A Survey on Steganography and Steganalysis Techniques in Secret Communication

Md Amiruzzaman*

West Chester University, West Chester, PA 19383, USA
mamiruzzaman@wcupa.edu

Received: June 13, 2022; Accepted: August 21, 2022; Published: August 31, 2022

Abstract

This paper presents a survey on classical steganographic algorithms and detection techniques (i.e., steganalysis techniques) along with their mathematical definitions. A little to no studies found that tried to provide mathematical definitions of these algorithms. Wheras, mathematical definitions and examples of different existing algorithms could help researchers to develop better understand how previous algorithms and their limitations. In-depth understaning is important to improve existing and develop new algorithms. Also, this paper presents a comparison chart of existing algorithms to show advantages and disadvantages of each steganographic algorithm in terms of steganalysis or detection techniques.

Keywords: steganography, steganalysis, secret communication, cybersecurity, information hiding

1 Introduction

Steganography and Steganalysis both are equally important concepts in information hiding or secret communication and cybersecurity [1, 2]. Hence, steganography and steganalysis are advancing same time [2, 3]. According to Amiruzzaman [2], whenever a steganographic method has been proposed, the method is about to be broken soon by new steganalysis methods. Therefore, researchers those are working in the filed of steganography is trying to develop new methods fully or partially secure from the existing steganalysis techniques [4]. However, it is not possible all the time to be able to take all the security issues into account and solve them in one method or algorithm [2]. Steganography is known to be one of the oldest arts or techniques for information hiding to establish a secure covert communication channel [5]. However, digital steganography techniques is not easy to understand and implement [2], [6], [7]. It has become one of the interesting topics of interests in cybersecurity research.

The most important goal of digital steganography is to conceal the existence of a secret message from attackers [1, 8, 9]. Therefore, it is important for researchers know strength and weakness of their steganographic schemes [2, 3]. There are several steganalysis or detection (or attacks) techniques available in the literature [10]. Among them statistical detection (or attack) [7] is one of the most popular and effective attacks in steganographic world [11]- [12]. So, the objective of any steganographic scheme is make sure that the scheme is secure against known steganalysis–at least secure against most steganalysis. In recent years, digital image became popular platform for secure communication or steganographic techniques [1], [2], [10], [7]. One of the simple solutions for digital image steganography against first

Research Briefs on Informaiton & Communication Technology Evolution (ReBICTE), Vol. 8, Article No. 7 (August 31, 2022)
DOI:*10.22667/ReBiCTE.2022.08.31.007*

*Corresponding author: Tel: +1-610-436-3230, Web: https://amiruzzaman.github.io/

order steganalyis (i.e., attack) is keeping the same or similar histogram of the data close enough to the original histogram [7]. However, keeping the same shape of a magnitude histogram is not easy to achieve as long as the coefficient magnitudes are modified [2].

One branch of steganography methods is continuously inventing schemes to preserve the original histogram perfectly [4, 13, 14]. For example, Least Significant Bit (LSB) modifying methods, e.g., OutGuess [6] method can preserve the original histogram almost perfectly, but not completely intact. The Outguess method modifies half of the nonzero coefficients and corrects the distorted histogram by adjusting with the rest of unused coefficients [2]. In general, perfect preservation is not possible because of unideal data patterns [15].

F5 Steganography technique [7] tries to narrow the gap between the original and the modified histograms by decrementing nonzero Joint Photographic Experts Group (JPEG) coefficients towards 0 and applying matrix embedding and permutative straddling. JPEG is a commonly used method of lossy compression for digital images. In a separate study, Sallee [16] modeled the marginal distribution of Discrete Cosine Transform (DCT) coefficients, in JPEG-compressed images, by the generalized Cauchy distribution. Thus, the embedded message is adapted to the generalized Cauchy distribution using arithmetic coding. Arithmetic coding transforms unevenly distributed bit streams into shorter and uniform ones. This procedure is known as Model-Based or MB1 [2]. One weak point of this method is that the block artifact increases with growing size of the payload. Model-Based2 or MB2 has presented a method to overcome this weakness [17]. The MB2 embeds a message in the same way as MB1 does, but its embedding capacity is only half of that of MB1. The other half of the nonzero DCT coefficients is reserved for de-blocking purpose.

Figure 1 illustrates a typical histogram of a JPEG image.

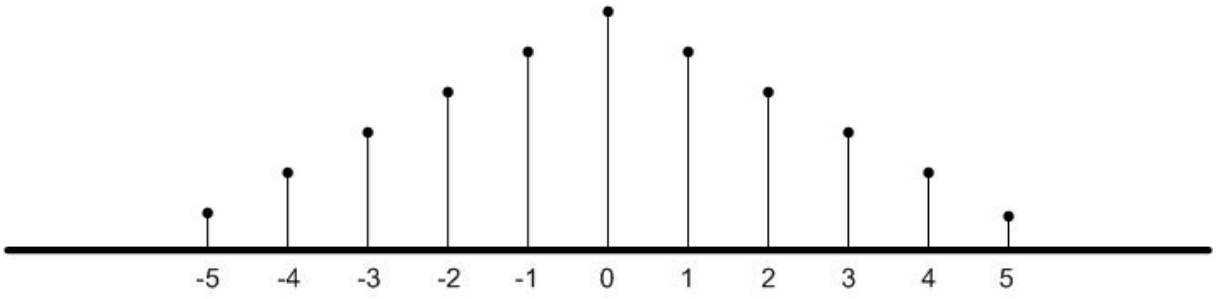

Figure 1: A Typical Histogram of a JPEG Image.

Preserving the perfect shape of the histogram of a stego image has been a primary target in the field of steganography [2, 8, 18]. In this paper a set of classical steganographic algorithms are presented to show how different algorithms tried to secure their scheme against histogram attack or first order statistical attack.

2 Important Terms

2.1 Covert Channel

In the literature, often the words “covert channel” and “steganography” is used interchangeably [2, 5]. So, it is important to define covert channel. A covert channel is a mean of communication (see Fig. 2),

where both the sender and the receiver collude to leak information.

In general, the channel itself is not intended to be used for. So, during the encoding process, the encoder function replaces the LSBs and substitutes with the secret message bits. The selection of LSBs may depend on the steganographic algorithm. The embedding operation of LSB-based steganography can be defined as, communication purposes. Therefore, the covert communication may be in violation of a mandatory access control security policy.

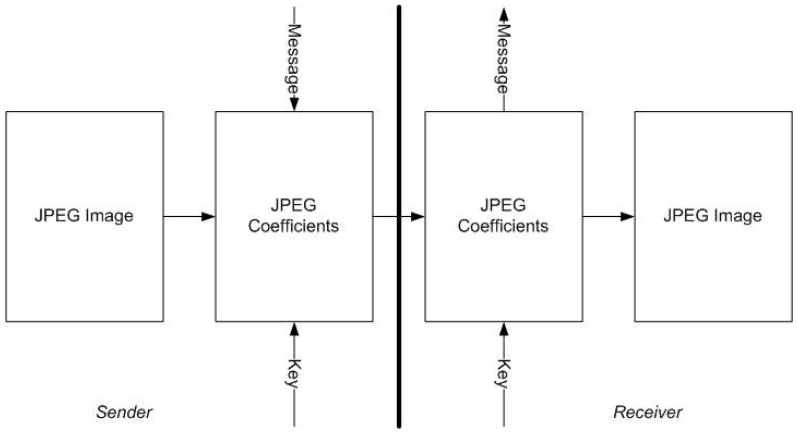

Figure 2: Working Procedure of JPEG Steganographic Algorithm.

2.2 Steganographic Function

Let $C_1, C_2, \cdots, C_n$ be a set data transmission units, and $B_1, B_2, \cdots, B_k$ be the set of covert data units to be transmitted ($k \leq n$). The steganographic function $\psi(.)$ embeds B_j into C_i and produces C_i' in such away that C_i' retains the basic characteristics of C_i. The receiver, on the other hand extract B_j from C_i' by applying a reverse function $\phi(.)$, i.e.,

$$C_j' = \psi(C_j, B_i) \qquad and \qquad B_i = \phi(C_j') \tag{1}$$

Note that functions $\phi(.)$ and $\psi(.)$ are not necessarily inverse of each other mathematically, rather $\psi(.)$ is the inverse transformation of $\psi(.)$.

2.3 Least Significant Bit

Least Significant Bit (LSB) is often used in covert channel to hide information [2]. In, LSB based image steganography, a researcher alters the LSB to embed the hidden message bit.

So, during the encoding process, the encoder function replaces the LSBs and substitutes with the secret message bits. The selection of LSBs may depend on the steganographic algorithm. The embedding operation of LSB-based steganography can be defined as,

$$C_i' = 2\lfloor C_i \rfloor + B_i \tag{2}$$

where, C_i is the coefficient, B_i is secret message bit, and C_i' is the modified coefficient after hiding the message bit B_i.

2.4 Hiding Rate

The hiding rate of steganography is a ratio between the number of secret bits that can be embedded over the total number of coefficients (also known as capacity),

$$\text{Hiding rate} = \frac{\text{Number of secret bits}}{\text{Capacity of encoding}} \tag{3}$$

2.5 Encoding

In this paper encoding referred to technique to hide data in the digital image. It is a function (e.g., $\psi(C,B)$) that modifies digital image coefficients C to embed or hide B.

2.6 Decoding

In this paper decoding referred to technique to extract data from digital image. It is a function (e.g., $\psi^{-1}(C')$) that extarcts B from modified digital image coefficients C'.

2.7 Image quality

2.7.1 Peak-Signal-to-Noise Ratio

Image quality is measured by Peak-Signal-to-Noise Ratio (PSNR), it is a good statistical tool to check quality of original and modified image [8,19]. If f is a reference or original image and f' is the modified or stego image (i.e., image modified to hide information in it), then PSNR can be computed by comparing two images (i.e., f and f') as,

$$PSNR(f,f') = 10\log_{10}\left(\frac{255^2}{MSE(f,f')}\right) \tag{4}$$

where, Mean Squared Error (MSE) is a function that helps to calculate deviation between two images (i.e., f and f'). The MSE can be computed as,

$$MSE(f,f') = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\left(f_{ij} - f'_{ij}\right) \tag{5}$$

where, M and N is the height and width of a image. The equation assumes that f and f' have same height and width.

2.7.2 Structural Similarity Index Measure

Another well-known image deviation measure tool is called Structural Similarity Index Measure (SSIM). This statistical tool measures the distortion between the reference f or original image and modified or stego image f'. The SSIM can be calculated as [20],

$$SSIM(f,f') = l(f,f')c(f,f')s(f,f') \tag{6}$$

where, functions $l(.)$, $c(.)$, and $s(.)$ can be defined as,

$$\begin{aligned} l(f,f') &= \frac{2\mu_f\mu_{f'}+C_1}{\mu_f^2+\mu_{f'}^2+C_1} \\ c(f,f') &= \frac{2\sigma_f\sigma_{f'}+C_2}{\sigma_f^2+\sigma_{f'}^2+C_2} \\ c(f,f') &= \frac{\sigma_{ff'}+C_3}{\sigma_f\sigma_{f'}+C_3} \end{aligned} \tag{7}$$

where, C_1, C_2, and C_3 are constant, μ is mean luminance, and σ is standard deviation or contrast. Note that, if $\mu_f = \mu'_f$, then mean luminance is maximal (i.e., equal to 1).

3 JPEG Steganalysis

Coefficients of JPEG image can be used to create histogram. The histogram can be used to analyze the quality of the compressed image, such as, brightness, contrast, etc. Note that, two different images may have same or similar histogram. The JPEG image coefficient histogram can be defined as,

$$H(i) = \sum_{j=0}^{i} h(j), 0 \leq i < n, \tag{8}$$

where, $n = 1, 2, 3, \cdots$ or total number of coefficients.

JPEG image coefficient histogram is bell shaped and symmetrical in both sides [21].

During the compression JPEG image creates 8×8 coefficients blocks. JPEG coefficients are also known as Discrete Cosine Transform (DCT) coefficients. DCT coefficients are divided into two groups: DC coefficient and AC coefficient. DC coefficient is the coefficient with zero frequency in both dimensions, and AC coefficients are remaining 63 coefficients with non-zero frequencies.

Neighboring blocks, share properties of an image, such as, edge, hue, etc. Therefore, the difference between neighboring blocks are not so significant. Similarly the difference between neighboring pixels are not huge either. Based on JPEG image properties, there are a few steganalysis are often used to detect covert communication.

Following paragraphs describe a few popular image steganalysis:

3.1 Histogram Attack

Let, $P(.)$ is the probability function and $C_i \in \mathbb{Z}$ is the coefficients, then histogram attack can be defined as,

$$\begin{aligned} P(C=1) > P(C=2) > P(C=3) > P(C=4) \\ P(C=1) - P(C=2) > P(C=2) - P(C=3) \\ P(C=2) - P(C=3) > P(C=3) - P(C=4) \end{aligned} \tag{9}$$

If an image does not follow the histogram properties, then that situation indicates that the image is modified or indicates the presence of secret information.

3.2 Blocking attack

The difference between two neighboring blocks should be close to zero. So, the blocking attack can be defined as,

$$block_i - block_{i+1} \approx 0 \tag{10}$$

If two neighbouring blocks are significantly different from each other, then that means the image is modified to hide secret information.

3.3 Chi-Square Attack

The χ^2 test is often used to detect stochastically independent shades [22, 23]. After replacing LSBs, the corresponding gray values $2i$ and $2i+1$ tend to e equal, so, for the n_j number of pixel with gray value j can be tested using χ^2 test as,

$$\chi^2 = \sum_{i=1}^{k} \frac{[n_{2i} - (n_{2i} + n_{2i+1})/2]^2}{(n_{2i} + n_{2i+1})/2} \tag{11}$$

and

$$p = 1 - \frac{1}{2^{(k-1)/2}\tau[(k-1)/2]} \int_0^{\chi^2} \exp(-\frac{t}{2}) t^{\frac{k-1}{2}-1} dt \tag{12}$$

where, p is the probability that the distribution of $n2i$ and $n2i+1$ are equal. If the probability p indicates that both $n2i$ and $n2i+1$ are equal, and experimental image shows differently, then the experimental image is altered image.

4 JPEG Steganography

Ase mentioned earlier, among all covert channels, the JPEG is the most popular medium for steganography. There are different techniques available for steganography [1]. However, only a few algorithms work with JPEG compressed images.

JPEG-based steganography algorithms work only and only with the DCT coefficients, that are integer values in the range of $[-2^{p-1} \cdots 2^{p-1}]$ for a p-bit encoding. Few other compression techniques work with DCT quantization values, considering the rounding error [1]. JPEG methods do not adjust the quantization errors. Therefore, there is no easy way to control the modification distortion.

5 Existing Steganographic Algorithms

5.1 JSteg Algorithm

The JSteg algorithm was invented by Derek Upham [24]. It provides the basic idea of information hiding in the coefficients of a JPEG image [25]. The JSteg algorithm hides data in the positive and negative values of the AC coefficients of a JPEG image, except the coefficients with values 0 and 1 (see Eq. 13). Let $C = [C_1 C_2 \cdots C_n]$ be a vector representing the coefficient of a JPEG image in the carrier medium, $C = [C_1' C_2' \cdots C_n']$ be a vector representing the coefficient in steganogram representation, and $B = [B_1 B_2 \cdots B_k]$ be a binary vector representing the steganographic values, where,

$$C_j' = \psi(C_j, B_i), \quad B_i = \phi(C_j'), \quad B_i \in \{0,1\}, \quad 1 \le i \le k,\ 1 \le j \le n \tag{13}$$

such that ϕ and ψ are encoding and decoding functions of JSteg algorithm, respectively.

5.1.1 JSteg Encoding

The encoding function of JSteg can be formulated as,

$$\underset{1\le i\le k,\ 1\le j\le n,\ i\le j}{C'_j = \psi(C_j, B_i)} = \begin{cases} C_j + [C_j + B_i]_2 & \text{if } B_i = 1, \quad 0 > C_j > 1 \\ C_j - [C_j + B_i]_2 & \text{if } B_i = 0, \quad 0 > C_j > 1 \\ \text{Skip} & \text{Otherwise} \end{cases} \tag{14}$$

where $[x]_2 = (x \mod 2)$.

5.1.2 JSteg Decoding

For decoding, after receiving C'_i, the receiver decodes B_i by simply modulo(2) operation as shown in Equation (15).

$$\underset{1\le i\le k,\ 1\le j\le n,\ i\le j}{B_i = \phi(C'_j)} = \begin{cases} 1 & \text{if } [C'_j]_2 = 1, \\ 0 & \text{Otherwise} \end{cases} \tag{15}$$

Example 1 (JSteg Encoding/decoding)**.** Consider *B*=[1 0 1 0 0 1] and *C*=[-4 -3 -2 -1 0 1 2 3], then *C′*=[-3 -4 -1 -2 0 1 2 3]. Decoding
$C' \equiv B \mod 2$, so, obtained *B*=[1 0 1 0 0 1], as the encoding and decoding both ignores coefficients 0 and 1.

5.1.3 Limitation of JSteg

The Jsteg algorithm poses two significant limitations. First, the Jsteg algorithm skips the coefficients with value 0 and 1. As a result, the upon skipping 0 and 1 coefficients, the hiding capacity of the algorithm decreases significantly. Second, the JSteg algorithm can not preserve the shape of JPEG coefficients histogram. Therfore, it can be detected by the first order statistics and χ^2 test.

5.2 F3 Algorithm

The F3 algorithm [7] is considered as an improvement to the JSteg algorithm. F3 uses the idea of shrinkage. During the shrinkage process, F3 does not embed (hide) any message bits. Fewer coefficients are modified without hiding any secret message bits.

As in Jsteg, F3 skips coefficients with value 0. It alters ± 1 to 0, while the hidden bit is 0. i.e, either 1 or -1 will be modified to 0. The decoder skips 0's during the decoding process.

Let as before, $C = [C_1 C_2 \cdots C_n]$ be a vector representing the coefficient of a JPEG image in the carrier medium, $C = [C'_1 C'_2 \cdots C'_n]$ be a vector representing the coefficient in steganogram representation, and $B = [B_1 B_2 \cdots B_k]$ be a binary vector representing the steganographic values (see Eq. 16).

5.2.1 F3 Encoding

The encoding algorithm of F3 can be formulated as,

$$\underset{1\le i\le k,\ 1\le j\le n,\ i\le j}{C'_j = \psi(C_j, B_i)} = \begin{cases} C_j - [C_j + B_i]_2 & \text{if } C_j > 0 \\ C_j + [C_j + B_i]_2 & \text{if } C_j < 1 \\ \text{Skip} & \text{Otherwise} \end{cases} \tag{16}$$

5.2.2 F3 Decoding

For decoding, after receiving C'_i, as in Steg, the receiver decodes B_i by simply modulo(2) operation as shown in Equation (17).

$$\underset{1\le i\le k,\ 1\le j\le n,\ i\le j}{B_i = \phi(C'_j)} = \begin{cases} 1 & \text{if } [C'_j]_2 = 1, \\ 0 & \text{Otherwise} \end{cases} \tag{17}$$

The F3 decoder skips coefficients with value 0. This is mainly due to the shrinkage process during the encoding period.

Example 2 (F3 encoding/decoding)**.** Consider B=[1 0 1 1 0 1] and C=[-3 -2 -1 0 1 2 3],
then C'=[-3 -2 -1 0 1 2 3].
B=[1 0 1 1 0 1], as the F3 encoding and decoding both ignores coefficient 0.

5.2.3 Limitation of F3

The F3 algorithm does not modify the coefficients with value 0. It also modifies ± 1 to 0, during the shrinkage, without hiding anything bits of information. This results an increase in number of 0's during encoding and waste of capacity. Like the JSteg algorithm, the F3 algorithm can not preserve the shape of JPEG coefficients histogram, and easily detected by the first order statistics and χ^2 test.

5.3 F4 Algorithm

The F4 algorithm is considered as an upgrade version of F3 algorithm. The detail of F4 algorithm has been described in [7]. The F4 algorithm continued with the shrinkage idea introduced in F3. As in F3 algorithm, F4 skips the coefficient with value 0. However, unlike F3, F4 algorithm alters $+1$ to 0, while the hidden message bit is 0, and alters -1 to 0, while the message bit is 1 (see Eq. 18).

5.3.1 F4 Encoding

Like F3, during the shrinkage process, F4 does not hide any message bits. The encoding process of F4 can be formulated as,

$$\underset{1\le i\le k,\ 1\le j\le n,\ i\le j}{C'_j = \psi(C_j, B_i)} = \begin{cases} C_j - [C_j + B_i + 1]_2 & \text{if } C_j > 0 \\ C_j + [C_j + B_i]_2 & \text{if } C_j < 1 \\ \text{Skip} & \text{Otherwise} \end{cases} \tag{18}$$

5.3.2 F4 Decoding

The decoding process, after receiving C_i' is shown in Equation (19).

$$\underset{1\le i\le k,\ 1\le j\le n,\ i\le j}{B_i = \phi(C_j')} = \begin{cases} \left[C_j'\right]_2 & \text{if } C_j' > 0 \\ \left[C_j'+1\right]_2 & \text{if } C_j' < 0 \\ \text{Skip} & \text{Otherwise} \end{cases} \tag{19}$$

The F4 decoder skips coefficients with value 0, and decoding of positive coefficients are different than negative coefficients.

Example 3 (F4 Encoding/decoding)**.** Consider B=[0 1 0 1 0 1] and C=[-3 -2 -1 0 1 2 3], then C'=[-3 -2 -1 0 1 2 3]. The encoding process is illustrated in Figure 19.

5.3.3 Limitation of F4

This algorithm produces extra number of zeros by the shrinkage process in the modified JPEG image. Extra zeros are created without hiding any secret message on it. This as in F3 reduces the hiding capacity.

Like JSteg and F3 algorithm, F4 algorithm can not preserve the shape of JPEG coefficient histogram. It can be detected by the first order statistics and χ^2 test.

5.4 F5 Algorithm

The F5 algorithm extends the functionality of F4 with two distinct features, matrix encoding and permutative straddling. The latter refers to scattering the entire message by permutation to equalize the spread of embedded data. F5 steganography hides p number of bits by modifying one coefficient out of $\alpha = 2^p - 1$ coefficients of a JPEG image. It splits the coefficients $C = [C_1 C_2 \cdots C_n]$ into $\beta = \frac{\lfloor n \rfloor}{\alpha}$ subsets denoted by $S = \left[S_1 S_2 \cdots S_\beta\right]$. Then each subset is encoded separately to form $S' = [S_1' S_2' \cdots S_\beta']$.

5.4.1 F5 Encoding

F5 divides the set of coefficients $C = [C_1 C_2 \cdots C_n]$ into β subsets each have α number of coefficients. Consider embedding the binary vector $B_i = [b_1, b_2, \cdots, b_k]$ into subset $S_i = [s_1, s_2, \cdots, s_m]$. We define matrix $H_{k\times m}$ that defines dependency between bits of B_i and the index of coefficients in S_i. For example, for a 2-bit and 3-bit embedding,

$$H_{2\times 3} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix} \qquad H_{3\times 7} = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

To find which coefficient in S_i should be modified, we use

$$t = H \times S_i^T - B_i^T$$

.

5.4.2 F5 Decoding

The decoder receives S'_i and decodes B_i with,

$$B_i = H \times S'^T_i$$

Example 4 (F5 Encoding/Decoding)**.** Let $S_i = \{-2, 3, -2\}$ and $B_i = \{1, 1\}$, then

$$H = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}$$

To find the position to modify,

$$t = H \times S_i^T - B_i^T = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix} \times \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} - \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

This corresponds to the first column of H, and hence the first coefficient in S_i (-2) needs to be modified to -1.

For the decoding process, the decoder receives $S'_i = \{-1, 3, -2\}$ as input and decodes B_i as,

$$B_i = H \times S'^T_i = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix} \times \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

5.4.3 Limitation of F5

This algorithm is same as F4 algorithm, except the inclusion of matrix embedding. Thus all the limitations of F4 algorithm remains the same in F5 algorithm [5]. The main advantage of F5 is its matrix embedding that provides less modifications. For example, out of 3 nonzero coefficients, F5 modifies only one coefficient to hide 2 bits.

5.5 Model Based Steganography (MBS) Algorithm

The idea behind MBS [16] is maximize the coding capacity while minimize the risk of detection. It is driven from a statistical model in which x is an instance of a random variable X distributed according to a statistical model P_X.

Let C denote the coefficients of JPEG image, where C_j as an instance of a random variable C. Let P_C be the probability distribution of C over the transmissions of C_j. The instance C_j is divided into two distinct parts; $C_{i\alpha}$ and $C_{i\beta}$. The part $C_{i\beta}$ will be replaced by $C'_{i\beta}$. In other words, $C_{i\beta}$ will be modified to $C'_{i\beta}$ in order to hide a secret message bits B_j.

Considering these two distinct parts as instances of two dependent random variables $C_{i\alpha}$ and $C_{i\beta}$. MBS method uses a model distribution (i.e., $\hat{P}_C$) of these two distinct instances. It then estimates the distribution of C over possible values for $C_{i\beta}$ conditioned on the current value for $C_{i\alpha}$.

5.5.1 MBS Encoding

Consider $C_j = (C_{j\alpha} \cup C_{j\beta})$,

$$\underset{1 \le i \le k,\ 1 \le j \le n,\ i \le j}{C'_j = \psi(C_j, B_i)} = \begin{cases} C_{j\alpha} \cup [C_{j\beta} + B_i + 1] & \text{if } C_{j\beta} > 0 \\ C_{j\alpha} \cup [C_{j\beta} + B_i] & \text{if } C_{j\beta} < 0 \\ \text{Skip} & \text{Otherwise} \end{cases} \tag{20}$$

5.5.2 MBS Decoding

For decoding, after receiving C'_i, the receiver decodes B_i by simply modulo(2) operation as shown in Equation (21).

$$\underset{1\leq i\leq k,\ 1\leq j\leq n,\ i\leq j}{B_i = \phi(C'_j)} = \begin{cases} 1 & \text{if } [C'_j]_2 = 1, \\ 0 & \text{Otherwise} \end{cases} \tag{21}$$

A steganogram c_i is split into parts c_α and c_α. A parametric model $\hat{P}_C$ is used to calculate the same probability distribution over possible c_β sequences that was used in the encoding process. c'_β is then fed into the entropy encoder which uses these probabilities to return the original message b_i.

Example 5 (MBS Encoding/Decoding)**.** Let $C_j = \{-1,-2,-1,1,1,2\}$ and, $B_i = \{1,1,0,1,0,1\}$. The modulo 2 of C_j is $C_j = \{1,0,1,1,1,0\}$ and $C'_j = \{-1,-1,-2,1,2,1\}$. For decoding, it is sufficient to get the modulo(2) operation on C'_j, which is $B_i = \{1,1,0,1,0,1\}$.

5.5.3 Limitation of MBS

Given only half of the coefficients can be used to hide the secret bits, the capacity of MBS is lower than its counter part. However, it avoid statistical detection better than the others. In MBS method maximum half of the coefficients can be used to hide the secret bits. Therefore, hiding capacity is lower than other algorithms.

5.6 Outguess Algorithm

The idea behind the Outguess algorithm is to measure the maximum length of a randomly spread secret code that can be embedded in an image while making sure the original histogram remain intact. Let h_d, $d = \cdots,-2,-1,0,1,2,\cdots$ denote the histogram corresponding to the DCT coefficient C and let P be the number of coefficients in C that are different from 0 and 1,

$$P \sum_{i\neq 0, i\neq 1} h_i$$

The Outguess algorithm tries to calculate the maximum length of a randomly spread secret code that can be encoded into the image while trying to keep the original histogram intact. Therefor, the histogram $[h_{2i}, h_{2i+1}]$ will be adjusted to:

$$\begin{array}{lcl} h_{2i} & \longrightarrow & h_{2i} - \alpha(h_{2i} - h_{2i+1}) \\ h_{2i+1} & \longrightarrow & h_{2i+1} + \alpha(h_{2i} - h_{2i+1}) \end{array} \tag{22}$$

where, $2\alpha = b/P$, b is the number of embedded secret code. For example when $h_{2i} > h_{2i+1}$, after embedding, there must be sufficient number of coefficients with value $2i+1$ to make the necessary adjustment. Therefore $h_{2i+1} - 2\alpha h_{2i+1} = \alpha(h_{2i} - h_{2i+1})$, hence,

$$\alpha_i = \frac{h_{2i+1}}{h_{2i+1} + h_{2i}}$$

The maximum message size that can be embedded with the necessary correction is, $b_{max} = 2\alpha P$, where $\alpha = \min_i \alpha_i$.

5.6.1 Limitation of Outguess

To hide one bit of secret message, this algorithm modifies two coefficient. Because of extensive modification, modified JPEG image quality decreases significantly. Can be detected by blocking factor analysis. In keeping the histogram shape intact, the algorithm modifies half of the coefficients to hide the message and adjust the second half to keep the shape as it was. This results the image to suffer from visual quality.

5.6.2 Outguess encoding/decoding

Let $C_j = \{\cdots, 5, 4, 3, 4, 5, 3, \cdots\}$. There are 6 non-zero coefficients, $P = 6$. The maximum capacity for Outguess algorithm would be $6/2 = 3$, and the maximum number of hidden bits $b_{\max} = 3$. The encoder divides C_j into two sets, C_{ja} and C_{jb}. C_{ja} will be modified to hide 3 bits, and C_{jb} will be modified to compensate the modification to keep the original histogram intact.

Example 6 (Outguess encoding/decoding)**.** Let $B_i = \{1, 1, 0\}$. A modulo 2 operation on C_j gives $\{\cdots, 1, 0, 1, |\, 0, 1, 1, \cdots\}$. Therefore, $C'_j = \{\cdots, 5, 3, 4, |\, 3, 5, 4, \cdots\}$. To decode the receiving coefficients $C'_j = \{\cdots, 5, 3, 4, |\, 3, 5, 4, \cdots\}$, modulo 2 operation on C'_j gives $C'_j = \{1, 1, 0, |\, 1, 1, 0\}$.

5.7 Limitations of JSteg, F3, F4 Algorithms

The first order statistical detection is one of the popular detection techniques for JPEG steganography detection [18]. The first order statistical detection uses magnitude of a the histogram of a JPEG coefficients to detect anomalies. The magnitude of the histogram follows a bell shape curve, sometimes with different magnitude. Let x_i represent the frequency associated with a coefficient, then

$$x_i > x_{i+1} \tag{23}$$

The JSteg, F3, F4 algorithms are suffering to maintain the bell shape histogram. The F5 algorithm has tried to overcome the problem but failed to do that because F5 is nothing but an advanced version of F4 algorithm (i.e., matrix embedding of F4 algorithm). When the receiving encoded image does not follow the bell shape histogram, that is a good indication of abnormality in the coding.

5.8 Modified matrix Embedding

The Modified matrix Embedding(MME) algorithm is an improved version of the F5 algorithm, it works the exact same way as the F5 algorithm [26]. However, the MME algorithm considers about rounding error during modification to reduce distortion of the stego image. This algorithm finds the candidate coefficient's position as same as F5 algorithm, and alters the coefficient with the help of Eq. 29 and modify the coefficient using the Eq. 24.

$$C'_i = \begin{cases} -2, \text{ if } r_i \leq 0, \text{ and } C_i = -1 \\ C_i + 1, \text{ if } r_i \leq 0, \text{ and } C_i \neq -1 \\ 2, \text{ if } r_i > 0, \text{ and } C_i = 1 \\ C_i - 1, \text{ if } r_i > 0, \text{ and } C_i = 1 \end{cases} \tag{24}$$

One of the most important advantages of the MME algorithm is that it can reduce the distortion. This is possible because the MME algorithm modifies the candidate coefficient in a way that provides the least distortion.

5.9 Minimum Distortion Embedding

A recent study presented a alorithm called Minimum Distortion Embedding (MME) [5]. Similar to MME, the MDE algorithm allows finding the candidate which provides the least amount of distortion. Hence, mathematically and theoretical MDE algorithm outperforms both F5 and MME algorithms.

First, the MDE algorithm gathers all the non-zero AC coefficients as an array ($D_i \in \mathbf{Z}$), and divides the array into small coefficient blocks B_i, where i = 1, 2, 3 $\cdots$, z, z is total number of block or total number of secret message bits (b_i). So, the B_z can be obtain by dividing the coefficient array by the number of secret message bits (see Eq. 25). If the number of secret message bits is α then,

$$B_z = \lfloor \frac{D_n}{\alpha} \rfloor \tag{25}$$

Second, it finds the coefficient that produces the least distortion for a block B_i

$$C_{min} = \min\{r_i, \{r_i, C_i \in B_i\}\} \tag{26}$$

Third, it finds the best candidate in the block and modifies the coefficient following the rule explained in Eq. 24.

Example 7 (MDE Encoding)**.** Let, a block be [5 2 3 1 -2 -5 -1], so, after adding all the coefficients the sum becomes 3, which is an odd number. So, if this block needs to hide 1 as a secret message bit, then nothing needs to be done for this case.

However, if the block needs to hide 0 as a secret message bit, then the sum value needs to be modified to an even number (i.e., 2 or 4). This can be done by modifying any of the coefficients. Either add 1 or subtract one would do the trick. However, because MDE algorithm tries to reduce the distortion as much as possible, therefore it looks for the coefficient that produces least distortion and ± 1 to it.

Example 8 (MDE encoding further explained)**.** Suppose a block B_i size is 5, and non-zero AC coefficients are before rounding -0.6994, 0.8534, 1.7352, 1.6229, -2.6861, and after rounding the DCT coefficients became as -1, 1, 2, 2, -3.

So, for the given block B_i the rounding errors would be as, -0.3006, 0.1466, 0.2648, 0.3771, -0.3139. Now, if modifications made by following the Eq. 24, then because of might look like as (-1-1), (1+1), (2-1), (2-1), (-3+1). Then, the error between original coefficients (before rounding) and modified coefficients (after modifying) would be -1.3006, 1.1466, -0.7352, -0.6229, and 0.6861. So, clearly the best candidate is the second to last coefficient.

6 Comparison of Classical Stegnagraphic Algorithms

This paper reviewed the basis steganographic algorithms. These algorithms have been evolved over the past decade in order to avoid breaking or detection, while increasing the encoding capacity. Table 1 compares the strengths and weaknesses of these algorithms in terms of both coding capacity and detection strength.

Table 1: Comparison of Existing Steganographic Algorithms

Algorithm	Capacity	Advantages	Detection Strength
JSteg	All nonzero AC coefficients except 1's	Simple encoding and decoding	Histogram analysis, χ^2 analysis
F3	All nonzero AC coefficients, except modification by shrinkage	Almost all nonzero coefficients cab be used	Histogram analysis, First order statistics, χ^2 analysis
F4	All nonzero AC coefficients, except modification by shrinkage	Almost all nonzero coefficients cab be used	First order statistics, χ^2 analysis
F5	Almost 13%, all nonzero AC coefficients, except modification by shrinkage	Matrix embedding techniques helps to reduce number of modification	CuSum analysis
Model Based	All nonzero AC coefficients	Keeps the histogram shape, based on probability theory	JPEG blocking factor analysis
Outguess	$\frac{1}{2}$ nonzero AC coefficients	Preserve the histogram shape	JPEG blocking factor analysis
MME	Almost 13%, all nonzero AC coefficients, except modification by shrinkage	Produces less distortion than F5 algorithm	Feature analysis
MDE	Depends on the block size	Produces less distorted stego image	Yet to analyze

7 Conclusion and Future Research Directions

One of the biggest challenges in covert channel or steganographic data hiding is that limited scope to hide and conceal the existence of the modification [5, 27]. While, so many algorithms developed over time to hide data, but most of them are slight variations of previous algorithms. Because, researchers do not provide mathematical models and example to explain their work. This paper explained mathematical model for classical steganograhpic algorithms. Understanding mathematical models will help researchers to gain in-depth knowledge in steganographic field of research. Also, by analyzing these mathematical models, they could identify steps in these algorithm to improve and make them more secure. Future work for this work would be to compare these algorithms with newer algorithms and provide and comparative analysis.

These existing algorithms and steganalysis suggest that researchers should focus on controlling quality of the setgo image. It seems that matrix embedding or distortion control algorithms are much harder to break than the others [27]. While, matrix embedding provides less modification, however, adding distortion control helps to control the quality of the stego image further. In the future, researchers should try to minimize the modification effect as JPEG image already suffer from quantization effect, an uncontrolled modification could reveal the existence of secret message.

References

[1] M. Amiruzzaman, H. Peyravi, M. Abdullah-Al-Wadud, and Y. Chung. Concurrent covert communication channels. In *Advances in Computer Science and Information Technology, Proc. of the 2020 International Conference on Advanced Computer Science and Information Technology (AST'10) and the 2010 International Conference on Advanced Communication and Networking (ACN'10), Miyazaki, Japan*, volume 6059 of *Lecture Notes in Computer Science*, pages 203–213. Springer, Berlin, Heidelberg, June 2010.

[2] M. Amiruzzaman. Steganographic covert communication channels and their detection. MS thesis, Kent State University, 2011.

[3] Md Amiruzzaman, Mohammad Abdullah-Al-Wadud, and Yoojin Chung. An analysis of syndrome coding. In *Proc. of the 2010 International Conference on Information Security and Assurance (ISA'10), Miyazaki, Japan*, volume 76 of *Communications in Computer and Information Science*, pages 37–50. Springer Berlin, Heidelberg, June 2010.

[4] M Amiruzzaman, H. Peyravi, M. Abdullah-Al-Wadud, and Y. Chung. An improved steganography covert channel. In *Proc. of the 2009 International Conference on Advanced Software Engineering and Its Applications (ASEA'09), Jeju Island, Korea*, volume 59 of *Communications in Computer and Information Science*, pages 176–187. Springer, Berlin, Heidelberg, December 2009.

[5] M Amiruzzaman and R. M. Nor. Hide secret information in blocks: Minimum distortion embedding. In *Proc. of the 7th International Conference on Signal Processing and Integrated Networks (SPIN'20), Noida, India*, pages 107–112. IEEE, February 2020.

[6] N. Provos. Defending against statistical steganalysis. In *Proc. of the 10th USENIX Security Symposium (Security'01), Washington, D.C., USA*. USENIX Association, August 2001.

[7] A. Westfeld. F5 — A steganographic algorithm. In *Proc. of the 2001 International Workshop on Information Hiding (IH'01), Pittsburgh, PA, USA*, volume 2137 of *Lecture Notes in Computer Science*, pages 289–302. Springer, Berlin, Heidelberg, April 2001.

[8] M. Y. Valandar, M. J. Barani, P. Ayubi, and M. Aghazadeh. An integer wavelet transform image steganography method based on 3d sine chaotic map. *Multimedia Tools and Applications*, 78(8):9971–9989, 2019.

[9] Z. Qu, Z. Cheng, and X. Wang. Matrix coding-based quantum image steganography algorithm. *IEEE Access*, 7:35684–35698, 2019.

[10] J. Fridrich, M. Goljan, and D. Soukal. Perturbed quantization steganography with wet paper codes. In *Proc. of the 2004 Workshop on Multimedia and Security (MM&Sec'04), Magdeburg Germany*, pages 4–15. ACM, September 2004.

[11] J. Fridrich, M. Goljan, and D. Hogea. Attacking the outguess. In *Proc. of Multimedia and Security Workshop at ACM Multimedia 2002, Juan-les-Pins, France*. ACM, December 2002.

[12] J. Fridrich, M. Goljan, and D. Hogea. Steganalysis of JPEG images: Breaking the F5 algorithm. In *Proc. of the 5th International Workshop on Information Hiding (IH'02), Noordwijkerhout, The Netherlands*, volume 2578 of *Lecture Notes in Computer Science*, pages 310–323. Springer, Berlin, Heidelberg, October 2002.

[13] B. He, Y. Chen, Y. Zhou, Y. Wang, and Y. Chen. A novel two-dimensional reversible data hiding scheme based on high-efficiency histogram shifting for jpeg images. *International Journal of Distributed Sensor Networks*, 18(3):15501329221084226, 2022.

[14] G. Peter, A. Sherine, Y. Teekaraman, R. Kuppusamy, and A. Radhakrishnan. Histogram shifting-based quick response steganography method for secure communication. *Wireless Communications and Mobile Computing*, 2022:1505133, March 2022.

[15] M. Amiruzzaman, M. Abdullah-Al-Wadud, and Y. Chung. Minimum distortion data hiding. In *Proc. of the 2010 International Conference on Advanced Software Engineering and Its Applications (ASEA'10), Jeju Island, Korea*, volume 117 of *Communications in Computer and Information Science*, pages 151–163. Springer, Berlin, Heidelberg, December 2010.

[16] P. Sallee. Model-based steganography. In *Proc. of the 2nd International Workshop on Digital Watermarking (IWDW'03), Seoul, Korea*, volume 2939 of *Lecture Notes in Computer Science*, pages 154—167. Springer, Berlin, Heidelberg, October 2003.

[17] P. Sallee. Model-based methods for steganography and steganalysis. *International Journal of Image and Graphics*, 5(1):167–190, 2005.

[18] H. Zhang, T. Zhang, and H. Chen. Revisiting weighted stego-image steganalysis for pvd steganography. *Multimedia Tools and Applications*, 78(6):7479–7497, 2019.

[19] A. Chatterjee and A. K. Das. Secret communication combining cryptography and steganography. In *Proc. of the 2016 International Conference on Advanced Computing and Intelligent Engineering (ICACIE'16), Bhubaneswar, Odisha, India*, volume 563 of *Advances in Intelligent Systems and Computing*, pages 281–291. Springer, Singapore, December 2018.

[20] A. Hore and D. Ziou. Image quality metrics: Psnr vs. ssim. In *Proc. of the 20th International Conference on Pattern Recognition (ICPR'10), Istanbul, Turkey*, pages 2366–2369. IEEE, August 2010.

[21] M. Amiruzzaman and H. J. Kim. Secure steganographic method. In *Proc. of the 5th International Conference on Visual Information Engineering (VIE'08), Xi'an, China*, pages 141–145. IET, July 2008.

[22] S. A. Nie, G. Sulong, R. Ali, and A. Abel. The use of least significant bit (lsb) and knight tour algorithm for image steganography of cover image. *International Journal of Electrical & Computer Engineering*, 9(6):5218–2526, 2019.

[23] A. Chatterjee and S. K. Pati. Data hiding with digital authentication in spatial domain image steganography. In *Proc. of the 2019 International Conference on Computational Intelligence in Pattern Recognition (CIPR'19), Shibpur, India*, volume 999 of *Advances in Intelligent Systems and Computing*, pages 897–907. Springer, Singapore, Januray 2019.

[24] D. Upham. Jsteg source. `http://zooid.org/~paul/crypto/jsteg/`. [Online; Accessed on August 10, 2022].

[25] A. Westfeld and A. Pfitzmann. Attacks on steganographic systems - breaking the steganographic utilities ezstego. In *Proc. of the 1999 International Workshop on Information Hiding (IH'99), Dresden, Germany*, volume 1768 of *Lecture Notes in Computer Science*, pages 61–75. Springer, Berlin, Heidelberg, September-October 1999.

[26] Y. Kim, Z. Duric, and D. Richards. Modified matrix encoding technique for minimal distortion steganography. In *Proc. of the 2006 International Workshop on Information Hiding (IH'06), Alexandria, VA, USA*, volume 4437 of *Lecture Notes in Computer Science*, pages 314–327. Springer, Berlin, Heidelberg, July 2006.

[27] T. D. Nguyen and H. Q. Le. A secure image steganography based on modified matrix encoding using the adaptive region selection technique. *Multimedia Tools and Applications*, 81:25251—-25281, 2022.

[28] R. Rosenholtz and A. B. Watson. Perceptual adaptive jpeg coding. In *Proc. of the 3rd IEEE International Conference on Image Processing (ICIP'96), Lausanne, Switzerland*, pages 901–904. IEEE, September 1996.

Appendix

The JPEG image compression is a complex process, it goes through JPEG encoder and decoder during the transformation before obtaining the DCT coefficients [2]. In the encoder, an input image is divided into 8×8 blocks (i.e., JPEG block). Let, $f(i, j)$, where $i, j = 0, 1, \cdots, N-1$ of a $N \times N$ image channel block and $F(i, j)$, where $i, j = 0, 1, \cdots, N-1$ of a DCT transformation of the $N \times N$ image channel block [1].

The DCT coefficient $F(0,0)$ is known as DC coefficient and the rest of the 63 coefficients of an 8×8 block are known as AC coefficients [2,5]. The quantization matrix is denoted as Q, and before rounding the coefficients are expressed as,

$$F'(i, j) = \frac{F(i, j)}{Q(i, j)} \tag{27}$$

and after rounding the coefficients becomes integer [28] as described in Eq. 28.

$$F''(i, j) = \lfloor F'(i, j) \rfloor \tag{28}$$

clearly there is a difference between $F'(i, j)$ and $F''(i, j)$ because of the rounding operation, which can be expressed as,

$$r_i = F'(i, j) - F''(i, j) \tag{29}$$

Author Biography

Md Amiruzzaman is an Assistant Professor in the Department of Computer Science at West Chester University. Prior to joining WCU, he worked as a software developer for almost 10 years for several companies. He has also held the position of Assistant Professor at Kent State University. He has completed his Bachelor's Degree in Computer Science from National University. Along with that, he has completed three Master's degrees, majoring in Computer Engineering in 2008 from Sejong University, Computer Science in 2011 from Kent State University and Technology in 2015, also from Kent State University. He received his Ph.D. degrees from Kent State University in 2016 (Mathematics Edu), 2019 (Evaluation and Measurement) and 2021 (Computer Science). In the past, he has worked as a Research Assistant at Sejong University and Korea University. Along with that, he gained the opportunity to teach at both National University and Korea University. His research interests include Visual Analytics of urban data, Data Mining, Machine Learning, Deep Learning and Data Hiding.