Hindawi
Journal of Computer Networks and Communications
Volume 2018, Article ID 9475142, 10 pages
https://doi.org/10.1155/2018/9475142

Research Article

A New Method of Image Steganography Using 7th Bit of a Pixel as Indicator by Introducing the Successive Temporary Pixel in the Gray Scale Image

Kamaldeep Joshi, Swati Gill, and Rajkumar Yadav

Department of Computer Science and Engineering, University Institute of Engineering and Technology, Maharshi Dayanand University, Rohtak, India

Correspondence should be addressed to Kamaldeep Joshi; kamalmintwal@gmail.com

Received 25 December 2017; Accepted 11 April 2018; Published 1 August 2018

Academic Editor: Nam Tuan Nguyen

As the internet has become the medium for transferring the sensitive information, the security of the transferred message has become the utmost priority. Image steganography has emerged out as the eminent tool of information hiding that ensures the security of the transmitted data. Image files provide high capacity, and their frequency of availability over the internet is also high. In this paper, a method of image coding is proposed that hides the information along a selected pixel and on the next value of the selected pixel, that is, pixel + 1. One bit is hidden at the selected pixel, and the second bit is hidden on the pixel +1 value. On the basis of the 7th bit of the pixels of an image, a mathematical function is applied at the 7th bit of the pixels, which generates a temporary variable (pixel + 1). The 7th bit of the selected pixel and 7th bit of pixel + 1 are used for information hiding and extraction. On the basis of a combination of these two values, two bits of the message can be hidden on each pixel. After implementation, the efficiency of the method is checked on the basis of parameters like PSNR and MSE, and then comparison with some already proposed techniques was done. This proposed image steganography showed interesting, promising results when compared with other existing techniques.

1. Introduction

Internet has emerged as the most convenient and efficient medium for communication. Through internet, messages can be transferred in a fast and cheap way in various fields like government offices, private sector, military, and medical areas [1]. Many times, confidentiality of the transferred message needs to be maintained. To ensure that the message is transferred securely and safely over the network, a suitable method is needed. Steganography proves as a trustable method for achieving this aim. In steganography, the data are hidden in the cover media. The cover medium can be in the form of image file, text file, video file, or audio file. Steganography is defined as a science or art of hiding the message inside some cover medium [2, 3]. The word steganography is built up of two words of ancient Greek origin "steganos" meaning "covered, concealed, or protected" and "graphie" meaning "writing." The concept of steganography is not new; its usage can be seen in the past also. Historical records depict that around 440 BC, Herodotus sent secret messages using the concept of steganography. In ancient times, Greeks also wrote messages on wood and covered them with wax. The concept of invisible ink was also used during the period of World War II. According to Greek history, secret messages were written on the bald scalp of the slaves, and after the growth of hair on their heads, they were sent as messengers.

The most popular medium used is image files because of their high capacity and easy availability over the internet [4]. At the sender's side, the image used for embedding the secret message is called cover image, and the secret information that needs to be protected is called a message. As soon as data

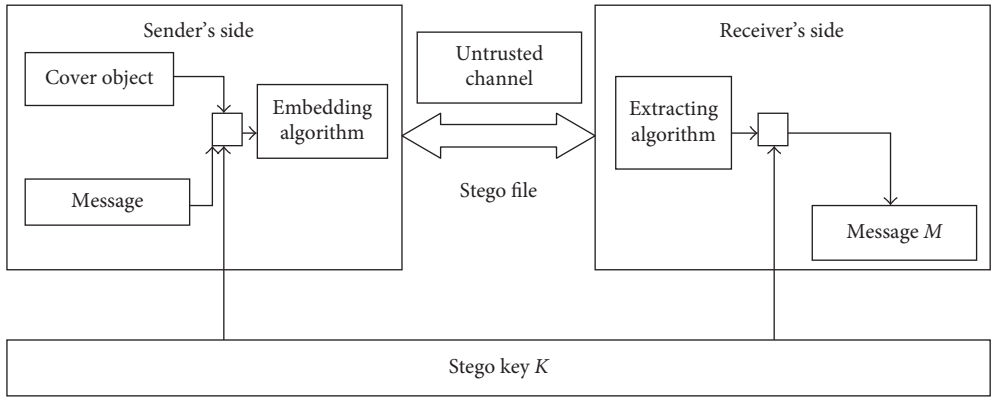

Figure 1: Steganography system.

are embedded using some appropriate embedding algorithm, then it is called stego image. This stego image is transferred to the receiver, and he extracts out the secret message using extraction algorithm [5]. Another data hiding technique, cryptography is also used for secure transmission of messages over the internet, but steganography is becoming more popular because of its advantages over cryptography. Cryptography hides the exact meaning of message from the third party whereas steganography hides the very existence of the message itself. Figure 1 shows the Steganography System.

2. Literature Review

Steganography has emerged as a glowing research area in which various methods have been proposed in several carrier media. Here, we are going to present the brief overview of some already proposed methods, and at last, we will be presenting the comparison work with these methods.

LSB method [6] provides the very basic idea of steganography in an easy manner. This method states that the secret message bits can be placed by replacing the least significant bits of the pixels of the image. It allows 100% insertion of message binary bits in the pixels of an image with a very minute change of +1 or −1 in the value of the pixels [7]. This method was vulnerable to attack as the message was present at LSB, and by only picking LSBs, the intruder can access the data. Quantization noise can also destroy the data present on LSB [8]. So, this method can be easily decoded by the intruder and is also not immune to the noise and compression techniques. Also, this method allows only single bit insertion of message data inside the particular pixel. Let us try to understand with the help of the following example.

Suppose the message string to be sent over the internet is 10010101, and the value of continuous pixels are as follows:

01101000 10101001 01101000 11110000 00011101 10000001 11110000 10101010.

Then after inserting the message, pixels would be as follows:

01101001 10101000 01101000 11110001 00011100 10000001 11110000 10101011.

This method was vulnerable to attack as the message was present at LSB, and by only picking LSBs, the intruder can access the data. Singh et al. proposed a method based on first and second bit plane. In this method, on the combination of 1st and 2nd bit plane the message was hidden. The main result of the method was that the probability of message insertion at a pseudorandom location at first chance was 50%. The probability was 50% when there was no need to change the pixel value. The probability was 12.5% when a change in pixel value was required. Batra and Rishi [9] proposed a method in which the message was hidden using the 6th, 7th, and 8th bits of a pixel in a grayscale image. This method overcomes the limitation of the Singh et al.'s method. The main result of the method was that the probability of message insertion at a pseudorandom location at first chance is 85.93%. The probability when the message was not changed was 43.18%. As the result shows, this method does not provide 100% message insertion rate. In FMM (Five Modulus method) [10], the cover image was divided into N blocks with block size $k*k$ pixels where k is the size of the window. Each pixel in these blocks was modified such that the pixel of the block is divisible by 5. The beauty of this method was that the message was scattered over the entire image. The limitation of this method is the hiding capacity that is low. In some cases, the average message hidden capacity is below 1 bit per pixel. Bailey and Curran [11] have presented the facts of the Stego Color Cycle (SCC) method. This is the advancing method of LSB. Here also LSB of pixels of color images is used for insertion of secret message binary bits. Insertion is done in a cyclic way by choosing the LSB of the red channel of first pixel and then LSB of the green channel of second pixel and then LSB of the blue channel of third pixel, and this cycle repeats in same cyclic order for all the pixels. This method also allows 100% insertion for RGB images, but for its simple cyclic order, it can be easily decoded by the intruder. Further, some more techniques are proposed that remove the fallacies of this method to some extent. Gutub [12] presented the pixel indicator method. This method is applied to RGB images in which two channels of the image are used for storing the data on the basis of the value of the third channel that acts as the indicator channel.

Table 1: Pixel indicator method.

Indicator channel	One channel	Second channel
00	No message	No message
01	No message	2 bits of message
10	2 bits of message	No message
11	2 bits of message	2 bits of message

A sequential order is used for choosing the indicator channel, that is, RGB, RBG, GBR, GRB, BRG, and BGR. Table 1 shows the PIT technique.

This method provides high-capacity data insertion; 2 bits and 4 bits of secret message can be hidden inside single pixel. Along with this, it provides high capability against decoding of the message by the intruder. The disadvantage associated with this method is that it does not provide 100% insertion as one channel is utilized for the indicator. Wu and Tsai [13] have proposed pixel value difference method. In this technique, the cover image is bifurcated into nonoverlapping blocks. Two consecutive pixels are placed inside each block. The difference of the consecutive pixels of each block is calculated, and this difference value is found to be small in smooth areas of the image whereas its value is large in edge areas. Using this technique, large amount of data can be placed in edge areas as compared to smooth areas. The amount of secret message bits that can be embedded depends upon the range of difference value; range is always taken in powers of 2 because the message has to be inserted in binary form. This method provides high embedding capacity and perceptibility but the stego image can be more distorted if the difference in pixel values is more.

LSB-S method [14] also provides two layers of security. First layer provides cryptographic security, and second method uses steganographic security. Joshi et al. [15, 16] proposed two methods based on XOR operation. The first method used two bits of the cover media, and the second one used three bits of the cover media. The author claimed 100% chances of message insertion.

3. Proposed Work

The aim of the proposed scheme is to develop a secure and robust technique for transfer of messages so that private and important information can be sent over the network in a protected manner without being vulnerable to any kind of attacks by an unintended recipient. The proposed method works on gray images. A mathematical function is applied to the 7th bit of the pixels. The 7th bits of the selected pixel and pixel + 1 value are extracted, and on the basis of a combination of these two values, 2 bits of the message can be extracted from each pixel. There can be four possible combinations 00, 01, 10, and 11. This method provides various advantages such as two bits of message storage in each pixel and nondependency of the technique on the 8th bit. There can be a change of +2 and −2 at maximum in the pixel value while inserting the data in the image file. This method helps in tackling the limitations of steganography to a higher extent.

4. Proposed Algorithm

Let the length of the message is known to sender as well as the receiver. Let I be the cover image of $R*C$ pixels, S be the N-bit secret message, x be the pixel value of I, and s be the bit of secret message, then the image matrix can be represented by (1), and S can be represented by (2).

$$I = \left\{ x_{ij} \middle| 1 <= i <= R,\ 1 <= j <= C,\ x_{ij} \in \{0,\ 1, \ldots,\ 255\} \right\}, \tag{1}$$

$$S = \left\{ s_N \middle| 1 <= N <= n,\ s_N \in \{0,\ 1\} \right\}. \tag{2}$$

Let S is the message to be hidden, Y is the cover media, K is the stego key used for insertion and retrieval of the message, E and D are the insertion and retrieval algorithms, respectively, and Y' is the stego file.

The insertion process may be given by the following equation:

$$Y' = E_K(S,\ Y). \tag{3}$$

The insertion algorithm is given in Pseudocode 1. At the other end, the reverse process is carried out and the message is extracted using the algorithm in Pseudocode 2. The message is separated from the cover image by (4). The retrieval process may be given by the following equation:

$$X = D_K(Y'). \tag{4}$$

The data can be inserted by the method given as follows.

5. Example of the Proposed Method

The proposed method can be explained with a suitable example. Let us assume that the secret message to be embedded is $m = \{01110110\}$, and the four pixel values being selected are $p = \{72, 95, 86, 58\}$.

5.1. At Sender's Side

$$\begin{aligned} P_1 &= 72\,(010010\ \mathbf{0}\ 0), \\ P_1 + 1 &= 73\,(010010\ \mathbf{0}\ 1). \end{aligned} \tag{5}$$

The 7th bits of P_1 and $P_1 + 1$ form the pair "00" but initial two message bits to be inserted are "01." Therefore, we need to add +1 to the value of P_1. Hence,

$$P_1' = \mathbf{73}\,(72 + 1), \tag{6}$$

where P_1' is the stego pixel.

Now, add +1 to the second pixel, that is,

$$\begin{aligned} P_2 &= 86\,(010101\ \mathbf{1}\ 0), \\ P_2 + 1 &= 87\,(010101\ \mathbf{1}\ 1). \end{aligned} \tag{7}$$

The 7th bits of P_2 and $P_2 + 1$ combine to form the pair "11" and the 3rd and 4th message bits are also "11." Hence value of

$$P_2' = P_2 = \mathbf{86}. \tag{8}$$

```
(1) a1 = enter the message;
(2) N = length (a1) *8;                          %Length calculates the message length in bits
(3) Binarystring = dec2bin (a1, 8);              %Message in bits
(4) I = Read image;                              %Insert Image in I variable
(5) [r, c] = size (I);                           %r and c give the row and column of the image
(6) X = zeros (r, c);                            %Initialize a temporary matrix, X, to ZERO
(7) Let K = 1;                                   %Initialize a variable K
(8) Let a = 1;                                   %Initialize a variable a
(9) Repeat L = 1: r                              %Repeat L = 1 to r
(10)   Repeat m = 1: c                             %Repeat L = 1 to c
(11)     A = get 2nd bit of I, (I (L, m), 2);      %A is the 2nd bit of I
(12)     Q = I (L, m) +1;                          %Q is the next pixel of the image I
(13)       B = get 2nd bit of Q, (Q, 2);           %B is the 2nd bit of Q
(14)         If (K < N)
(15)           M1 = binarystring (a);              %M1 is the first message bit
(16)           M2 = binarystring (a + 1);          %M2 is the second message bit
(17)           if ((A==0) && (B==0))
(18)             if ((M1==0) && (M2==0))
(19)               X (L, m) = I (L, m);            %No change in pixel value and drop I in X
(20)             end
(21)             if ((M1==0) && (M2==1))
(22)               X (L, m) = I (L, m) +1;         %Pixel value is incremented by 1
(23)             end
(24)             if ((M1==1) && (M2==0))
(25)               X (L, m) = I (L, m) −1;         %Pixel value is decremented by 1
(26)             end
(27)             if ((M1==1) && (M2==1))
(28)               X (L, m) = I (L, m) +2;         %Pixel value is incremented by 1
(29)             end
(30)           end
(31)           if ((A==0) && (B==1))
(32)             if ((M1==0) && (M2==0))
(33)             X (L, m) = I (L, m) −1;
(34)             end
(35)             if ((M1==0) && (M2==1))
(36)             X (L, m) = I (L, m);
(37)             end
(38)             if ((M1==1) && (M2==0))
(39)           X (L, m) = I (L, m) +2;
(40)             end
(41)             if ((M1==1) && (M2==1))
(42)           X (L, m) = I (L, m) +1;
(43)             end
(44)           end
(45)             if ((A==1) && (B==0))
(46)           if ((M1==0) && (M2==0))
(47)             X (L, m) = I (L, m) +1;
(48)             end
(49)             if ((M1==0) && (M2==1))
(50)               X (L, m)=I (L, m) +2;
(51)             end
(52)             if ((M1==1) && (M2==0))
(53)              X (L, m) = I (L, m);
(54)             end
(55)               if ((M1==1) && (M2==1))
(56)               X (L, m) = I (L, m) −1;
(57)               end
(58)           end
(59)             if ((A==1) && (B==1))
(60)               if ((M1==0) && (M2==0))
(61)                 X (L, m) = I (L, m) +2;
(62)               end
```

Pseudocode 1: Continued.

```
(63)                    if ((M1==0) && (M2==1))
(64)                       X (L, m) = I (L, m) −1;
(65)                    end
(66)                    if ((M1==1) && (M2==0))
(67)                       X (L, m) = I (L, m) +1;
(68)                    end
(69)                    if ((M1==1) && (M2==1))
(70)                       X (L, m) = I (L, m);
(71)                    end
(72)                 end
(73)                 K = K + 2;
(74)                 a = a + 2;
(75)              else
(76)                 X (L, m) = I (L, m);
(77)              end
(78)           end
(79) end
(80) Convert X matrix to an image, that is, stego.tif
```

Pseudocode 1: Insertion of the Message.

```
(1) S = Read stego.tif image                    %Read the stego image in S matrix
(2) [row, col] = size (S);                      %Find row and column of the stego image
(3) load N;                                     %Load the message length
(4) k = 1;                                      %Initialize the value of k by 1
(5) m = 0;                                      %Initialize m matrix to zero
(6) Repeat i = 1: row
(7)    Repeat j = 1: col
(8)       if k < N
(9)          m (k) = 2nd bit of S, i.e. (S (i, j),2);
(10)         m (k + 1) = 2nd bit of the next pixel of S, i.e.(S (i, j) +1, 2);
(11)         k = k + 2;
(12)      end
(13)   end
(14) end
(15) m;
(16) M = reshape (m,8,[]);
(17) Convert M into a character stream;
```

Pseudocode 2: Retrieval of the Message.

Now, third pixel value, that is

$$P_3 = 95\,(010111\ \mathbf{1}\ 1),$$
$$P_3 + 1 = 96\,(011000\ \mathbf{0}\ 0). \tag{9}$$

The 7th bits of P_3 and $P_3 + 1$ combine to form the pair "10" but the 5th and 6th message bits are "01" Hence, we need to add +2 to the pixel value.

$$P_3' = P_3 + 2 = 95 + 2 = \mathbf{97}. \tag{10}$$

Now, third pixel is

$$P_4 = 58\,(001110\ \mathbf{1}\ 0),$$
$$P_4 + 1 = 59\,(001110\ \mathbf{1}\ 1). \tag{11}$$

The 7th bits of P_4 and $P_4 + 1$ combine to form the pair "11," but the 7th and 8th message bits to be inserted are "10," Therefore, +1 is needed to be done.

$$P_4' = P_4 + 1 = \mathbf{59}. \tag{12}$$

Now, the value of pixels in the stego image that is transferred over the internet is

$$P' = \{73,\ 86,\ 97,\ 59\}. \tag{13}$$

5.2. At Receiver's Side. The set of selected pixels is {73, 86, 97, 59}.

Now,

$$P_1' = 73\,(010010\ \mathbf{0}\ 1),$$
$$P_1' + 1 = 74\,(010010\ \mathbf{1}\ 0). \tag{14}$$

The 7th bits combined to form the message bit "**01**."

Now, the second pixel value is

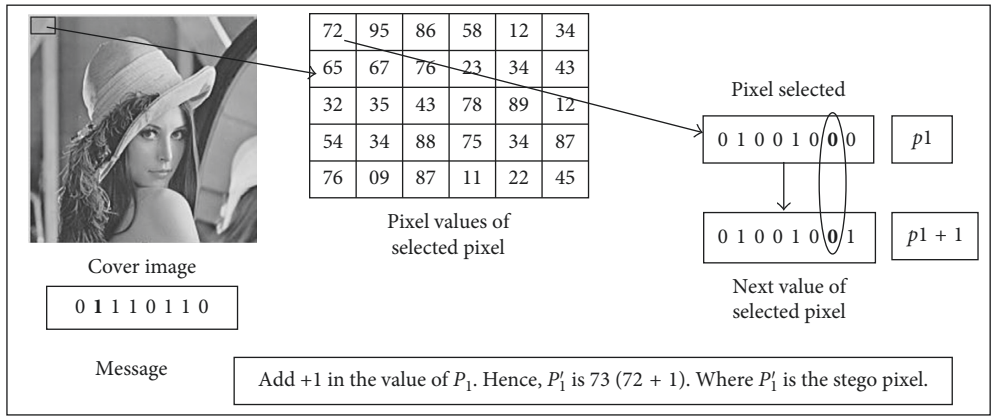

Figure 2: Diagram of the proposed method.

Table 2: Performance analysis of the proposed method using (256∗256) images.

Image name	Message size	PSNR	MSE
Lena	Two kilobytes	55.4015	0.1875
Baboon	Two kilobytes	55.4189	0.1867
Home	Two kilobytes	55.4771	0.1842
Girl	Two kilobytes	55.3071	0.1916
Clock	Two kilobytes	55.3331	0.1904
Cameraman	Two kilobytes	55.4602	0.1850

Table 3: Performance analysis of the proposed method using (256∗256) images.

Image name	Message size	PSNR	MSE
Lena	Four kilobytes	52.4009	0.3741
Baboon	Four kilobytes	52.3961	0.3745
Home	Four kilobytes	52.4091	0.3734
Girl	Four kilobytes	52.3211	0.3810
Clock	Four kilobytes	52.3882	0.3752
Cameraman	Four kilobytes	52.3926	0.3748

$$P_2' = 86\,(010101\ \mathbf{1}\ 0),$$
$$P_2' + 1 = 87\,(010101\ \mathbf{1}\ 1). \tag{15}$$

The 7th bits combine to form the pair of message bits "**11**."

Now, the third pixel value is

$$P_3' = 97\,(011000\ \mathbf{0}\ 1),$$
$$P_3' + 1 = 98\,(011000\ \mathbf{1}\ 0). \tag{16}$$

The 7th bits combined to form the message bit "**01**."

Fourth pixel value is

$$P_4' = 59\,(001110\ \mathbf{1}\ 1),$$
$$P_4' + 1 = 60\,(001111\ \mathbf{0}\ 0). \tag{17}$$

The 7th bits combined to form the message bit "**10**."

Hence, the received message stream at the receiver's side is {01110110}.

The diagrammatic representation of the proposed method is shown in Figure 2.

Table 4: Performance analysis of the proposed method using (256∗256) images.

Image name	Message size	PSNR	MSE
Lena	Six kilobytes	50.6416	0.5609
Baboon	Six kilobytes	50.6195	0.5638
Home	Six kilobytes	50.6268	0.5629
Girl	Six kilobytes	50.5525	0.5726
Clock	Six kilobytes	50.6094	0.5651
Cameraman	Six kilobytes	50.6463	0.5603

Table 5: Performance analysis of the proposed method using (256∗256) images.

Image name	Message size	PSNR	MSE
Lena	Eight kilobytes	49.3701	0.7517
Baboon	Eight kilobytes	49.3804	0.7500
Home	Eight kilobytes	49.3446	0.7562
Girl	Eight kilobytes	49.2966	0.7646
Clock	Eight kilobytes	49.3483	0.7555
Cameraman	Eight kilobytes	49.3941	0.7476

6. Experimental Results and Analysis

The efficiency is checked on the basis of two parameters, that is, PSNR [17] (peak signal to noise ratio) and MSE (mean square error). Obtained values show the high efficiency of the proposed method:

$$\mathrm{MSE} = \frac{1}{R \times C} \sum_{i=1}^{R} \sum_{j=1}^{C} \left(x_{ij} - x_{ij}'\right)^2, \tag{18}$$

where R and C represent the dimensions of the image matrix, x_{ij} represents the original image, and x_{ij}' represents the stego image.

$$\mathrm{PSNR} = 10 \log_{10}\left[\frac{I^2}{\mathrm{MSE}}\right] (\mathrm{dB}), \tag{19}$$

where I represents the maximum possible value of the pixel in an image. PSNR is measured in decibel.

The results of the proposed method with different images and different message sizes are given in Tables 2–6.

Table 6: Performance analysis of the proposed method using (256∗256) images.

Image name	Message size	PSNR	MSE
Lena	Ten kilobytes	48.4066	0.9385
Baboon	Ten kilobytes	48.4114	0.9374
Home	Ten kilobytes	48.3779	0.9447
Girl	Ten kilobytes	48.3404	0.9529
Clock	Ten kilobytes	48.3666	0.9472
Cameraman	Ten kilobytes	48.4299	0.9344

Table 7: Comparison of the proposed method with other techniques based on PSNR by hiding 8 KB of data in images of resolution (256∗256).

Image name	Classic LSB method [18]	SCC method [18]	PIT [18]	FMM [18]	CST [18]	Proposed method
Lena	42.51	42.60	42.30	43.57	55.92	49.37
Baboon	54.73	47.97	46.89	44.55	48.95	49.38
House	52.04	52.89	51.07	67.55	51.17	49.35
Couple	48.40	47.91	46.58	46.25	55.91	49.32
Trees	56.27	49.76	48.60	46.12	38.54	49.29
Moon	56.02	47.26	46.39	45.82	47.49	49.33
Average of 150 images	**45.28**	**41.83**	**41.22**	**41.97**	**37.38**	**49.34**

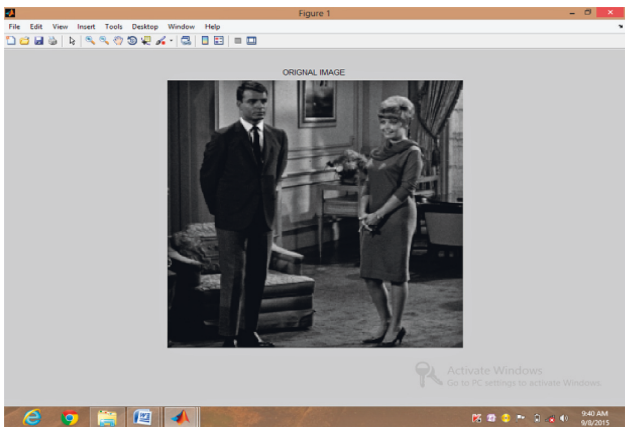

Figure 3: Original image of couple.

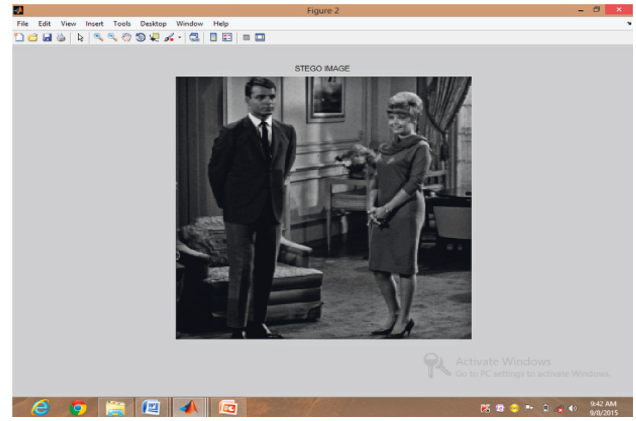

Figure 4: Stego image of couple.

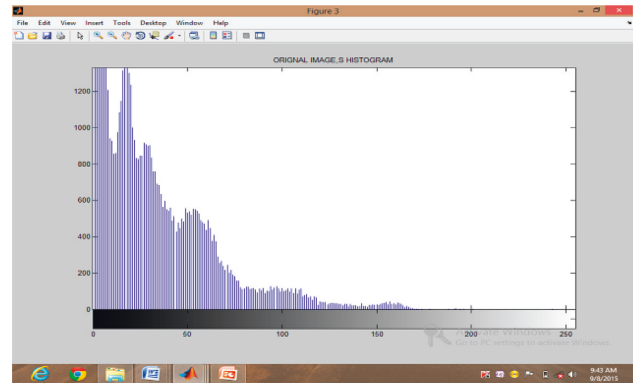

Figure 5: Histogram of original image of couple.

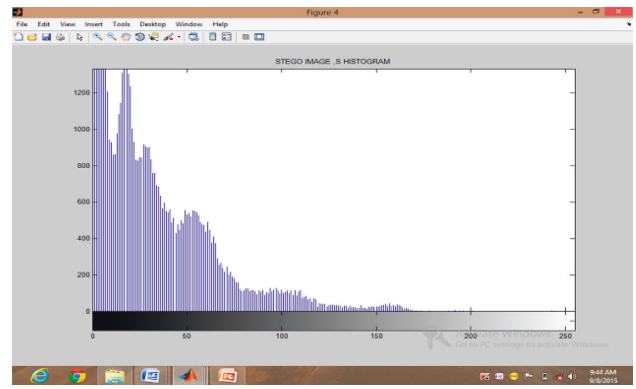

Figure 6: Histogram of stego image of couple.

7. Comparison of the Proposed Method with Other Techniques

The PSNR value of the proposed method is compared with different techniques, and the corresponding results are depicted in Table 7. 8 KB message data are taken in binary form and applied to the standard images of resolution (256∗256). These images are taken from USC-SIPI-ID dataset. Table 7 consists of the result of PSNR values of different techniques when applied on different images; these values of PSNR for the other method are taken from [17].

The LSB method can be easily hacked down. The Stego Color Cycle method is only an extension of the LSB method that is applied to the three-color components of the image. PIT method does not allow insertion of data at each pixel. The proposed method tries to overcome the drawbacks of all these methods.

8. Histogram Results for the Images

Histogram results of few images are depicted by applying the proposed method using a 2 KB length of the message. The original image and stego image along with their corresponding histograms are shown in Figures 3–22.

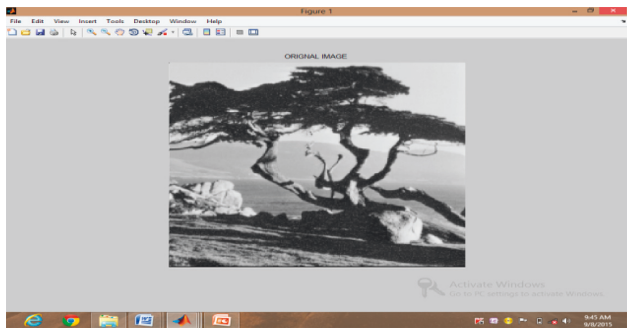

Figure 7: Original image of trees.

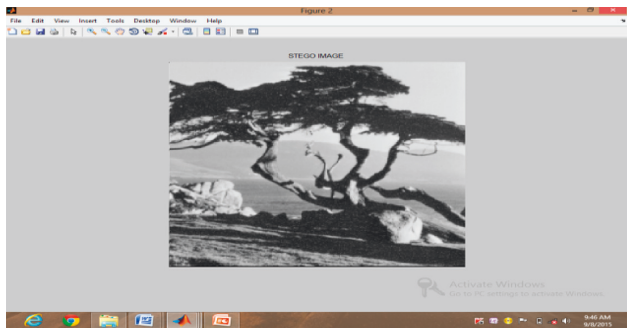

Figure 8: Stego image of trees.

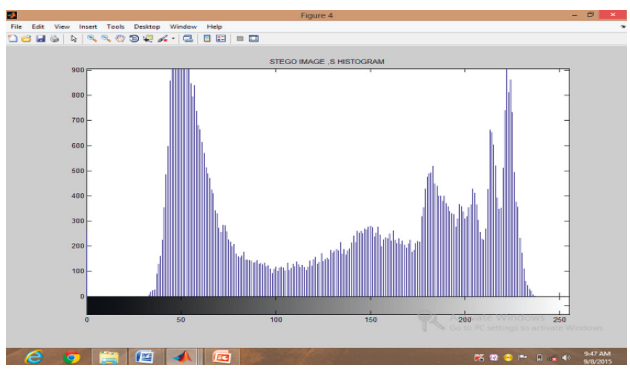

Figure 9: Histogram of original image of trees.

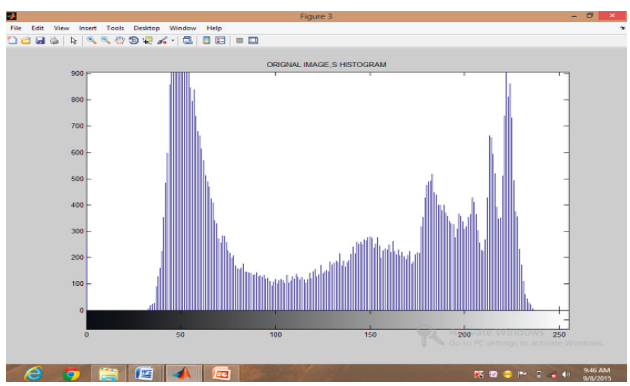

Figure 10: Histogram of stego image of trees.

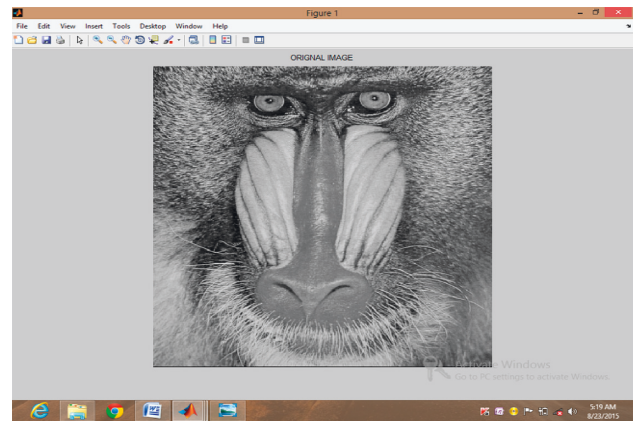

Figure 11: Original image of baboon.

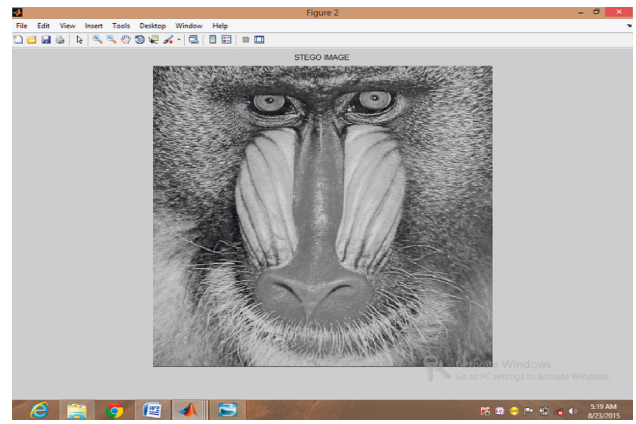

Figure 12: Stego image of baboon.

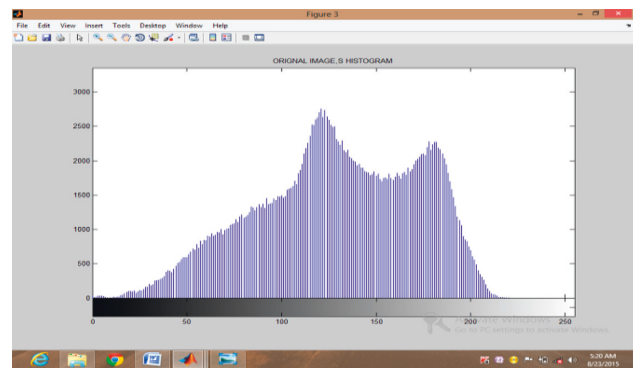

Figure 13: Histogram of original image of baboon.

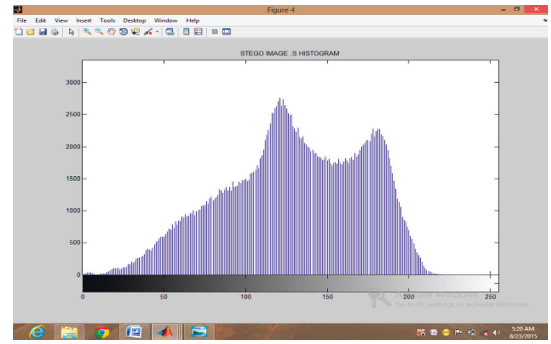

Figure 14: Histogram of stego image of baboon.

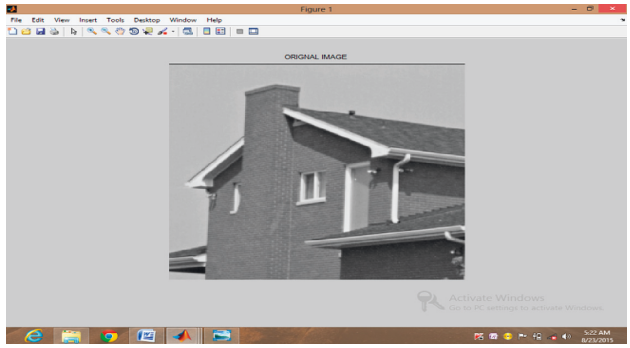

Figure 15: Original image of house.

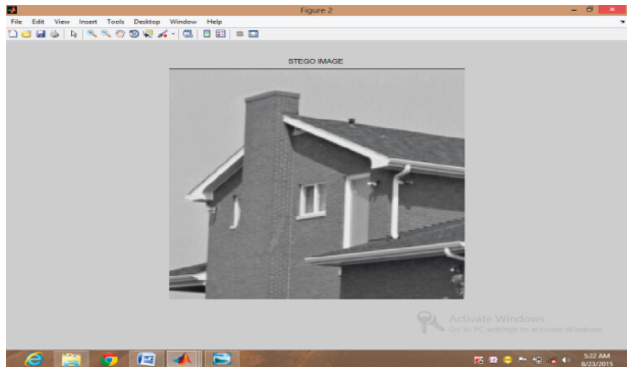

Figure 16: Stego image of house.

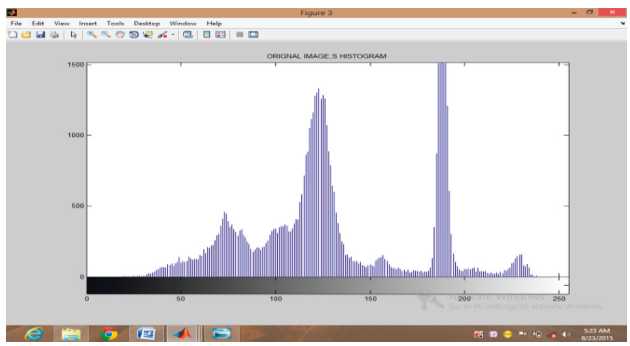

Figure 17: Histogram of original image of house.

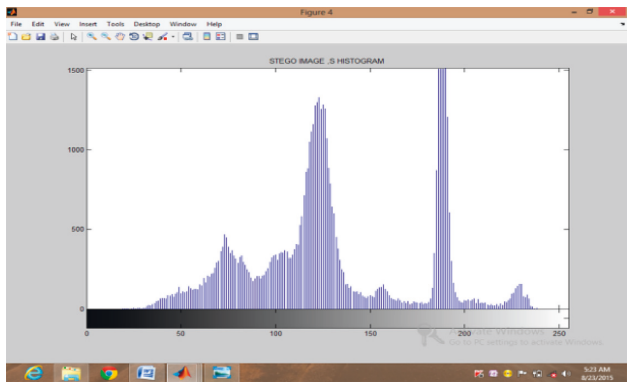

Figure 18: Histogram of stego image of house.

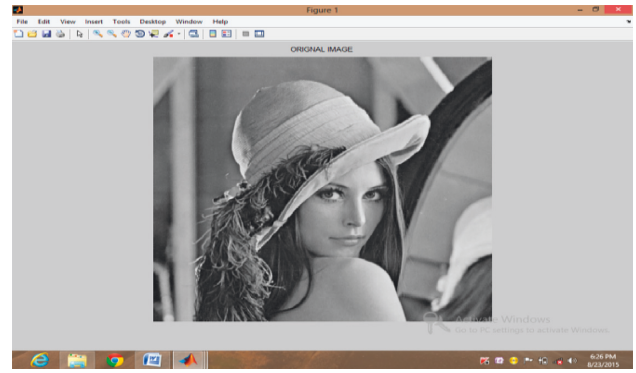

Figure 19: Original image of Lena.

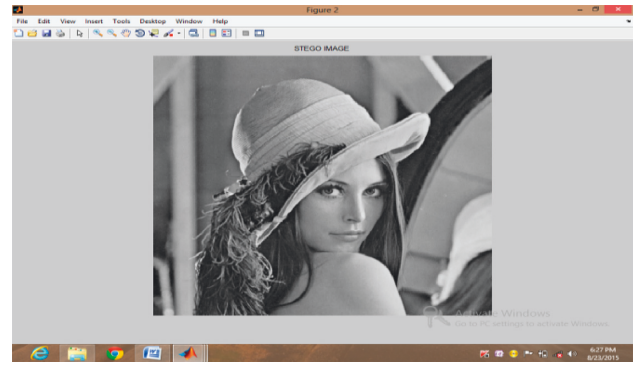

Figure 20: Stego image of Lena.

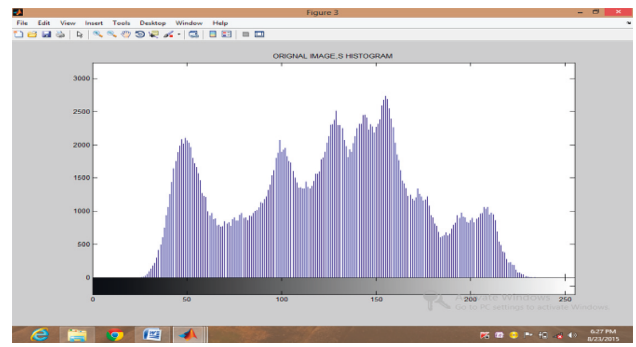

Figure 21: Histogram of original image of Lena.

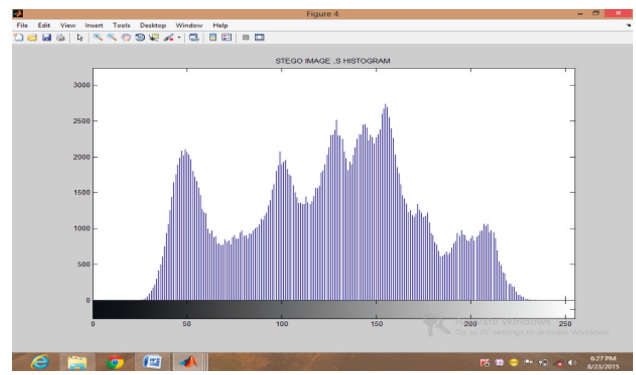

Figure 22: Histogram of stego image of Lena.

9. Conclusion and Future Scope

The above discussed steganographic method allows high capacity of data to be hidden inside the gray carrier image. Each pixel stores two bits of message bit inside the pixel, whereas other methods like LSB allow only one bit of message hiding inside every pixel. Our method does not entertain its dependency over the 8th bit as that is found in the case of LSB method. Another advantage that this method comes up is 100% insertion of data inside the selected pixel, whereas methods like "6th, 7th bit" allow only 50% insertion approximately. A very simple mathematical function of comparing the bits is used. One of the major requirements of steganography is to send the secret message inside the carrier image without creating much difference to the original image. Our technique also fulfills this requirement up to a higher extent. A maximum change of +2 or −2 is entertained while transferring the stego image. In the same manner, message can be extracted at the receiver side by using the same method on the stego image. One of the major demands of the good steganography method is to provide good PSNR and MSE values. Our method provides high PSNR and low MSE values when compared with other methods.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

[1] V. Potdar and E. Chang, "Gray level modification steganography for secret communication," in *Proceedings of the IEEE International Conference on Industrial Informatics*, Berlin, Germany, 2004.

[2] K. H. Jung, "Dual image based reversible data hiding method using neighboring pixel value differencing," *Imaging Science Journal*, vol. 63, no. 7, pp. 398–407, 2015.

[3] S. Atawneh and P. Sumari, "Hybrid and blind steganographic method for digital images based on DWT and chaotic map," *Journal of Communications*, vol. 8, no. 11, pp. 690–699, 2013.

[4] W. Bender, "Techniques for data hiding," *IBM Systems Journal*, vol. 35, no. 3-4, pp. 313–336, 1996.

[5] F. A. P. Petitcolas, R. J. Anderson, and M. G. Kuhn, "Information hiding: a survey," *Proceedings of the IEEE*, vol. 87, no. 7, pp. 1062–1078, 1999.

[6] N. F. Johnson and S. Jajodia, "Exploring steganography: seeing the unseen," *IEEE Computer*, vol. 31, no. 2, pp. 26–34, 1998.

[7] R. J. Anderson, "Stretching the limit of steganography in information hiding," *Springer Lecture Notes in Computer Science*, vol. 1174, pp. 39–48, 1996.

[8] P. Singh, S. Batra, and H. R. Sharma, "Evaluating the performance of message hidden in first and second bit plane," *WSEAS Transaction on Information Science and Technology*, vol. 2, no. 8, pp. 1220–1222, 2005.

[9] S. Batra and R. Rishi, "Insertion of message in 6th, 7th and 8th bit of pixel values and its retrieval in case intruder changes the least significant bit of image pixels," *International Journal of Security and Its Applications*, vol. 4, no. 3, pp. 1–10, 2010.

[10] F. A. Jassim, "A novel steganography algorithm for hiding text in image using five modulus method," 2013, https://arxiv.org/abs/1307.0642.

[11] K. Bailey and K. Curran, "An evaluation of image based steganography methods," *Multimedia Tools and Applications*, vol. 30, no. 1, pp. 55–88, 2006.

[12] A. A.-A. Gutub, "Pixel indicator technique for RGB image steganography," *Journal of Emerging Technologies in Web Intelligence*, vol. 2, no. 1, 2010.

[13] D. C. Wu and W. H. Tsai, "A steganographic method for images by pixel-value differencing," *Pattern Recognition Letters*, vol. 24, no. 9-10, pp. 1613–1626, 2003.

[14] K. Joshi and R. Yadav, "A new LSB-S image steganography method blend with cryptography for secret communication," in *Proceedings of the Third International Conference on Image Information Processing (ICIIP)*, pp. 86–90, Waknaghat, India, December 2015.

[15] K. Joshi, P. Dhankhar, and R. Yadav, "A new image steganography method in spatial domain using XOR," in *Proceedings of the Annual IEEE India Conference (INDICON)*, pp. 1–6, IEEE, New Delhi, India, December 2015.

[16] K. Joshi and R. Yadav, "New approach toward data hiding using XOR for image steganography," in *Proceedings of the Ninth International Conference on Contemporary Computing (IC3)*, pp. 1–6, IEEE, Noida, India, August 2016.

[17] K. Joshi, R. Yadav, and S. Allwadhi, "PSNR and MSE based investigation of LSB," in *Proceedings of the International Conference on Computational Techniques in Information and Communication Technologies (ICCTICT)*, pp. 280–285, IEEE, New Delhi, India, March 2016.

[18] M. Khan, S. Muhammad, M. Irfan, R. Seungmin, and B. W. Sung, *A Novel Magic LSB Substitution Method (M-LSB-SM) Using Multilevel Encryption and Achromatic Component of an Image*, Springer, Berlin, Germany, 2015.