The 5th International Conference on Electrical Engineering – Boumerdes (ICEE-B)
October 29-31, 2017, Boumerdes, Algeria.

An Improved Approach for LSB-Based Image Steganography using AES Algorithm

Sofyane Ladgham Chikouche, Noureddine Chikouche
Computer Science Department
University of M'sila, Algeria

Abstract— **The steganography is the art of hidden; its main aim is to pass unnoticed data in another data. There are many types of data that used in steganography, such as message, image, and video. In this work, we are interested in hiding a message inside an image. Our work focuses on the study of three approaches based on least significant bit (LSB) techniques that mean put the bits of the message in the least significant bits in each pixel of the image. Moreover, we propose an improved approach for LSB-based image steganography. In this approach, we reduce the length of hidden message by Deflate algorithm which is a lossless data compression algorithm that combines the LZ77 algorithm and the Huffman algorithm. Another important characteristic of our approach is to protect the reduced hidden data by AES (Advanced Encryption Standard) algorithm. Our experiment results show that our improved approach is most effective compared to existing approaches.**

Keywords-component; Image steganography; LSB; Advanced Encryption Standard; Cryptography

I. INTRODUCTION

The steganography is one of among old sciences as the cryptography. Its main objective is to pass unnoticed information in another message. In the modern literature, the aim of steganography is to hide a secret data in a medium file so that an intruder who controls the communication does not remark existing a hidden message behind the medium file. The medium file can be an image, a sound, a video, etc. Image steganography is the most common form of steganography that used in various applications for hiding text in image, video in image, and audio in image. In image steganography, the principal posed question is: how to hide a secret message inside an image by steganography techniques?

The secret information can be concealed using two insertion domains: spatial domain and frequency domain. We are interested in the spatial domain because the changes of the image bits do not appear to the human eye. The spatial domain performs the dissimilation in the bits of the pixels of the holder image. The LSB (Least Significant Bit) technique is one of the existing spatial techniques. It is based on hiding a secret message in the least significant bits of the pixels in the image so that the distortions brought by the insertion process remain imperceptible. Indeed, for the human eye, variations in the value of the LSB are almost imperceptible.

In our work, we study three techniques based on the algorithm the least significant bit (LSB). Moreover, we propose an improved approach for LSB-based image steganography. In our approach, we reduce the length of hidden message by Deflate algorithm which is a lossless data compression algorithm that couples the LZ77 algorithm and the Huffman algorithm. Another important characteristic of our approach is to protect the reduced hidden data by AES (Advanced Encryption Standard) algorithm. In addition, we implement the studied approaches and our approach by Java language. Our experiment results show that our improved approach is more effective than existing approaches.

The rest of this paper is structured as follows: Section II presents AES algorithm and compression algorithms. Section III presents related works. Section IV describes our improved approach. The experiment results and the discussion of the results obtained are presented in Section IV. Finally, the paper ends with a general conclusion.

II. PRELIMINARIES

A. Advanced Encryption Standard

Advanced Encryption Standard (AES) [1] is the strongest symmetric algorithm until now. It is a symmetric encryption algorithm that supports data block of 128-bit and variable key sizes of 128, 192 and 256 bits. In AES, we can use only one key for both encryption and decryption that can be used by sender and receiver. The secrecy maintained by the key has secured and authentication has maintained the key itself.

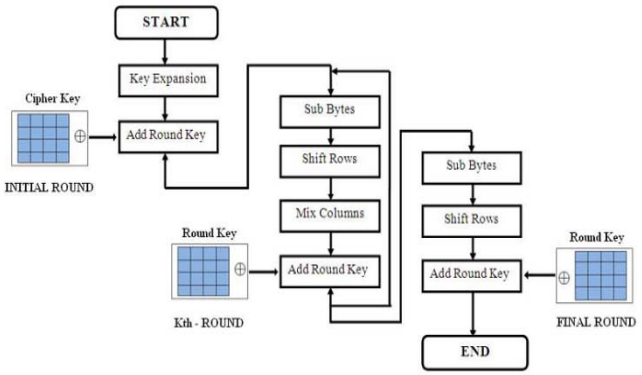

Figure 1: AES algorithm

In AES, input data is arranged in 4x4 arrays of bytes called a State, with four rows and four columns consisting of 16 bytes in total. AES uses a round function that is composed of four

978-1-5386-0686-5/17/$31.00 ©2017 IEEE

different byte-oriented transformations. The AES algorithm is presented in Figure 1.

B. Compression Algorithms

The deflation algorithm used by gzip (also zip and zlib) Deflate is a lossless data compression algorithm that couples the LZ77 algorithm and the Huffman coding

LZ77 algorithm: LZ77 is a lossless data compression algorithm [2] proposed by Abraham Lempel and Jacob Ziv in 1977 and 1978. This algorithm is universal; it does not need to know the statistics of the data to be compressed as is the case with the coding From Huffman. Here, the compression is done in one pass and is of course without loss of data.

Compression is achieved by the equivalence of strings. More precisely, we use a "window" which is itself separated into two parts, to move in the text, which thus contains the previously coded portion and a portion of text that remains to be compressed. The idea is quite simple: when we find equivalence between the part to be compressed and the part already read, instead of rewriting the equivalence, we write only at the output the couple (position, length) which indicates or is l Equivalence and its length in the part already read. We then add the uncompressed character according to the equivalence, to the couple. We make this choice especially for reasons of time! Indeed, the context causes that if it has not been coded in the read expression, it is because there is a good chance that it is not compressed afterwards. This avoids wasting valuable time searching for equivalence.

Huffman coding: Huffman coding [2] is a process in which the length of the encoded symbols varies in the opposite direction to their probability of occurrence (i.e. the high probability symbols will have a short code while the low probability symbols will have a Long code). The model provides the coding process with the probability or frequency counter for each symbol encountered in the input stream. The algorithm creates variable length codes over an integer number of bits depending on the probabilities provided by the model.

The Huffman technique is based on the construction of a binary decoding tree. This tree is elaborated ascending starting from the leaves of the tree and ascending to the root.

III. RELATED WORKS

In a survey of proposed approaches for LSB-based image steganography, we can find many techniques proposed using various algorithms (e.g. symmetric-key algorithm, asymmetric-key algorithm, compression algorithms, etc.) such as [3-9]. Our work is articulated on study three important approaches.

Least Significant Bit Substitution Technique (LSB): In this technique, the LSBs of the pixel values of cover-image are modified according to bits of message [3]. The simplest of LSB steganography techniques is LSB replacement for all pixels of an image. Since only LSB is changed, the difference between the cover (i.e. original) image and the stego-image is hardly noticeable.

Figure 2 shows an explicative diagram to hide the text message with least significant bit substitution technique**.**

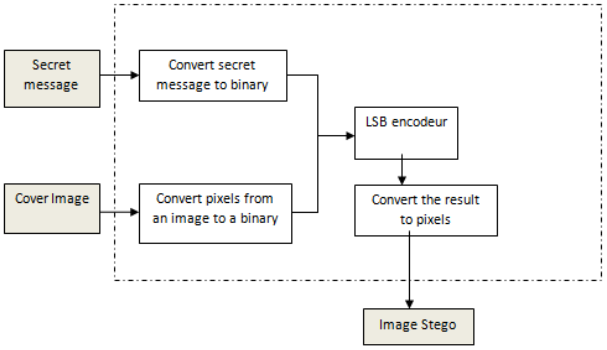

Figure 2: Diagram to hide the text message (LSB technique).

Pseudo-Random LSB Encoding Technique: In this technique, a random-key is used to choose the pixels randomly where message bits will be stored [4]. This implies that the message bits more difficult to find for an intruder. Moreover, the coloured image has three spaces (RGB) the data can be hidden in the LSB of any colour space of the randomly selected pixels. With the use of this technique, it will be difficult for the attacker to identify the pattern in which message bits are hidden, as no particular pattern is followed for embedding subsequent message bits. At transmitter side, a random key is used to randomise the cover-image and embeds the message bits into the LSB of the pixels. This random key is used as a seed for pseudo-random number generator for selecting pixel locations in an image for hiding the secret message bits.

Figure 3 shows an explicative diagram to hide the text message with pseudo-random LSB encoding technique.

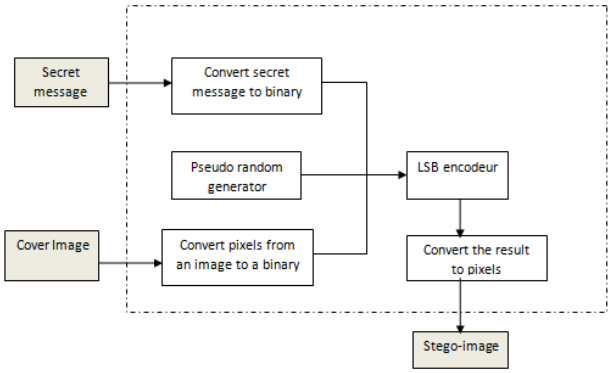

Figure 3: Diagram to hide the text message (Pseudo-Random LSB technique).

Distortion Technique This technique is a modification of LSB substituting technique. In this technique, the modification in LSB of the pixel value is done if the value of the secret bit is 1 else pixel value will remain unchanged which is unlike LSB technique in which every pixel value irrespective of 0 or 1 will modify pixel value [5]. It uses an approach similar to Pseudorandom LSB with the different in cover-pixels are used

978-1-5386-0686-5/17/$31.00 ©2017 IEEE

for information hiding. A Pseudorandom number generator is used to make this selection. To embed bit 1, a random value *x* is added or subtracted from pixel's value. The value of *x* is so chosen that minimum change in cover-image occurs. On receiver side the stego-image is compared with the original image. If the pixel differs, the corresponding message bit is 1; otherwise, the bit is 0**.**

Figure 4 shows an explicative diagram to hide the text message with distortion technique.

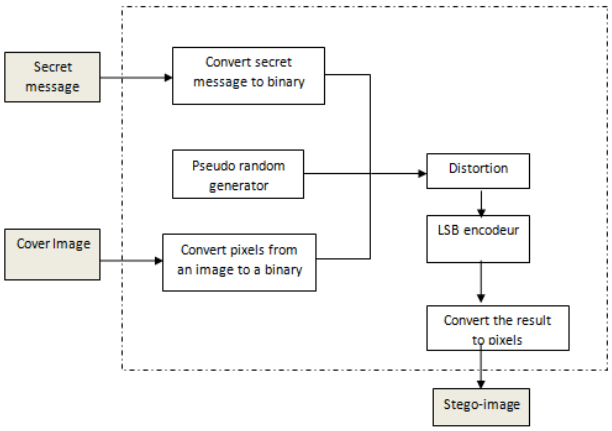

Figure 4: Diagram to hide the text message (distortion technique).

IV. OUR IMPROVED APPROACH

Our approach is based on modification of the pseudo-random LSB encoding technique which touches only the message, the message passes through three phases before injected into cover-image: (1) the first phase is to encrypt the message using AES algorithm (Advanced Encryption Standard) to secure the message. The choose AES cryptosystem because it is fast and secure. (2) The second phase is compressing the message with deflation algorithm of lossless compression to get more information storage. (3) The last phase is converting the message into binary mode then injects it inside the cover-image. Next, we use a random key to choose the pixels randomly where message bits will be stored; this will make the message bits more difficult to find for an intruder. Moreover, the coloured image has an RGB colour model. The data can be hidden in the LSB of any colour space of the randomly selected pixels.

Embedding Algorithm:

Figure 5 shows an explicative diagram to hide the text message.

Step 1: Get the cover-image and secret message.

Step 2: Encrypt the secret message with encryption algorithm AES (Advanced Encryption Standard) then compressed the message with the compression algorithm (deflation).

Step 3: Divide the message into bits, which may be available as text or binary data.

Step 4: Initialize the random key and randomly identify the pixels of cover-image. This random key is basically a seed which is used to generate the same set of random values every time by using random number generator.

Step 5: LSB of randomly located pixels will be modified as per the values of message bits. First of all the message are inserted at the LSB of the Red space's pixels. Later, the process is repeated in green and blue spaces.

Step 6: The modified pixel value is fed back to its respective position. As per the size of message data LSBs of image pixels are modified.

Step 7: save and send the stego-image.

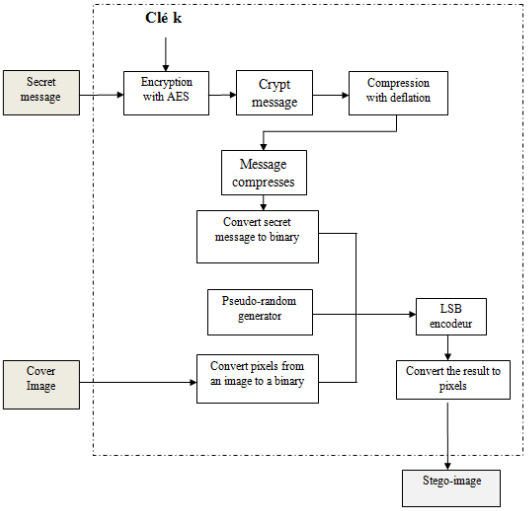

Figure 5: Diagram to hide the text message (our approach).

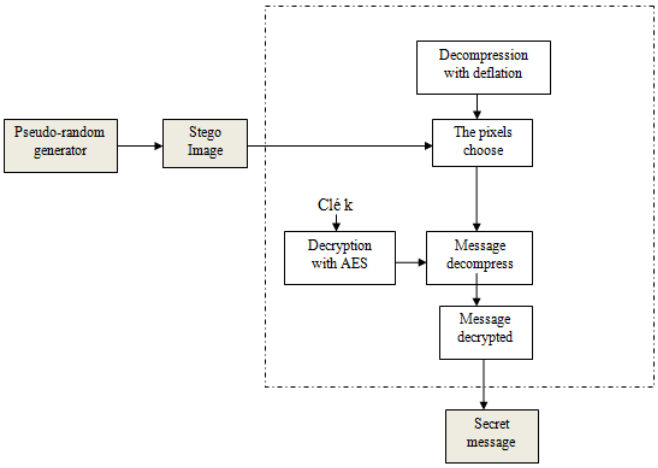

Figure 6: Diagram to retrieve the text message (our approach).

Algorithm to Retrieve Text Message:

Figure 6 shows an explicative diagram for retrieving the text message.

Step 1: Input the stego-image.

Step 2: Initialize the random key and randomly identify the pixels of cover-image. This random key is same at both ends.

Step 3: Read the LSB of each identified pixel of stego-image.

978-1-5386-0686-5/17/$31.00 ©2017 IEEE

Step 4: Decompress each byte with the compression algorithm (deflation) then do the decryption operation with the AES algorithm.
Step 5: Convert each 8 bits into character, which becomes a secret message.

V. EXPERIMENTAL RESULTS AND DISCUSSION

A. Experimental results

In this work, we review three different steganography algorithms for a hidden message inside a cover-image, all these algorithms based on the LSB technique. However, we propose an improved method which is based on combination between Pseudo-Random LSB (Bit Substitution Technique) Encoding technique and cryptographic algorithm. To evaluate the performance of each algorithm, we implement them in Java environment.

We choose image of type TIFF format because the size of the stego-image is no change in comparison with the original image. The use of the TIFF format is lossless for archiving images.

Using the studied algorithms, we have hidden a message in a colour image. We compare the studied techniques with our approach in terms of histogram of the original image and the image-stego and the percentage of change between the original image and the stego-image.

1- Least Significant Bit Substitution (LSB) Algorithm:

In each Figure (1, 2, 3, and 4):
(a) : Original colour image,
(b): Stego colour image,
(c): Histogram the original image,
(d): Histogram stego-image.

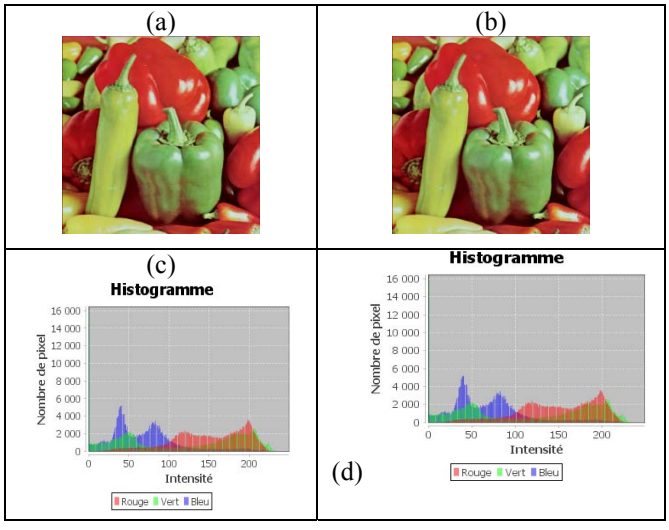

Figure 4: Results obtained for the LSB techniques.

Table 1: Results obtained for the LSB techniques.

Designation	**Value**
size of image in pixels	262144
size of image in bits	2691456
size of message in bits	3264
Number of bits changed	**1614**
Percentage change	**0.05%**

In each table (1, 2, 3, and 4):
Size of image is the number of pixels image.
Size of image in bit = the number of pixels * 24.
Size of message in bit = the number of characters * 8.
Percent change is = Number of bits changed divided by the size of the image in bits multiple 100.

2- Pseudo-Random LSB Encoding Algorithm:

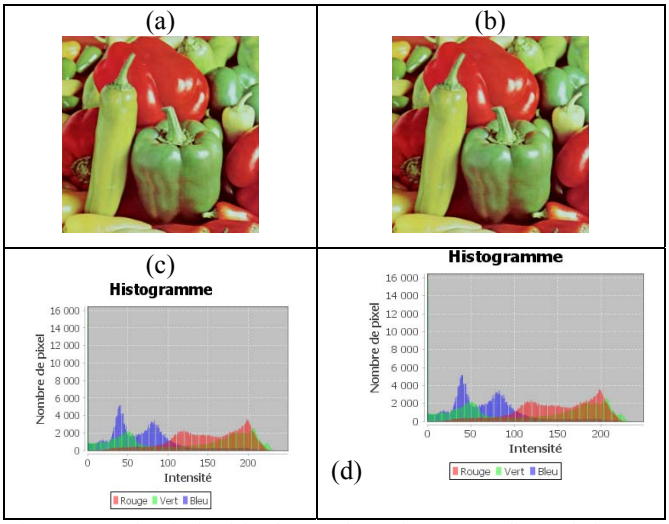

Figure 5: Results obtained for the pseudo-random LSB coding technique.

Table 2: Results obtained for the pseudo-random LSB coding technique.

Designation	**Value**
size of image in pixels	262144
size of image in bits	2691456
size of message in bits	3264
Number of bits changed	**1612**
Percentage change	**0.025%**

3- Distortion Technique:

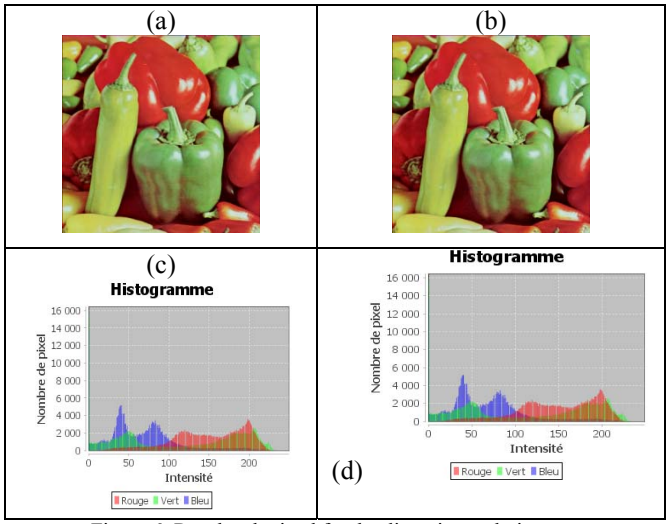

Figure 6: Results obtained for the distortion technique

Table 3: Results obtained for the distortion technique

Designation	Value
size of image in pixels	262144
size of image in bit	2691456
size of message in bit	3264
Number of bits changed	**2136**
Percentage change	**0.033%**

4- Our Improved approach:

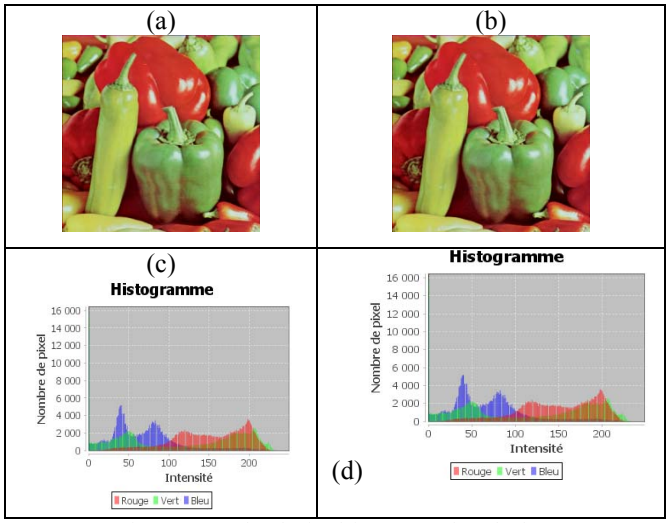

Figure 7: Results obtained for proposed method

Table 4: Results obtained for proposed method.

Designation	Value
size of Image in pixels	262144
size of Image in bit	2691456
size of message in bit	3264
size of Message encrypted in bits	**4608**
size of Message encrypted and compressed in bits	**3840**
Percentage of compresstion	**17%**
Number of bits changed	**3749**
Percentage change	**0.059%**
Security	**Security of the AES key size is 256 bits**

B. Discussion

In this section, we compare the studied approaches and our improved approach. For each approach, we present its advantages and its limitations.

Concerning the Least Significant Bit (LSB) technique, this method consists of modifying the least significant bit of the pixels encoding the image. The advantages of this technique are: the picture quality of cover-image is hardly affected; hiding capacity is good, and it is very simple in implementation. But the disadvantages of this technique are: robustness is less the hidden data is subject to alternation due image manipulation, detection of secret data is easy because of easy algorithm, more information storage requires large image size thus requires high transmission rate due to large size of stego-image.

About the pseudo-random LSB encoding technique, a random key is used to randomly select the pixels where the message bits will be stored. Then, the intruder finds a difficulty to find the secret message. The advantages of this technique are: degradation of cover-image will be very low as the pixels identified are at distant from one another and embedding capacity is good. But the disadvantages of this technique are: access to key *i.e.* seed, can easily detect the location of pixels in cover-image, and thus easily reveal the secret message, more information storage requires large image size thus requires high transmission rate due to large size of stego-image.

Concerning the distortion technique, a pixel property of the cover-image is modified according to a secret message, and then the deformation of the distorted original image contains secret information. The advantages of this technique are: cover-image will be very little modified thus shows no visual difference from stego-image and embedding capacity is good, involves very little complexity. But the disadvantages of this technique are: original image is required at receiver side for retrieving the stored information, thus needs to be sent with stego-image, In case attacker retrieves original cover-image also, retrieval the message is easily detected. It requires high transmission rate due to large size of images.

Concerning our proposed approach, it is a hybrid between LSB-based steganography and cryptography. The advantages

978-1-5386-0686-5/17/$31.00 ©2017 IEEE

of this technique are : degradation of the cover-image will be very low as the pixels identified are at distant from one another, embedding capacity is good, hiding capacity is good, involves very little complexity, and the hiding message is protected by using the AES (Advanced Encryption Standard) algorithm with key size 256 bits.

VI. CONCLUSION

In this work we examined the different steganography algorithms to hide a message inside a cover-image, all these algorithms are based on the LSB technique. However, we proposed an improved method based on the combination of the Pseudo-random LSB (Bit Substitution Technique) and the cryptographic algorithm. We choose AES algorithm with size key 256 which is the strongest symmetric algorithm until now and it is very fast compared to asymmetric-key algorithms.

Based on our implementation of these approaches in Java environment, we compared the studied techniques with our approach in terms of histogram of the original image and the image-stego and the percentage of change between the original image and the stego-image. Our experiment results show that our improved approach is most effective compared to existing approaches.

In this work, we hide a secret message of type text in a medium file of type image. Further study about hiding an image and a video inside an image will be done soon.

REFERENCES

[1] NIST, "Advanced Encryption Standard," http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf, 2001 (accessed March 24, 2017).

[2] An Explanation of the DEFLATE Algorithm : http://www.gzip.org/deflate.html (accessed March 7, 2017).

[3] Ali K. Hmood B. B. Zaindan, "an overview on hiding information techniques in images," journal of applied sciences, vil. 10, no. 18, 2010.

[4] J. Hossain, "Information-Hiding Using Image Steganography with Pseudorandom Permutation", *Bangladesh Research Publications Journal*, vol. 9, no. 3, pp. 215-225, pp. 215-225, 2014.

[5] C. P. Sumathi, T. Santanam and G. Umamaheswari, "A Study of Various Steganographic Techniques Used for Information Hiding," *International Journal of Computer Science & Engineering Survey (IJCSES)*, vol. 4, no. 6, pp. 9-25, 2013.

[6] S. M. Masud Karim, M. S. Rahman and M. I. Hossain, "A new approach for LSB based image steganography using secret key," *14th International Conference on Computer and Information Technology (ICCIT 2011)*, Dhaka, 2011, pp. 286-291.

[7] D. Samidha and D. Agrawal, "Random image steganography in spatial domain," *2013 International Conference on Emerging Trends in VLSI, Embedded System, Nano Electronics and Telecommunication System (ICEVENT)*, 2013, pp. 1-3.

[8] S. N. Gowda, "An advanced Diffie-Hellman approach to image steganography," *2016 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*, 2016, pp. 1-4.

[9] Z. Y. Al-Omari and A. T. Al-Taani, "Secure LSB steganography for coloured images using character-colour mapping," *2017 8th International Conference on Information and Communication Systems (ICICS)*, 2017, pp. 104-110.

978-1-5386-0686-5/17/$31.00 ©2017 IEEE