An Improved Steganographic Technique Using LSB Replacement on a Scanned Path Image

B. Karthikeyan, S. Ramakrishnan, V. Vaithiyanathan, S. Sruti, and M. Gomathymeenakshi
(Corresponding author: B. Karthikeyan)

School of Computing, SASTRA University, Thanjavur, Tamilnadu, India
(Email: karthikeyan@it.sastra.edu)

(Received June 3, 2012; revised and accepted Sep. 22, 2012)

Abstract

Transfer of secret data is an important aspect in the arena of developing technology. Privacy of data is one of the major tasks required now days. In this paper, we dealt with steganographic method using randomly generated key along with raster scan, which may include horizontal or vertical pattern, access of image pixels in order to hide a plaintext in an image. The method deployed is also suitable for long plaintext insertion in order to have no visible change in the original image. The process of scanning the image pixels strengthens the security level of the plaintext to be inserted. Optimal Pixel Adjustment Process (OPAP) is applied to the procured stegoimage in order to enhance the image quality.

Keywords: EBCDIC, plaintext, raster scan, stegnography, stegoimage

1 Introduction

Communication is an important aspect in Information Technology nowadays. Cryptographic techniques along with the assurance that the data is secret are the most needed. The method is known as steganography. Steganography [1] means art of communiqué. Image steganography is supposed to be the most secured one than hiding the secret data in another text, by invisible ink or audio because the visual power of humans are limited. Steganography is proposed for secrecy and is not logical but only to the persons having the key.

Secret key Steganography [2, 13] is a system where key exchange must be preceding to communication. Secret key along with the plaintext is embedded in the cover information. One with the key only can do the overturn process and access the secret data. The lead in this method is that though it is susceptible to interception, only the one with the key can way in the secret data.

In this paper, the aim is to get a stegoimage by embedding the plaintext using random key generation mechanism and by reading the cover image pixel values using raster scan. By means of Extended Binary Coded Decimal Interchange Code (EBCDIC) code, the plaintext is transformed into integers, followed by permutation using the key. The plaintext thus obtained undergoes eXclusive NOR (XNOR) operation with the key and is embedded in the image. Using pseudorandom generator for key generation and access through Raster Scan of the cover image for every plaintext is the noteworthy aspect that increases the security level of communication. OPAP improves the quality of stegoimage after Least Significant Bit (LSB) substitution.

2 Literature Survey

The prime aim of steganography is to hide data into a cover medium, in such a way that it cannot be easily detected. Breaking the secret information from the cover medium irrespective of the security level provided, lies in the brilliance of the person attacking the message. Conventionally linguistic steganography was employed; it involves the process of converting the natural language into machine readable language with pre-shared key. Wayner [12] proposed linguistic steganography in his work, where the plaintext is kept covert without any change in syntactic and semantic meaning. The problem with this approach is that the information can be recovered easily by simple brute force approach.

Later steganography is improvised to the non-linguistic technique which focused on embedding the plaintext in an image. This procedure performs permutation, eXclusive OR (XOR) operation on plaintext with the key and finally LSB replacement with the image pixels [11]. But this process is applicable only to a gray level image which fails in case of color images.

The advanced version of the previous method is suggested in [6]. This centered on increasing the security by making use of pseudo-randomized key and also applicable to color level images. Although it is not up to the grade of enhancing the complexity for steganalysis, reading the image pixels in different pattern.

Though there is progressing innovations in strengthening the security of plaintext, steganalyst equally

defines algorithm which identifies the hidden messages by examining the difference LSB in the regular and singular groups [3, 7, 8]. This sort of attacking is also handled by improving the quality of the stegoimage by applying OPAP.

3 Proposed Work

The flowchart as in Figure 1 explains how the secret plaintext is embedded in the cover image upon sharing the key between authenticated users. Let the plaintext be PT of 256 characters. EBCDIC code conversion is applied in order to get the integer values of plaintext. It is of the form: PT=PT (i, j) where i=1 to 16 and j=1 to 16.

Generated key by the pseudorandom generator is of the form: KE (i, j) where i=1 to 16 and j=1 to 16, where every value ranges from 1 to 256. The forthcoming algorithm explains about the pseudorandom generator for key generation.

3.1 Randomization

A list of random numbers is produced when input is specified in the pseudorandom generator. Every time, the values are produced without any repetition [6].

Algorithm 1: Pseudorandom Generation

1: Get the number of keys to be generated and store it in x.
2: **for** i ← 1 to x
3: **for** j ← 1 to x
4: Generate a random number 'y' within the 'x' using nextInt() function in Java [4].
5: If y is not in KE then include y in KE array.
6: **end for**
7: **end for**

Operations like permutation and XNOR operation can be applied on the plaintext as well as the key generated randomly [14]. This increases the security of the plaintext which is nothing but the secret data to be embedded.

3.2 Combinations

Permutation is done on plaintext using the key and is denoted as PP (i, j) where i=1 to 16 and j=1 to 16. Let KE (i, j) = C. We can assume that C is of the form: C=16y+z, Integers y and z ranges from 1 to 256. When z = 0, the last row y^{th} column element of the PT will be placed as the i^{th} row j^{th} column element of the matrix PP. On the other hand, when z≠ 0, the $(z+1)^{th}$ row y^{th} column element of the PT will be placed as the i^{th} row j^{th} column element of PP. The above method is the procedure for permutation.

3.3 XNOR Operations

XNOR operation increases the security as usually an eavesdropper applies XOR operation if the parameters are known. Binary equivalent of both plaintext (PP) and key are stored. XNOR operation is performed between the plaintext (PP) and the key matrix (KE), which are placed in the same position. Thus we XNOR the resulting plaintext (PP) along with the key (KE) is given in Equation (1).

$$TP = \overline{PP \oplus KE} \quad (1)$$

3.4 Formations of Stego Image

The next step is to insert the modified plaintext (TP) into the cover image. Let the cover image IM be of the form IM (i, j) where i= 1 to 256 and j= 1 to 256. The following procedure is feasible for both color and gray level image. The color level image cannot be processed directly. For our ease, color level image is converted to gray scale and then processed. Simplicity in processing the image is achieved if the conversion is done. Any tool can be used for conversion. An example syntax for conversion in MATLAB [12] is IM=rgb2gray(HI), HI is the pixel values of the color image [5, 15]. Thus IM has the equivalent pixel values for gray level image.

In order to increase the security, we apply Raster scan methods for accessing the pixels of the cover image. We here use both horizontal and vertical interlaced scan methods to have a discussion about their efficiency. The image IM is accessed by raster scan method and the order of pixel values to be processed after the scan is stored in the IM (i , j).

Pixel value of every cover image pixel is 8 bits. Human visual system should not be able to detect the changes in stegoimage. Thus the insertion of plaintext must be in a very accurate manner. The procedure for inserting the plaintext in the image is as follows:

Let KE (1, 1) = s. Let us focus the thought on the s^{th} column of IM, i.e., j = s. Convert PP_{11} into its binary equivalent so that we get 8 binary bits in the form of string. Then having the first six-bits of each value of IM_{ir}, i=1 to 4, AND operation is done on the last two bits of the gray image value and the first two bits of the PP_{11}. The logical operation is for acquiring additional data security from steganalysis through brute force approach. This follows with concatenating the resultant two bits in the 1^{st} row, the next 2 bits of PP_{11} after the AND operation in the next row, etc., till we reach the 4^{th} row and wear out the eight binary bits. The embedding of the secret data into image happens column by column till the entire secret data is embedded. The final image with the secret data is termed as stegoimage. It is sent to the other authenticated user along with key.

3.5 Retrieval of Plaintext

One secret key is used for inserting as well as retrieving the plaintext in secret key steganography. Decryption is reverse process of encryption with the same key. The end user can

use the key already decided between the parties to be shared in order to get the plaintext from stegoimage. Secrecy is not maintained if key is exposed.

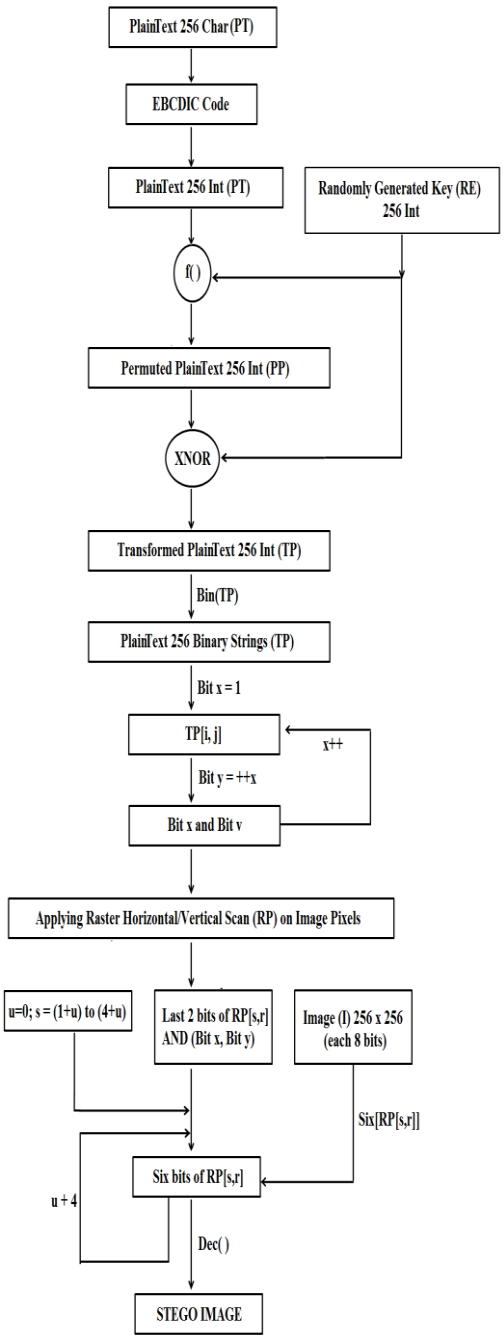

Figure 1: Flowchart

Steps involved in decryption: Binary equivalents of the stegoimage pixel values are taken. Two bits at the last which is concatenated at every pixel of the stegoimage are composed to get the plaintext in binary format which are converted into characters. Figure 2 depicts the original image where the plaintext of 256 characters has to be inserted. Figure 3 depicts the stegoimage formed after plaintext insertion with pseudorandomised key, XNOR and logical operation.

Figure 2: Original image

Figure 3: Stego image

3.6 Results

This paper gives an improved technique for LSB replacement steganography. Randomized key generation and scanning technique strengthens the security of the encryption process. The logical XNOR operation along with permutation allows the plaintext to get embedded column wise in the cover image. We have used OPAP to enhance the quality of the image. The Mean Square Error (MSE) was found to be less on the stegoimage where OPAP has been applied. Further we replace the last two bits of pixel values with plaintext in the concealing technique. It is noticed that Most Significant Bit (MSB) substitution leads to easy visualization of the changes made in the cover image after inserting plaintext. The comparison is given in Table 1.

The scanning of image pixels improvises the security of the stegoimage being transferred. This paper makes use of

raster scanning techniques. The efficiency of scanning process is judged by three factors: Activity Measure (AM), average run length (AR) and sum of differences (SD) [9]. The activity measure given in Equation (2) is the count of the number of transitions in a linear.

$$AM = \sum_{i=1}^{n-1} \binom{n}{k} \delta(current, previous) \quad (2)$$

where δ(x,y) = 0, x = y,
1, x != y.

Table1: MSE comparison

S.No	ERROR METRIC		
	Image Name 256 x 256	Normal MSE	MSE after OPAP
1.	Dolphin	0.00137	0.00088500
2.	Obama	0.00151	0.00090025
3.	Jerry Lang	0.00148	0.00086975
4.	Pizza	0.0010	0.00090026

The second parameter being average run length given in Equation (3) is identified by calculating the average of the number of occurrences of each symbol in a linear sequence.

$$AR = \sum c \times \left(c \times n/L\right) \quad (3)$$

where L - Length of sequence;
n - Number of occurrences ;
c - Run Length.

The sum of differences given in Equation (4) is acquired by adding the absolute valued of the difference of the current pixel and the previous pixel over the whole sequence.

$$SD = \sum_{i=1}^{n-1} \mathrm{mod}(current - previous) \quad (4)$$

4 Conclusion

The overall summarization of the paper centers around strengthening the security by processing the original image pixel values through the scanning methods. The paper focuses on entrenching the valuable plaintext in an image with the help pseudo randomized key matrix. Initially permutation of the plaintext with the key is made by inserting the later into the plaintext column by column. This process further increases the randomization of the plaintext when it is inserted into the image.

The errors in stegoimage is reduced by the operations of XNOR followed by AND. Insertion of permuted plaintext into the image is performed after the image is scanned either by raster horizontal scan or raster vertical scan.

References

[1] D. Bret, *A detailed look at Steganographic Techniques*, US:SANS institute, 2002.

[2] C. Eric, *Hiding in plain sight, Stegnography and the art of Covert Communication*, US: Wiley publishing, 2003.

[3] J. Fridrich, M. Golijan, and R. Du, "Reliable detection of lsb steganography in color and grayscale image," *IEEE Multimedia*, vol. 8, pp. 22-28, 2001.

[4] S. Herbert, *Java: The Complete Reference*, US: McGraw Hill, 2006.

[5] M. S. Hwang, K. F. Hwang, and C. C. Chang, "A time-stamping protocol for digital watermarking," *Applied Mathematics and Computation*, vol. 169, pp. 1276-1284, 2005.

[6] B. Karthikeyan, V. Vaithiyanathan, B. Thamotharan, M. Gomathymeenakshi, and S. Sruti, "LSB replacement steganography in an image using pseudorandomised key generation," *Research Journal of Applied Sciences, Engineering & Technology*, vol. 4, no. 5, pp. 491-494, 2012.

[7] G. Manikandan, M. Kamarasan, P. Rajendiran,and R. Manikandan, "A hybrid approach for security enhancement by modified crypto-stegno scheme," *European Journal of Scientific Research*, vol. 60, no. 2, pp. 224-230, 2011.

[8] G. Manikandan, N. Sairam, and M. Kamarasan, "A hybrid approach for security enhancement by compressed crypto-stegno scheme," *Research Journal of Applied Sciences, Engineering and Technology*, vol. 4, no. 6, pp. 608-614, 2012.

[9] J. Nandhakishore, *A study of scanning paths for BWT based compression*, West Virgina: Lane department of Computer Science and Electrical Engineering, 2004.

[10] P. Rudra, *Getting Started with MATLAB: A Quick Introduction for Scientists and Engineers*, India: Oxford Press, 2002.

[11] V. U. K. Sastry, and Ch. Samson, "Key based stegnography in a gray level image involving permutation and XOR operation," *Journal of Global Research in Computer Science*, vol. 2, no. 6, pp. 53-56, 2011.

[12] P. Wayner, "Mimic Functions," *Cryptologia*, vol. XVI, no. 3, pp. 193-214, 1992.

[13] C. C. Wu, M. S. Hwang, and S. J. Kao, "A new approach to the secret image sharing with steganography and authentication," *Imaging Science Journal*, vol. 57, no. 3, pp. 140-151,2009.

[14] N. I. Wu and M. S. Hwang, "Data hiding: Current status and key issues" *International Journal of Network Security*, vol. 4, no. 1, pp. 1-9, Jan. 2007.

[15] N. I. Wu, C. M. Wang, C. S. Tsai, and M. S. Hwang, "A certificate-based watermaking scheme for coloured images," *The Images Science Journal*, vol. 56, pp. 326-332, 2008.

B. Karthikeyan is currently working as Assistant Professor in School of Computing, SASTRA University. He received his M.Tech degree from Manonmaniam Sundaranar University. Presently he is pursuing Ph.D in the area of Image Processing. His research area includes Image Processing and Steganography.

S. Ramakrishnan is currently working as Assistant Professor in School of Computing, SASTRA University. He received his B.E degree from Madurai Kamaraj University and M.E from Anna University. Presently he is pursuing Ph.D in the area of Image Processing. His research area includes Image Processing and Computer Graphics.

V. Vaithiyanathan is currently as Associate Dean (Research) in School of Computing, SASTRA University. He obtained his Ph.D degree from Alagappa University, Karaikudi. He has published more than 20 papers in reputed journals. His area of interest includes Image Processing, Cryptography, Steganography and Soft computing.

S. Sruti is pursuing her Bachelor of Technology degree in Information Technology at SASTRA University, Tamil Nadu, India. Her research area includes Cryptography and Steganography. She published a paper on her research area in a reputed Journal.

M. Gomathymeenakshi is pursuing Bachelor of Technology degree in Information Technology at SASTRA University, Tamil Nadu, India. Her research area includes Image Processing and Steganography. She published a paper on her research area in a reputed journal.