempt to model the noise as accurately as possible, then use it in embedding. The seeker would then be forced to make an attempt at discerning the source of the noise in a test image. This creates an "additive noise arms race" with each side attempting to out pace the other. This scenario and solutions are discussed next.

6.3.1 Would the Real Noise Please Stand Up?

The seeker knows that the hider may be hiding data using a noise model mimicking naturally present noise. This is a difficult situation for the seeker as the problem becomes: is the noise natural? This is an interesting problem in that in the most basic sense, one would need to perform analysis on the noise in an image. In blind steganalysis the seeker has no access to the coverimage, thus the noise must be separated from the image. This amounts to the denoising of an image. If the

hider can generate a pseudo-noise that is sufficiently close to real-world noise, the differences could be masked by estimation errors in extracting the noise.

In addition, as the noise models improve, the pseudo-noise will approach being indistinguishable from the real-world noise. Generally, this means if we ignore the denoising problem and assume we are able to perfectly recover a potential stegonoise, it will be impossible to determine whether the noise is natural or man-made.

6.3.2 Guilty Until Proven Innocent

In a system where the pseudo-noise is indistinguishable from the real-world noise, the seeker has no choice but to flag every message containing such noise. This method would produce a large number of false positives. The hope in this case is that the number of such systems is fairly low. For example, a specific user sending a large number of apparently scanned photographs may be suspicious. As data hiding methodologies continue to advance, it is very likely that the number of systems in this category will grow.

6.4 The Future of Detection Steganalysis

This section presents a commentary on difficulty of blind steganalysis for detecting covert communications, as well as discussing a paradigm shift for the goals of such steganalysis.

6.4.1 Seeing Is Not Believing

The cat and mouse game of hiding and seeking presented in the previous section is not limited to additive noise. The difficulty in steganalysis is that the seeker only sees what the hider wants them to. This means that any statistic in the image should be regarded as potentially misleading. I believe that the hider has an insurmountable advantage in this simple game. In the case where the hider has complete access to the detection algorithms of the seeker, they will be able to construct a hiding scheme that is able to evade detection. Fortunately, this goal is counter to the greedy nature of the hider, in that he would like to stuff as many bits as possible into each image.

6.4.2 The New Goal of Detection Steganalysis

By in large the scales are heavily tipped in favor of the hider. So much in fact, that it is likely there will never be a "catch all" detection scheme. Because of this it is important to shift the goal of detection steganalysis.

Detection steganalysis should be viewed as a preventative measure. As it is impossible to catch a well designed scheme carrying a properly sized message, the goal should be to develop general functions that expose the presence of hidden information. Each of these functions can be seen as a further constraint on the hider. With each of these constraints the usable payload is decreased. By adding to this library of analysis functions, successful information hiding becomes more and more impractical- although never impossible.

New methods of steganalysis, such as presented in this thesis, should not be viewed as a nail in a coffin, but rather an additional weight around the neck of covert messaging.

LITERATURE CITED

[1] J. J. Harmsen and W. A. Pearlman, "Steganalysis of additive noise modelable information hiding," in *Proc. SPIE Electronic Imaging 5022*, Santa Clara, CA, Jan. 21–24, 2003.

[2] B. Schneier, *Applied Cryptography*, 2nd ed. New York, NY: John Wiley & Sons, Inc., 1996.

[3] J. Massey, "An introduction to contemporary cryptology," *Proceedings of the IEEE*, vol. 76, no. 5, pp. 533– 549, May 1988.

[4] G. Simmons, Ed., *Contemporary Cryptology: The Science of Information Integrity*. Piscatoway, NJ: IEEE Press, 1992.

[5] D. Kahn, *The History of Steganography*, ser. Lecture Notes in Computer Science. New York: Springer-Verlag, 1996, vol. 1174.

[6] S.-J. Lee and S.-H. Jung, "A survey of watermarking techniques applied to multimedia," in *ISIE*, 2001, pp. 272–277.

[7] M. L. Miller, I. J. Cox, J.-P. M. G. Linnartz, and T. Kalker, "A review of of watermarking principles and practices," in *Digital Signal Processing for Multimedia Systems*, K. K. Parhi and T. Nishitani, Eds. IEEE, 1999, pp. 461–485.

[8] I. J. Cox, J. Kilian, F. T. Leighton, and T. Shamoon, "Secure spread spectrum watermarking for multimedia," *IEEE Trans. on Image Processing*, vol. 6, no. 12, pp. 1673–1687, Dec. 1997.

[9] D. Kundur and D. Hatzinakos, "Digital watermarking for telltale tamper proofing and authentication," vol. 87, no. 7, pp. 1167 –1180, July 1999.

[10] C. Cachin, *An Information-Theoretic Model for Steganography*, ser. Lecture Notes in Computer Science. New York: Springer-Verlag, 1998, vol. 1525.

[11] S. Katzenbeisser and F. A. P. Petitcolas, "Defining security in steganographic systems," in *SPIE Security and Watermarking of Multimedia Contents IV*.

[12] I. S. Moskowitz, G. E. Longdon, and L. Chang, "A new paradigm hidden in steganography," in *WNSP: New Security Paradigms Workshop*. ACM Press, 2001, pp. 41–50.

[13] J. Zollner and et al, *Modeling the Security of Steganographic Systems*, ser. Lecture Notes in Computer Science. Berlin, Germany: Springer-Verlag, 1998, vol. 1525.

[14] J. Fridrich, M. Goljan, and R. Du, "Detecting LSB steganography in color, and gray-scale images," *IEEE Trans. Multimedia*, vol. 8, no. 4, pp. 22–28, Oct. 2001.

[15] H. Farid, "Detecting steganographic messages in digital images," Dartmouth College, Tech. Rep. TR2001-412.

[16] ——, "Detecting hidden messages using higher-order statistical models," in *Proc. IEEE International Conference on Image Processing*, Rochester, NY, Sept. 20–23, 2002, pp. 23–23.

[17] J. Fridrich, M. Goljan, and R. Du, "Steganalysis based on jpeg compatibility," in *SPIE Multimedia Systems and Applications IV*, Denver, CO, Aug. 20–24, 2001.

[18] N. F. Johnson and S. Jajodia, *Steganalysis of Images Created Using Current Steganographic Software*, ser. Lecture Notes in Computer Science. Berlin, Germany: Springer-Verilog, Apr. 1998, vol. 1525.

[19] S. Lyu and H. Farid, "Detecting hidden messages using higher-order statistics and support vector machines," in *5th International Workshop on Information Hiding*, Noordwijkerhout, The Netherlands, 2002.

[20] R. M. Gray and D. L. Neuhoff, "Quantization," *IEEE Trans. on Information Theory*, vol. 44, no. 6, pp. 2325 – 2383, Oct. 1998.

[21] G. E. Healey and R. Kondepudy, "Radiometric ccd camera calibration and noise estimation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 16, no. 3, pp. 267–276, Mar. 1994.

[22] C. E. Shannon, "Communication in the presence of noise," *Proceedings of the I.R.E.*, vol. 37, pp. 10–21, Jan. 1949.

[23] J. Fridrich and M. Goljan, "Digital image steganography using stochastic modulation," in *Proc. SPIE Electronic Imaging*, Santa Clara, CA, Jan. 21–24, 2003.

[24] J. Woods and H. Stark, *Probability and Random Processes With Applications to Signal Processing*, 3rd ed. Upper Saddle River, NJ: Prentice-Hall, 2001.

[25] A. V. Oppenheim, R. W. Schafer, and J. R. Buck, *Discrete-Time Signal Processing*, 2nd ed. Englewood Cliffs, NJ: Prentice Hall, 1999.

[26] A. Westfeld and A. Phitzmann, "Attacks on steganographic systems," in *Proceedings 3rd Information Hiding Workshop*, Dresden, Germany, Sept. 28-Oct. 1 1999, pp. 61–75.

[27] D. S. Mitrinović, J. E. Pečarić, and A. M. Fink, *Classical and New Inequalities in Analysis*. Dordrecht, The Netherlands: Kluwer Academic Publishers, 1993.

[28] C. Kurak and J. McHugh, "A cautionary note on image downgrading," in *Computer Security Applications Conference*, San Antonio, TX, Dec. 1992.

[29] L. M. Marvel, C. G. Boncelet, Jr, and C. T. Retter, "Spread spectrum image steganography," *IEEE Trans. Image Processing*, vol. 8, no. 8, pp. 1075–1083, Aug. 1999.

[30] R. L. Pickholtz, D. L. Schilling, and L. B. Milstein, "Theory of spread spectrum communications — a tutorial," *IEEE Trans. Comm.*, vol. COM-30, no. 5, pp. 855–884, May 1982.

[31] F. Alturki and R. Mersereau, "A novel approach for increasing security and data embedding capacity in images for data hiding applications," in *Information Technology: Coding and Computing*, Las Vegas, NV, Apr. 2–4, 1997, pp. 228–233.

[32] J. S. Lim, *Two-Dimensional Signal Processing*. Englewood Cliffs, NJ: Prentice Hall, 1990.

[33] D. R. Brillinger, "Fourier analysis of stationary processes," *Proceedings of the IEEE*, vol. 62, no. 12, pp. 1628–1643, Dec. 1974.

[34] W. A. Pearlman, "Quantization error bounds for computer-generated holograms," Information Systems Laboratory, Stanford University, Stanford,CA, Tech. Rep. 6503-1, Aug. 1974.

[35] R. Franzen. (2002, Mar. 27,) Kodak lossless true color image suite: PhotoCD PCD0992. Available: http://sqez.home.att.net/thumbs/Thumbnails.html.

[36] R. O. Duda, P. E. Hart, and H. G. Stork, *Pattern Classification*, 2nd ed. New York, NY: Wiley-Interscience, 2000.

[37] J. B. Bednar and T. L. Watt, "Alpha-trimmed means and their relationship to the median filters," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-32, pp. 145–153, Feb. 1984.

APPENDIX A
Spread Spectrum Image Steganography

Spread spectum image steganography (SSIS) was introduced in [29]. It serves to create a Gaussian sequence that contains the message bits. This sequence is added to the coverimage to produce the stegoimage. The following discussion describes the algorithm.

A.1 SSIS Embedding

To embed a message,

$$\mathbf{m} = \{m_1, m_2, \ldots, m_l\},\ m_i \in \{-1, 1\}.$$

We first use the key to create the a sequence of realizations of a uniform random variable distributed as $\mathcal{U}(0, 1)$,

$$\mathbf{u} = \{u_1, u_2, \ldots, u_l\}.$$

We create a second random sequence based on $\mathbf{u}$ as follows,

$$u_i' = \begin{cases} u_i + 0.5,\ 0 \leq u_i < 0.5, \\ u_i - 0.5,\ 0.5 \leq u_i \leq 1. \end{cases} \tag{A.1}$$

The stegonoise sequence, $\mathbf{s} = \{s_1, s_2, \ldots, s_l\}$, is then created as,

$$s_i = \begin{cases} \Phi^{-1}(u_i),\ m_i \ = -1, \\ \Phi^{-1}(u_i'),\ m_i \ = 1. \end{cases} \tag{A.2}$$

where $\Phi^{-1}(\cdot)$ is defined to be the inverse cumulative distribution function for a Gaussian variable. The purpose of the transformation of (A.1) is to maximize the difference between the values of s_i based on m_i. For example, in a direct modulation scheme, using a pseudo-noise sequence of $\mathcal{N}(0, \sigma^2)$ many of the modulated values

are near zero. This small separation causes decoding errors due to the estimation step discussed in the recovery section. The transforms seeks to prevent this by separating the values of the stegonoise.

Finally the stegonoise is added to the coverimage to produce the stegoimage,

$$\mathbf{X}_s = round(\mathbf{X}_c + \mathbf{s}). \tag{A.3}$$

A.2 SSIS Recovery

To recover the message, an estimate of the coverimage is made using an alpha-trimmed mean filter[37],

$$\widehat{\mathbf{X}_c} = atm\left(\mathbf{X}_s\right). \tag{A.4}$$

This allows for the estimation of the stegonoise,

$$\widehat{\mathbf{s}} = \mathbf{X}_s - \widehat{\mathbf{X}_c}. \tag{A.5}$$

Finally the message bits are recovered as,

$$\widehat{m_i} = sign\left(\widehat{s_i} \cdot \Phi^{-1}(u_i')\right). \tag{A.6}$$

APPENDIX B
Gaussian Multivariate

This derivation of the Gaussion Multivariate may be found in further detail in [36].

We begin with the class conditional *pdf* of the observation, $\mathbf{x} \in \Re^d$, given it is of class ω_i. The distribution is a d dimensional Gaussian with mean $\boldsymbol{\mu}_i$ and covariance matrix $\boldsymbol{\Sigma}_i$. This is denoted, $p(\mathbf{x}|\omega_i)$,

$$p(\mathbf{x}|\omega_i) = \frac{1}{(2\pi)^{d/2}\,|\boldsymbol{\Sigma}_i|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_i)^t \boldsymbol{\Sigma}_i^{-1}(\mathbf{x}-\boldsymbol{\mu}_i)}. \tag{B.1}$$

To classify an observation, $\mathbf{x}$, we will choose the class that maximizes the posterior probability. That is we choose class ω_i where,

$$\omega_i^* = \arg\max_i p(\mathbf{x}|\omega_i). \tag{B.2}$$

As the logarithm function is monotonically increasing we can replace (B.2) with,

$$\omega_i^* = \arg\max_i \ln p(\mathbf{x}|\omega_i). \tag{B.3}$$

We denote the likelihood function of ω_i as $g_i(\mathbf{x})$. It is defined as the natural logarithm of $p(\mathbf{x}|\omega_i)$. Using the properties of the natural logarithm we can write the discriminant as,

$$g_i(\mathbf{x}) = \ln p(\mathbf{x}|\omega_i) \tag{B.4a}$$

$$= -\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_i)^t \boldsymbol{\Sigma}_i^{-1}(\mathbf{x}-\boldsymbol{\mu}_i) - \frac{d}{2}\ln 2\pi - \frac{1}{2}\ln|\boldsymbol{\Sigma}_i|. \tag{B.4b}$$

To classify a sample $\mathbf{x}$, the discriminant function for each distribution is evaluated with $\mathbf{x}$ and the discriminant function yielding the largest value is chosen as the class.

For two categories, we have two discriminant functions, $g_1(\mathbf{x})$ and $g_2(\mathbf{x})$. By evaluating both functions with the observation vector, $\mathbf{x}$, we decide ω_1 if $g_1(\mathbf{x}) >$

$g_2(\mathbf{x})$, else we decide ω_2.

APPENDIX C
Error Bounds

C.1 Average Error

The probability of error is,

$$P(error|\mathbf{x}) = \begin{cases} P(\omega_1|\mathbf{x}), & \text{we decide } \omega_2 \\ P(\omega_2|\mathbf{x}), & \text{we decide } \omega_1 \end{cases} \tag{C.1}$$

As we are using the Bayes decision rule, we choose ω_1 if $P(\omega_1|\mathbf{x}) > P(\omega_2|\mathbf{x})$, else decide ω_2. We have,

$$P(error|x) = \min[P(\omega_1|x), P(\omega_2|x)] \tag{C.2}$$

C.2 Chernoff Bound

To find an upper bound for the error, we make use of,

$$\min[a, b] \leq a^{\beta} b^{1-\beta} \quad \forall\ a, b \geq 0 \text{ and } 0 \leq \beta \leq 1. \tag{C.3}$$

Applying the inequality to (C.2) we find,

$$P(error) \leq P^{\beta}(\omega_1) P^{1-\beta}(\omega_2) \int p^{\beta}(\mathbf{x}|\omega_1) p^{1-\beta}(\mathbf{x}|\omega_2) d\mathbf{x} \quad \forall\ 0 \leq \beta \leq 1. \tag{C.4}$$

This can be written as,

$$P(error) \leq P^{\beta}(\omega_1) P^{1-\beta}(\omega_2) e^{-k(\beta)}, \tag{C.5}$$

where,

$$k(\beta) = \frac{\beta(1-\beta)}{2} (\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1)^t [\beta\boldsymbol{\Sigma}_1 + (1-\beta)\boldsymbol{\Sigma}_2]^{-1} (\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1) + \frac{1}{2} \ln \frac{|\beta\boldsymbol{\Sigma}_1 + (1-\beta)\boldsymbol{\Sigma}_2|}{|\boldsymbol{\Sigma}_1|^{\beta} |\boldsymbol{\Sigma}_2|^{(1-\beta)}}. \tag{C.6a}$$

The value of β which maximizes $k(\beta)$ is found numerically. Evaluating (C.5) with the optimal β gives an upper bound on the error.