A New Approach for LSB Based Image Steganography using Secret Key

S. M. Masud Karim, Md. Saifur Rahman, Md. Ismail Hossain
Computer Science and Engineering Discipline
Khulna University, Khulna 9208, Bangladesh.
masud@cse.ku.ac.bd, saifur_cseku@yahoo.com, ismailcseku@yahoo.com

Abstract

This paper introduces a best approach for Least Significant Bit (LSB) based on image steganography that enhances the existing LSB substitution techniques to improve the security level of hidden information. It is a new approach to substitute LSB of RGB true color image. The new security conception hides secret information within the LSB of image where a secret key encrypts the hidden information to protect it from unauthorized users. In general, in LSB methods, hidden information is stored into a specific position of LSB of image. For this reason, knowing the retrieval methods, anyone can extract the hidden information. In our paper, hidden information is stored into different position of LSB of image depending on the secret key. As a result, it is difficult to extract the hidden information knowing the retrieval methods. We have used the Peak Signal-to-Noise Ratio (PSNR) to measure the quality of the stego images. The value of PSNR gives better result because our proposed method changes very small number of bits of the image. The obtained results show that the proposed method results in LSB based image steganography using secret key which provides good security issue and PSNR value than general LSB based image steganography methods.

Keywords: cover-image, steganography, stego-image, LSB.

I. INTRODUCTION

In the past, people used hidden tattoos or invisible ink to uncover steganographic content. Today, computer and network technologies provide easy to use communication channels for steganography. But privacy and anonymity is a concern for most people on the internet. Image Steganography allows for two parties to communicate secretly and covertly. Steganography is a technique to hide information from the observer to establish an invisible communication [1]. Generally a steganographic system consists of cover media into which the secret information is embedded. The embedding process produces a stego medium by replacing the information with data from hidden message. To hide hidden information, steganography gives a large opportunity in such a way that someone cannot know the presence of the hidden message. The goal of modern steganography is to keep its information undetectable [2].

Generally secret information is stored into the specific position of Least Significant Bit (LSB) of a cover image which is the carrier to embed messages [1, 2, 3, 4]. Anyone can ensure that the specific position of LSB contains secret information. So it is easy to recover the secret information for anyone by using retrieval method. The main intention of image steganography is to ensure security of hidden information. For security purpose, we have introduced a new approach of LSB based image steganography. Here we are adding a secret key which ensure the security of hidden information. The insertion of hidden information is totally controlled by the secret key. This secret key decides the appropriate position of hidden information. It is very difficult to retrieve the hidden information without the same secret key. So by using a secret key, we can increase the security level of the hidden information in LSB based image steganography.

There are a number of researches available describing features of image steganography. Many steganographic methods have been proposed [2, 3, 4, 5, 6]. The most common of these is replacing *least significant bits* (LSB) of the pixels with the secret message. A well-known LSB based image steganography is presented in [3] and that proposed an adaptive method based on inter pixel relationship. This method greatly enhanced the stego image quality. It is possible to recover the secret information for anyone by applying the retrieval method. Another LSB based image steganography is presented in [2] and proposed three efficient steganographic methods that utilize the neighbourhood information to estimate the amount of data to be embedded into an input pixel of cover image and that embed a fixed three bits of information in smooth areas and a variable number of bits are embedded into the edged areas. This method uses some pixels of the image to store too many bits of hidden information but other pixels remain unchanged. As a result, some pixels are distorted roughly but other pixels become unused. They also did not provide any security issue and it is possible to recover the secret information for anyone.

In this paper, we proposed an efficient LSB based steganographic method that utilizes the secret key to hide the information into an input pixel of cover image without producing perceptible distortions. Here a bit of hidden information is placed in either LSB of Green or Blue matrix of a specific pixel which is decided by the secret key. So anyone cannot exactly make a decision that the bit of hidden information is placed in either LSB of Green or Blue matrix.

987-161284-908-9/11/$26.00 © 2011 IEEE

As a result, the security level of image steganography is attained.

II. Literature Review

The simplest approach to hiding data within an image is called least significant bit (LSB) insertion. For 24-bit true color image, the amount of changes will be minimal and indiscernible to the human eye. As an example, suppose that we have three adjacent pixels (nine bytes) with the following RGB encoding:

10010101	00001101	11001001
10010110	00001111	11001010
10011111	00010000	11001011

Now suppose we want to hide the following 9 bits of data **101101101**. If we overlay these 9 bits over the LSB of the 9 bytes above, we get the following (where bits in bold have been changed) pixels:

10010101	00001100	11001001
10010111	00001110	11001011
10011111	00010000	11001011

The following formula provides a very generic description of the pieces of the steganographic process:

cover image + *hidden information* = *stego image*

In this perspective, the *cover image* is the main image in which the *hidden information* will be embedded. The resultant image is the *stego image* (which will, of course. be the same type of image as the *cover image*).

To measure the quality of stego image, Peak Signal-to- Noise Ratio (PSNR) is calculated. PSNR is a statistical measurement used for digital image or video quality assessment [2]. PSNR is most easily defined via the mean squared error (MSE) which for two $m \times n$ monochrome images I and K where one of the images is considered a noisy approximation of the other is defined as [3]:

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i,j) - K(i,j)]^2 \quad (1)$$

The PSNR is defined as:

$$PSNR = 10.log_{10}\left(\frac{MAX_I^2}{MSE}\right) = 20.log_{10}\left(\frac{MAX_I}{\sqrt{MSE}}\right) \quad (2)$$

Larger PSNR indicates better quality of the image or in other terms lower distortion. The larger the PSNR value the smaller the possibility of visual attack by human eye [2] [3].

III. Proposed Methods

In this paper, we have taken the binary representation of the hidden information and overwrite the LSB of each byte within the cover image. Here we have introduced a secret key to protect the hidden information. The following formula, we have used in our proposed method is:

cover image + *secret key* + *hidden information* = *stego image*

The secret key is converted into one dimensional (1D) circular array bit stream as shown in below

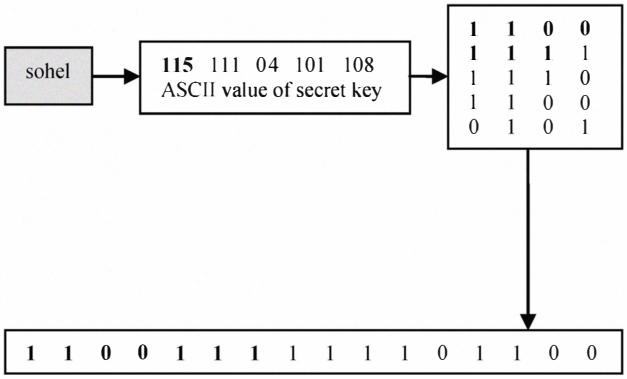


Fig. 1 1D array representation of a secret key

A 24-bit color scheme uses 24 bits per pixel and each byte represents the intensity of the three primary colors red, green, and blue (RGB), respectively. So, a cover image can be split into three matrices as shown in Fig. 2.

The hidden information is converted from decimal to binary. Each pixel is converted into 8 bit binary value. Then the 2D array is reshaped into a 1D array. This 1D array matrix is also called bit stream of hidden information. The process to convert the hidden information into 1D array is shown in Fig. 3.

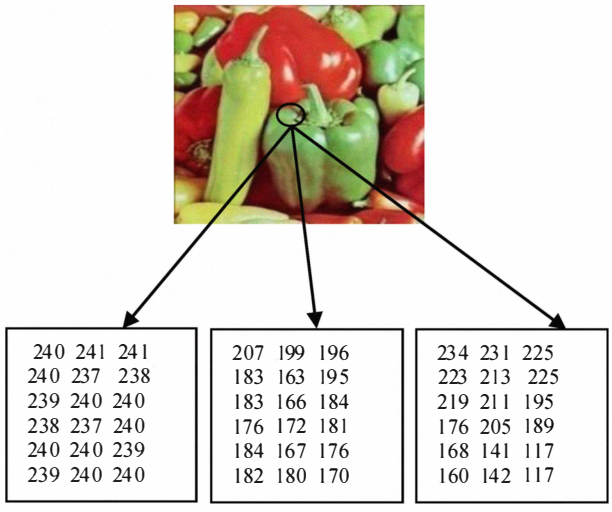


Fig. 2 RGB matrix representation of a cover image

A. Hiding Technique of Hidden Information

To hide hidden information we have to take a cover image. This cover image is divided into three matrices (Red, Green and Blue) as shown in Fig. 2. The secret key is converted into

1D array of bit stream. Secret key and Red matrix are used only for decision making to replace hidden information into either Green matrix or Blue matrix. Each bit of secret key is XOR with each LSB of Red matrix. The resulting XOR value decides that the 1 bit of hidden information will be placed with either LSB of Green matrix or Blue matrix. The same process will be continued until the hidden information is finished. The flow chart to hide hidden information into cover image is shown in Fig. 4.

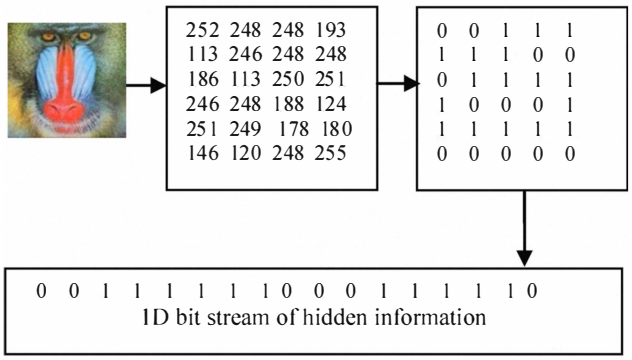


Fig. 3 1D array representation of hidden information

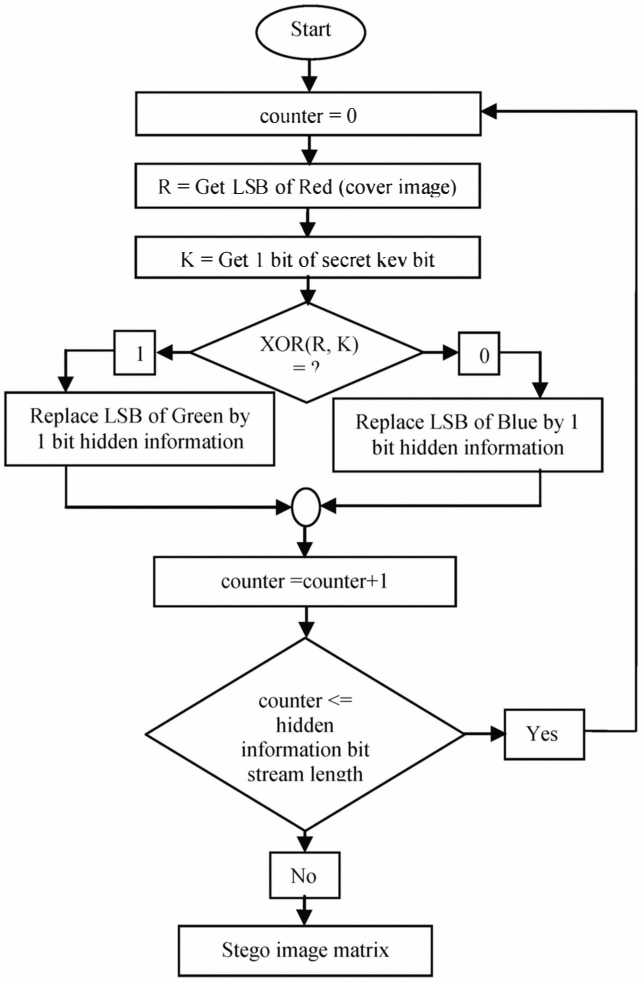


Fig. 4 Flow Chart to hide hidden information into cover image

At Fig. 5, the LSB of Red matrix of pixel 1 is 0 and the first bit of secret key is 1. The XOR value of 0 and 1 is 1. In our method, if the XOR value is 1 then the LSB of Green matrix is replaced by the first bit of hidden information. If the XOR value is 0 then the LSB of Red matrix is replaced by the first bit of hidden information. The 1D array of secret key is circular. The substitution process will be continued depending on the length of hidden information's 1D array.

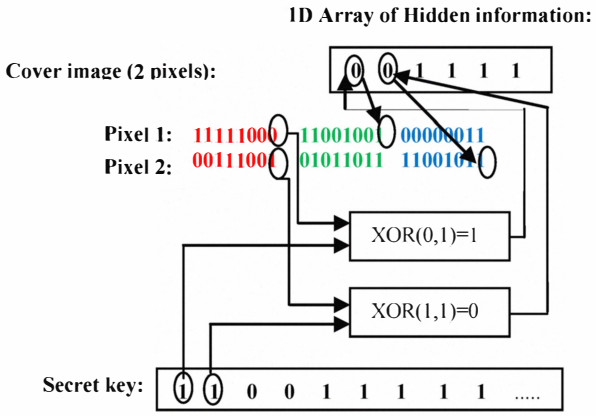


Fig. 5 1D array representation of hidden information

At Fig. 5, the LSB of Red matrix of pixel 1 is 0 and the first bit of secret key is 1. The XOR value of 0 and 1 is 1. In our method, if the XOR value is 1 then the LSB of Green matrix is replaced by the first bit of hidden information. If the XOR value is 0 then the LSB of Red matrix is replaced by the first bit of hidden information. The 1D array of secret key is circular. The substitution process will be continued depending on the length of hidden information's 1D array.

B. Recovery Technique of Hidden Information

To recover the hidden information, we have to take a stego image. This stego image is divided into three matrices (Red, Green and Blue) as shown in Fig. 2. Then we have to know the secret key. The secret key is converted into 1D array bit stream. Each bit of secret key is XOR with the each LSB of Red matrix of the stego image.

The resulting XOR value decides that 1 bit of hidden information is stored in either LSB of Green matrix or Blue matrix of the stego image. The length of hidden information is stored in the first row of stego image during the hiding process. The recovery process will be continued depending on the length of hidden information bit stream. The flow chart to recover hidden information from stego image is shown in Fig. 6.

At Fig. 7, the LSB of Red matrix of pixel 1 is 0 and the first bit of secret key is 1. The XOR value of 0 and 1 is 1. In our method, if the XOR value is 1 then the hidden bit can be found at LSB of Green matrix. And if the XOR value is 0 then the hidden bit can be found at LSB of Blue matrix. This bit is picked and stored into a 1D array. Finally the 1D array is reshaped into 2D array to form actual hidden information. The process to recover hidden information from stego image is shown in Fig. 7.

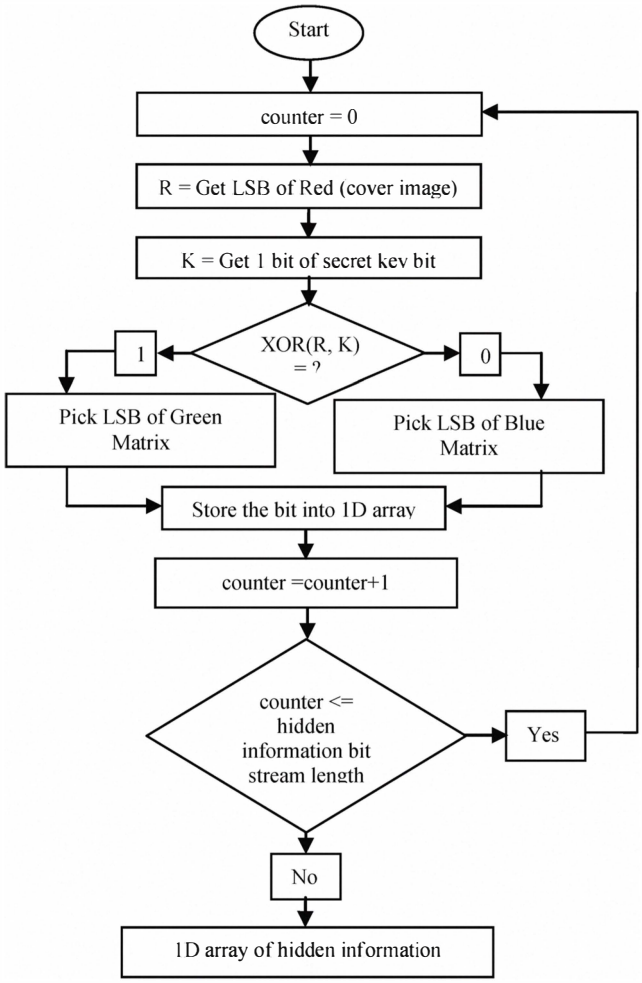


Fig. 6 Flow Chart to recover hidden information from stego image

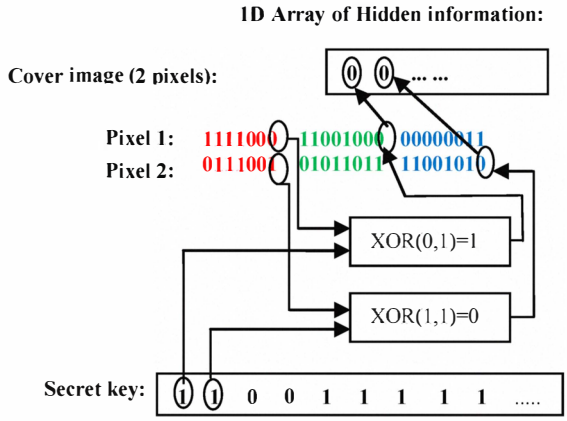


Fig. 7 Process to recover hidden information from stego image

IV. Experimental Result And Discussion

A. Experimental Results

Experimental results are given in this section to demonstrate the performance of our proposed method. We used some standard RGB (true color) images as the cover image. Small size image is used as the hidden information.

(a)Lena (b) Baboon (c) Peppers

Fig. 8 Original cover image

The hidden information used in our proposed method is shown below:

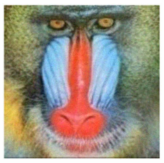
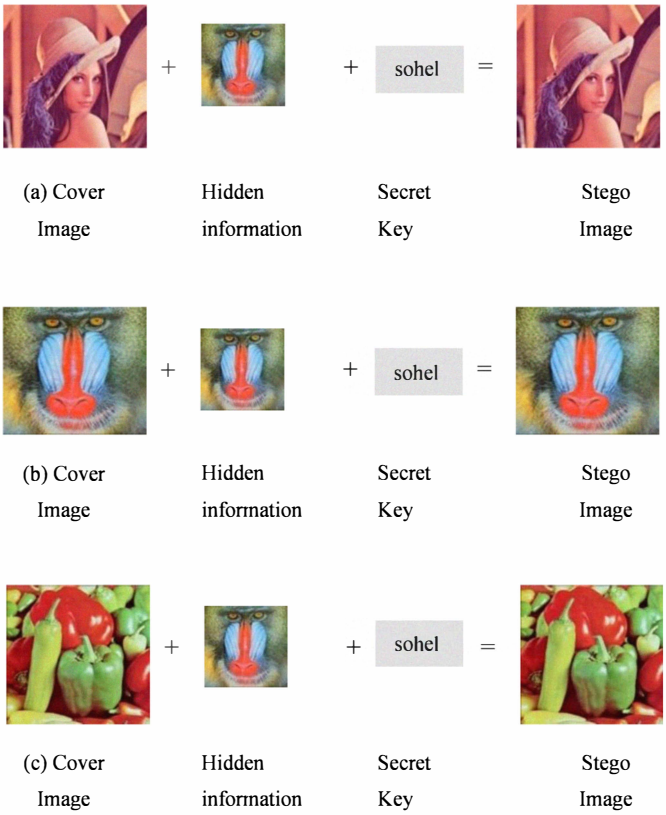


Fig. 9 Hidden information

The procedure to get stego image from cover image by using our proposed method is shown below:



(a) Cover Image | Hidden information | Secret Key | Stego Image

(b) Cover Image | Hidden information | Secret Key | Stego Image

(c) Cover Image | Hidden information | Secret Key | Stego Image

Fig. 10 Stego image produced by using hidden information

Three standard RGB (true color) images, named Lena, Baboon and Peppers are used as cover image. These images are shown in Fig. 8. The hidden information which is used to hide into cover image is shown in Fig. 9. Hidden information is inserted into cover image with secret key. The resulting image is called stego image. The procedure to get stego image from cover image is shown in Fig. 10. These stego images, named Lena, Baboon and Peppers are shown in Fig. 11. The distortions take place in the stego images due to embedding a large amount of secret message using our proposed method are undisclosed to human eye. The cover image used in our proposed method is shown in Fig. 8.
The stego images resulted from our proposed method is shown in Fig. 11.

(a)Lena (b) Baboon (c) Peppers

Fig. 11 Resulted stego image

Using equation (1) and (2), we calculated PSNR value of stego image. These PSNR values are displayed in Table I.

Table I The experimental results for the proposed method

Images	PSNR (in dB)
Lena	53.7618
Baboon	53.7558
Peppers	53.7869

B. Comparison

The experimental results by applying our proposed method on different standard images are compared with other methods we have reviewed. Na-I Wu's method [3] modifies more pixels (almost all of the pixels in an image) of an image in hiding information. Four Neighbor, Eight Neighbor and Diagonal Neighbor methods [2] modify 3 or more bits of a pixel. But for the same capacity our proposed scheme modifies only one bit of each pixel.

Table II Comparison results with NA-I WU's method and four neighbor methods

Cover Images	PSNR (in dB) in Na-I Wu's method	PSNR (in dB) in Four Neighbor method	PSNR (in dB) in our method
Lena	34.3962	41.1468	53.7618
Baboon	30.413	36.5154	53.7558
Peppers	33.7496	41.0315	53.7869

Another point is that, without secret key no one able to know the exact position where the hidden information is placed. Because each bit of hidden information is placed either at LSB of Green matrix or Blue matrix. So, to extract the hidden information, the secret key is must be needed. As we changed only one bit of LSB of Green or Blue at any pixel of the cover image, so the resulting PSNR value of our proposed method is better than other methods. So the proposed method provides an effective way to implant hidden information into the cover image without producing clear distortion.

V. Conclusion

The experimental results show that the proposed method is an effective way to integrate hidden information reporting without significant distortion. And it is very difficult for the unauthorized users to identify the changes in stego image. The use of the secret key gives a way to secure the information from illegal user.
In our proposed method, we used a secret key to hide hidden information into cover image. This process provides a new dimension for image steganography. It is very difficult to recover the hidden information for third party without knowing the secret key. Our proposed method provides better PSNR value where larger PSNR indicates better quality of the image or in other terms lower distortion.

References

[1] F. Hartung and M. Kutte "*Information hiding-a survey,*" Proceedings of the IEEE: Special Issue on Identification and Protection of Multimedia Content, Volume: 87 Issue: 7, pp. 1062 – 1078, July. 1999.

[2] M. Hossain, S.A. Haque, F. Sharmin, *"Variable Rate Steganography in Gray Scale Digital Images Using Neighborhood Pixel Information"*, Proceedings of 2009

12th International Conference on Computer and Information Technology (ICCIT 2009) 21-23 December 2009, Dhaka, Bangladesh.

[3] Na-I Wu, "*A Study on Data Hiding for Gray-Level and Binary Images"*, http://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&ved=0CBwQFjAA&url=http%3A%2F%2Fethesys.lib.cyut.edu.tw%2FETD-db%2FETD-search%2Fgetfile%3FURN%3Detd-0707104-144705%26filename%3Detd-0707104-144705.pdf&ei=yMavTr7LOoSBhQet3pHRAg&usg=AFQjCNFzfbb-TMOJ3fg_Qvv8DsUfJY8qwA ,Accessed on March 2009.

[4] G.J. Simmons, "*The Prisoners' problem and the subliminal channel*," in proc. CRYPTO'83, pp. 51-67, 1983.

[5] N. Nabavian *"CPSC 350 Data Structures*: *Image Steganography"*, nabav100@chapman.edu, November 2007.

[6] T. Morkel, J.H.P. Eloff, M.S. Olivier, "*An overview f image steganography"*, http://mo.co.za/open/stegoverview.pdf, Accessed on January 2009.

[7] Sellars, Duncan, "*Introduction to Steganography*", http://www.cs.uct.ac.za/courses/CS400W/NIS/papers99/dsellars/stego.html.

[8] Mendall, Ronald, "*Steganography-Electronic Spycraft*, *Earthweb Networking and Communications",* September 2000, http://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=2&ved=0CBsQFjAB&url=http%3A%2F%2Fwww.csie.mcu.edu.tw%2F~s9170464%2FSteganography.doc&ei=j8qvTpD3GoiohAfFtoXDAg&usg=AFQjCNGu-W3k9B-GZMdr-8GVUs4SIxPUdQ