An Improved 2LSB Steganography Technique for Data Transmission

Mr. Dipak U. Chaudhari[1], Dr. Sahebrao B. Bagal[2]

[1]M.E. (E & Tc), Late G.N. Sapkal, College of Engineering, Nashik, Maharashtra, India
[2]Principal and Guide, Late G.N. Sapkal, College of Engineering, Nashik, Maharashtra, India

Abstract - *In this paper message bits are embedded randomly to achieve the higher security by combining cryptography and steganography. Cryptography protects the contents of a message, steganography can be said to protect both messages and communicating parties. 2LSB technique is used to improve data capacity in an image. Steganography means the study of invisible communication. In Steganography usually hide the existence of the communicated data in such a way that it remains confidential & it maintains secrecy between two communicating parties. This paper investigates how the parity of data can be used effectively to hide a secret message randomly in the image.*

Key Words: Cryptography, Steganography, MSE, PSNR, Two LSB, Random insertion.

1. INTRODUCTION

In today's computer world, it is very important to keep secret information secret, private information private, and when profits are involved, protect the copyrights of data. To accomplish these difficult tasks, new methods based on the principle of steganography is being developed and used. Many times, a cryptographic message transition gives unwanted attention. The cryptographic concept may be restricted for use. Steganography is the combination of art and science of communicating in a way which hides the existence of the communication.

Cryptography hides the content of a private message from a unwanted people, whereas steganography even conceals the real message. Steganography is commonly different than cryptography or watermarking [3]. The relation between them is related in many ways, the fundamental difference between them in the way they are defined and application of them for the different problems to which they are applied. Steganography is different than cryptography & must not be confused between both, where we transform the message so as to make it meaning obscure to unwanted people who intercept it. So, the definition of breaking the system is different [1].

This technique is based on RGB images. The two least significant bits of the red channel will be used as an indication to the existence of hidden data in green and blue channels. Before embedding the data bits, the parity of the data has been checked [5]. Then according to the content of red component data is placed into the pixel. The selection of pixels to embed was crucial since two controlling elements are used for modification of pixel. The technique used random insertion of bits; every pixel doesn't carry the message bits, so it is difficult to detect presence of information in the pixel.

In cryptography, the structure of a message is jumbled to make it meaningless and unintelligible unless the decryption key is detected. In other word, steganography prevents an unwanted recipient from suspecting that the data exist. A steganographic system thus embeds hidden content in unexceptional cover media so as not to produce an eavesdropper's suspicion. [1] As an example, it is possible to embed a text inside an image or an audio file.

2. PROPOSED SCHEME

The system is broken in cryptography when the hacker can read the secrete message. To break a steganographic system attacker have to detect that steganography has been used and he is detect to read the embedded message. RGB image consist of 3 colors red, green & blue. Image component is given by equation (1). Here R(x, y) is used as a controlling element & pair of data bits is embedded into G(x, y) & B(x, y). For any sequence of message bit pairs (m1, m2) (m3, m4). .. .. (m i-1,mi) ,we will compare the message bits with current G(x, y) and B(x, y) and by using table 2 and 3 data bits will get added into pixels. Whether to embed odd or even parity data is decided by performing modulus operation on R(x, y) which is obtained by equation (2).

$$F(x, y) = R(x, y) + G(x, y) + B(x, y) \qquad ...(1)$$

$$(R(x, y) + 2) \bmod (4) = 0 \qquad ...(2)$$

If equation (2) satisfies the condition, difference between message bit pairs & two least significant bits of G(x, y) is calculated using equation (3). If OE is less than ±2, even parity data is embedded into G(x, y).

$$OE = G\,(b1, b0) - (mi\text{-}1, mi) \qquad ...(3)$$

Where, b0 and b1 are two least significant bits of pixel. Same technique is used for embedding in B(x, y). OE decides whether to embed or not. If equation (2) does not satisfy the condition, then pair of odd data bits is embedded. OE is used to control embedding data rate. The OE variable affects probability of embedding payload. For maximum efficiency, value of OE must be in between -1 to +1. The following steps are used to embed data in pixels:

Table -1: Embedding scheme

2 LSBs of Red	2 LSBs of Green	2 LSBs of Blue
00	Add even parity data	Add even parity data
01	Add odd parity data	Add odd parity data
10	Add even parity data	Add even parity data
11	Add odd parity data	Add odd parity data

Table 1 shows the embedding scheme for message bits if two least significant bits of red color is divisible by 2 then and only then embed the even parity bits of the message Otherwise place the odd parity bits in to the green color and blue color component of that particular pixel. According to the following steps,

If data bits are 00 then,

$$G(x, y) = \begin{cases} G(x, y) & \text{for } g00, g10, g11 \\ G(x, y) - 1 & \text{for } g01 \end{cases} \qquad \ldots(4)$$

If data bits are 11then

$$G(x, y) = \begin{cases} G(x, y) & \text{for } g00, g01, g11 \\ G(x, y) + 1 & \text{for } g10 \end{cases} \qquad \ldots(5)$$

$$For\ R(x, y),\ mod(2)\ != 0 \qquad \ldots(6)$$

If data bits are 01 then

$$G(x, y) = \begin{cases} G(x, y) & \text{for } g01, g11 \\ G(x, y) -1 & \text{for } g10 \\ G(x, y) + 1 & \text{for } g00 \end{cases} \qquad \ldots(7)$$

If data bits are 10 then

$$G(x, y) = \begin{cases} G(x, y) & \text{for } g10, g00 \\ G(x, y) - 1 & \text{for } g11 \\ G(x, y) + 1 & \text{for } g01 \end{cases} \qquad \ldots(8)$$

Equation (4), (5), (7) and (8) are used to embed the message bits in to green color. Same equations can be used to embedding in blue color. We have placed the message bits in B(x, y) after performing the operations on G(x, y).

If the contents of R=00, OR 10 and at the same time data bit s are 00 or 1

Table -2: Embedding scheme when R = 00 OR 10

	d00	d11
For G=00	G=G	DON'T ADD
For G=01	G=G-1	DON'T ADD
For G=10	DON'T ADD	G=G+1
For G=11	DON'T ADD	G=G

If the contents of R=01 OR 11 and at the same time data bits are 00 or 10

Table -3: Embedding scheme when R = 01 OR 11

	d01	d10
For G=00	G=G + 1	DON'T ADD
For G=01	G=G	G=G+1
For G=10	G=G -1	G=G
For G=11	DON'T ADD	G=G – 1

If equation (6) satisfies the condition & data bits are having even parity. Then equation (9) can be used to embed the data. In this case pixel value will be changed without embedding data bits. This is undesired operation which helps to degrade the image. But embedding capacity is increases. Since some of the pixel doesn't carry any information, it is difficult to detect the pixel which carries the information. When contents of R are 00 or 11 then add the message bits without checking any condition. It has been observed that if we use parity check for R= 00 or 11, around 35 to 40% pixels get wasted, which doesn't carry the message bits. Our aim is to provide higher security with higher embedding rate.

3. EXPERIMENTAL RESULTS

First, different images of same size have been taken for inserting the same message. And then different messages of same size have been taken to insert in one image. Some experiments are also performed on different size of images and different size of messages. All pixels will not participate in embedding of message, as embedding of data is depends on the parity of message bits; the algorithm is tested by two ways. Experimental result shows that 65 to 74 percent of pixels contained data.

For embedding data bits in to image OE must be less than ±2, it is observed that equation (2) limits the embedding rate. This is because of random insertion. These Images are taken from www.google.com.

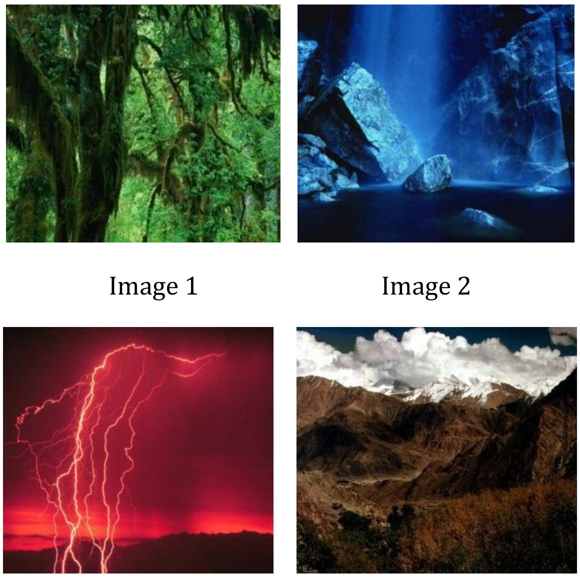

Image 1 Image 2

Image 3 Image 4

Fig -1: Cover images

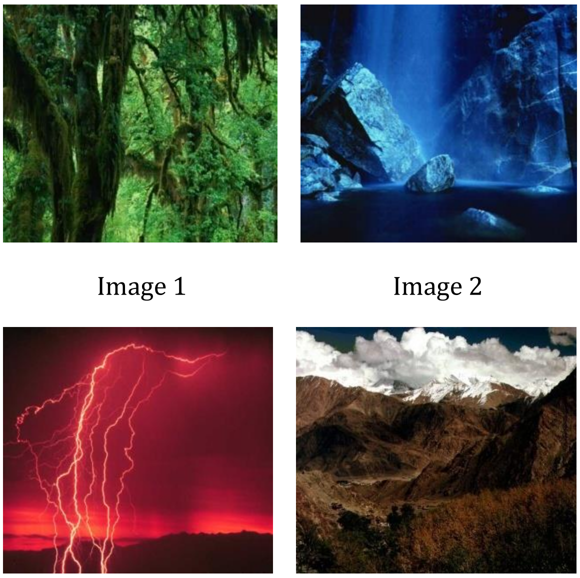

Image 1 Image 2

Image 3 Image 4

Fig -2: Stego images

3. CALCULATION OF PSNR

In order to avoid the stego-image from being suspected of hiding secret information in, the quality of the stego-image should not be degraded significantly. The mean squared error between the cover image and the stego-image (embedding distortion) can be used as one of the measures to assess the relative perceptibility of the embedded text. [3][6]. Usually, the PSNR (Peak Signal-to-Noise Ratio) formula is used to evaluate the distortion between the pre-processing image and the post-processing image.

$$\text{MSE} = \left[\frac{1}{X \; x \, Y}\right] \sum_{I}^{X} \sum_{J}^{Y} \left[X_{IJ} - X_{IJ}^{\prime}\right]^{2}$$

$$\text{PSNR} = 10 \log_{10}[\,(2^{n} - 1) / \text{MSE}]$$

The **x** and y stand for image's height and width, respectively. The Xij and Xiij represent the preprocessing image pixel value in position (i, j) and the post-processing image pixel value in position (i, j), respectively. Theoretically, if the distortion between the preprocessing image and the post processing image is small, the value of PSNR comes out larger. Therefore, a larger value of PSNR means that the processed image has better quality. Usually, if the PSNR value is greater than or equal to 30 db, the distortion between the original image and the processed image is not suspicious to the human eye the experiment is performed on different images to calculate PSNR. The figure 1 shows the cover images and figure 2 shows the respective stego images. Cover images are taken from google.com. Table 4 illustrates the PSNR values and mean square error.

Table -3: PSNR Values of the images

Image	MSE	PSNR
Image 1	0.7216	39.2678
Image 2	0.8814	36.4343
Image 3	0.7346	37.5528
Image 4	0.7118	42.7283

4. CONCLUSION

In the project, the two least significant bits of pixel will used to embed the data bits therefore the capacity of the algorithm will be better. Encryption and steganography can achieve separate goals. Encryption encodes data such that an unintended recipient cannot determine its intended meaning. Since Two LSB's are used for embedding. Present method is having high embedding rate. The capacity of the algorithm is better. Experimental result shows that PSNR ratios of images are founds to be more than 30dB. Thus processed images are not suspicious to the human eye, along with this advantage the random insertion method is used. And thus the transmission of data is highly secured.

REFERENCES

[1] Ishwarjot Singh ,J.P Raina,“ Advance Scheme for Secret Data Hiding System using Hop feld & LSB” International Journal of Computer Trends and Technology (IJCT) – volume 4 Issue 7–July 2013.

[2] G. Manikandan, M. Kamarasan and N. Sairam “A Hybrid Approach for Security Enhancement by Compressed Crypto-Stegno Scheme “, Research Journal of Applied Sciences, Engineering and Technology 4(6): 608-614, 2012

[3] Dumitrescu, S. W. Xiaolin and Z. Wang, 2003. Detection of LSB steganography via sample pair analysis. In: LNCS, Vol. 2578, Springer-Verlag, New York, pp: 355 -372.

[4] J. Kaur and Sanjeev Kumar, ” Study and Analysis of Various Image Steganography Techniques” IJST Vol.2 Issue 3, September 2011 .

[5] Adnan Gutub, Abdulaziz Tabakh, Ayed Al-Qahtani “Triple-A: Secure RGB Image Steganography Based on Randomzation”, International Confernce on Computer Systems and Applicatons (AICCSA-2009), pp: 400-403, 10-13 May 2009.

[6] Sharp T., “An implementation of key-based digital signal Steganography” in Pro. Information Hiding Workshop,Springer LNCS 2137, pp. 13–26, 2001.

[7] Jarno Mielikainen, "LSB Matching Revisited", Signal Processing Letters, IEEE, Publication : May 2006 Volume : 13, Issue : 5, pp. 285- 287.

[8] Bailey, K., and Curran, K., “An Evaluation of Image Based Steganography Methods”, Journal of Multimedia Tools and Applications, Vol. 30, No. 1, pp. 55-88, 2006.

[9] J. Mielikainen, "LSB Matching Revisited" Signal Processing Letters, IEEE, Publication : May 2006 Volume : 13, Issue : 5, pp. 285- 287.

[10] Sanjeev Kumar & Jagvinder Kaur, ” Study and Analysis of Various Image Steganography Techniques” IJCST Vol.2, Issue 3, September 2011

[11] Ahn, L.V. and N.J. Hopper, 2004. Secret-key steganography. In Lecture Notes in Computer Science.Vol. 3027 /2005 of Advances in Cryptology - EUROCRYPT 2004, pp: 323–341. Springer-Verlag Heidelberg.

[12] Liu, Fenlin, Yang, Chunfang., Luo, Xiangyang., and Zeng, Ying., “Pixel Group Trace Model-Based Quantitative Steganalysis for Multiple LSB Steganography”, IEEE Transactions on Information Forensics and Security, Vol. 8, No. 1, January 2013.

[13] N.F Johnson and Jajodia, S., “Exploring Steganography:Seeing the Unseen”, Computer Journal, February 2008.

[14] Shabir A. Parah, G.M. Bhat, Javaid A. Sheikh, “Data Hiding in Intermediate Significant Bit Planes, A High Capacity Blind Steganographic Technique”, Intenational Conference on Emerging Trends in Science, Engineering and Technology , pp.192-197, July 2012.

[15] Lee ; Chen, and L.H., “High capacity image Steganographic model”, Visual Image Signal Processing, 147:03,June 2008

[16] Nagaraj V., Dr. Vijayalakshmi V. and Dr. Zayaraz G., “Modulo based Image Steganography Technique against Statistical and Histogram Analysis”, IJCGA Special Issue on“Network Security and Cryptography” NSC, 2011.