Hiding Data in Images Using PCP
Souvik Bhattacharyya and Gautam Sanyal

Abstract—In recent years, everything is trending toward digitalization and with the rapid development of the Internet technologies, digital media needs to be transmitted conveniently over the network. Attacks, misuse or unauthorized access of information is of great concern today which makes the protection of documents through digital media a priority problem. This urges us to devise new data hiding techniques to protect and secure the data of vital significance. In this respect, steganography often comes to the fore as a tool for hiding information. Steganography is a process that involves hiding a message in an appropriate carrier like image or audio. It is of Greek origin and means "covered or hidden writing". The goal of steganography is covert communication. Here the carrier can be sent to a receiver without any one except the authenticated receiver only knows existence of the information. Considerable amount of work has been carried out by different researchers on steganography. In this work the authors propose a novel Steganographic method for hiding information within the spatial domain of the gray scale image. The proposed approach works by selecting the embedding pixels using some mathematical function and then finds the 8 neighborhood of the each selected pixel and map each bit of the secret message in each of the neighbor pixel coordinate position in a specified manner. Before embedding a checking has been done to find out whether the selected pixel or its neighbor lies at the boundary of the image or not. This solution is independent of the nature of the data to be hidden and produces a stego image with minimum degradation.

Keywords—Cover Image, LSB, Pixel Coordinate Position (PCP), Stego Image.

I. INTRODUCTION

STEGANOGRAPHY is the art and science of hiding information by embedding messages within other, seemingly harmless messages. Steganography means “covered writing” in Greek. As the goal of steganography is to hide the presence of a message and to create a covert channel, it can be seen as the complement of cryptography, whose goal is to hide the content of a message. Another form of information hiding is digital watermarking, which is the process that embeds data called a watermark, tag or label into a multimedia object such that watermark can be detected or extracted later to make an assertion about the object. The object may be an image, audio, video or text only. A famous illustration of steganography is Simmons’ Prisoners’ Problem [15]. Alice and Bob are in jail, locked up in separate cells far apart from each other, and wish to devise an escape plan. They are allowed to communicate by means of sending messages via trusted couriers, provided they do not deal with escape plans. But the couriers are agents of the warden Eve (who plays the role of the adversary here) and will leak all communication to her. If Eve detects any sign of conspiracy, she will thwart the escape plans by transferring both prisoners to high-security cells from which nobody has ever escaped. Alice and Bob are well aware of these facts, so that before getting locked up, they have shared a secret codeword that they are now going to exploit for embedding hidden information into their seemingly innocent messages, which gives the birth of steganography principle. Alice and Bob succeed if they can exchange information allowing them to coordinate their escape and Eve does not become suspicious. The warden is free to examine all communication exchanged between Alice and Bob can either be active or passive. An active warden will try to alter the communication with the suspected hidden information deliberately in order to remove the information where as a passive warden takes the note of covered communication, informs the others and allows the message to pass through. An assumption can be made based on this model is that if both the sender and receiver share some common secret information then the corresponding steganography protocol is known as the then the secret key steganography where as pure steganography means that there is none prior information shared by sender and receiver. If the public key of the receiver is known to the sender, the steganographic protocol is called public key steganography [2], [3] and [9]. For a more thorough knowledge of steganography methodology the reader may see [13], [16]. Some Steganographic model with high security features has been presented in [4], [5] and [6]. Although all digital file formats can be used for steganography, but the image and audio files are more suitable because of their high degree of redundancy [16]. Fig. 1 below shows the different categories of file formats that can be used for steganography techniques.

A block diagram of a generic image steganographic system is given in Fig. 2.

A message is embedded in a digital image (cover image) through an embedding algorithm, with the help of a secret key. The resulting stego image is transmitted over a channel to the receiver where it is processed by the extraction algorithm using the same key. During transmission the stego image, it can be monitored by unauthenticated viewers who will only notice the transmission of an image without discovering the existence of the hidden message. Capacity, security, and robustness are the three main aspects affecting steganography and its usefulness. Capacity refers to the amount of data bits.
that can be hidden in the cover medium. Security relates to the ability of an eavesdropper to figure the hidden information easily. Robustness is concerned about the resist possibility of modifying or destroying the unseen data.

In this work a specific image based steganographic method for gray level image has proposed. In this method instead of embedding the secret message into the cover image a mapping technique has been used to generate the stego image. This method is capable of extracting the secret message without checking the cover image.

This paper has been organized as following sections: Section II describes some related works, Section III deals with proposed method. Algorithms are discussed in Section IV and Experimental results are shown in Section V. Section VI contains the analysis of the results and Section VII draws the conclusion.

II. RELATED WORKS

A. Data Hiding by LSB

Various techniques about data hiding have been proposed in literatures. One of the common techniques is based on manipulating the least-significant-bit (LSB) [7], [8] and [12], [14] planes by directly replacing the LSBs of the cover-image with the message bits. LSB steganography techniques will either change the value of each pixel by +1 or -1 or it will make the pixel value remain unchanged. This is depending on the value of the hidden data bit and the lsb of the corresponding pixel value. LSB methods typically achieve high capacity but unfortunately LSB insertion is vulnerable to slight image manipulation such as cropping and compression. For example, converting a GIF or a BMP image, which reconstructs the original message exactly (lossless compression), to a JPEG format, which does not (lossy compression), and then converting back, can destroy the data in the LSBs. Thus improving the stego image quality is one of the major research areas about LSB.

B. Data Hiding by PVD

The pixel-value differencing (PVD) method proposed by Wu and Tsai [17] can successfully provide both high embedding capacity and outstanding imperceptibility for the stego-image. The pixel-value differencing (PVD) method segments the cover image into non overlapping blocks containing two connecting pixels and modifies the pixel difference in each block (pair) for data embedding. A larger difference in the original pixel values allows a greater modification. The hiding algorithm is described below:

1) Calculate the difference value \( d_i \) for each block of two consecutive pixels \( p_i \) and \( p_{i+1} \) such that \( d_i = p_{i+1} - p_i \)
2) Find the optimal \( R_i \) of the \( d_i \) such that \( R_i = \min(u_i - k) \) where \( u_i \geq k, k = |d_i| \) and \( R_i \in [l_i, u_i] \) where \( i=1,2,...,n \).
3) Compute t bits of secret data which are hidden in each \( d_i \) i.e. each block of two consecutive pixels is defined as \( t = \log_2 w_i \) where \( w_i \) is the width of the \( R_i \).
4) Read t bits binary secret data one by one according to Step 3, and then transforms t into decimal value b. For instance, assume a binary secret data is 100, then \( b = 4 \).
5) Calculate the new difference value \( d_i' \) using \( d_i' = l_i + b \) if \( d_i \geq 0 \) and \( d_i' = (l_i + b) \) if \( d_i < 0 \).
6) \( p_i \) and \( p_{i+1} \) are modified to hide t secret data by the following formula:

\[
(p'_i, p'_{i+1}) = \begin{cases} 
(p_i - \lfloor m/2 \rfloor, (p_{i+1} - \lfloor m/2 \rfloor)) & \text{if } d_i \text{ odd} \\
(p_i - \lfloor m/2 \rfloor, (p_{i+1} - \lceil m/2 \rceil)) & \text{if } d_i \text{ even}
\end{cases}
\]

where \( m = d_i' - d_i \), Finally, we compute the values of \( (p'_i, p'_{i+1}) \) which represent the secret data.
7) Repeat Steps 1-6, until all secret data are hidden into the cover image and the stego-image is obtained.

In the extraction phase, the original range table is necessary. It is used to partition the stego-image by the same method as used to the cover image. The extraction phase is implemented as follows:

1) Calculate the difference value \( d_i' \) between each two successive pixels for each block \( (p'_i, p'_{i+1}) \) from the following formula: \( d_i' = |p'_i - p'_{i+1}| \)
2) Find the optimum \( R_i \) of the \( d_i' \) just as in Step 2 in the hiding phase.
3) Obtain b’ by subtracting \( l_i \) from \( d_i' \). The b’ value represents the value of the secret data in decimal.

Based on PVD method, various approaches have also been proposed. Among them Chang et al. [11], proposes a new method using tri-way pixel-value differencing which is better than original PVD method with respect to the embedding capacity and PSNR.

C. Data Hiding by GLM

In 2004, Potdar et al.[10] proposes GLM (Gray level modification) technique which is used to map data by modifying the gray level of the image pixels. Gray level modification Steganography is a technique to map data (not embed or hide it) by modifying the gray level values of the image pixels. GLM Steganography uses the concept of odd and even numbers to map data within an image. It is a one-to-one mapping between the binary data and the selected pixels in an image. From a given image a set of pixels are selected based on a mathematical function. The gray level values of those pixels are examined and compared with the bit stream that is
to be mapped in the image. Initially, the gray level values of the selected pixels (odd pixels) are made even by changing the gray level by one unit. Once all the selected pixels have an even gray level it is compared with the bit stream, which has to be mapped. The first bit from the bit stream is compared with the first selected pixel. If the first bit is even (i.e., 0), then the first pixel is not modified as all the selected pixels have an even gray level value. But if the bit is odd (i.e., 1), then the gray level value of the pixel is decremented by one unit to make its value odd, which then would represent an odd bit mapping. This is carried out for all bits in the bit stream and each and every bit is mapped by modifying the gray level values accordingly.

\[
\begin{align*}
 & P_{r,c}^{\prime} = P_{r,c} \oplus m_k \quad (1) \\
 & I = 1, 2, \ldots, 8.
\end{align*}
\]

\( j < N, P_{ij} \in \{0, 1, \ldots, 255\} \). Let MSG be the n bit secret message represented as MSG = \((m_i) \mid 0 \leq i \leq n, m_i \in \{0, 1\}\). A pixel \( P_{rc} \) can be selected randomly with row \( r \) and column \( c \). Next step is to find the 8 neighbors \( P_{r',c}^{\prime} \) of the pixel \( P_{rc} \) such that \( r' = r + k, c' = c + k, -1 \leq k \leq 1 \). Row array (ROWA) and the column array (COLA) of the coordinate position can be formed by inserting the last bit (binary) of the row position of the pixel at POS1 in row array and last bit (binary) of the column position of the pixel at POS1 at column array. Proceed this way for the remaining positions. Considering the secret message as a long bit stream, each bit of every byte of the secret message will be embedded in each LSB of the neighbor pixel using the following formula:

\[
P_{r',c',i}(POSI) = (ROWA_{POS1} \oplus COLA_{POS1} \oplus m_I) \quad (1)
\]

\( I = 1, 2, \ldots, 8. \)

The embedding process will be finished when all the bits of every byte of secret message are mapped or embedded. A flowchart in figure 6 describes the proposed embedding scheme.

2) Data Extraction Method: The process of extraction proceeds by selecting those same pixel with their neighbors. Again form the ROWA and COLA using the same method described during embedding operation. Message bit can be extracted using the following formula:

\[
m_I = P_{r',c',i}(POSI) \oplus (ROWA_{POS1} \oplus COLA_{POS1}) \quad (2)
\]

\( I = 1, 2, \ldots, 8. \)

The extracting process will be finished when all the bits of every byte of secret message are extracted. A flowchart in figure 8 describes the proposed scheme.

One important point needs to be kept in mind that a specific order for selecting the neighbor pixels has to be maintained for embedding / mapping process and also for the process of extraction other wise it would not be possible for retrieve the data in proper sequence. This sequence has been shown in Figure 10.
3) Analysis of the Method: In this section, the general operations of data hiding of the proposed method have been described. As previously mentioned, C be the original 8 bit gray scale image of size N x N i.e. C = (P\textsubscript{ij} | 0 ≤ i < N, 0 ≤ j < N, P\textsubscript{ij} ∈ {0, 1, ..., 255}) and MSG be the n bit secret message represented as MSG = (m\textsubscript{i} | 0 ≤ i < n, m\textsubscript{i} ∈ {0, 1}). Suppose the modified message (MSGNEW) bit is to be embedded into the k right most LSBs of the pixels of the cover image C. Here MSGNEW can be represented as MSGNEW = (m\textsubscript{i} | 0 ≤ i < n, m\textsubscript{i} ∈ {0, 1}), although for this problem k is always one and m\textsubscript{i} can be represented as m\textsubscript{i} = ROW\textsubscript{APOS}\textsubscript{I} \oplus COL\textsubscript{APOS}\textsubscript{I} \oplus m\textsubscript{I}, I = 1, 2, ..., 8. The embedding process will be finished when all the bits of every bytes of secret message are mapped or embedded. The modified pixel value of P\textsubscript{r}'\textsubscript{c}' of the stego image S can be written as P\textsubscript{r}'\textsubscript{c}' = P\textsubscript{r}\textsubscript{c} - P\textsubscript{r}\textsubscript{c} mod 2 + m\textsubscript{i}. At the time of extraction, given the stego image S, the embedded messages can be readily extracted without referring to the original cover-image. Using the same sequence as in the embedding process, the set of pixels P\textsubscript{r}1', P\textsubscript{r}2', ..., P\textsubscript{r}n', are selected from the stego image S and the message can be obtained using the following formula:

\[ m\textsubscript{I} = P\textsubscript{r}i'(\text{POS}I) \oplus (ROW\text{APOS}i \oplus COL\text{APOS}i), I = 1, 2, ..., 8. \]

Next we will see that the resulting message bit computed from P\textsubscript{r}c' is identical with the original message bit. From Eq. (2), we can get m\textsubscript{I} = P\textsubscript{r}i'(\text{POS}I) \oplus (ROW\text{APOS}i \oplus COL\text{APOS}i) = (ROW\text{APOS}i \oplus COL\text{APOS}i \oplus m\textsubscript{I}) \oplus (ROW\text{APOS}i \oplus COL\text{APOS}i) = m\textsubscript{I} (using a \oplus a = 0).
Pixel Selection Method: The algorithm for selection of pixel for embedding is described below:

1. Let i = 2, j = 2.
2. while (j ≠ N).
3. begin
4. Mark pixel (i, j).
5. if (i ≥ N)
6. begin
7. i = 2, j = j + 3
8. End
9. End
10. End

![Sample of Selected Pixel](image)

**V. EXPERIMENTAL RESULTS**

In this section the authors present the experimental results of the proposed method based on two benchmarks techniques to evaluate the hiding performance. First one is the capacity of hiding data and another one is the imperceptibility of the stego image, also called the quality of stego image. The quality of stego-image should be acceptable by human eyes. The authors also present a comparative study of the proposed methods with the existing methods like PVD, GLM and the methods proposed by Ahmad T et al. by computing embedding capacity, mean square error (MSE) and peak signal-to-noise ratio (PSNR). The authors also compute the normalized cross correlation coefficient for computing the similarity measure between the cover image and stego image. In this section experimental result of stego-image are shown based on two well known images: Lena and Pepper images.

A comparison of the embedding capacity has been illustrated in figure 12.

**TABLE I**

<table>
<thead>
<tr>
<th>Image Size</th>
<th>Data Size (Pl. Mb)</th>
<th>Data Size (PVD)</th>
<th>Data Size (glm)</th>
<th>Data Size (prop)</th>
</tr>
</thead>
<tbody>
<tr>
<td>512x512</td>
<td>2048</td>
<td>2493</td>
<td>1764</td>
<td>1764</td>
</tr>
<tr>
<td>256x256</td>
<td>8192</td>
<td>10077</td>
<td>7253</td>
<td>7253</td>
</tr>
<tr>
<td>128x128</td>
<td>3276</td>
<td>40033</td>
<td>28560</td>
<td>28560</td>
</tr>
</tbody>
</table>

**TABLE II**

<table>
<thead>
<tr>
<th>Image Size</th>
<th>Data Size (Pl. Mb)</th>
<th>Data Size (PVD)</th>
<th>Data Size (glm)</th>
<th>Data Size (prop)</th>
</tr>
</thead>
<tbody>
<tr>
<td>512x512</td>
<td>2048</td>
<td>2493</td>
<td>1764</td>
<td>1764</td>
</tr>
<tr>
<td>256x256</td>
<td>8192</td>
<td>10077</td>
<td>7253</td>
<td>7253</td>
</tr>
<tr>
<td>128x128</td>
<td>3276</td>
<td>40033</td>
<td>28560</td>
<td>28560</td>
</tr>
</tbody>
</table>

Fig. 12. Comparison of embedding capacity

**For PVD method all the images used are of size 512x512.**

Peak Signal to Noise Ratio (PSNR): PSNR measures the quality of the image by comparing the original image or cover image with the stego-image, i.e. it measures the percentage of the stego-data to the image percentage. The PSNR is used to evaluate the quality of the stego-image after embedding the secret message in the cover. Assume a cover image C(i, j) that contains N by N pixels and a stego image S(i, j) where S is generated by embedding / mapping the message bit stream. Mean squared error (MSE) of the stego image as follows:

\[
MSE = \frac{1}{N \times N} \sum_{i=1}^{N} \sum_{j=1}^{N} [C(i, j) - S(i, j)]^2
\]

The PSNR is computed using the following formulae:

\[
PSNR = 10 \log_{10} \frac{255^2}{MSE} \text{ db}
\]

In Figure 15 a comparative study of PSNR of various methods has been shown.

Similarity Measure: For comparing the similarity between cover image and the stego image, the normalized cross correlation coefficient (r) has been computed. In statistics, correlation indicates the strength and direction of a linear relationship between two random variables. In general statistical usage, correlation or co-relation refers to the departure of two random variables from independence. In this broad sense there are several coefficients, measuring the degree of correlation, adapted to the nature of the data. A number of different coefficients are used for different situations. The best known is the Pearson correlation coefficient.

![Comparison of the Lena and Pepper images](image)

**Fig. 13. A) Cover Image B) Stego Image of Lena**

**Fig. 14. A) Cover Image B) Stego Image of Pepper**
product-moment correlation coefficient, which is obtained by dividing the covariance of the two variables by the product of their standard deviations. The correlation coefficient \( \rho_{x,y} \) between two random variables \( X \) and \( Y \) with expected values \( \mu_x \) and \( \mu_y \) and standard deviations \( \sigma_x \) and \( \sigma_y \) is defined as

\[
\rho_{x,y} = \frac{\text{cov}(x, y)}{\sigma_x \sigma_y} = \frac{E((X - \mu_x)(Y - \mu_y))}{\sigma_x \sigma_y}
\]

where \( E \) is the expected value operator and \( \text{cov} \) means covariance. Since \( \mu_x = E(X) \), \( \sigma_x^2 = E[(X - E(X))^2] = E(X^2) = E^2(X) \) and likewise for \( Y \), and since \( E[(X - E(X))(Y - E(Y))] = E(XY) - E(X)E(Y) \), we may also write

\[
\rho_{x,y} = \frac{E(xy) - E(x)E(y)}{\sqrt{E(x^2) - E^2(x)} \sqrt{E(y^2) - E^2(y)}}
\]

The value of correlation is 1 in the case of an increasing linear relationship, -1 in the case of a decreasing linear relationship, and some value in between in all other cases, indicating the degree of linear dependence between the variables. The closer the coefficient is to either -1 or 1, the stronger the correlation between the variable. If the variables are independent then the correlation is 0, but the converse is not true because the correlation coefficient detects only linear dependencies between two variables.

Cross correlation is a standard method of estimating the degree to which two series are correlated. Consider two series \( x(i) \) and \( y(i) \) where \( i=0,1,2,\ldots,N-1 \). The cross correlation \( r \) at delay \( d \) is defined as

\[
r = \frac{\sum [(x(i) - mx)(y(i-d) - my)]}{\sqrt{\sum (x(i) - mx)^2} \sqrt{\sum (y(i-d) - my)^2}}
\]

where \( mx \) and \( my \) are the means of the corresponding series. The cross-correlation is used for template matching is motivated through the following formula

\[
r = \sum f(x,y) t(x-u,y-v)
\]

where \( f \) is the image and the sum is over \( x, y \) under the window containing the feature \( t \) positioned at \( u, v \).

Similarity measure of two images can be done with the help of normalized cross correlation generated from the above concept using the following formula:

\[
r = \frac{\sum (C(i,j) - m_1)(S(i,j) - m_2)}{\sqrt{\sum (C(i,j) - m_1)^2} \sqrt{\sum (S(i,j) - m_2)^2}}
\]

Here \( C \) is the cover image, \( S \) is the stego image, \( m_1 \) is the mean pixel value of the cover image and \( m_2 \) is the mean pixel value of stego image. It has been seen that the correlation coefficient computed here for both Lena image and Pepper image is one which indicates the both the cover image and stego image are of highly correlated i.e. both of these two images are same. Besides comparison through histogram technique has been done. It has been observed that the histogram of the cover image and the stego image is identical.
VI. ANALYSIS OF THE RESULTS

In this article the authors proposed an efficient image based steganography approach for hiding information in a gray scale image. Comparison has been shown with some existing methods like PVD, GLM and the technique proposed by Ahmad T et al. From the experimental results in can be seen that although the embedding capacity of this proposed method is low compared to PVD, GLM and other technique but the similarity measures proves that the proposed method is best among these four methods which ensures that cover image and the stego image generated after embedding /mapping the secret message bit stream is almost identical. This property enables the proposed method to avoid steganalysis also. As the message bits are not directly embedded at the pixels of the cover image, steganalysis may be able to find out the embedded bits but can not be able to extract the original message bits. Besides PSNR value of the proposed method for various size of the image is better compared to other method in most of the cases.

VII. CONCLUSION

The work dealt with the techniques for steganography as related to gray scale image. A new and efficient steganographic method for embedding secret messages into images without producing any major changes has been proposed. This property enables the method to avoid steganalysis. This method also capable of extracting the secret message without the cover image. This approach may be modified to work on color images also.

REFERENCES