High Security Data Hiding Using Image Cropping and LSB Least Significant Bit Steganography

Khalid A. Al-Afandy
Department of Computer Science & Engineering
Faculty of Electronic Engineering, Menoufia University
Menouf, Egypt
Khalid_yuosif@yahoo.com

El-Sayed M. EL-Rabaie
Department of Electronic and Communication Engineering
Faculty of Electronic Engineering, Menoufia University
Menouf, Egypt
Srabie1@yahoo.com

Osama S. Faragallah, Ahmed ELmhalawy
Department of Computer Science & Engineering
Faculty of Electronic Engineering, Menoufia University
Menouf, Egypt
osam_sal@yahoo.com, Ahmed.elmahalawy@el-eng.menofia.edu.eg

Gh. M. El-Banby
Department of Industrial Electronics and Control Engineering
Faculty of Electronic Engineering, Menoufia University
Menouf, Egypt
ghadabanby@yahoo.com

***Abstract*—A high security data hiding approach using image cropping and Least Significant Bit (LSB) steganography is proposed. The predefined certain secret coordinate crops are extracted from the cover image. The secret text message is divided into parts with the same image crops. Each part of the secret text message is embedded into an image crop with secret sequence using LSB approach. The embedding is done using the cover image of three color channels. The stego image is given by reassembling the image and the stego crops. A detailed comparative study is performed between the proposed approach and the other state-of-the-art approaches. This comparison is based on visualization to detect any degradation in stego image, difficulty of extracting the embedded data by any unauthorized viewer, Peak Signal-to-Noise Ratio (PSNR) of stego image, and the embedding algorithm CPU time. Experimental results shows that the proposed approach is more secure compared with the other traditional approaches.**

Keywords—Steganography;Least Significant Bit (LSB); Cropping.

I. INTRODUCTION

In the last years the Internet has been considered as a suitable medium for transferring digital data and multimedia. Its main advantage is the availability to almost everyone and data can be received within a few seconds. The main disadvantage of using the Internet is the weak data security, because data can be monitored by any unauthorized viewers. That is why steganography should be used. Steganography is a technique for embedding a secret data into a cover image. Any unauthorized user can view the stego image but only authorized users can extract the secret data. Any steganography approach must be secure to avoid any unauthorized access. It is divided into two main steps, the first step is the embedding algorithm and the second step is the extraction algorithm [1].

The Least Significant Bit (LSB) [1] is a widely used steganography algorithm. It is based on converting characters of the secret text message into a string of binary bits. The original algorithm uses a gray-scale cover image by embedding up to three bits from the secret text message into the least three significant bits of the cover image pixels [1]. This algorithm was adapted to work on color images by using the three color channels. The eight bits are divided into three bits in one color channel, three bits in another color channel and two bits in the last color channel with many variations of the sequence used [5-8].

Another adaptation of the LSB is to use it only on a crop from the cover image [2]. It is based on extracting a crop from the cover image and then embedding the secret text message into this crop using LSB approach. The stego image is obtained by reassembling the image and the stego crop [2-4]. The crop coordinates must be known to the receiver to be able to extract the message.

Breaking the security of the embedded message, allowing unauthorized users to view it or detect that the image contains a secret message is discussed in [9]. That is why the steganography algorithm must be highly secure.

The target of this paper is to present a more secure steganography approach. It is based on extracting predefined crops from the cover image with predefined certain secret coordinates (eg. four crops). The secret text message is divided into the same predefined parts (four parts). The secret text message parts are embedded into the cover image crops by a secret sequence using LSB approach. The embedding is done using three color channels of the image. Three bits are embedded into the red color channel, three bits into the green color channel, and two bits into the blue color channel. Extracting the secret text message is impossible without knowing the image stego crops, the secret sequence of embedding, and the color channels used for embedding. So, the unauthorized viewers cannot monitor the secret message, and hence, the proposed approach is highly secure approach.

978-1-5090-0751-6/16/$31.00 ©2016 IEEE

The rest of this paper is organized as follows. The literature review is presented in section II. Section III shows the proposed approach. The simulation results are explained in detail in section IV, and section V presents the conclusions followed by the relevant references.

II. Literature Review

A. Least Significant Bit (LSB)

LSB algorithm is the simplest and widely used with steganography technique. It is based on embedding the secret text message bits into the least three significant bits of the cover image pixels [8]. The least significant bits of the cover image are used to hide the secret text message [5]. The LSB steganography approach can be classified into two main approaches, LSB replacement and LSB matching [8]. LSB replacement is the simplest. It is based on replacing the least three bits of the cover image pixels with each up to three bits of the message data values that need to be hidden [5]. The basic LSB approach is given by

$$C = \{ X_{ij} \mid 0 \le i < M_c, 0 \le j < N_c \} \quad (1)$$

$$X_{ij} \in \{ 0,1,2,3,\ldots,255 \} \quad (2)$$

$$M = \{ m_i \mid 0 \le I < N, m_i \in \{0,1\} \} \quad (3)$$

where C is the original 8-bit gray-scale cover image of $M_c \times N_c$ pixels, and M is the n-bits secret message.

The n-bits secret message M is embedded into k-least significant bits of cover image C, and the secret message M is rearranged to form a conceptually k-bits virtual image M^* represented as:

$$M^* = \{ m^*_i \mid 0 \le i < n^*, m^*_i \in \{0,1,2,\ldots,2^{k-1}\} \} \quad (4)$$

where $n^* < M_c \times N_c$

Mapping between the secret message $M = \{ m_i \}$ and the embedded message $M^* = \{ m^*_i \}$ is defined as:

$$m^*_i = \sum_{j=0}^{k-1} m_{i \times k+j} \times 2^{k-1-j} \quad (5)$$

A subset of n^* pixels $\{ x_{11}, x_{12}, \ldots, x_{1n} \}$ is chosen from the cover image C in a predefined sequence. Embedding is done by replacing the LSBs of x_{1i} by m^*_i. mathematically, the pixel value x_{1i} of the chosen pixel for storing the message m^*_i is modified to form the stego pixel x^*_{1i} as follows:

$$x^*_{1i} = x_{1i} - x_{1i} \bmod 2^k + m^*_i \quad (6)$$

The embedding algorithm is changed slightly to be adapted to RGB color images. The mathematical models (1) to (6) are used for each color channel pixels. The 8-bits of the secret text message are divided into three parts for embedding into color channels with the sequence, three bits in the red color channel, three bits in the green color channel, and two bits in the blue color channel. It must be noted that the authorized receivers must have the same color channels 8-bits order used for embedding the secret text message into the color cover image to be able to extract the secret message [1,5-8]. Fig. 1. shows the LSB in RGB color cover image.

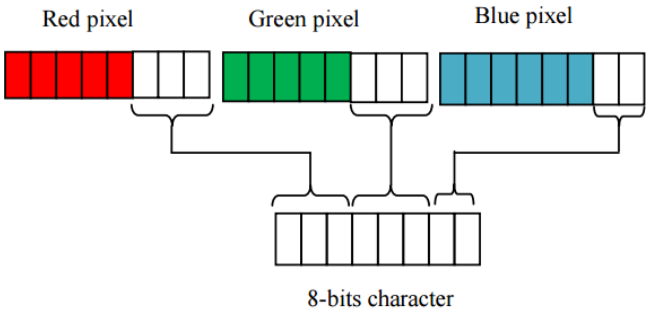


Fig. 1. LSB in RGB color cover image.

B. LSB on Cover Image Crop

LSB on cover image crop is applied according to [4]. It is based on extracting a crop from the cover color image with predefined secret coordinates. The secret text message is embedded into this crop using LSB approach explained in section *A*. The stego image is given by reassembling the image and the stego crop. It must be noted that authorized receiver must have the cover image crop coordinates and the bits order into color channels to extract the secret message [2-4]. Fig. 2. (a) shows the cover image and Fig. 2 (b) displays the cover image crop that the secret message is embedded into.

(a) (b)

Fig. 2. (a) cover image, (b) cover image crop.

III. The Proposed Approach

The goal of the proposed approach is to ensure that no one except the authorized viewer can monitor or extract the secret text message. This approach is based on cropping the cover image into a predefined number of crops with certain secret coordinates (eg. four crops). The secret text message is divided into parts with the same cover image crops (four parts). Each secret text message part is embedded into an image crop by a secret sequence using the LSB approach. Embedding is done in color channels by the sequence, three bits in the red color channel, three bits in the green color channel, and two bits in the blue color channel. Finally, the stego crops are assembled to get the stego image. It must be noted that the secret message cannot be extracted without knowing the cover image stego crops coordinates, and the bits sequence into color channels. Fig. 3. (a) shows the cover image and Fig. 3. (b) displays the four crops that the secret message are embedded into.

(a)

(b)

Fig. 3. (a) cover image, (b) four crops from cover image.

LSB multi-crop color image embedding algorithm:

Step1 Read the cover RGB color image.
Step2. Read the secret text message.
Step3. Divide the secret message into four parts.
Step4. Extract four crops from the cover image with certain coordinates.
Step5. Convert each part from the secret text message into binary format.
Step6. Embed each part of the secret message into a crop using a known sequence ordered as follows, three bits in the red color channel, three bits in the green color channel, and two bits in the blue color channel.
Step7. Assemble the four stego crops and the cover image to get the stego image.

LSB multi-crop color image extracting algorithm:

Step1. Read the stego RGB color image.
Step2. Extract four crops from the stego image with the same crops coordinates in the embedding algorithm.
Step3. Read LSB from the four stego crops with the same color channels bits order in the embedding algorithm.
Step4. Convert the binary data to text.
Step5. Collect the four parts of the secret text message with the same sequence in the embedding algorithm that gives the secret text.

IV. Simulation Results

All tests have been performed using an Intel® core™i5 CPU M450 @2.4GHz with 6GB Memory, running Windows 7 64-bit operating system and using MATLAB 8. The image used is an RGB color JPEG images with size 512 × 512, resolution 96 × 96 dpi and bit depth 24. There are four main tests to determine the performance of any stegnanography approach; visual test to determine any degradation in quality or colors compared to original image, the Peak Signal-to-Noise Ratio (PSNR) of the stego image, the embedding algorithm CPU time, and the secret text message extraction complexity.

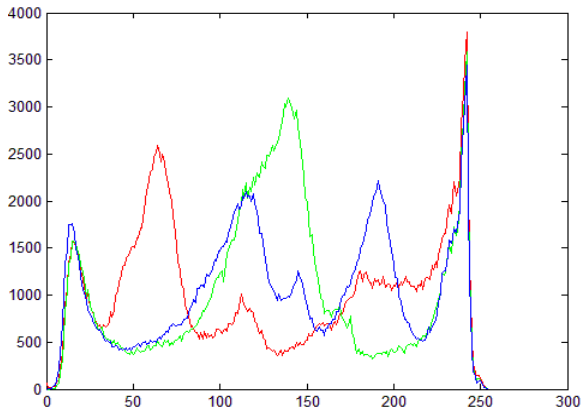
The PSNR can be calculated as follows:

$$MSE = \frac{1}{M \times N} \sum_{x=0,y=0}^{M-1,N-1} (A_s(x,y) - A(x,y))^2 \quad (7)$$

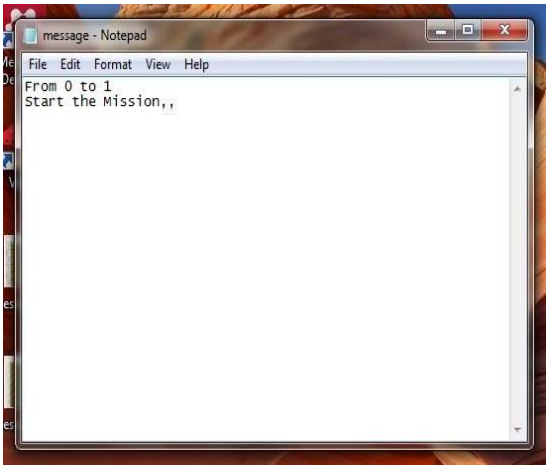
$$PSNR\ (DB) = 10\, log_{10} \frac{255^2}{MSE} \quad (8)$$

where A is the original image, A_s is the stego image, and M, N are the dimensions of the original and the stego image. It must be noted that the highest PSNR result is the best result. Visualization test results for the proposed approach compared with other state-of-the-art approaches are shown in Fig. 5. The PSNR and the embedding algorithm CPU time are shown in Table I and Fig. 6.

(a)



(b)



(c)

Fig. 4. (a) Cover image (Rokayya), (b) cover image histogram, (c) text file message.txt.

TABLE I. PSNR AND CPU TIME FOR THE PROPOSED APPROACH AND OTHER STATE OF THE ART APPROACHES.

	LSB	LSB on crop	Proposed
PSNR	62.5699	63.3480	62.5332
CPU time (sec)	0.6552	0.39	0.4524

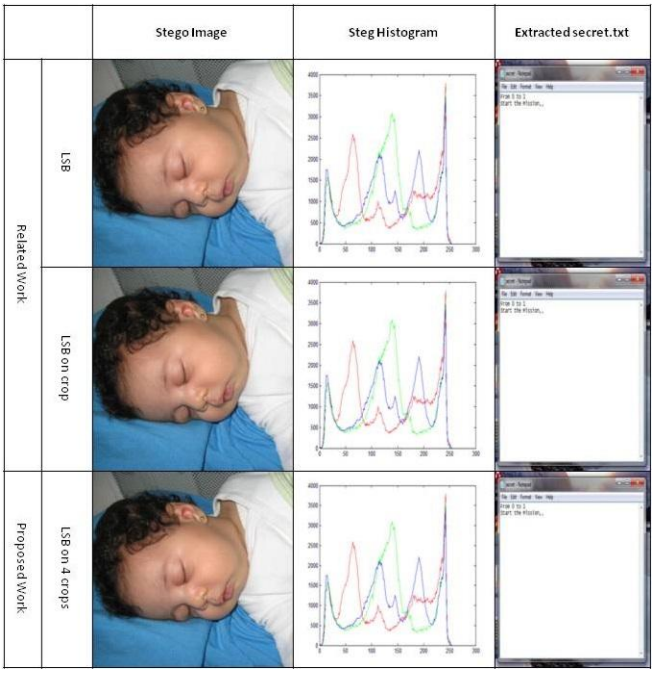
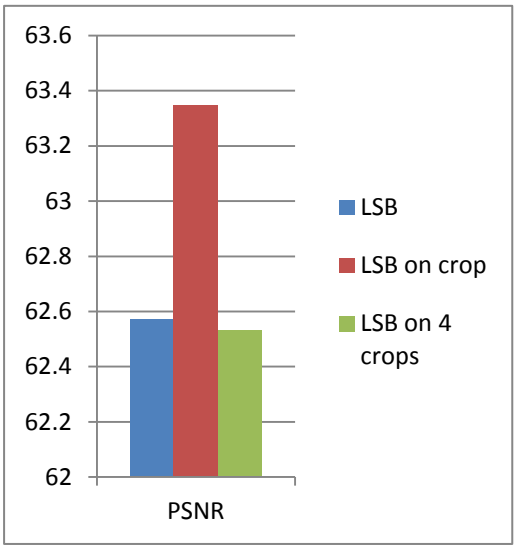
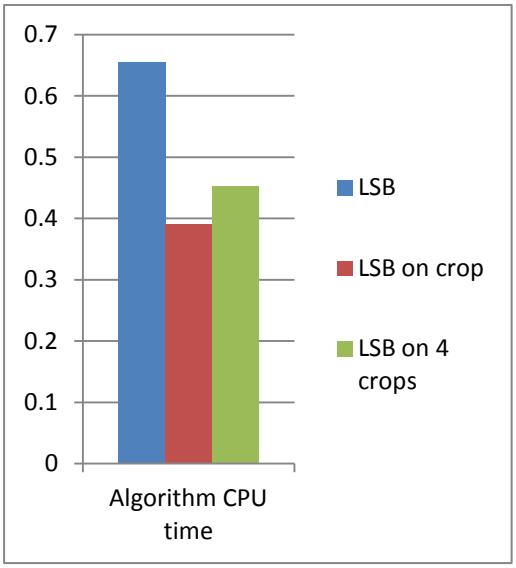


Fig. 5. Visualization test results for the proposed approach compared with other state of the art approaches.



(a)



(b)

Fig. 6. (a) PSNR for stego image, (b) CPU time in seconds.

As shown from the visualization test, the stego images for the proposed approach and the other compared approaches in this paper do not have any degradation in quality. The evaluation results illustrate that the PSNR of the stego image for the proposed approach is in the range of the other state-of-the-art approaches. The CPU time for the proposed approach is in the range of the other state-of-the-art approaches, the difference is a fraction of a second. Extracting the secret text message in the proposed approach is more complex, secure, and it could not be monitored by unauthorized users.

It must be clarified that the proposed approach here is based on multi-crops. So it is possible to choose any number of crops and it can be used to hide images or texts by converting them into bit stream and then embedding this stream into the cover image using the proposed approach in this paper.

It must be noted that increasing the number of crops means increasing the CPU time and decreasing the PSNR.

V. CONCLUSIONS

This paper proposed a highly secure data hiding approach using cropping image and Least Significant Bit (LSB) steganography. It is based on dividing the secret text message into four parts and extracting four crops from the cover color image with certain secret coordinates. Each message part is embedded into an image crop using a predefined secret sequence. The crops reassembled with the cover image giving the stego image. This proves to be a more secure method for data hiding and at the same time more complex in secret data extraction. The experimental results demonstrated that the proposed approach PSNR and CPU time are within the same range of other similar approaches, yet it proved to be more secure.

References

[1] Jassim, Firas A., "A novel steganography algorithm for hiding text in image using five modulus method", arXiv preprint arXiv, Vol. 72, No. 17, PP. 39-44, 2013.

[2] Bandyopadhyay, Debiprasad, et al., "A Novel Secure Image Steganography Method Based on Chaos Theory in Spatial Domain", International Journal of Security, Privacy and Trust Management (IJSPTM), Vol. 3, No. 1, PP. 11-22, 2014.

[3] Jain, Nitin, Sachin Meshram, and Shikha Dubey., "Image Steganography Using LSB and Edge–Detection Technique", International Journal of Soft Computing and Engineering (IJSCE), Vol. 2, No. 3, PP. 217-222, 2012.

[4] RAKHI1 & VIJAY PRAKASH SINGH., " DATA HIDING IN SKIN TONE OF IMAGES USING STEGANOGRAPHY", International Journal of Electronics and Communication Engineering (IJECE), Vol. 2, No. 4, PP. 105-112, 2014.

[5] Goel, Stuti, Arun Rana, and Manpreet Kaur., "Comparison of image steganography techniques", International Journal of Computers and Distributed Systems, Vol. 3, No. 1, PP. 20-30, 2013.

[6] Lwin, Thandar, and SUWAI PHYO., "Information Hiding System Using Text and Image Steganography", International Journal of Scientific Engineering and Technology Research, Vol. 3, No. 4, PP. 1972-1977, 2014.

[7] Krati Vyas, B.L.Pal, "A PROPOSED METHOD IN IMAGE STEGANOGRAPHY TO IMPROVE IMAGE QUALITY WITH LSB TECHNIQUE", International Journal of Advanced Research in Computer and Communication Engineering, Vol. 3, No. 1, PP. 5246-5251, 2014.

[8] Rawat, Deepesh, and Vijaya Bhandari., "Steganography Technique for Hiding Text Information in Color Image using Improved LSB Method", International Journal of Computer Applications, Vol. 67, No. 1, PP. 22-25, 2013.

[9] Thiyagarajan, P., G. Aghila, and V. Prasanna Venkatesan., "Stego-Image Generator (SIG)-Building Steganography Image Database", Advances in Digital Image Processing and Information Technology Springer Berlin Heidelberg, PP. 257-267, 2011.