Image Steganography and Steganalysis: Concepts and Practice

Rajarathnam Chandramouli[1], Mehdi Kharrazi[2], and Nasir Memon[3]

[1] Department of Electrical and Computer Engineering
Stevens Institute of Technology, Hoboken, NJ 12345,USA
mouli@stevens-tech.edu
[2] Department of Electrical and Computer Engineering
Polytechnic University, Brooklyn, NY 11201, USA
mehdi@isis.poly.edu
[3] Department of Computer and Information Science
Polytechnic University, Brooklyn, NY 11201, USA
memon@poly.edu

Abstract. In the last few years, we have seen many new and powerful steganography and steganalysis techniques reported in the literature. In the following paper we go over some general concepts and ideas that apply to steganography and steganalysis. Specifically we establish a framework and define notion of security for a steganographic system. We show how conventional definitions do not really adequately cover image steganography and an provide alternate definition. We also review some of the more recent image steganography and steganalysis techniques.

1 Introduction

Steganography refers to the science of "invisible" communication. Unlike cryptography, where the goal is to secure communications from an eavesdropper, steganographic techniques strive to hide the very presence of the message itself from an observer. Although steganography is an ancient subject, the modern formulation of it is often given in terms of the *prisoner's problem* [1] where Alice and Bob are two inmates who wish to communicate in order to hatch an escape plan. However, all communication between them is examined by the warden, Wendy, who will put them in solitary confinement at the slightest suspicion of covert communication. Specifically, in the general model for steganography, illustrated in Figure 1, we have Alice wishing to send a secret message m to Bob. In order to do so, she "embeds" m into a *cover-object* c, to obtain the *stego-object* s. The stego-object s is then sent through the public channel. In a *pure steganography* framework, the technique for embedding the message is unknown to Wendy and shared as a secret between Alice and Bob. However, it is generally not considered as good practice to rely on the secrecy of the algorithm itself. In *private key steganography* Alice and Bob share a secret key which is used to embed the message. The secret key, for example, can be a password used to seed a pseudo-random number generator to select pixel locations in an image

T. Kalker et al. (Eds.): IWDW 2003, LNCS 2939, pp. 35–49, 2004.

cover-object for embedding the secret message (possibly encrypted). Wendy has no knowledge about the secret key that Alice and Bob share, although she is aware of the algorithm that they could be employing for embedding messages. In *public key steganography*, Alice and Bob have private-public key pairs and know each other's public key. In this paper we restrict our attention to private key steganography.

The warden Wendy who is free to examine all messages exchanged between Alice and Bob can be passive or active. A *passive* warden simply examines the message and tries to determine if it potentially contains a hidden message. If it appears that it does, she suppresses the message and/or takes appropriate action, else she lets the message through without any action. An *active* warden, on the other hand, can alter messages deliberately, even though she does not see any trace of a hidden message, in order to foil any secret communication that can nevertheless be occurring between Alice and Bob. The amount of change the warden is allowed to make depends on the model being used and the cover-objects being employed. For example, with images, it would make sense that the warden is allowed to make changes as long as she does not alter significantly the subjective visual quality of a suspected stego-image. In this paper we restrict our attention to the passive warden case and assume that no changes are made to the stego-object by the warden Wendy.

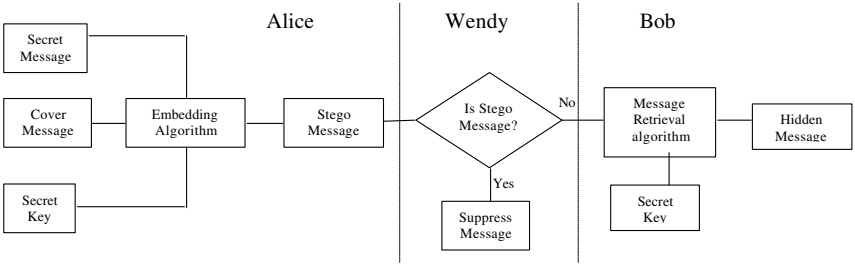

Fig. 1. Framework for Secret Key Passive Warden Steganography. Alice embeds secret message in cover image (left). Wendy the warden checks if Alice's image is a stego-image (center). If she cannot determine it to be so, she passes it on to Bob who retrieves the hidden message based on secret key (right) he shares with Alice.

It should be noted that the general idea of hiding some information in digital content has a wider class of applications that go beyond steganography. The techniques involved in such applications are collectively referred to as *information hiding*. For example, an image printed on a document could be annotated by metadata that could lead a user to its high resolution version. In general, metadata provides additional information about an image. Although metadata can also be stored in the file header of a digital image, this approach has many limitations. Usually, when a file is transformed to another format (e.g., from

TIFF to JPEG or to bmp), the metadata is lost. Similarly, cropping or any other form of image manipulation destroys the metadata. Finally, metadata can only be attached to an image as long as the image exists in the digital form and is lost once the image is printed. Information hiding allows the metadata to travel with the image regardless of the file format and image state (digital or analog).

A special case of information hiding is *digital watermarking*. Digital watermarking is the process of embedding information into digital multimedia content such that the information (the watermark) can later be extracted or detected for a variety of purposes including copy prevention and control. Digital watermarking has become an active and important area of research, and development and commercialization of watermarking techniques is being deemed essential to help address some of the challenges faced by the rapid proliferation of digital content. The key difference between information hiding and watermarking is the absence of an active adversary. In watermarking applications like copyright protection and authentication, there is an active adversary that would attempt to remove, invalidate or forge watermarks. In information hiding there is no such active adversary as there is no value associated with the act of removing the information hidden in the content. Nevertheless, information hiding techniques need to be robust against accidental distortions.

Unlike information hiding and digital watermarking, the main goal of steganography is to communicate securely in a completely undetectable manner. That is, Wendy should not be able to distinguish in any sense between cover-objects (objects not containing any secret message) and stego-objects (objects containing a secret message). In this context, *steganalysis* refers to the body of techniques that aid Wendy in distinguishing between cover-objects and stego-objects. It should be noted that Wendy has to make this distinction without any knowledge of the secret key which Alice and Bob may be sharing and sometimes even without any knowledge of the specific algorithm that they might be using for embedding the secret message. Hence steganalysis is inherently a difficult problem. However, it should also be noted that Wendy does not have to glean anything about the contents of the secret message m. Just determining the existence of a hidden message is enough. This fact makes her job a bit easier.

Given the proliferation of digital images, and given the high degree of redundancy present in a digital representation of an image (despite compression), there has been an increased interest in using digital images as cover-objects for the purpose of steganography. For a good survey of image steganography techniques, the reader is referred to [2]. The development of techniques for image steganography and the wide-spread availability of tools for the same have led to an increased interest in steganalysis techniques for image data. The last two years, for example, have seen many new and powerful steganalysis techniques reported in the literature. Many of such techniques are specific to different embedding methods and indeed have shown to be quite effective in this regard. However, our intention here is not to present a comprehensive survey of different embedding techniques and possible ways to detect them. Instead we focus on some general concepts

and ideas that apply across different techniques and cover-media. The rest of this paper is organized as follows: in section 2 we first establish a formal framework and define the notion of security for a steganographic system. We point out how conventional definitions do not really adequately cover image steganography (or steganography using any multimedia object for that matter) and provide alternate definitions. In section 3, we go over the more recent steganography and steganalysis techniques and in section 4 we conclude.

2 Steganographic Security

In this section we explore the topic of steganographic security. Some of the earlier work on this topic was done in [3,4,5]. Here, a steganographic system is considered to be insecure if the warden Wendy is able to prove the existence of a secret message. In other words, if she can distinguish between cover-objects and stego-objects, assuming she has unlimited computing power. Let P_C denote the probability distribution of cover-objects and P_S denote the probability distribution of stego-objects. Cachin [3] defines a steganographic algorithm to be ϵ-secure ($\epsilon \geq 0$) if the relative entropy between the cover-object and the stego-object probability distributions (P_C and P_S, respectively) is at most ϵ, i.e.,

$$D(P_C||P_S) = \int P_C \cdot log\frac{P_C}{P_S} \leq \epsilon \tag{1}$$

From this equation we note that $D(.)$ increases with the ratio $\frac{P_C}{P_S}$ which in turn means that the reliability of steganalysis detection will also increase. A steganographic technique is said to be *perfectly secure* if $\epsilon = 0$ (i.e. $P_C = P_S$). In this case the probability distributions of the cover and stego-objects are indistinguishable. Perfectly secure steganography algorithms (although impractical) are known to exist [3].

We observe that there are several shortcomings in the ϵ-secure definition presented in Eq. (1). Some of these are listed below.

- The ϵ-secure notion as presented in [3] assumes that the cover and stego-objects are vectors of independent, identically distributed (i.i.d.) random variables. This is not true for many real-life cover signals such as images. One approach to rectify this problem is to put a constraint that the relative entropy computed using the n-th order joint probability distributions must be less than, say, ϵ_n and then force the embedding technique to preserve this constraint. But, it may then be possible, at least in theory, to use $(n+1)$st order statistics for successful steganalysis. This line of thought clearly poses several interesting issues:
  - Practicality of preserving nth order joint probability distribution during embedding for medium to large values of n.
  - Behavior of the sequence $\{\epsilon_n\}$ depends on the cover message as well as the embedding algorithm. If this sequence exhibits a smooth variation then, for a desired target value, say, $\epsilon = \epsilon^*$, it may be possible to precompute a value of $n = n^*$ that achieves this target.

Of course, even if these nth order distributions are preserved, there is no guarantee that embedding induced perceptual distortion will be acceptable. If this distortion is significant, then it is not even necessary to use a statistical detector for steganalysis!

- While the ϵ-secure definition may work for random bit streams (with no inherent statistical structure), for real-life cover-objects such as audio, image, and video, it seems to fail. This is because, real-life cover-objects have a rich statistical structure in terms of correlation, higher-order dependence, etc. By exploiting this structure, it is possible to design good steganalysis detectors even if the first order probability distribution is preserved (i.e., $\epsilon = 0$) during message embedding. If we approximate the probability distribution functions using histograms, then, examples such as [6] show that it is possible to design good steganalysis detectors even if the histograms of cover and stego are the same.
- Consider the following embedding example. Let X and Y be two binary random variables such that $P(X = 0) = P(Y = 0) = 1/2$ and let them represent the host and covert message, respectively. Let the embedding function be given by the following:

$$Z = X + Y \bmod 2. \tag{2}$$

We then observe that $D(P_Z||P_X) = 0$ but $E(X - Z)^2 = 1$. Therefore the non-zero mean squared error value may give away enough information to a steganalysis detector even though $D(.) = 0$.

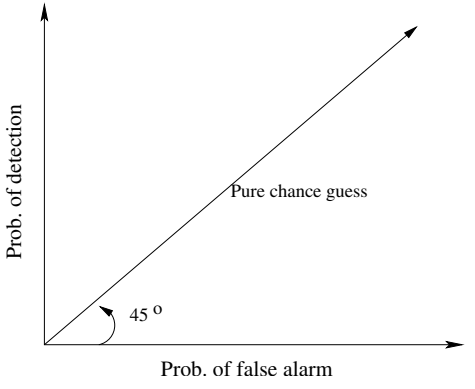

Fig. 2. Detector ROC plane.

Given these arguments, is there an alternative measure for stego security that is perhaps more fundamental to steganalysis? In the rest of this section we present an alternate definition of steganographic security. In our new definition,

the *false alarm probability* (α=P(detect message present|message absent)) and the *detection probability* (β=P(detect message present| message present) play an important role. A steganalysis detector's receiver operating characteristic (ROC) is a plot of α versus β. Points on the ROC curve represent the achievable performance of the steganalysis detector. The average error probability of steganalysis detection is given by,

$$P_e = (1-\beta)P(\text{message embedded}) + \alpha P(\text{message not embedded}). \quad (3)$$

If we assume $P(\text{message embedded}) = P(\text{message not embedded})$ then, from Eq. (3),

$$P_e = \frac{1}{2}\left[(1-\beta)+\alpha\right] \quad (4)$$

Note that, α and β are detector dependent values. For example, for a chosen value of α, β can be maximized by using a Neyman-Pearson statistical detector [7] or, both α and β can be fixed and traded-off with the number of observations required for detection by using Wald's sequential probability ratio test [8]. Observe from Eq. (4) that, if $\alpha = \beta$ then $P_e = 1/2$ as shown in Fig. 2. That is, the detector makes purely random guesses when it operates or forced to operate on the 45 degree line in the ROC plane. This means that the detector does not have sufficient information to make an intelligent decision. Therefore, if the embedder forces the detector to operate on the 45 degree ROC line by employing appropriate algorithms and/or parameters, then we say that the stego message is secure and obtain the following definitions.

Definition 1 *A stego embedding algorithm is said to be $\gamma_{\mathcal{D}}$-secure w.r.t. a steganalysis detector $\mathcal{D}$ if $|\beta_{\mathcal{D}} - \alpha_{\mathcal{D}}| \leq \gamma_{\mathcal{D}}$, where $0 \leq \gamma_{\mathcal{D}} \leq 1$.*

Definition 2 *A stego embedding algorithm is said to be perfectly secure w.r.t. a steganalysis detector $\mathcal{D}$ if $\gamma_{\mathcal{D}} = 0$.*

Clearly, from these definitions we can think of embedding and steganalysis as a zero sum game where the embedder attempts to minimize $|\beta - \alpha|$ while the steganalyst attempts to maximize it.

3 Steganalysis

There are two approaches to the problem of steganalysis, one is to come up with a steganalysis method specific to a particular steganographic algorithm. The other is developing techniques which are independent of the steganographic algorithm to be analyzed. Each of the two approaches has it's own advantages and disadvantages. A steganalysis technique specific to an embedding method would give very good results when tested only on that embedding method, and might fail on all other steganographic algorithms. On the other hand, a steganalysis

method which is independent of the embedding algorithm might preform less accurately overall but still provide acceptable results on new embedding algorithms. These two approaches will be discussed below and we will go over a few of the proposed techniques for each approach.

Before we proceed, one should note that steganalysis algorithms in essence are called successful if they can detect the presence of a message. The message itself does not have to be decoded. Indeed, the latter can be very hard if the message is encrypted using strong cryptography. However, recently there have been methods proposed in the literature which in addition to detecting the presence of a message are also able to estimate the size of the embedded message with great accuracy. We consider these aspects to be extraneous and only focus on the ability to detect the presence of a message.

3.1 Embedding Algorithm Specific Steganalysis Techniques

We first look at steganalysis techniques that are designed with a particular steganographic embedding algorithm in mind. Steganographic algorithms could be divided into 3 categories based on the type of the image used as the cover medium, i.e. Raw images (for example bmp format), Palette based images (for example GIF images), and finally JPEG images.

Raw Images are widely used with the simple LSB embedding method, where the message is embedded in a subset of the LSB (least significant bit) plane of the image, possibly after encryption. It is well known that an image is generally not visually affected when its least significant bit plane is changed. Popular steganographic tools based on LSB like embedding [9,10,11], vary in their approach for hiding information. Some algorithms change LSB of pixels visited in a random walk, others modify pixels in certain areas of images, or instead of just changing the last bit they increment or decrement the pixel value.

An early approach to LSB steganalysis was presented in [12] by Westfeld and Pfitzmann. They note that LSB embedding induces a partitioning of image pixels into Pairs of Values (PoV's) that get mapped to one another. For example the value 2 gets mapped to 3 on LSB flipping and likewise 3 gets mapped to 2. So (2, 3) forms a PoV. Now LSB embedding causes the frequency of individual elements of a PoV to flatten out with respect to one another. So for example if an image has 50 pixels that have a value 2 and 100 pixels that have a value 3, then after LSB embedding of the entire LSB plane the expected frequencies of 2 and 3 are 75 and 75 respectively. This of course is when the entire LSB plane is modified. However, as long as the embedded message is large enough, there will be a statistically discernible flattening of PoV distributions and this fact is exploited by their steganalysis technique. The length constraint, on the other hand, turns out to be the main limitation of their technique. LSB embedding can only be reliably detected when the message length becomes comparable with the number of pixels in the image. In the case where message placement is known, shorter messages can be detected. But requiring knowledge of message placement

is too strong an assumption as one of the key factors playing in the favor of Alice and Bob is the fact that the secret message is hidden in a location unknown to Wendy.

A more direct approach for LSB steganalysis that analytically estimates the length of an LSB embedded message in an image was proposed by Dumitrescu et. al. [13]. Their technique is based on an important statistical identity related to certain sets of pixels in an image. This identity is very sensitive to LSB embedding, and the change in the identity can quantify the length of the embedded message. This technique is described in detail below, where our description is adopted from [13].

Consider the partition of an image into pairs of horizontally adjacent pixels. Let $\mathcal{P}$ be the set of all these pixel pairs. Define the subsets X, Y and Z of $\mathcal{P}$ as follows:

- X is the set of pairs $(u, v) \in \mathcal{P}$ such that v is even and $u < v$, or v is odd and $u > v$.
- Y is the set of pairs $(u, v) \in \mathcal{P}$ such that v is even and $u > v$, or v is odd and $u < v$.
- Z is the subset of pairs $(u, v) \in \mathcal{P}$ such that $u = v$.

After having made the above definitions, the authors make the assumption that statistically we will have

$$|X| = |Y|. \tag{5}$$

This assumption is true for natural images as the gradient of intensity function in any direction is equally likely to be positive or negative.

Furthermore, they partition the set Y into two subsets W and V, with W being the set of pairs in $\mathcal{P}$ of the form $(2k, 2k+1)$ or $(2k+1, 2k)$, and $V = Y - W$. Then $\mathcal{P} = X \cup W \cup V \cup Z$. They call sets X, V, W and Z as *primary sets*.

When LSB embedding is done pixel values get modified and so does the membership of pixel pairs in the primary sets. More specifically, given a pixel pair (u, v), they identify the following four situations:

00) both values u and v remain unmodified;
01) only v is modified;
10) only u is modified;
11) both u and v are modified.

The corresponding change of membership in the primary sets is shown in Figure 3.

By some simple algebraic manipulations, the authors finally arrive at the equation

$$0.5\gamma p^2 + (2|X'| - |\mathcal{P}|)p + |Y'| - |X'| = 0. \tag{6}$$

where $\gamma = |W| + |Z| = |W'| + |Z'|$. The above equation allows one to estimate p, i.e the length of the embedded message, based on X', Y', W', Z' which can all be measured from the image being examined for possible steganography. Of course it should be noted that we cannot have $\gamma = 0$, the probability of which for natural images is very small.

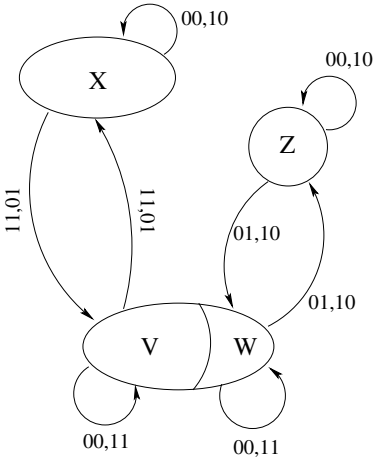

Fig. 3. State transition diagram for sets X, V, W, Z under LSB flipping.

In fact, the pairs based steganalysis described above was inspired by an effectively identical technique, although from a very different approach, called RS-Steganalysis by Fridrich et. al. in [14] that had first provided remarkable detection accuracy and message length estimation even for short messages. However, RS-Steganalysis does not offer a direct analytical explanation that can account for its success. It is based more on empirical observations and their modelling. It is interesting to see that the Pair's based steganalysis technique essentially ends up with exactly the same steganalyzer as RS-Steganalysis.

Although the above techniques are for gray scale images, they are applicable to color images by considering each color plane as a gray scale image. A steganalysis technique that directly analyzes color images for LSB embedding and yields high detection rates even for short messages was proposed by Fridrich, Du and Long [15]. They define pixels that are "close" in color intensity to be pixels that have a difference of not more than one count in any of the three color planes. They then show that the ratio of "close" colors to the total number of unique colors increases significantly when a new message of a selected length is embedded in a cover image as opposed to when the same message is embedded in a stego-image (that is an image already carrying a LSB encoded message). It is this difference that enables them to distinguish cover-images from stego-images for the case of LSB steganography.

In contrast to the simple LSB method discussed above, Hide [11] increments or decrements the sample value in order to change the LSB value. Thus the techniques previously discussed for LSB embedding with bit flipping do not detect Hide. In order to detect embedded messages by Hide, Westfeld [16] proposes a similar steganalysis attack as Fridrich, Du and Long [15] were it is argued that since the values are incremented or decremented, 26 neighboring colors for

each color value could be created, were as in a natural image there are 4 to 5 neighboring colors on average. Thus by looking at the neighborhood histogram representing the number of neighbors in one axis and the frequency in the other one would be able to say if the image carries a message. This is clearly seen in Fig. 4.

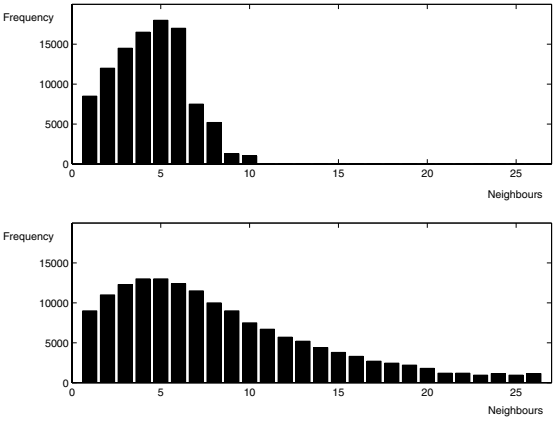

Fig. 4. Neighborhood histogram of a cover image (top) and stego image with 40 KB message embedded (bottom)[16]

Palette Based Images like GIF images, are another popular class of images for which there have been a number of steganography methods proposed [17,18, 19]. Perhaps some of the earliest steganalysis work in this regard was reported by Johnson and Jajodia [20]. They mainly look at palette tables in GIF images and anomalies caused therein by common stego-tools that perform LSB embedding in GIF images. Since pixel values in a palette image are represented by indices into a color look-up table which contains the actual color RGB value, even minor modifications to these indices can result in annoying artifacts. Visual inspection or simple statistics from such stego-images can yield enough tell-tale evidence to discriminate between stego and cover-images.

In order to minimize the distortion caused by embedding, EzStego [17] first sorts the color pallet so that the color differences between consecutive colors is minimized. It then embeds the message bits in the LSB of the color indices in the sorted pallet. Since pixels which can modified due to the embedding process get mapped neighboring colors in the palette, which are now similar, visual artifacts are minimal and hard to notice. To detect EzStego, Fridrich [6] argues that a vector consisting of color pairs, obtained after sorting the pallet, has considerable structure due to the fact there a small number of colors in pallet images. But the embedding process will disturb this structure, thus after the embedding the

entropy of the color pair vector will increase. The entropy would be maximal when the maximum length message is embedded in to the GIF image. Another steganalysis techniques for EzStego were proposed by Westfeld [12], but the technique discussed above provides a much higher detection rate and a more accurate estimate of the message lengths.

JPEG Images are the the third category of images which are used routinely as cover medium. Many steganalysis attacks have been proposed for steganography algorithms [21,22,23] which employ this category of images. Fridrich [6] has proposed attacks on the F5 and Outguess algorithms, both of which work on jpeg images. F5 [23] embeds bits in the DCT coefficients using matrix embedding so that for a given message the number of changes made to the cover image is minimized. But F5 does alter the histogram of DCT coefficients. Fridrich proposes a simple technique to estimate the original histogram so that the number of changes and length of the embedded message could be estimated. The original histogram is simply estimated by cropping the jpeg image by 4 columns and then re-compressing the image using the same quantization table as used before. As is evident in Fig 5, the resulting DCT coefficient histogram would be a very good estimate of the original histogram. Although no analytical proof is given for the estimation method, steganalysis based on this simple technique preforms very well.

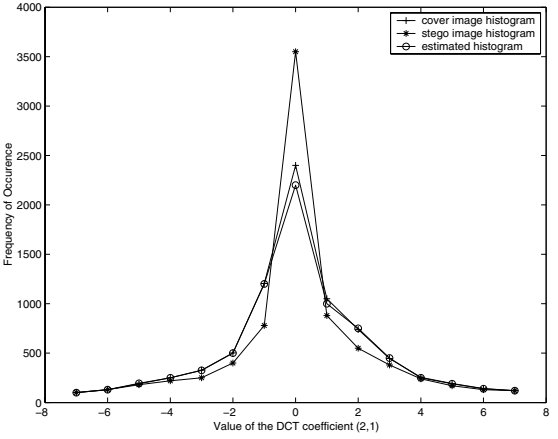

Fig. 5. The effect of F5 embedding on the histogram of the DCT coefficient (2,1).[6]

A second technique proposed by Fridrich [6] deals with the Outguess [21] embedding program. Outguess first embeds information in LSB of the DCT coefficients by making a random walk, leaving some coefficients unchanged. Then it adjusts the remaining coefficient in order to preserve the original histogram

of DCT coefficients. Thus the previous steganalysis method where the original histogram is estimated will not be effective. On the other hand when embedding messages in a clean image, noise is introduced in the DCT coefficient, therefore increasing the spatial discontinuities along the 8x8 jpeg blocks. Given a stego image if a message is embedded in the image again there is partial cancellation of changes made to the LSB of DCT coefficients, thus the increase in discontinuities will be smaller. This increase or lack of increase in the discontinuities is used to estimate the message size which is being carried by a stego image.

3.2 Universal Steganalysis Techniques

The steganalysis techniques described above were all specific to a particular embedding algorithm. A more general class of steganalysis techniques pioneered independently by Avcibas et. al. [24,25,26] and Farid [27], are designed to work with any steganographic embedding algorithm, even an unknown algorithm. Such techniques have subsequently been called *Universal Steganalysis* techniques or *Blind Steganalysis Techniques*. Such techniques essentially design a classifier based on a training set of cover-objects and stego-objects arrived at from a variety of different algorithms. Classification is done based on some inherent "features" of typical natural images which can get violated when an image undergoes some embedding process. Hence, designing a feature classification based universal steganalysis technique consists of tackling two independent problems. The first is to find and calculate features which are able to capture statistical changes introduced in the image after the embedding process. The second is coming up with a strong classification algorithm which is able to maximize the distinction captured by the features and achieve high classification accuracy.

Typically, a good feature should be accurate, consistent and monotonic in capturing statistical signatures left by the embedding process. Prediction accuracy can be interpreted as the ability of the measure to detect the presence of a hidden message with minimum error on average. Similarly, prediction monotonicity signifies that the features should ideally be monotonic in their relationship to the embedded message size. Finally, prediction consistency relates to the feature's ability to provide consistently accurate predictions for a large set of steganography techniques and image types. This implies that the feature should be independent on the type and variety of images supplied to it.

In [26] Avcibas et. al. develop a discriminator for cover images and stego images, using an appropriate set of Image Quality Metrics (IQM's). Objective image quality measures have been utilized in coding artifact evaluation, performance prediction of vision algorithms, quality loss due to sensor inadequacy etc. In [26] they are used not as predictors of subjective image quality or algorithmic performance, but specifically as a steganalysis tool, that is, as features used in distinguishing cover-objects from stego-objects.

To select quality metrics to be used for steganalysis, the authors use Analysis of Variance (ANOVA) techniques. They arrive at a ranking of IQM's based on their F-scores in the ANOVA tests to identify the ones that responded most consistently and strongly to message embedding. The idea is to seek IQM's that

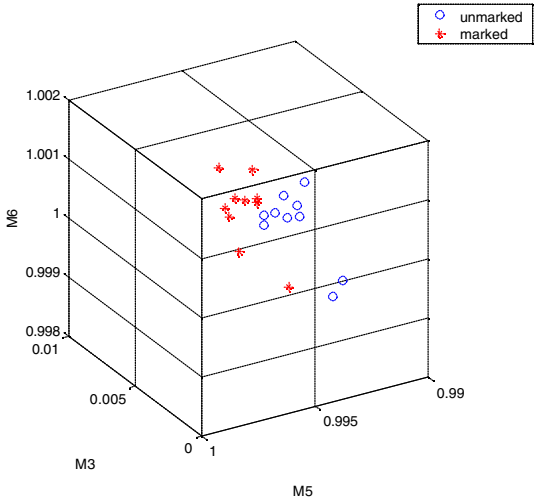

Fig. 6. Scatter plot of 3 image quality measures showing separation of marked and unmarked images.

are sensitive specifically to steganography effects, that is, those measures for which the variability in score data can be explained better because of some treatment rather then as random variations due to the image set. The rationale of using several quality measures is that different measures respond with differing sensitivities to artifacts and distortions. For example, measures like mean-square-error respond more to additive noise, whereas others such as spectral phase or mean square HVS-weighted (Human Visual System) error are more sensitive to pure blur; while the gradient measure reacts to distortions concentrated around edges and textures. Similarly embedding techniques affect different aspects of images. Fig.6 shows separation in the feature plane between stego images and cover images, for 3 example quality metrics.

A second technique proposed by Avcibas et. al. [24] looks at seventh and eight bit planes of an image and calculates several binary similarity measures. The approach is based on the fact that correlation between contiguous bit-planes is effected after a message is embedded in the image. The authors conjecture that correlation between the contiguous bit planes decreases after a message is embedded in the image. In order to capture the effect made by different embedding algorithms several features are calculated. Using the obtained features a MMSE linear predictor is obtained which is used to classify a given image as either as a cover image or an image containing hidden messages.

A different approach is taken by Farid et. al[27] for feature extraction from images. The authors argue that most of the specific steganalysis techniques concentrate on first order statistics, i.e. histogram of DCT coefficients, but simple counter measure could keep the first order statistics intact thus making the ste-

ganalysis technique useless. So they propose building a model for natural images by using higher order statistics and then show that images with messages embedded in them deviate form this model. Quadratic mirror filters (QMF) are used to decompose the image, after which higher order statistics such as mean, variance, kurtosis, and skewness are calculated for each subband. Also the error obtained from an optimal linear predictor of coefficient magnitudes of each subband is used as a second set of features.

In all of the above methods, the calculated features are used to train a classifier, which in turn is used to classify clean and stego images. Different classifiers have been employed by different authors, Avcibas et. al. uses a MMSE Linear predictor, where as Farid et. al[27] uses a Fisher linear discriminant [28] and also a Support Vector Machine (SVM) [29] classifier. SVM classifiers seem to have much better performance in terms of classification accuracy compared to linear classifiers since they are able to classify non-linearly separable features. All of the above authors have reported good accuracy results in classifying images as clean or containing hidden messages after training with a classifier. Although, direct comparison might be hard as is in many classification problems, due to the fact that the way experiments are setup or conducted could be very different and thus could effect the overall results.

4 Conclusions

The past few years have seen an increasing interest in using images as cover media for steganographic communication. There have been a multitude of public domain tools, albeit many being ad-hoc and naive, available for image based steganography. Given this fact, detection of covert communications that utilize images has become an important issue. There have been several techniques for detecting stego-images that have been developed in the past few years. In this paper we have reviewed some fundamental notions related to steganography using image media, including security. We also described in detail a number of steganalysis techniques techniques that are representative of the different approaches that have been taken.

References

1. G. Simmons, "The prisoners problem and the subliminal channel," *CRYPTO*, pp. 51–67, 1983.
2. N. F. Johnson and S. Katzenbeisser, "A survay of steganographic techniques," *in S. Katzenbeisser and F. Petitcolas (Eds.): Information Hiding, pp 43–78. Artech House, Norwood, MA.*, 2000.
3. C. Cachin, "An information-theoretic model for steganography," *2nd International Workshop Information Hiding*, vol. LNCS 1525, pp. 306–318, 1998.
4. J. Zollner, H. Federrath, H. Klimant, A. Pfitzman, R. Piotraschke, A. Westfeld, G. Wicke, and G. Wolf, "Modeling the security of steganographic systems," *2nd Information Hiding Workshop*, pp. 345–355, April 1998.

5. R. Chandramouli and N. Memon, "Steganography capacity: A steganalysis perspective," *To appear in SPIE Security and Watermarking of Multimedia Contents V*, vol. 5020, 2003.
6. J. Fridrich, M. Goljan, D. Hogea, and D. Soukal, "Quantitive steganalysis of digital images: Estimating the secret message lenght," *ACM Multimedia Systems Journal, Special issue on Multimedia Security*, 2003.
7. R. Chandramouli and N. Memon, "Analysis of lsb image steganography techniques," *IEEE Intnl. Conf. on Image Processing*, vol. 3, pp. 1019–1022, 2001.
8. ——, "On sequential watermark detection," *IEEE Transactions on Signal Processing*, vol. 51, no. 4, pp. 1034–1044, April 2003.
9. F. Collin, "Encryptpic," *http://www.winsite.com/bin/Info?500000033023.*
10. G. Pulcini, "Stegotif," *http://www.geocities.com/SiliconValley/9210/gfree.html.*
11. T. Sharp, "Hide 2.1, 2001," *http://www.sharpthoughts.org.*
12. A. Westfeld and A. Pfitzmann, "Attacks on steganographic systems," *Information Hiding. 3rd International Workshop*, p. 61–76, 1999.
13. S. Dumitrescu, X. Wu, and N. Memon, "On steganalysis of random lsb embedding in continuous-tone images," *IEEE International Conference on Image Processing, ROchester, New York.*, September 2002.
14. J. Fridrich, M. Goljan, and R. Du, "Detecting lsb steganography in color and gray-scale images," *IEEE Multimedia Special Issue on Security*, pp. 22–28, October-November 2001.
15. J. Fridrich, R. Du, and L. Meng, "Steganalysis of lsb encoding in color images," *ICME 2000, New York, NY, USA.*
16. A. Westfeld, "Detecting low embedding rates," *Information Hiding. 5th International Workshop*, p. 324–339, 2002.
17. R. Machado, "Ezstego," *http://www.stego.com*, 2001.
18. M. Kwan, "Gifshuffle," *http://www.darkside.com.au/gifshuffle/.*
19. C. Moroney, "Hide and seek," *http://www.rugeley.demon.co.uk/security/hdsk50.zip.*
20. N. F. Johnson and S. Jajodia, "Steganalysis of images created using current steganography software," *in David Aucsmith (Eds.): Information Hiding, LNCS 1525, Springer-Verlag Berlin Heidelberg.*, pp. 32–47, 1998.
21. N. Provos, "Defending against statistical steganalysis," *10th USENIX Security Symposium*, 2001.
22. D. Upham, "Jpeg-jsteg," *ftp://ftp.funet.fi/pub/crypt/steganography/jpeg-jsteg-v4.diff.gz.*
23. A. Westfeld, "F5—a steganographic algorithm: High capacity despite better steganalysis," *Information Hiding. 4th International Workshop*, p. 289–302, 2001.
24. I. Avcibas, N. Memon, and B. sankur, "Steganalysis using image quality metrics." *Security and Watermarking of Multimedia Contents, San Jose, Ca.*, Feruary 2001.
25. ——, "Image steganalysis with binary similarity measures." *IEEE International Conference on Image Processing, ROchester, New York.*, September 2002.
26. ——, "Steganalysis using image quality metrics." *IEEE transactions on Image Processing*, January 2003.
27. H. Farid and S. Lyu, "Detecting hidden messages using higher-order statistics and support vector machines," *5th International Workshop on Information Hiding.*, 2002.
28. R. Duda and P. Hart, "Pattern classification and scene analysis," *John Wiley and Sons.*, 1973.
29. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery.*, pp. 2:121–167, 1998.