Image Steganography: Concepts and Practice

Mehdi Kharrazi[1], Husrev T. Sencar[2], and Nasir Memon[2]
[1]*Department of Electrical and Computer Engineering*
[2]*Department of Computer and Information Science*
Polytechnic University, Brooklyn, NY 11201, USA
{*mehdi, taha, memon*}*@isis.poly.edu*

In the last few years, we have seen many new and powerful steganography and steganalysis techniques reported in the literature. In the following tutorial we go over some general concepts and ideas that apply to steganography and steganalysis. We review and discuss the notions of steganographic security and capacity. Some of the more recent image steganography and steganalysis techniques are analyzed with this perspective, and their contributions are highlighted.

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. The general idea of hiding some information in digital content has a wider class of applications that go beyond steganography, Fig. 1. 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.

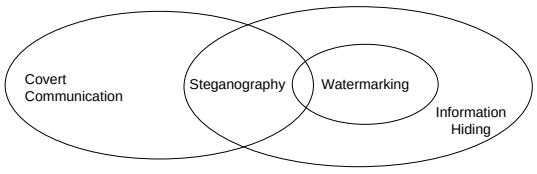

Fig. 1. Relationship of steganography to related fields.

Unlike information hiding and digital watermarking, the main goal of steganography is to communicate securely in a completely undetectable manner. Although steganography is an ancient art, first used against the persian by the romans, it has evolved much trough the years.

In the following tutorial we focus on some general concepts and ideas that apply across the field of steganography. The rest of this tutorial is organized as follows: in section 2 we first define the problem which steganography tries to address and introduce to the reader some terminologies commonly used in the field. In section 3 we go over different approaches in defining security. In section 4, the notion of steganographic capacity is discussed, section 5 goes over some embedding techniques, and in sections 6 some steganalysis techniques are reviewed. We conclude in section 7.

2. General Concepts

In this section we go over the concepts and definitions used in the field of steganography. We first start by going over the framework in which steganography is usually presented and then go over some definitions.

The modern formulation of steganography 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 Fig. 2, 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, and obtains a *stego-object* s. The stego-object s is then sent through the public channel. Thus we have the following definitions:

Cover-object: refers to the object used as the carrier to embed messages into. Many different objects have been employed to embed messages into for example images, audio, and video as well as file structures, and html pages to name a few.

Stego-object: refers to the object which is carrying a hidden message. so given a cover object, and a messages the goal of the steganographer is to produce a stego object which would carry the message.

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 considered that the algorithm in use is not secret but only the key used by the algorithm is kept as a secret between the two parties, this assumption is also known as Kerchoff's principle in the field of cryptography. The secret key, for example, can be a password used to seed a pseudo-random number generator to select pixel locations in an image 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.

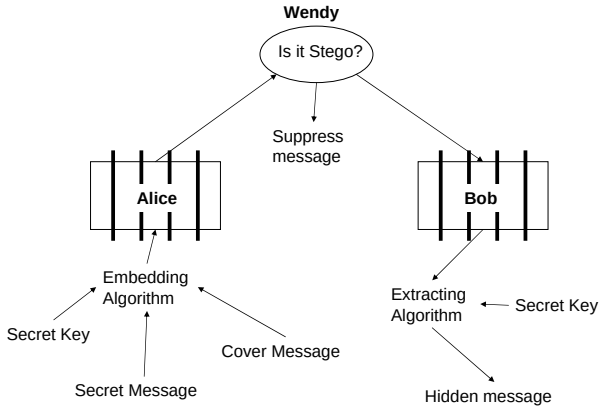

Fig. 2. General model for steganography.

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 tutorial we assume that no changes are made to the stego-object by the warden Wendy.

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.

The development of techniques for steganography and the wide-spread availability of tools for the same have led to an increased interest in steganalysis techniques. 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. We will review these techniques in the coming sections.

3. Steganographic Security

In steganography, unlike other forms of communications, one's awareness of the underlying communication between the sender and receiver defeats the whole purpose. Therefore, the first requirement of a steganographic system is its *undetectability*. In other words, a steganographic system is considered to be *insecure*, if the warden Wendy is able to differentiate between cover-objects and stego-objects.

There have been various approaches in defining and evaluating the *security* of a steganographic system. Zollner et al. [2] were among the first to address the undetectability aspect of steganographical systems. They provide an analysis to show that information theoretically secure steganography is possible if embedding operation has a random nature and the embedded message is independent from both the cover-object and stego-object. These conditions, however, ensure undetectability against an attacker who knows the stego-object but has no information available about the indeterministic embedding operation. That is, Wendy has no access to the statistics, distribution, or conditional distribution of the cover-object.

On the other hand, [3,4] approached steganographic security from a complexity theoretic point of view. Based on cryptographic principles, they

propose the design of encryption-decryption functions for steganographic embedding and detection. In this setting, the underlying distribution of the cover-objects is known by the attacker, and *undetectability* is defined in a conditional sense as the inability of a polynomial-time attacker (Wendy) to distinguish the stego-object from a cover-object. This model assumes that stego-object is a distorted version of the cover-object, however, it does not attempt to probabilistically characterize the stego object.

In [5], Cachin defined the first steganographic security measure that quantifies the information theoretic security of a stegosystem. His model assigns probability distributions to cover-object and stego-object under which they are produced. Then, the task of Wendy is to decide whether the observed object is produced according to known cover-object distribution or not. In the best case scenario, Wendy also knows the distribution of stego-object and makes a decision by performing a binary hypothesis test. Consequently, the detectability of a stegosystem is based on relative entropy between the probability distributions of the cover-object and stego-object, denoted by P_c and P_s, respectively, i.e.,

$$D(P_c||P_s) = \int P_c \log \frac{P_c}{P_s}. \tag{1}$$

From this equation, we note that $D(P_c||P_s)$ increases with the ratio $\frac{Pc}{Ps}$ which in turn means that the reliability of steganalysis detector will also increase. Accordingly, a stego technique is said to be perfectly secure if $D(P_c||P_s) = 0$ (P_c and P_s are equal), and ϵ-secure if the relative entropy between P_c and P_s is at most ϵ, $D(P_c||P_s) \leq \epsilon$. Perfectly secure algorithms are shown to exist, although they are impractical [5]. However, it should be noted that this definition of security is based on the assumption that the cover-object and stego-object are independent, identically distributed (i.i.d.) vectors of random variables.

Since Wendy uses hypothesis testing in distinguishing between stego-objects and cover-objects, she will make two types of errors, namely, type-I and type-II errors. A type-I error, with probability α occurs, when a cover-object is mistaken for a stego-object (false alarm rate), and a type-II error, with probability β, occurs when a stego-object is mistaken for a cover-object (miss rate). Thus bounds on these error probabilities can be computed using relative entropy, thereby relating steganographic security to detection error probabilities. Cachin [5] obtains these bounds utilizing the facts that deterministic processing can not increase the relative entropy between two distributions, say, P_c and P_s, and hypothesis testing is a form

of processing by a binary function that yields α (P(detect message present | message absent)) and β (P(detect message absent | message present)). Then, the relative entropy between distributions P_c and P_s and binary relative entropy of two distributions with parameters (α,$1-\alpha$) and (β, $1-\beta$) need to satisfy

$$d(\alpha, \beta) \leq D(P_c||P_s), \tag{2}$$

where $d(\alpha, \beta)$ is expressed as

$$d(\alpha, \beta) = \alpha \log \frac{\alpha}{1-\beta} + (1-\alpha) \log \frac{1-\alpha}{\beta}. \tag{3}$$

Then, for an ϵ-secure stegosystem we have

$$d(\alpha, \beta) \leq \epsilon. \tag{4}$$

Consequently, when the false alarm rate is set to zero ($\alpha = 0$), the miss rate is lower bounded as $\beta \geq 2^{-\epsilon}$. It should be noted that the probability of detection error for Wendy is defined as

$$P_e = \alpha P(\text{message absent}) + \beta P(\text{message present}). \tag{5}$$

Based on above equations, for a perfectly secure stegosystem, $\alpha + \beta = 1$, and when a cover-object is equally likely to undergo embedding operation, then $P_e = \frac{1}{2}$. Hence, Wendy's decisions are unreliable.

As one can observe, there are several shortcomings in the above definition of security. 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 these structures, it is possible to design good steganalysis detectors even if the first order probability distribution is preserved (i.e., $\epsilon = 0$) during the embedding process. 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 the cover image is and the stego image 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:

$$Z = X + Y \bmod 2. \tag{6}$$

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 $D(.) = 0$.

One attempt to overcome the limitations of i.i.d. cover-object model was made by Wang et al. [7] where they extended Cachin's results to multivariate Gaussian case, assuming that cover-object and stego-object are vectors of length N with distributions $P_{\mathbf{c}^N}$ and $P_{\mathbf{s}^N}$, respectively. In the multivariate case, similar to i.i.d. case, undetectability condition requires that the distribution of cover-object is preserved after embedding. However, when this is not possible, the degree of detectability of a stegosystem will depend on the deviation from the underlying distribution and the covariance structure of the cover-object. If the cover-object is jointly Gaussian with zero mean and covariance matrix $R_{\mathbf{c}^N}$, among all distributions (with zero mean and covariance matrix $R_{\mathbf{s}^N}$) the Gaussian distribution for the stego-object minimizes the relative entropy. Then, the detectability of stegosystem can be quantified based on the relative entropy as

$$D(P_{\mathbf{c}^N}||P_{\mathbf{s}^N}) = \frac{1}{2}\left(tr(\hat{R}) - \log(\hat{R} + I_N)\right) \approx \frac{1}{4}tr(\hat{R}^2) \tag{7}$$

where tr(.) denotes the trace of a matrix, I_N is the $N \times N$ identity matrix, and $\hat{R} = R_{\mathbf{c}^N} R_{\mathbf{s}^N}^{-1} - I_N$. Consequently, Wendy's detection error probability, P_e can be lower bounded as [7]

$$P_e > \frac{1}{2}\exp^{-\frac{D(P_{\mathbf{c}^N}||P_{\mathbf{s}^N}) + D(P_{\mathbf{s}^N}||P_{\mathbf{c}^N})}{2}} \tag{8}$$

assuming both hypotheses are equally likely, i.e., $P_e = \frac{1}{2}\alpha + \frac{1}{2}\beta$.

Although [7] addressed the inherent limitation of the ϵ-secure notion of Cachin, [5], by considering non-white cover-objects, due to analytical tractability purposes they limited their analysis to cover-objects that are generated by a Gaussian stationary process. However, as stated before, this is not true for many real-life cover-objects. One approach to rectify this problem is to probabilistically model the cover-objects or their transformed versions or some perceptually significant features of the cover-object and 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 $(n + 1)$th 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 ϵ_n depends on the cover message as well as the embedding algorithm. If it varies monotonically with n then, for a desired target value, say, $\epsilon = \epsilon^*$, it may be possible to pre-compute 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 distortions will be acceptable. If such distortions are significant, then it is not even necessary to use a statistical detector for steganalysis!

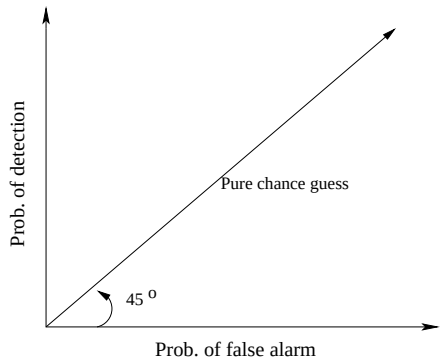

Fig. 3. Detector ROC plane. (Figure taken from [8])

From a practical point of view, Katzenbeisser et al. [9] propose the idea of using an indistinguishability test to define the security of a stegosystem. In their model, Wendy has access to cover-object and stego-object generation mechanisms and uses them consecutively to learn the statistical features of both objects to distinguish between them, rather than assuming their true probability distributions are available. In a similar manner, Chandramouli et al. [8] propose an alternative measure for steganographic security. Their definition is based on the false alarm probability (α), the detection probability ($1 - \beta$), and the steganalysis detector's receiver operating characteristic (ROC) which is a plot of α versus $1 - \beta$. Points on the ROC curve represent the achievable performance of the steganalysis detector. The average error probability of steganalysis detection is as defined in Eq. (5). Assuming P(message present)=P(message absent) and setting $\alpha = 1 - \beta$, then $P_e = 1/2$ and ROC curve takes the form shown in Fig. 3. That is, the detector makes purely random guesses when it operates or forced to operate on the 45 degree line in the ROC plane. Then, the stegano-

graphic security can be defined in terms of the deviation of the steganalysis detector's operation curve from the 45 degree ROC line. Correspondingly, a stegosystem can be defined to be $\gamma_{\mathcal{D}}$-secure with respect to a steganalysis detector $\mathcal{D}$ when $|1 - \beta_{\mathcal{D}} - \alpha_{\mathcal{D}}| \leq \gamma_{\mathcal{D}}$ where $0 \leq \gamma_{\mathcal{D}} \leq 1$ and $\gamma_{\mathcal{D}} = 0$ refers to the perfect security condition, similar to the ϵ-security notion of Cachin [5].

4. Steganographic Capacity

Steganographic capacity refers to the maximum amount (rate) of information that can be embedded into a cover-object and then can be reliably recovered from the stego-object (or a distorted version), under the constraints of undetectability, perceptual intactness and robustness, depending on whether Wendy is active or passive. Compared to data hiding systems, stegosystems have the added core requirement of undetectability. Therefore, the steganographic embedding operation needs to preserve the statistical properties of the cover-object, in addition to its perceptual quality. On the other hand, if Wendy suspects of a covert communication but cannot reliably make a decision, she may choose to modify the stego-object before delivering it. This setting of steganography very much resembles to data hiding problem, and corresponding results on data hiding capacity can be adapted to steganography [10].

As discussed in the previous section, the degree of undetectability of a stegosystem is measured in terms of a distance between probability distributions $P_{\mathbf{c}^N}$ and $P_{\mathbf{s}^N}$, i.e., $D(P_{\mathbf{c}^N}||P_{\mathbf{s}^N}) \leq \epsilon$ where $\epsilon = 0$ is the perfect security condition. Let $d(\mathbf{c}^N, \mathbf{s}^N)$ be a perceptual distance measure defined between cover-object $\mathbf{c}^N$ and stego-object $\mathbf{s}^N$. When the warden is passive, the steganographic capacity C_p of a perfectly secure stegosystem with embedding distortion limited to P is defined, in terms of random vectors $\mathbf{s}^N$ and $\mathbf{c}^N$, as

$$C_p = \{\sup H(\mathbf{s}^N|\mathbf{c}^N) : P_{\mathbf{c}^N} = P_{\mathbf{s}^N} \text{ and } \frac{1}{N} E[d(\mathbf{c}^N, \mathbf{s}^N)] \leq P\} \qquad (9)$$

where $E[.]$ denotes the expected value and supremum is taken over all $P_{\mathbf{s}^N|\mathbf{c}^N}$ for the given constraints. In [10], Moulin et al. discuss code generation (embedding) for a perfectly secure stegosystem with binary i.i.d. cover-object and Hamming distortion measure, and provide capacity results. However, generalization of such techniques to real life cover-objects is not possible due to two reasons. First is the simplistic i.i.d. assumption, and second is the utilized distortion measure as there is no trivial relation

between bit error rate and reconstruction quality.

In order to be able to design practical stegosystems, the perfect security condition in Eq. (9) can be relaxed by replacing it with the ϵ-security notion. One way to exploit this is by identifying the perceptually significant and insignificant parts of the cover-object $\mathbf{c}^N$, and preserving the statistics of the significant component while utilizing the insignificant component for embedding. For this, let there be a function g(.) such that $d(\mathbf{c}^N, g(\mathbf{c}^N)) \approx 0$ and $g(\mathbf{c}^N) = g(\mathbf{s}^N)$. Then, Eq. (9) can be modified as

$$C_p = \{\sup H(\mathbf{s}^N|\mathbf{c}^N) : P_{g(\mathbf{c}^N)} = P_{g(\mathbf{s}^N)} \text{ and } \frac{1}{N}E[(d(\mathbf{c}^N, \mathbf{s}^N)] \leq P\} \quad (10)$$

where $D(P_{\mathbf{c}^N}||P_{\mathbf{s}^N}) \leq \epsilon$. This approach requires statistical modelling of the cover-object or of some features of it, which will be modified during embedding. For example, [11,12,13] observe the statistical regularity between pairs of sample values in an image, and provide a framework for (ϵ-secure) embedding in least significant bit (LSB) layer. Similarly, Sallee [14] models AC components of DCT coefficients by Generalized Cauchy distribution and uses this model for embedding. In the same manner, wavelet transformed image coefficients can be marginally modelled by Generalized Laplacian distribution [15]. This approach, in general, suffers due to the difficulty in modelling the correlation structure via higher order joint distributions which is needed to ensure ϵ-security.

In the presence of an active warden, the steganographic capacity can be determined based on the solution of data hiding capacity with the inclusion of undetectability or ϵ-security condition. Data hiding capacity has been the subject of many research works, see, [16,17,18,19,20,21,22,23,24,25] and references therein, where the problem is viewed as a channel communication scenario with side information at the encoder. Accordingly, the solution for the data hiding capacity requires consideration of an auxiliary random variable u that serves as a random codebook shared by both embedder and detector. Let the distorted stego-object be denoted by y, and assume cover-object and stego-object are distorted by amounts P and D during embedding operation and attack, respectively. Since undetectability is the central issue in steganography, we consider the additional constraint of $P_c = P_s$. Then, the steganographic capacity for the active warden case, C_a, is derived, in terms of i.i.d. random variables c, u, s, and y, as

$$C_a = \{\sup I(u, y) - I(u, c) : P_c = P_s, E[(d(c, s)] \leq P, \text{and} E[(d(s, y)] \leq D\} \quad (11)$$

where supremum is taken over all distributions $P_{u|c}$ and all embedding func-

tions under the given constraints. The computation of the steganographic capacity of practical stegosystems, using Equations (9)-(11), still remains to be an open problem due to lack of true statistical models and for reasons of analytical tractability.

Chandramouli et al. [13], from a practical point of view, make an alternative definition of steganographic capacity based on the γ-security notion given in the previous section [8]. They define steganographic capacity from a detection theoretic perspective, rather than information theoretic, as the maximum message size that can be embedded so that a steganalysis detector is only able to a make a perfectly random guess about the presence/absence of a covert message. This indicates that the steganographic capacity in the presence of steganalysis varies with respect to the steganalysis detector. Therefore, its formulation must involve parameters of the embedding function as well as that of the steganalysis detector. Assuming N is the number of message carrying symbols, and $\alpha_{\mathcal{D}}^{(N)}$ and $1-\beta_{\mathcal{D}}^{(N)}$ are the corresponding false alarm and detection probabilities for a steganalysis detector $\mathcal{D}$, the steganographic capacity is defined as

$$N_{\gamma}^{*} = \{\max N \text{ subject to } |1-\beta_{\mathcal{D}}^{(N)} - \alpha_{\mathcal{D}}^{(N)}| \leq \gamma_{\mathcal{D}}\} \text{ symbols.} \qquad (12)$$

Based on this definition, [13] provide an analysis on the capacity of LSB steganography and investigate under what conditions an observer can distinguish between stego-images and cover-images.

5. Techniques for Image Steganography

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. Therefore we have limited our discussion to the case of images for the rest of this tutorial. We should also note that there have been much more work on embedding techniques which make use of the transform domain or more specifically JPEG images due to their wide popularity. Thus to an attacker the fact that an image other that that of JPEG format is being transferred between two entities could hint of suspicious activity.

There have been a number of image steganography algorithm proposed, these algorithm could be categorized in a number of ways:

- Spatial or Transform, depending on redundancies used from either domain for the embedding process.

- Model based or ad-hoc, if the algorithm models statistical properties before embedding and preserves them, or otherwise.
- Active or Passive Warden, based on whether the design of embedder-detector pair takes into account the presence of an active attacker.

In what follows we go over algorithm classified into 3 different sections, based on the more important characteristics of each embedding technique. Although some of the techniques which we will discuss below have been successfully broken by steganalysis attacks, which we will go over in Section 6.

5.1. *Spatial Domain Embedding*

The best widely known steganography algorithm is based on modifying the least significant bit layer of images, hence known as the *LSB technique*. This technique makes use of the fact that the least significant bits in an image could be thought of random noise and changes to them would not have any effect on the image. This is evident by looking at Fig. 4. Although the image seems unchanged visually after the LSBs are modified, the statistical properties of the image changes significantly. We will discuss in the next section of this tutorial how these statistical changes could be used to detect stego images created using the LSB method.

In the LSB technique, the LSB of the pixels is replaced by the message to be sent. The message bits are permuted before embedding, this has the effect of distributing the bits evenly, thus on average only half of the LSB's will be modified. Popular steganographic tools based on LSB embedding [26,27,28], 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 [28].

Fridrich et al. [29] proposed another approach for embedding in spatial domain. In their method, noise that statistically resemble common processing distortion, e.g., scanner noise, or digital camera noise, is introduced to pixels on a random walk. The noise is produced by a pseudo random noise generator using a shared key. A *parity function* is designed to embed and detect the message message signal modulated by the generated noise.

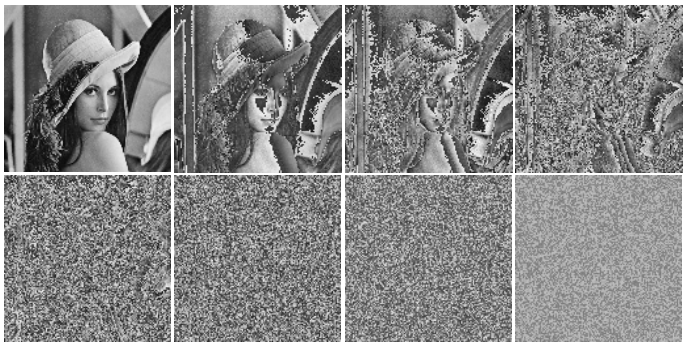

Fig. 4. Bitplane decomposition of image Lena.

5.2. *Transform Domain Embedding*

Another category for embedding techniques for which a number of algorithms have been proposed is the transform domain embedding category. Most of the work in this category has been concentrated on making use of redundancies in the DCT (discrete cosine transform) domain, which is used in JPEG compression. But there has been other algorithms which make use of other transform domains such as the frequency domain [30].

Embedding in DCT domain is simply done by altering the DCT coefficients, for example by changing the least significant bit of each coefficient. One of the constraints of embedding in DCT domain is that many of the 64 coefficients are equal to zero, and changing two many zeros to non-zeros values will have an effect on the compression rate. That is why the number of bit one could embed in DCT domain, is less that the number of bits one could embed by the LSB method. Also the embedding capacity becomes dependent on the image type used in the case of DCT embedding, since depending on the texture of image the number of non-zero DCT coefficients will vary.

Although changing the DCT coefficients will cause unnoticeable visual artifices, they do cause detectable statistical changes. In the next section,

we will discuss techniques that exploit these statistical anomalies for steganalysis. In order to minimize statistical artifacts left after the embedding process, different methods for altering the DCT coefficients have been proposed, we will discuss two of the more interesting of these methods, namely the F5 [31] and Outguess [32] algorithms.

F5 [31] embedding algorithm was proposed by Westfeld as the latest in a series of algorithms, which embed messages by modifying the DCT coefficients. For a review of jsteg, F3 and F4 algorithms that F5 is built on, please refer to [31]. F5 has two important features, first it permutes the DCT coefficients before embedding, and second it employs matrix embedding.

The first operation, namely permuting the DCT coefficients has the effect of spreading the changed coefficients evenly over the entire image. The importance of this operation becomes evident when a small message is used. Let's say we are embedding a message of size m, then if no permutation is done and coefficients are selected in the order they appear, then only the first m coefficients are used. Thus the first part of the image get's fully changed after embedding, and the rest of the image remains unchanged. This could facilitate attacks on the algorithm since the amount of change is not uniform over the entire image. On the other hand when permutation is done, the message is spread uniformly over the image thus the distortion effects of embedding is spread equally and uniformly over the entire image.

The second operation done by F5 is matrix embedding. The goal of matrix embedding is to minimize the amount of change made to the DCT coefficients. Westfeld [31], takes n DCT coefficients and hashes them to k bits. If the hash value equals to the message bits then the next n coefficients are chosen and so on. Otherwise one of the n coefficients is modified and the hash is recalculated. The modifications are constrained by the fact that the resulting n DCT coefficients should not have a hamming distance of more than d_{max} from the original n DCT coefficients. This process is repeated until the hash value matches the message bits. So then given an image, the optimal values for k and n could be selected.

Outguess [32], which was proposed by Provos, is another embedding algorithm which embeds messages in the DCT domain. Outguess goes about the embedding process in two separate steps. First it identifies the redundant DCT coefficients which have minimal effect on the cover image, and then depending on the information obtained in the first steps, chooses bits in which it would embed the message. We should note that at the time Outguess was proposed, one of its goals was to overcome steganalysis attacks which look at changes in the DCT histograms after embedding. So Provos,

proposed a solution in which some of the DCT coefficients are left unchanged in the embedding process, afterwards these remaining coefficients are adjusted in order preserve the original histogram of DCT coefficients. As we will see in the steganalysis section both F5 [31], and Outguess [32] embedding techniques have been successfully attacked.

As mentioned before, another transform domain which has been used for embedding is the frequency domain. Alturki et al. [30] propose quantizing the coefficients in the frequency domain in order to embed messages. They first decorrelate the image by scrambling the pixels randomly, which in effect whitens the frequency domain of the image and increases the number of transform coefficients in the frequency domain thus increasing the embedding capacity. As evident from Fig. 5, the result is a salt and pepper image where it's probability distribution function resembles a gaussian distribution. The frequency coefficients are then quantized to even or odd multiples of the quantization step size to embed zeros or ones. Then the inverse FFT of the signal is taken and descrambled. The resulting image would be visually incomparable to the original image. But statistically the image changes and as the authors show in their work, the result of the embedding operation is the addition of a gaussian noise to the image.

5.3. *Model Based Techniques*

Unlike techniques discussed in the two previous subsections, model based techniques try to model statistical properties of an image, and preserve them in the embedding process. For example Sallee [14] proposes a method which breaks down transformed image coefficients into two parts, and replaces the perceptually insignificant component with the coded message signal. Initially, the marginal statistics of quantized (non-zero) AC DCT coefficients are modelled with a parametric density function. For this, a low precision histogram of each frequency channel is obtained, and the model is fit to each histogram by determining the corresponding model parameters. Sallee defines the offset value of coefficient within a histogram bin as a *symbol* and computes the corresponding *symbol probabilities* from the relative frequencies of symbols (offset values of coefficients in all histogram bins).

In the heart of the embedding operation is a non-adaptive arithmetic decoder which takes as input the message signal and decodes it with respect to measured symbol probabilities. Then, the entropy decoded message is embedded by specifying new bin offsets for each coefficient. In other words, the coefficients in each histogram bin are modified with respect to

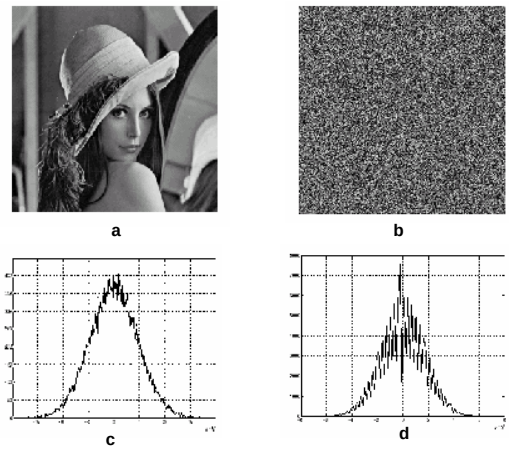

Fig. 5. Frequency domain embedding. a) Original image, b) scrambled image, c) histogram of DFT coefficients, and d) histogram of DFT coefficients after quantization. (Figure taken from [30])

embedding rule, while the global histogram and symbol probabilities are preserved. Extraction, on the other hand, is similar to embedding. That is, model parameters are determined to measure symbol probabilities and to obtain the embedded symbol sequence (decoded message). (It should be noted that the obtained model parameters and the symbol probabilities are the same both at the embedder and detector). The embedded message is extracted by entropy encoding the symbol sequence.

Another model based technique was proposed by Radhakrishnan et al. [33], in which the message signal is processed so that it would exhibit the properties of an arbitrary cover signal, they call this approach data masking. As argued if Alice wants to send an encrypted message to Bob, the warden Wendy would be able to detect such a message as an encrypted stream since it would exhibit properties of randomness. In order for a secure channel to achieve covertness, it is necessary to preprocess the encrypted stream at the end points to remove randomness such that the resulting stream defeats statistical tests for randomness and the stream is reversible at the other end.

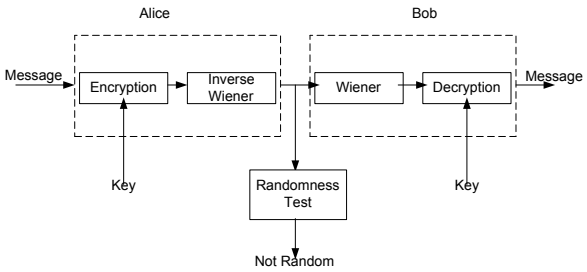

Fig. 6. Proposed System for Secure and Covert Communication.(Figure taken from [33])

The authors propose Inverse Wiener filtering as a solution to remove randomness from cipher streams as shown in Fig 6. Let us consider the cipher stream as samples from a wide sense stationary (WSS) Process, E. We would like to transform this input process with high degree of randomness to another stationary process, A, with more correlation between samples by using a linear filter, H. It is well known that the power spectrum of a WSS input, $A(w)$, to a linear time invariant system will have the output with the power spectrum $E(w)$ expressed as

$$E(w) = |H(w)|^2 A(w). \tag{13}$$

If $E(w)$ is a white noise process, then $H(w)$ is the whitening filter or Wiener filter. Since the encrypted stream is random, its power spectral density is flat and resembles the power spectral density of a white noise process. Then, the desired Wiener filter can be obtained by spectral factorization of $(E(w)/A(w))$ followed by selection of poles and zeros to obtain the minimum phase solution for $H(w)$. The authors discuss how the above method could be used with audio as cover-object in [33], and more recently with images as cover-object in [34].

6. 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 al-

gorithms. 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, and 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.

6.1. *Technique Specific Steganalysis*

We first look at steganalysis techniques that are designed with a particular steganographic embedding algorithm in mind. As opposed to the previous section, were the embedding algorithms were categorized depending on the approach taken in the embedding process, here we categorize the steganographic algorithms depending on the type of image they operate on, which includes Raw images (for example bmp format), Palette based images (for example GIF images), and finally JPEG images.

6.1.1. *Raw 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. An early approach to LSB steganalysis was presented in [11] by Westfeld et al. 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. [12]. 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 [12].

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

$$|X| = |Y|. \tag{14}$$

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 Fig. 7.

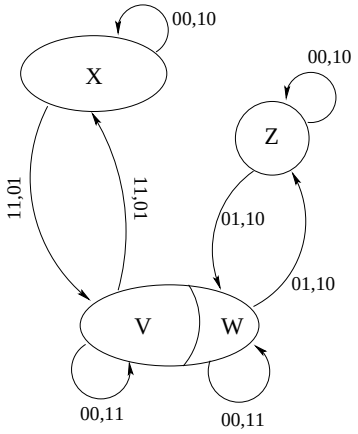

Fig. 7. State transition diagram for sets X, V, W, Z under LSB flipping.(Figure taken from [12])

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

$$0.5\gamma p^2 + (2|X'| - |\mathcal{P}|)p + |Y'| - |X'| = 0. \tag{15}$$

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.

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 [35] 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 et al. [36]. 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, Hide [28] 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 [37] proposes a similar steganalysis attack as Fridrich et al. [36] 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 8.

6.1.2. *Palette Based Images*

Pallete based images, like GIF images, are another popular class of images for which there have been a number of steganography methods proposed [38,39,40]. Perhaps some of the earliest steganalysis work in this regard was reported by Johnson et al. [41]. 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

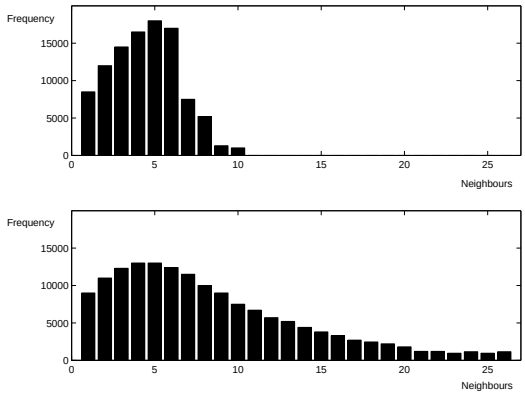

Fig. 8. Neighborhood histogram of a cover image (top) and stego image with 40 KB message embedded (bottom). (Figure taken from [37])

and cover-images.

In order to minimize the distortion caused by embedding, EzStego [38] 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 [11], but the technique discussed above provides a much higher detection rate and a more accurate estimate of the message lengths.

6.1.3. *JPEG Images*

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 [32,42,31] which employ this category of images. Fridrich [6] has proposed attacks on the F5 and Outguess algorithms, both of which were covered in the previous section. F5 [31] 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, at the same time it spreads the message over the cover image. 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 recompressing the image using the same quantization table as used before. As is evident in Fig 9, the resulting DCT coefficient histogram would be a very good estimate of the original histogram.

Intuitively, effect of the cropping operation could be reasoned as follows. In a natural image, characteristics are expected to change smoothly with respect to spatial coordinates. That is, image features computed in a portion of image will not change significantly by a slight shift in the computation window. In the same manner, the statistics of the DCT coefficients computed from a shifted partitioning of an image should remain roughly unchanged. However, since in F5, DCT coefficients are tailored by the embedder, cropping of the image (shift in the partitioning) will spoil the the structure created by embedding process, thereby, the coefficient statistics will vary and estimate the original structure.

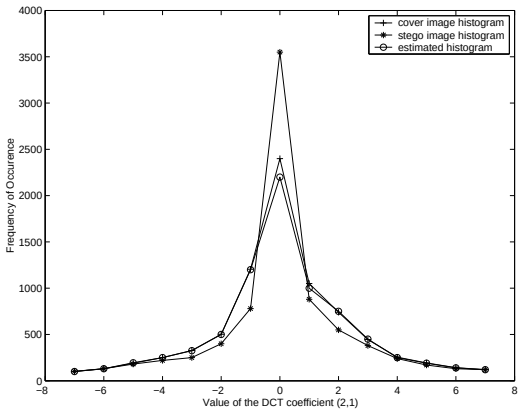

Fig. 9. The effect of F5 embedding on the histogram of the DCT coefficient (2,1). (Figure taken from [6])

A second technique proposed by Fridrich [6] deals with the Outguess [32] 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 LSBs 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. In a related work Wang et al. [43] use a statistical approach and show how embedding in DCT domain effects differently the distribution of neighboring pixels which are inside blocks or across blocks. These differences could be used to distinguish between clean and stego images.

6.2. *Universal Steganalysis*

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. [44,45,46] and Farid et al. [47,48], are designed to work with any steganographic embedding algorithm, even an unknown algorithm. Such techniques have subsequently been called *Universal Steganalysis* or *Blind Steganalysis Techniques*. Such approaches essentially design a classifier based on a training set of cover-objects and stego-objects obtained from a variety of different embedding 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, monotonic, and consistent in capturing statistical signatures left by the embedding process. Detection accuracy can be interpreted as the ability of the measure to detect the presence of a hidden message with minimum error on average. Similarly, detection monotonicity signifies that the features should ideally be mono-

tonic in their relationship to the embedded message size. Finally, detection consistency relates to the feature's ability to provide consistently accurate detection 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 [46] 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 [46] 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.

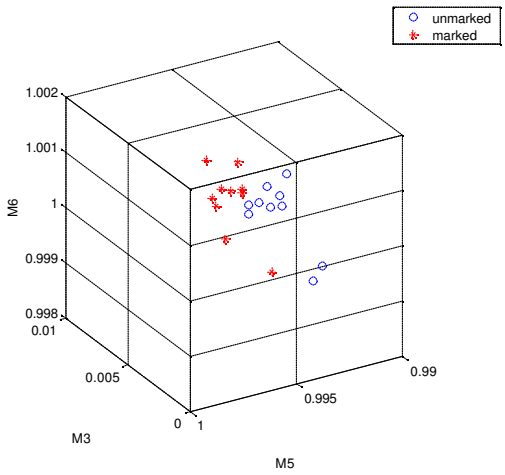

Fig. 10. Scatter plot of 3 image quality measures showing separation of marked and unmarked images. (Figure takenh from [46])

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 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. 10 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. [44] 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 a cover image or an image containing hidden messages.

A different approach is taken by Farid et. al [47,48] 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 measures could keep the first order statistics intact thus making the steganalysis 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, skewness, and kurtosis are calculated for each subband. Additionally the same statistics are calculated for the error obtained from an optimal linear predictor of coefficient magnitudes of each subband, as the second part of the feature set.

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. use a MMSE Linear predictor, where as Farid et al. [47,48] uses a Fisher linear discriminant [49] and also a Support Vector Machine (SVM) [50] 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 vary.

7. Conclusion

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. In this tutorial we have reviewed some fundamental notions related to steganography and steganalysis.

Although we covered a number of security and capacity definitions, there has been no work successfully formulating the relationship between the two from the practical point of view. For example it is understood that as less information is embedded in a cover-object the more secure the system will be. But due to difficulties in statistical modelling of image features, the security versus capacity trade-off has not been theoretically explored and quantified within an analytical framework.

We also reviewed a number of embedding algorithms starting with the earliest algorithm proposed which was the LSB technique. At some point LSB seemed to be unbreakable but as natural images were better understood and newer models were created LSB gave way to new and more powerful algorithms which try to minimize changes to image statistics. But with further improvement in understanding of the statistical regularities and redundancies of natural images, most of these algorithms have also been successfully steganalysed.

In term of steganalysis, as discussed earlier, there are two approaches, technique specific or universal steganalysis. Although finding attacks specific to an embedding method are helpful in coming up with better embedding methods, their practical usage seems to be limited. Since given an image we may not know the embedding technique being used, or even we might be unfamiliar with the embedding technique. Thus universal steganalysis techniques seem to be the real solution since they should be able to detect stego images even when a new embedding technique is being employed.

References

1. G. Simmons, "The prisoners problem and the subliminal channel," *CRYPTO*, pp. 51–67, 1983.
2. 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.
3. N. J. Hopper, J. Langford, and L. von Ahn, "Provably secure steganography," *Advances in Cryptology: CRYPTO 2002, August*, 2002.
4. L. Reyzin and S. Russell, "More efficient provably secure steganography," 2003.
5. C. Cachin, "An information-theoretic model for steganography," *2nd International Workshop Information Hiding*, vol. LNCS 1525, pp. 306–318, 1998.
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. Y. Wang and P. Moulin, "Steganalysis of block-structured text," *Proceedings of SPIE*, 2004.
8. R. Chandramouli and N. Memon, "Steganography capacity: A steganalysis perspective," *SPIE Security and Watermarking of Multimedia Contents V*, vol. 5020, 2003.
9. S. Katzenbeisser and F. A. P. Petitcolas, "Defining security in steganographic systems," *Proceedings of the SPIE vol. 4675, Security and Watermarking of Multimedia Contents IV.*, pp. 50–56, 2002.
10. P. Moulin and Y. Wang, "New results on steganography," *Proc. of CISS*, 2004.
11. A. Westfeld and A. Pfitzmann, "Attacks on steganographic systems," *3rd International Workshop on Information Hiding.*, 1999.
12. 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.
13. R. Chandramouli and N. Memon, "Analysis of lsb image steganography techniques," *IEEE International Conference on Image Processing*, vol. 3, pp. 1019–1022, 2001.
14. P. Sallee, "Model-based steganography," *International Workshop on Digital Watermarking, Seoul, Korea.*, 2003.
15. E. P. Simoncelli, "Modeling the joint statistics of images in the wavelet domain," *Proceedings of the 44th Annual Meeting*, 1999.
16. R. J. Barron, B. Chen, and G. W. Wornell, "The duality between information embedding and source coding with side information and its implications - applications," *IEEE Transactions on Information Theory.*
17. A. Cohen and A. Lapidoth, "On the gaussian watermarking game," *International Symposium on Information Theory*, June 2000.
18. P. Moulin and M. Mihcak, "A framework for evaluating the data-hiding capacity of image sources," *IEEE Transactions on Image Processing*, vol. 11, no. 9, pp. 1029–1042.

19. R. Zamir, S. Shamai, and U. Erez, "Nested lattice/linear for structured multiterminal binning," *IEEE Transactions on Information Theory*, 2002.
20. J. Chou, S. S. Pradhan, L. E. Ghaoui, and K. Ramchandran, "A robust optimization solution to the data hiding problem using distributed source coding principles," *Proceedings SPIE: Image and Video Communications and Processing*, vol. 3974, 2000.
21. R. Chandramouli, "Data hiding capacity in the presence of an imperfectly known channel," *SPIE Proceedings of Security and Watermarking of Multimedia Contents II*, vol. 4314, pp. 517–522, 2001.
22. P. Moulin and J. Sullivan, "Information theoretic analysis of information hiding," *To appear in IEEE Transactions on Information Theory*, 2003.
23. M. Ramkumar and A. Akansu, "Information theoretic bounds for data hiding in compressed images," *IEEE 2nd Workshop on Multimedia Signal Processing*, pp. 267–272, Dec. 1998.
24. ——, "Theoretical capacity measures for data hiding in compressed images," *SPIE Multimedia Systems and Application*, vol. 3528, pp. 482–492, 1998.
25. R. Chandramouli, "Watermarking capacity in the presence of multiple watermarks and partially known channel," *SPIE Multimedia Systems and Applications IV*, vol. 4518, pp. 210–215, Aug. 2001.
26. F. Collin, "Encryptpic," *http://www.winsite.com/bin/Info?500000033023.*
27. G. Pulcini, "Stegotif," *http://www.geocities.com/SiliconValley/9210/gfree.html.*
28. T. Sharp, "Hide 2.1, 2001," *http://www.sharpthoughts.org.*
29. J. Fridrich and M. Goljan, "Digital image steganography using stochastic modulation," *SPIE Symposium on Electronic Imaging, San Jose, CA,*, 2003.
30. F. Alturki and R. Mersereau, "Secure blind image steganographic technique using discrete fourier transformation," *IEEE International Conference on Image Processing, Thessaloniki, Greece.*, 2001.
31. A. Westfeld, "F5a steganographic algorithm: High capacity despite better steganalysis," *4th International Workshop on Information Hiding.*, 2001.
32. N. Provos, "Defending against statistical steganalysis," *10th USENIX Security Symposium*, 2001.
33. R. Radhakrishnan, K. Shanmugasundaram, and N. Memon, "Data masking: A secure-covert channel paradigm," *IEEE Multimedia Signal Processing, St. Thomas, US Virgin Islands*, 2002.
34. R. Radhakrishnan, M. Kharrazi, and N. Memon, "Data masking: A new approach for steganography?" *To appear in the Journal of VLSI Signal Processing-Systems for Signal, Image, and Video Technology.*
35. 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.
36. J. Fridrich, R. Du, and L. Meng, "Steganalysis of lsb encoding in color images," *ICME 2000, New York, NY, USA.*
37. A. Westfeld, "Detecting low embedding rates," *5th International Workshop on Information Hiding.*, pp. 324–339, 2002.
38. R. Machado, "Ezstego," *http://www.stego.com*, 2001.

39. M. Kwan, "Gifshuffle," *http://www.darkside.com.au/gifshuffle/*.
40. C. Moroney, "Hide and seek," *http://www.rugeley.demon.co.uk/security/hdsk50.zip*.
41. 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.
42. D. Upham, "Jpeg-jsteg," *ftp://ftp.funet.fi/pub/crypt/steganography/jpeg-jsteg-v4.diff.gz*.
43. Y. Wang and P. Moulin, "Steganalysis of block-dct image steganography," *IEEE Workshop On Statistical Signal Processing*, 2003.
44. I. Avcibas, N. Memon, and B. sankur, "Steganalysis using image quality metrics." *Security and Watermarking of Multimedia Contents, San Jose, Ca.*, Feruary 2001.
45. ——, "Image steganalysis with binary similarity measures." *IEEE International Conference on Image Processing, Rochester, New York.*, September 2002.
46. ——, "Steganalysis using image quality metrics." *IEEE transactions on Image Processing*, January 2003.
47. S. Lyu and H. Farid, "Detecting hidden messages using higher-order statistics and support vector machines," *5th International Workshop on Information Hiding.*, 2002.
48. ——, "Steganalysis using color wavelet statistics and one-class support vector machines," *SPIE Symposium on Electronic Imaging, San Jose, CA,*, 2004.
49. R. Duda and P. Hart, "Pattern classification and scene analysis," *John Wiley and Sons.*, 1973.
50. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery.*, pp. 2:121–167, 1998.