Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IMPACT FACTOR: 6.017

IJCSMC, Vol. 6, Issue. 3, March 2017, pg.138 – 147

A Secure and Efficient Method of Hiding-Extracting Secret Message

Akram Naser Adam Aldakari, Prof. Hamza Al-Sewadi
Middle East University, Jordan-Amman

***Abstract:** This paper introduces a secure and efficient method of hiding-extracting secret message (SAEMHESM), this method is based on generating a private random key to hide/extract the secret message in/from color image, and the color image will be encrypted/decrypted color image using the gray image equivalent without losing the colors.*

The proposed SAEMHESM will be tested and implemented, and the results will be compared with other results obtained by other methods of hiding/extracting secrete message to insure that the proposed SAEMHESM method is more secure and can avoid hacking, and more efficient by minimizing the time needed for hiding and extracting the secrete message.

***Keywords:** Secrete message, Private Key, speedup, hiding time, extracting time, hacking time, holding image.*

1- Introduction

A secure message transferring and retrieval is a very important concern in data security and a secure method of data transmission is the needed of every time. A number of techniques and methods have been proposed and used for data security[1], and these method are facing some disadvantages like the poor security and lack of efficiency because of the big times needed to hide and extract the secrete message.

SAEMHESM a technique of embedding secrete message into digital color covering image without causing perceptual degradation of the color image when transmitting the secrete message and extracting the same message after retrieving the covering image [2] and [3].

Watermarking method of hiding/extracting secrete message is a process of hiding secrete data in a covering color image, where hidden message does not need to have any defined relation to the covering image [4] , [5] and [6].

Hiding or extracting secrete message in or from covering color image requires a watermark or a logo which is used as a secrete key [7]. The secrete key is symmetric and known by both the sender and receiver [6] and [7].

One of the easiest methods of secret message hiding and extracting in and from covering color image is the least significant bit (LSB). LSB method can be used to hide a secrete message in a covering color image, this involves replacement of LSB's of the covering image pixels with secrete message bits.

LSB method uses a standard mathematical procedure to hide or to extract the secrete message in or from the covering image, thus the LSB bits can be easily detected using nay hacking program, which means that LSB method has a very poor security level [3], and [7].

2- The proposed SAEMHESM method of steganography

Before describing the proposed method let use define some terminologies used in the method:

- Secrete message: A set of characters to be inserted into a color image and retrieved when needed, this message can be used as an author or the holder of the original color image. Here we can also use the color image to protect the message from the hackers.

- Private Key: One column matrix with size equal secretes message size. This key is to be generated randomly and converted to position values in the holding image.

- Hiding time: Time in seconds to insert the secrete message in color image.

- Extracting time: Time in seconds to extract the original secrete message from color image.

- Speedup: A value that describes the efficiency of the proposed method and it reflects how much the time was minimized and it is equal the result of division the time needed by another method of message hiding by the time needed to the proposed method, for efficient method this value must be greater than one.

- Hacking time: Time needed to unauthorized person (Hacker) to extract the secrete message.

- Holding image: Any color image with any size and any type(jpg, png, tif)

The proposed SAEMHESM method can be implemented in 2 phases:

1. Phase 1: Hiding secrete message in color image.

2. Phase 2: Extracting secrete message from holding color image.

Phase 1:

Hiding secret message:

This phase can be implemented applying the following steps:

1) Get the covering image (CI) and the secret message (SM).

2) Find the size of CI (SCI) and the length of the secrete message (LSM).

3) Generate a one column random key (RK) with size =LSM.

4) Calculate the private key (PK) by taking the fix value of the results of multiplication RK and SCI.

5) Save PK to be used in extraction sub phase.

6) Reshape CI to 1 column image (RI).

7) Use PK elements as a positions in covering image (RI) to hide a secrete message.

8) Reshape RI back to CHI.

9) Save the covering holding image (CHI).

Phase 2:

Extracting secret message:

This phase can be implemented applying the following steps:

1) Get the covering holding image (CHI).

2) Load PK.

3) Reshape CHI to 1 column image (RHI).

4) Use PK elements values as positions in RHI to retrieve the message.

3- Implementation and experimental results

The proposed method was implemented using matlab v7.0 and the results were compared with the results of implementing LSB and watermarking methods. Various secrete messages with different length were used and various holding images with different sizes and types were taken, figure 1 shows the original and the holding images using a secrete message of 100 byte length and 750*975*3 png image using the proposed method.

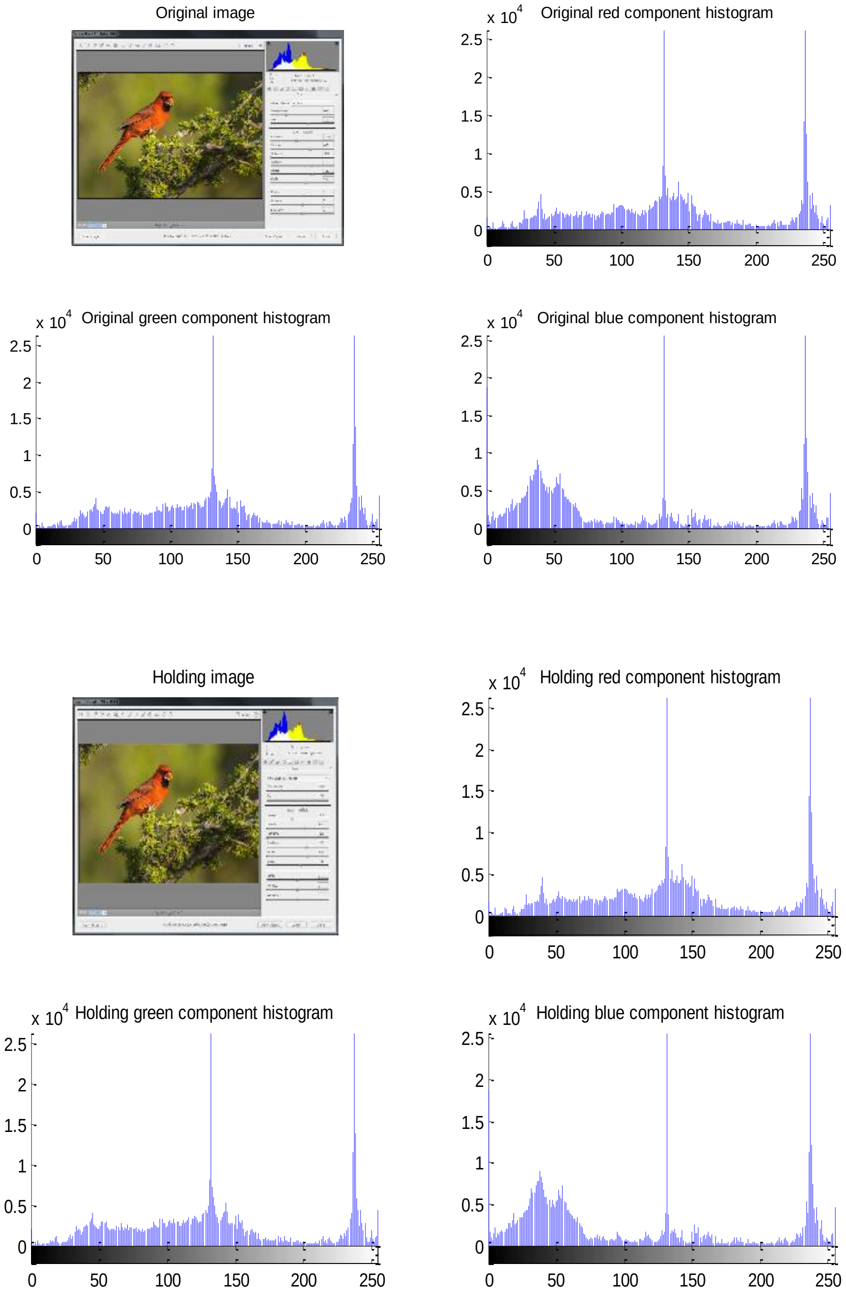

Figure 1: Original and holding images

The three methods were implemented using different messages and different images the results of implementations are shown in tables 1 through 4.

Table 1: Results1 (Image size=1655x 2498x3(jpg))

Text size(Byte)	LSB (1)		Watermarking(key=123) (2)		Proposed (3)	
	Hiding time(Seconds)	Extracting time	Hiding time(Seconds)	Extracting time	Hiding time(Seconds)	Extracting time
10	0.193000	0.134000	0.092000	0.071000	0.0060	0.0010
20	0.193000	0.134000	0.092000	0.071000	0.0060	0.0010
30	0.193000	0.134000	0.092000	0.071000	0.0060	0.0010
40	0.193000	0.134000	0.092000	0.071000	0.0060	0.0010
50	0.193000	0.134000	0.092000	0.071000	0.0060	0.0010
60	0.193000	0.134000	0.092000	0.071000	0.0060	0.0010
70	0.193000	0.134000	0.092000	0.071000	0.0070	0.0010
80	0.193000	0.134000	0.092000	0.071000	0.0070	0.0010
90	0.193000	0.134000	0.092000	0.071000	0.0070	0.0010
100	0.193000	0.134000	0.092000	0.071000	0.0070	0.0010
Speed up (3) with others Taking a message with 100 byte	27.5714	134	13.1429	71.0000	1	1

Table 2: Results2 (Image size=750x 975x3(png))

Text size(Byte)	LSB		Watermarking(key=123)		Proposed	
	Hiding time(Seconds)	Extracting time	Hiding time(Seconds)	Extracting time	Hiding time(Seconds)	Extracting time
10	0.115000	0.052000	4.800000	4.800000	0.001017	0.001017
20	0.115000	0.052000	4.800000	4.800000	0.001017	0.001017
30	0.115000	0.052000	4.800000	4.800000	0.001017	0.001017
40	0.115000	0.052000	4.800000	4.800000	0.001017	0.001017
50	0.115000	0.052000	4.800000	4.800000	0.001017	0.001017
60	0.115000	0.052000	4.800000	4.800000	0.001017	0.001017
70	0.115000	0.052000	4.800000	4.800000	0.001017	0.001017
80	0.115000	0.052000	4.800000	4.800000	0.001017	0.001017
90	0.115000	0.052000	4.800000	4.800000	0.001017	0.001017
100	0.115000	0.052000	4.800000	4.800000	0.001017	0.001017
Speed up (3) with others Taking a message with 100 byte	113.0777	51.1308	4.7198e+003	4.7198e+003	1	1

Table 3: Results3 (Image size=516x 600x3(jpg))

Text size(Byte)	LSB		Watermarking(key=123)		Proposed	
	Hiding time(Seconds)	Extracting time	Hiding time(Seconds)	Extracting time	Hiding time(Seconds)	Extracting time
10	0.071000	0.041000	0.177000	0.155000	0.000170	0.000170
20	0.071000	0.041000	0.177000	0.155000	0.000170	0.000170
30	0.071000	0.041000	0.177000	0.155000	0.000170	0.000170
40	0.071000	0.041000	0.177000	0.155000	0.000170	0.000170
50	0.071000	0.041000	0.177000	0.155000	0.000170	0.000170
60	0.071000	0.041000	0.177000	0.155000	0.000170	0.000170
70	0.071000	0.041000	0.177000	0.155000	0.000170	0.000170
80	0.071000	0.041000	0.177000	0.155000	0.000170	0.000170
90	0.071000	0.041000	0.177000	0.155000	0.000170	0.000170
100	0.071000	0.041000	0.177000	0.155000	0.000170	0.000170
Speed up (3) with others Taking a message with 100 byte	417.6471	241.1765	1.0412e+003	911.7647	1	1

Table 4: Results4 (Image size=320x 450x3(jpg))

Text size(Byte)	LSB		Watermarking(key=123)		Proposed	
	Hiding time(Seconds)	Extracting time	Hiding time(Seconds)	Extracting time	Hiding time(Seconds)	Extracting time
10	0.057000	0.037000	0.607000	0.607000	0.000120	0.000120
20	0.057000	0.037000	0.607000	0.607000	0.000120	0.000120
30	0.057000	0.037000	0.607000	0.607000	0.000120	0.000120
40	0.057000	0.037000	0.607000	0.607000	0.000120	0.000120
50	0.057000	0.037000	0.607000	0.607000	0.000120	0.000120
60	0.057000	0.037000	0.607000	0.607000	0.000120	0.000120
70	0.057000	0.037000	0.607000	0.607000	0.000120	0.000120
80	0.057000	0.037000	0.607000	0.607000	0.000120	0.000120
90	0.057000	0.037000	0.607000	0.607000	0.000120	0.000120
100	0.057000	0.037000	0.607000	0.607000	0.000120	0.000120
Speed up (3) with others Taking a message with 100 byte	475	308.3333	5.0583e+003	5.0583e+003	1	1

From the obtained experimental results we can see that the proposed method gives a higher efficiency comparing with the other two methods for both hiding and extracting secrete messages with different sizes in a holding images with different sizes and types.

As mentioned in [7] LSB and watermarking methods provide a poor security levels. The security level of using the proposed method depends on the following factors:

1) The secrete message length.
2) The holding image size.
3) The private key is to be randomly generated.
4) The maximum number in the private key matrix
5) The number of elements in the private key.

Suppose we want to hide a message of 5 characters in a holding image with size equal 1655* 2498*3, the proposed method will generate a private key likes (it may be changed):

12230891 5870666 11197273 5594287 9978076

Message character	Position in the reshaped holding image with size(12402570x1)
1	12230891
2	05870666
3	11197273
4	05594287
5	09978076

For simplicity of calculation let us consider the message of one character then:

The worst number of hacking processes=1

The best number of hacking processes=10^8.

The average number of hacking processes=(1+10^8)/2 =5*10^7

Hacking time=5*10^7*0.0010=50000 seconds

=833.3333 minutes

= 13.8889 hours

This time will rapidly increased for years by increasing the secrete message length.

Also we can achieve higher levels of security by encryption/decryption the holding image as proposed in [8] and [9].

Conclusions

A method of hiding/extracting secrete message in holding cover image was proposed; the experimental results showed that the proposed method provides a higher efficiency than LSB and watermarking methods.

The proposed method provides a high security level by making the process of secrete message hacking impossible especially for messages with long lengths.

References

[1] Mekha Jose, Hiding Image in Image Using LSB Insertion Method with Improved Security and Quality, International Journal of Science and Research (IJSR) ISSN (Online): 2319-7064.

[2] Reena M Patel,D J Shah , "Concealogram : Digital image in image using LSB insertion method", International journal of electronics and communication engineering & technology(IJECET), 2013.

[3] Nadeem Akhtar, Pragati Johri, Shahbaaz Khan, "Enhancing the security and quality of LSB based image steganography", 2013 5th International Conference on Computational Intelligence and Communication Networks.

[4] Mamta.Juneja , Parvinder S. Sandhu, "An improved LSB based Steganography with enhanced Security and Embedding/Extraction", 3rd International Conference on Intelligent Computational Systems (ICICS'2013) January 26-27, 2013 Hong Kong (China)J.

[5] S. M. Masud Karim, Md. Saifur Rahman, Md. Ismail Hossain, "A New Approach for LSB Based Image Steganography using Secret Key", Proceedings of 14th International Conference on Computer and Information Technology (ICCIT 201 I) 22-24 December, 201 I, Dhaka, Bangladesh.

[6] Morkel T., Eloff J. H. P., and Olivier M. S., "An Overview of Image Steganography", Information and Computer Security Architecture (ICSA) Research Group, University of Pretoria, South Africa, 2005.

[7] Ms. Nidhi Bux , Prof. K. J. Satao, Implementation of Watermarking Technique for Secured Transmission, International Journal of Advanced Research in Computer and Communication Engineering Vol. 4, Issue 8, August 2015.

[8] Jihad Nadir, Ziad Alqadi and Ashraf Abu Ein, Classification of Matrix Multiplication Methods Used to Encrypt-decrypt Color Image, International Journal of Computer and Information Technology (ISSN: 2279 – 0764) Volume 05 – Issue 05, September 2016.

[9] Majed O. Al-Dwairi, Ziad A. Alqadi, Amjad A. AbuJazar and Rushdi Abu Zneit, Optimized True-Color Image Processing, World Applied Sciences Journal 8 (10): 1175-1182, 2010 ISSN 1818-4952.