Secure Information Transfer through Image Steganography with High Data Hiding Capacity

Mr. Shreyansh Mishra[1], Mr. Sourav Mishra[2], Mr. Rohit Mundada[3], Prof. Pratik Shah[4].

[1,2,3]Student, Dept of E&TC Engineering, Dr. D.Y. Patil School of Engineering, Maharashtra, India
[4]Professor, Dept. of E&TC Engineering, Dr. D.Y. Patil School of Engineering, Maharashtra, India

---

ABSTRACT - *Innovation in modern technology and fast Internet have made information distribution over the world easy and economically. Such large distribution and exchange of information has made people to worry about their privacy and work. Prevention of important data from unauthorized users by concealing it in other non-secret data is Steganography. There are various techniques such as Genetic algorithm, Least Significant bit (LSB), Spread Spectrum, etc. used for digital image steganography, among which very few techniques deal with increasing imperceptibility and data embedding capacity together. This paper, reviews the modified version of the least-significant-bit (LSB) technique of Steganography resulting in improved imperceptibility and power to noise ratio (PSNR). The experimental results manifested that the proposed technique furnish an improvement in imperceptibility of stego-image at high data embedding rate in comparison to many other popular steganography techniques. The average PSNR value of 46.38 is exhibited by various stego-images at two bit per pixel data embedding rate.*

1.INTRODUCTION

Hiding secret data inside any form of digital media like images, audio, video, etc is Steganography. As the image steganography is very popular as it has an enormous amount of redundancy, which helps to hide the secret data without being noticed effectively. The secret data is embedded in the cover image and the resultant image obtained after inserting the secret data is called stegoimage. The main aim of image steganography is to hide the existence of secret communication; it is done by reducing the difference between stego-image and the cover image. The important parameters used for evaluating the performance of image steganography are robustness, imperceptibility, payload capacity, and security. Imperceptibility represents the similarity of stego-image with the cover image. Payload capacity is the capacity of data hiding which indicates the total number of bits hidden and successfully recovered by the Stego system. The ability of the embedded data to remain intact if the system undergoes transformation like linear and non-linear filtering, addition of random noise, rotation, scaling and compression, sharpening or blurring, lossy compression is called as Robustness. Security is the parameter which enlarges the capability of steganography technique to resist the steganalysis attacks. The image steganography techniques have been classified mainly into two groups i.e. Spatial Domain and Transform Domain. Also few new classification of steganographic techniques have evolved such as spread spectrum, statistical method, distortion technique, etc. Spatial domain techniques envelop bit-wise methods that apply bit insertion and noise manipulation. In Transform domain techniques also known as frequency domain, images are first transformed and then the message is embedded in the image. Spatial domain algorithms usually have a better visual quality of stego-images; however, their performance against statistical steganalysis attacks is poor. Spatial domain techniques might have good data embedding capacity, but it has an adverse effect on the visual quality of stego-image. Transform domain methods are good against statistical steganalysis attacks as the secret data is spread across the entire image through frequency domain coefficients. However, these techniques have small payload capacity and imperceptibility is lower. Significant amount of work has been carried out in steganography; both in spatial as well as transform domain, but limited work is presented on the use of evolutionary computation in image steganography. In this paper, we propose LSB based spatial domain image steganography technique with high data embedding capacity and imperceptibility. In the proposed technique, two-bit of secret data is embedded in each pixel of cover image. In the proposed technique we embed the modified secret data in LSB's of the cover image using LSB replacement steganography, but the data is not inserted sequentially. A sequence Generator (SG) is used to find the sequence in which the binary stream of secret data is to be embedded in cover image. The secret data is modified before embedding it in the cover image. After embedding secret data in cover image OPAP is used to further improve the quality of stego-image. For experimentation, standard test images are chosen and the performance of the proposed technique is compared with several other popular steganography techniques. PSNR and MSE parameters are used to evaluate the performance of the proposed technique with other techniques for same combinations of cover image and secret data image. The rest of the paper is organized as follows: Section 2 describes the proposed technique in detail along with Sequence generator and OPAP. Section 3 presents the experimental results and discussion finally paper is concluded in section 4.

1.1. BLOCK DIAGRAM

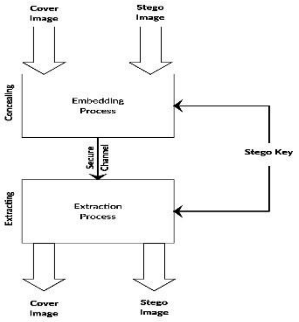


Fig – 1.1.1: Block Diagram

2. PROPOSED TECHNIQUE

As discussed in the previous section, the main aim of this technique is to find the best places in the cover image to embed modified secret data. This is done by finding a sequence which can generate maximum similarity between data to be embedded and LSB's of the cover image. For generation of sequences we use proposed algorithm, To understand the proposed technique knowledge of a few important preliminary concepts is required; these concepts will be discussed in the following sub-sections. Sequence generator is discussed in section 2.1. Section 2.2 explains optimal pixel adjustment procedure. In section 2.3. the process of embedding secret data in a cover image is explained and the process of extracting the secret data from the stego-image is explained in section 2.4.

2.1. Sequence Generator

The sequence generator algorithm is used generate a pseudo-random sequence of (X_1, X_2..., X_m) over an interval [0, M-1] as shown in equation 1. To generate the pseudo random sequence of M numbers the algorithm requires the seed value, the exponential factor the offset value and the no of bits: X_{n+1}= ((2^a+1)× (first)+c)mod(2^no. of Bits) (1)

Where X_{n+1} is the next integer value of the pseudorandom sequence, X_n is the present integer value or seed value in the initial state, a is the exponential factor, c is the offset value and the factor 2^ (no. of Bits) gives the length of the sequence. As the selection of a, c, and X_0 values affects the length of the random sequence, care is required to select all these parameters, such that we obtain the sequence of unique numbers in the given range. For example to generate a sequence of length 16, we can select the first = 16, a = 7, c = 7 and no. of bits = 4, thus we obtain the sequence as 7, 14, 5, 12, 3, 10, 1, 8, 15, 6, 13, 4, 11, 2, 9, 16.

The no. of bits used for our sequence generation is 4, to obtain sequence upto 65536 for 256×256 image and 262144 for 512×512 image.

The steps are as follows:

Step1: Declare & Initialize values of no. of bits, first, a, c and Seq (len), where len is the product of row and column matrix of the image.
Step2: Initialize seq (1) = first
Step3: Use the for loop to obtain sequence upto len

```
for k =1 to len
value=mod(((2^a)+1)*first+c,(2^no.of Bits))
first = value
 if value is 0
then value=len
end seq
(k)=value end
```

2.2. Optimal Pixel Adjustment Process

OPAP is applied to stego-image obtained after data embedding. OPAP is used to reduce the difference between the pixel values of cover image and stegoimage. Chan and Cheng [12] developed the idea of OPAP to improve the quality of stego-image after LSB steganography. The elementary concept of OPAP is as follows:

- Let pi, pi' and pi'' be the corresponding pixel values of the ith pixel in the original cover image C, the stego-image C' obtained by LSB replacement steganography and C'' the modified stego-image obtained after the OPAP.
- Let ∂i= pi' – pi be the embedding error between pi and pi' obtained after embedding k bits of secret message per pixel.
- Therefore -2k< ∂i< 2k
- The value of ∂i can be further divided into three intervals:

Interval 1: 2k-1< ∂i< 2k,
Interval 2: -2k-1 ≤ ∂i ≤ 2k-1,
Interval 3: -2k< ∂i< -2k-1

- Depending upon these three intervals OPAP modifies pi' to pi''.

Case 1: (2k-1< ∂i< 2k): If pi' ≥ 2k, then pi''= pi' - 2k; otherwise pi''= pi'
Case 2: (-2k-1 ≤ ∂i ≤ 2k-1): pi''= pi'
Case 3: (-2k< ∂i< -2k-1): If pi' < 256 - 2k,
then pi''= pi' + 2k; otherwise pi''= pi'

From the above discussion it is clear that OPAP modifies stego-image only if the embedding error ∂i is greater than 2k-1 or less than -2k-1. The proposed technique is developed for 2 bit per pixel data embedding hence OPAP will modify only those pixels whose ∂i is 3 or -3.

2.3. Embedding the Secret Data

In this subsection, the process of embedding the secret data in cover image is discussed. LSB replacement technique is used for data insertion. Two bits of secret data are embedded in each pixel of the cover image. The secret data is converted into two arrays which are one dimensional. One array has 4 MSB's of all secret data image pixels while another array has 4 LSB's. One bit of data from each array is inserted in the cover image using LSB replacement steganography. The order in which these secret data bits are inserted is dependent on the sequence generated by Sequence generator. The data insertion is carried out as per the sequence after obtaining the sequence. The algorithm for embedding secret data in cover image is given below. Input: Cover image C = {c_1, c_2, ..., $c_{(m \times n)}$}, Secret message image M = {m_1, m_2, ..., m(m/2) × (n/2)}
Output: Stego-image S = {s_1, s_2, ..., $s_{(m \times n)}$}.

1. Create two, one dimensional bit arrays sm_1 and sm_2 of length len = m × n.
2. Insert four MSB's of all pixels from M in consecutive locations of sm1 and similarly insert four LSB's of all pixels from Min consecutive locations of sm_2. sm1=$(M)_{8-5}$ sm2=$(M)_{4-1}$
3.Generate p sequences (seq) of length len using values of first=16, a=7 and c=7 and no. of Bits=4 for 256×256 embedding and 18 for 512×512 embedding.
4. Obtain Arranged data as:

```
For locate = 1: len
loc1=sm1(seq(locate));
        loc2=sm2(seq(locate));
        arrangedData1(locate)=loc1;
        arrangedData2(locate)=loc2;
                  end
```

4. Embed two bits of secret data in LSB's of cover image based on the sequence (seq).

```
count = 1 for i=1 to m  for j=1 to n  value1=testImage (
i,j);
cover_image(i,j)=cover_image(i,j)double(mod(cover_i
mage(i,j),4))+double(actualData(
c));
         c=c+1;
end
    end
count = count + 1;
```

6. Apply OPAP to stego-image obtained after step 5.

2.4. Extracting the Secret Data

The process of extracting the secret data from stego-image starts from the predefined locations of stego-image. After this next step is generating sequence used to embed the secret data, it can be done using sequence generator. After obtaining sequence each pixel is visited as per sequence generated by sequence generator and two bits of secret data are extracted from each pixel. The data extracted is stored in two separate arrays. To reconstruct the secret data image these arrays are used. Given below is the algorithm for extracting secret data from stego-image:

Input: Stego-image S = {s_1, s_1, …, $s_{(m \times n)}$}
Output: The Secret message image M = {m_1, m_2, …, m(m/2) × (n/2)}
1. Extract secret key from predefined pixels locations of stego-image.
2. Obtain the value of first, a and c from the secret key.
3. Generate a sequence of len numbers using sequence generator algorithm.
4. To extract and store secret data image, initialize two, one dimensional bit arrays of length lena's se_1 and se_2. 5. Extract two bits of data from LSB's of each pixel and store them in se_1 and se2 depending upon the value of seq.
6. Concatenate four bits of se_2 and se_1 to form one pixel of the secret image. Continue this process till m/2 × n/2 pixels are obtained.

The extraction of secret data from the stego-image is extremely difficult for eavesdroppers. To extract the secret data, first the secret key needs to be mined and further, the process of using secret key to generate sequence and extraction of data is necessary. Hence data extraction can only be done by person who has prior knowledge of the embedding technique.

3. RESULTS AND DISCUSSIONS

For experimentation, standard test images were chosen and the performance of the proposed technique was compared with several other popular steganography techniques. The performance of various techniques was evaluated using PSNR parameter for the same combinations of cover and secret data images. The resolution of cover image is 512×512 and that of secret data image is 256×256. Both the images are grey scale images. Software tool used to implement the proposed technique was Matlab 8.1. Equation 2 was used to find the MSE of stego-images, in equation 2 M and N represents number of rows and columns in the image respectively. Xij and Yij are pixel values of ijth location of original image and stego-image respectively. Equation 3 was used to find the PSNR value of stego-images.

$$MSE = \frac{1}{MN} \sum_{i=1}^{M} \times \sum_{j=1}^{N} (Xij - Yij)_2 \qquad (2)$$

$$PSNR = 10 \times \log_{10} \frac{255 \times 255}{MN} \qquad (3)$$

In Figure 3.1 all the test images used for experimentation are displayed. Lena, Jet, Pepper, Sailboat and Baboon are the images used as cover image and Figure 3.1(f) a test pattern is used as secret data image.

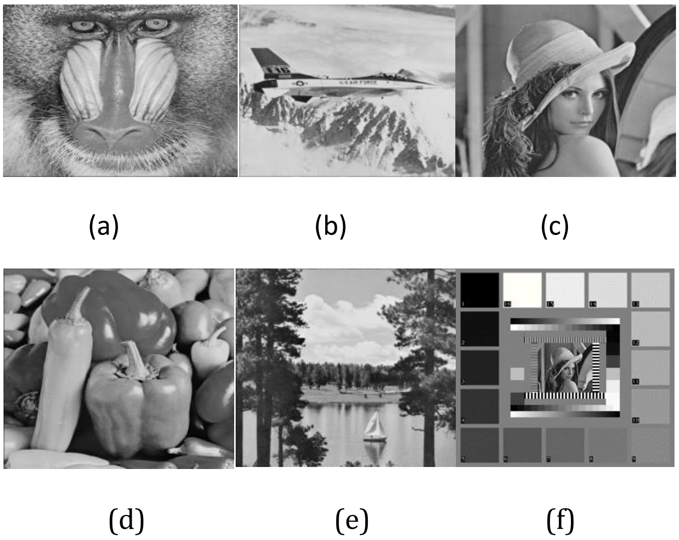

(a) (b) (c)
(d) (e) (f)

Fig – 3.1: Test images (a) - (e) Cover images (Baboon, Jet, Lena, Pepper and Sailboat). (f) - Secret message image (Test Pattern).

3.1 Imperceptibility Analysis

To measure the changes incorporated during the embedding of secret message into cover image, imperceptibility analysis is used. In imperceptibility, here the Peak signal to noise ratio (PSNR) is used to compare the cover image and the secret image. The test images used to compare the performance of the proposed technique with other popular steganography schemes are same. In imperceptibility analysis higher PSNR value ensures better performance of the steganography technique, as it guarantees a superior visual quality of stego-image. Table 1 displays PSNR values of stego-images obtained from the proposed technique and various other steganography techniques at same data embedding rate i.e. 2 bits per pixel. These results undoubtedly advocate the dominance of the proposed technique over other techniques. Hence we can state that the proposed scheme is exceedingly imperceptible. Figure 3.1.1 and 3.1.2 demonstrates the PSNR value of stegoimages obtained from the proposed and existing spatial domain steganography technique. The comparison between cover image and stego-image obtained from the proposed technique for Lena and Baboon image is shown in figure 3.1.3.

Table 3.1.1: PSNR values of the proposed technique and various other steganography methods for different stego-images.

Cover Image	Lin And Tsai [13]	Yang Et al. [14]	Chang Et al. [15]	Wu Et al. [16]	Kanan Et al. [7]	Proposed Technique
Lena	39.20	41.60	40.37	43.54	45.12	46.37
Jet	39.25	41.66	40.73	43.53	45.18	46.38
Pepper	39.17	41.56	39.30	43.56	45.13	46.37
Sailboat	39.18	41.51	38.86	43.55	45.10	46.38
Baboon	39.18	41.55	39.94	43.54	45.12	46.35

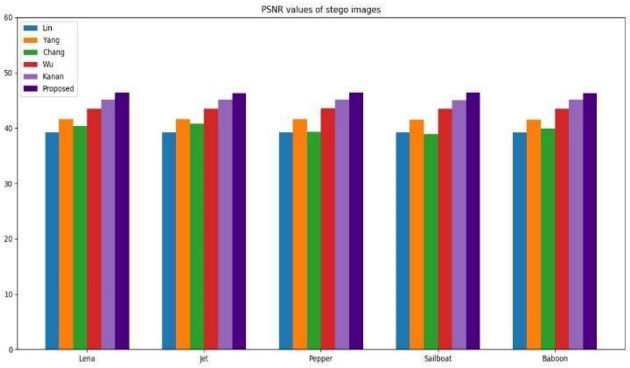
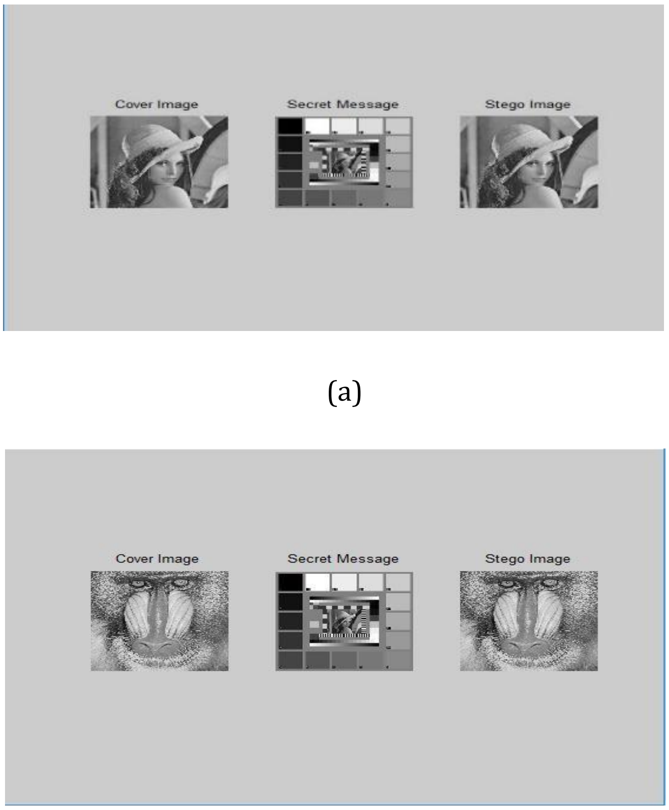


Fig. 3.1.2: PSNR values of the proposed technique and various other steganography methods for different stego-images.



(a)

(b)

Fig. 3.1.3 (a) & (b): Stego-images obtained after secret data embedding from proposed technique.

4. CONCLUSION

In this paper an imperceptible, high payload capacity, spatial domain image steganography technique is presented. The proposed scheme employs sequence generation method to find location and order to embed secret data in cover image. The results of proposed technique are compared with various other popular spatial domain steganography schemes. The superiority of the proposed technique is quite evident by the obtained results, both in subjective and objective analysis. The proposed method produces very small amount of change in stego-image, making it highly imperceptible and immensely challenging to identify the presence of steganography by visual inspection. The extraction of secret data from the stego-image is extremely difficult for eavesdroppers as it requires the knowledge.

REFERENCES

[1] Bedi P, Bansal P & Sehgal P, "Using PSO in a spatial domain based image hiding scheme with distortion tolerance", Computers & Electrical Engineering, Vol. 39, No. 1, (1013), pp. 640-654.

[2] Cheddad A, Condell J, Curran K & McKevitt P, "Digital image steganography: Survey and analysis of current methods", Signal processing, Vol. 90, No. 3, (1010), pp. 717-751.

[3] Subhedar MS &Mankar VH, "Current status and key issues in image steganography: A survey", Computer science review, Vol. 13, (1014), pp. 95-113.

[4] Li B, He J, Huang J & Shi, YQ, "A survey on image steganography and steganalysis", Journal of Information Hiding and Multimedia Signal Processing, Vol. 1, No. 1, (1011), pp. 141-171.

[5] Kanan HR &Bahram N, "A novel image steganography scheme with high embedding capacity and tunable visual image quality based on a genetic algorithm", Expert Systems with Applications, Vol. 41, No. 14, (1014), pp. 6113-6130.

[6] Wang S, Yang B &Niu X., "A secure steganography method based on genetic algorithm" Journal of Information Hiding and Multimedia Signal Processing, Vol. 1, No. 1, (1010), pp. 18-35.

[7] Maheswari SU &Hemanth DJ, "Performance enhanced image steganography systems using transforms and optimization techniques", Multimedia Tools and Applications, Vol. 76, No. 1, (1017), pp. 415436.

[8] Fontaine C, "Linear congruential generator", Encyclopedia of Cryptography and Security, Springer, Boston, MA, (1011), pp. 711-711.

[9] Shah PD &Bichkar RS, "A Secure Spatial Domain Image Steganography Using Genetic Algorithm and Linear Congruential Generator", International Conference on Intelligent Computing and

Applications, Advances in Intelligent Systems and Computing, Springer, Singapore, (1018), pp. 119-119.

[10] Al-Dmour H & Al-Ani A, "A steganography embedding method based on edge identification and XOR coding", Expert systems with Applications, Vol. 46, (1016), pp. 193-306.

[11] Al-Dmour H & Al-Ani A, "A steganography embedding method based on edge identification and XOR coding", Expert systems with Applications, Vol. 46, (1016), pp. 193-306.

[12] Subhedar MS &Mankar VH, "Current status and key issues in image steganography: A survey", Computer science review, Vol. 13, (1014), pp. 95-113.

[13] Lin C & Tsai W, "Secret image sharing with steganography and authentication", Journal of Systems and software, Vol. 73, No. 3, (2004), pp. 405414.

[14] Yang C, Chen T, Yu KH & Wang C, "Improvements of image sharing with steganography and authentication", Journal of Systems and software, Vol.

80, No. 7, (2007), pp.1070-1076.

[15] Chang C, Hsieh Y & Lin C, "Sharing secrets in stegoimages with authentication", Pattern Recognition, Vol. 41, No. 10, (2008), pp. 3130-3137.

[16] Wu C, Kao S & Hwang S, "A high quality image sharing with steganography and adaptive authentication scheme", Journal of Systems and Software, Vol. 84, No. 12, (2011), pp. 2196-2207.