Journal of Theoretical and Applied Information Technology
15th July 2018. Vol.96. No 13

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A NOVEL ZERO-ERROR METHOD TO CREATE A SECRET TAG FOR AN IMAGE

[1]JAMIL AL-AZZEH, [2]BILAL ZAHRAN, [3]ZIAD ALQADI, [4]BELAL AYYOUB AND [5]MAZEN ABU-ZAHER

[1]Associated Professor, PhD Degree in Computer Engineering, Computer Department, Al Balqa Applied University, Jordan
[2]Associated Professor, PhD Degree in Computer Engineering, Computer Department, Al Balqa Applied University, Jordan
[3] Professor, Computer Department, PhD Degree in Computer Engineering, Al Balqa Applied University, Jordan
[4]Assossiated Professor, PhD Degree in Computer Engineering, Computer Department, Al Balqa Applied University, Jordan
[5]Teacher. Computer Department, Master Degree in Computer Engineering ,Al Balqa Applied University, Jordan
E-mail: [1]jamil.azzeh@bau.edu.jo, [2] zahranb@yahoo.com, [3]natalia_maw@yahoo.com, [4]zahranb@yahoo.com, [5] mazen.abuzaher@fet.edu.jo.

ABSTRACT

Nowadays, privacy is a demand for every one specially when sending personal information over the internet using social media sites and applications. One way to ensure this privacy is by hiding our personal or secret information inside an image (cover image). Only how knows the hiding process can extract the information from the cover image. To ensure that the cover image must preserve its quality as possible so that no one doubt about it. The process of hiding a text in gray and color image (creating a tag) is used in many important applications. Generally, most methods have been proposed to solve the issue depend on least significant bit (LSB) criteria. LSB methods have a set of negatives and defects in addition to lack of safety of these methods, error ratio between the original image and the text-bearing image ranges from small to large; If the error rate is high, this will lead to distortion in the image which can be observed by the naked eye of the human. Accordingly, to reduce error rate the text message size must be small; the size of the hidden text in the image depends on the size of the image. In this paper, we introduced a novel approach to hide any text independently on the size of cover-image and make the size of the text unrestricted "as it can be larger than the cover-image size". Our approach provided a high degree of safety with zero error ratio regardless of the size of the text.

Keywords: *LSB, short message, covering image, PSNR, MSE.*

1. INTRODUCTION

There is a big difference between the hiding and encryption of information. In the first, the information is hidden so that the user will not be aware of and aware of the existence of that information, but in encryption, the user is aware that there is hidden information but coded incomprehensible. So the most appropriate way to build a strong protection system is to rely on the two technologies to make the process of penetrating the information more complex. This wonderful science has many applications that differ by the type of inclusion used (we will talk about the species in the next section). In this section we will review some of those applications in general. The idea of hiding messages and encryption has been used in wars between countries such as the Second World War, where messages are exchanged between the army in a way that appears to be normal for the enemy army, but it carries embedded meaning that can be understood only by those who are in the same Army. In addition spies use these techniques to send their messages without any doubt about its real contents. Other application is to protect the intellectual property of all types of electronic files by using Watermarking you can prove that you are the official owner of the image,

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

sound or video file. The watermark means adding certain information to the medium so that these additions do not affect its quality. With the expansion of digital media, the use of watermarks is becoming very common. It can be used, for example, to add a certain mark to a group of images that will be published on a website. This signifies the owner of these images to protect the copyright of this owner. The process of communication between people is one of the most important means that helped human growth. This process requires the confidentiality of transferred data in absolute secrecy.

Steganography is one of the methods used by humans in the field of data concealment. This method is used in Greek. Steganography means "covered" or "restricted". The emergence of computers, the speed of data transfer within networks and the possibility of transferring different types of data have allowed the possibility of concealing a particular type of data within another type, thus giving a higher degree of protection.

Whatever your purpose is: If you need to keep the confidentiality of a text especially when conducting the process of textual transmission over the Internet, give important images a secret tag that can be used to denote ownership of the image, discover the process of forging important documents by reference to the hidden text in the document. Or use the color image tag as an identifier or key to retrieve the color image [2- 4]; then one of the known methods that can be used is Steganography.

In earlier studies, a number of algorithms were proposed to hide text within gray images, some of which relied on the direct concealment of data on the image where part of the image was replaced by a secret message. These algorithms were characterized by high data storage, but were not resistant to sabotage. Other algorithms based on transforming the image into another form using the separate cosine functions and then embedding the secret message into the new image format [1]. These algorithms were resistant to sabotage, but they could not conceal a large volume of data.

The general mechanism of masking text message in any picture (steganography) is summarized as shown in Figure 1 [5, 6]. Basically Any text message is a set of letters and symbols whose number is specified by American Standard Code for Information Interchange "ASCII" code and each code is usually a value between zero and 255.

In this paper our work performs text hiding process inside a Gray or color image depending on ASCII code. The rest of this paper show some related work, our approach steps in addition to detailed comparison between our approach and another two famous approaches in term of error ratio and speed of processing.

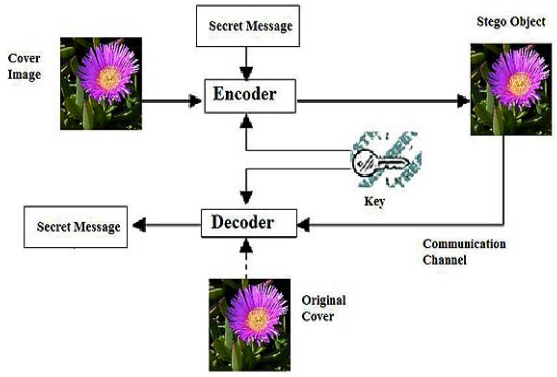

Figure 1: Mechanism Of Hiding Text In Color Image

2. RELATED WORK

Steganography is the art hiding messages in such a way that no one except sender and intended recipient can realize there is a hidden message. There are currently several ways to hide any text message in a color image; most of these methods are based on least significant bit (LSB) [16]. A complete survey of most known steganography methods can be found in [17, 18].

LSB dependent approaches are popular because that it is easy to implement it, despite its several disadvantages. To hide a secret message inside an image, a proper cover image is needed. LSB methods use bits of each pixel in the image, accordingly if we want to compress the cover

Journal of Theoretical and Applied Information Technology
15th July 2018. Vol.96. No 13

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

image it is necessary to use a lossless compression format, otherwise the hidden information will get lost in the transformation of a lossy compression algorithm. Using Gray images each pixel has one value which means that we can hide just one bit. On the other hand with 24-bit color image each pixel has three values Red, Green, Blue color component which can be used, so a total of 3 bits can be stored in each pixel.

The main disadvantage of standard LSB methods it is easy to detect the hidden message since it is simply the least significant bit in each pixel. To overcome this disadvantage many algorithms have been proposed to enhance LSB and make it less detectable and more secure [19, 20]. Other methods [21, 22], try to increase the amount of data that can be hiding in the cover image and pre-encrypt the message before hide it in the cover image.

Next two subsections discuss in detail the basic operation of two LSB standard methods. Each method has been implemented using messages with different length. The implementations record four values (mean square error, peak signal to noise ratio, hiding time, and retrieving time) that we will use to compare with our algorithm.

2.1- Least significant four bits method

In this method, the four least significant bits of a pixel in an image is used to store the lower half of the ASCII letter and the second half of the letter value is stored in the four least significant bits of the next pixel, so one ASCII code requires 2 pixels of a Gray image and one pixel for a color image to hide it and perform the hiding process such as shown in Figure 2 [7]. Extracting the letter from the image can be done as shown in Figure 3 [8, 9].

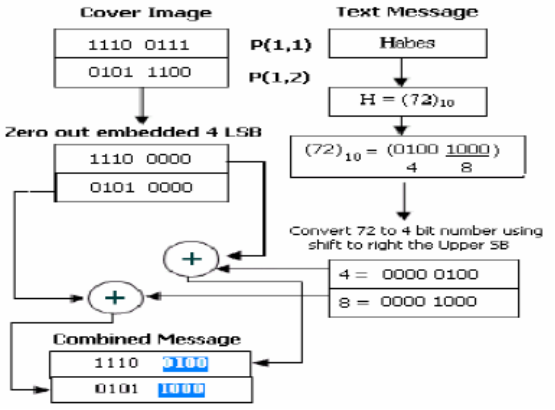

Figure 2: Hiding Using Least Significant Four Bits Method

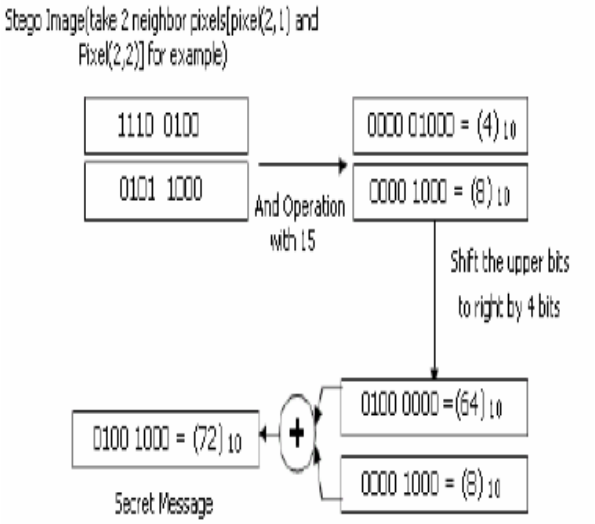

Figure 3: Extracting The Letter Using Least Significant Four Bits Method

2.2- Least significant bit method

The least significant bit (LSB)[13, 14] method came to reduce the distortion in the image by reducing the value of the mean square error (MSE) [8, 12] and increasing the value of peak signal to noise ratio (PSNR)[12, 13]. Having a reconstructed image we can decide the quality of it compared to the original image using MSE and PSNR measurements.

Mean Squared Error method is a way to measuring the similarity of tow images. It is compute an error signal by subtracting the test image from the reference, and then computing the average value of

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

the error signal. The mean-squared-error (MSE) is the simplest, and the most widely used.

One problem with mean-squared error is that it depends strongly on the image intensity scaling. Peak Signal-to-Noise Ratio (PSNR) avoids this problem by scaling the MSE according to the image range. Using PSNR, the higher the PSNR, the better the quality of the reconstructed image.

In Least significant bit method, 8 pixels are allocated for each letter code in the text message so that each bit of the text message is stored in the lower bit of the pixel. Thus, the size of the message here will be based on the size of the image divided by 8 in Gray image and divided by 3 in color image. Using these criteria we reduce the MSE value between the original picture and the hiding image. Because we change only one bit in pixels and the amount of change is only one or zero and the pixel value after loading is very close to the pixel value before loading. In this case it is difficult to distinguish the difference between the original image and the bearing image [10, 11]. Figure 4 shows the process of inserting the letter V in the Gray image pixels using LSB method.

A: The series of cover bytes before the embedding process

0	1	2	3
10111000	10000001	10100001	01111100
4	5	6	7
01110110	10010000	11010011	01010010

B: The resulting bytes after the embedding process

0	1	2	3
1011100(0)	1000000(1)	1010000[0]	0111110[1]
4	5	6	7
011101(0)	1001000[1]	1101001(1)	0101001(0)

A: Cover bytes before the embedding
B: Resulting bytes after the embedding
(=) The bit new value is identical to its original
[≠] The bit new value is different from its original

Figure 4: Hiding The Letter V

This method was implemented using various short and long messages with Gray and color cover image. The results of implementation are shown in Tables 1 for short messages and Table 2 for long messages. Table 1: LSB results for short messages, Table 2: LSB results for long messages.

We can note from the above results the following facts:

1 - There is an error and permanently no matter how long the message.

2- The error value increases by increasing the length of the message

3 - The image becomes distorted when hiding a long message and this distortion becomes visible and can be distinguished by the naked eye, which points that the image contains a hidden message.

4- Cover image size must be calculated carefully according to the text message that we need to hide. In Least significant four bits method, the cover image size must be greater than message size *2. On the other hand, with Least significant bit method cover image size must be greater than message size *8.

5- The method is unsafe because the method procedure is known.

3. THE PROPOSED METHOD

We have previously indicated that the text message contains a set of symbols whose value is limited to values from 0 to 255 which represents ASCII code that is identical to pixel value in the digital image which range from 0 “black color” to 255 “white color” for Gray images, and in color images the three values for Red, Green, and Blue components also range from 0 to 255.

Using our algorithm we need two files one represent the cover image which will contain the hidden message, the other is Tag file which contains the location of each character of the hidden message inside the cover image.

For example to hide the letter A “ASCII is 65” using our method; we should search for a pixel with value 65 which represent the letter A ASCII then store that pixel location in a tag file to return when retrieving. This indicates that no adjustment in the

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

cover image, which in turn gives the result of a zero error.

According to our method it is impossible to observe changes in the cover image with the naked eye or even with computer programs. The size of the cryptic message is unlimited because each ASCII code will be stored in the same pixel location whatever how many this code is repeated.

To insure the robustness of our method; more than 100 images have been scanned each of them has pixels of value from 0 to 255. Even though; if none of these values are available, the image can be calibrated using Histogram equalization method [15] so that the color values are naturally distributed, containing all values from 0 to 255.

Histogram equalization as shown in Figure 5 is an image processing method used to spread out the gray level values for any image so that the result image cover the overall values from 0 to 255.

The main use of histogram equalization is to enhance image appearance by adjusting its intensities. However, in our proposed method we use it to modify any image so that it covers the full range gray values from 0 to 255. Histogram equalization can be used for gray or colored image.

For gray images histogram equalization can be done in three steps [15]:

First step is computing the histogram of the image for each value from 0 to 255.

H_n = number of pixels with intensity n (1)

Where n= 0-255.

Second step is calculating the normalized sum of histogram.

P_n = number of pixels with intensity n / total number of pixels (2)

The final step is to transform the input image to an output image using the following equation. For each pixel value "k" find the output value F(k)

$$F(k) = 255 \times \sum_{n=0}^{k} p_n \quad (3)$$

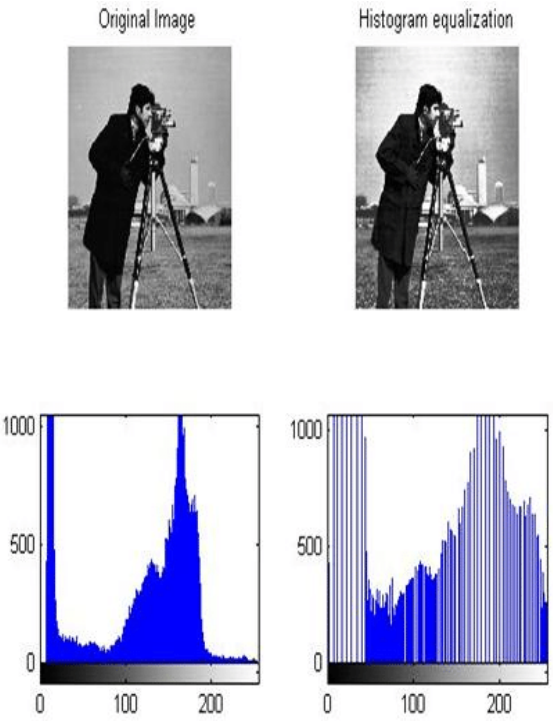

Figure 5: Histogram Equalization [15]

For RGB colored images the same steps can be done separately for each color channel.

The proposed method can be implemented in two phases:

- Phase one: Selecting the locations for the text message.
- Phase two: Using the locations as a tag (key) to identify the image by retrieving the text message.

Phase one:

This phase represents the stego generation using the selected cover image and the secret text. At the end of this stage the stego image must contain our secret message. Using tag vector which generated at this stage we can extract our secret message in phase two. Phase one can be implemented applying the following steps:

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

1. Read the cover image

2. Retrieve cover image size (number of rows, columns and colors)

3. Calibrate the image using histogram equalization "if needed"

4. Read the text message to be hidden (inserted)

5. Specify the size of the text message

6. Converting the 2D image matrix into a vector with one dimension. If the image is gray that mean the first entry in the vector is the value of the first pixel, the second entry is for the second pixel and so on till the last pixel in the image. On the other hand using color images, this mean that each pixel has three values "Red, Green, and Blue". When converting from 2D image to a vector, the first three entries in the vector will be the Red, Green, and Blue values for the first pixel, the next three entries will be for the second pixel in the image and so on.

7. Create a vector of locations that represent each value from 0 to 255

8. Take the first letter of the message and search in the vector image for the first location which contains the code of the letter and store this location in tag vector and so on for the second and third code till the last code

9. Save tag vector in separate file.

10. Convert the image from one-dimensional matrix into a two-dimensional matrix.

Phase two

This phase the secret message can be obtained from the stego image use tag vector. Phase two can be implemented applying the following steps:

1. Retrieve the cover image

2. Retrieve tag file.

3. Convert the two-dimensional image matrix into a one-dimensional matrix

4. Use values in tag file to retrieve symbols from the image.

Using the previous description, suppose we want to insert the message 'EXAMPLE' in a pre defined image, we have search the first pixels, whose values equal the letter value and keep the selected locations as a tag to identify the image or to retrieve the message, Table 3 shows the message symbols values and there locations in the image.

Table 3: Letters And There Locations

Message letter	Later value(ASCII)	Location (will be stored in tag file)
E	101	787
X	120	345
A	97	357
M	109	558
P	112	1631
L	108	294
E	101	787

We can note from the previous table that the same letter will be stored on the same location whatever how many time this letter is repeated "see letter E".

Another technique we used in our algorithm to increase security of the Tag file is to use more than one location for the same character. Suppose we have a cover image with gray values as shown in Figure 6. We want to insert the message 'EXAMPLE', first the proposed method will search the location of each character in the cover image. For a moderate image we can be sure that at least all character will has more than one location. So we will store more than one location for each character as shown in Table 4.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

5	101	30	77	70	87	97
13	100	30	99	255	210	33
55	0	20	120	100	210	112
50	17	101	108	120	108	109
17	13	97	1	4	16	109
215	101	1	0	3	112	108
255	215	111	112	70	80	4

Figure 6: Image sample

Table 4: Letters And There Multi Locations

Message letter	Later value(ASCII)	Location (will be stored in tag file)
E	101	2, 24
X	120	18, 26
A	97	7, 31
M	109	28, 35
P	112	21, 41
L	108	27, 42
E	101	2, 24

Accordingly we deduce that we can hide any message with any length without affecting the cover image appearance using Tag file. In addition if anyone gets the Tag file he cannot deduce the hidden message especially when using more one location for each character.

4. IMPLEMENTATION AND RESULTS DISCUSSION OF THE PROPOSED METHOD

To test our method messages with different length from 16 to 51200 bytes were treated. As shown in figures 7, 8, and 9, according to histogram and PSNR comparison between original image and stego images there are no changes in the image. Figure 7, Figure 8 and *Figure 9*

In addition, the proposed method was implemented using the same messages used in LSB method "see section 2.2" and the results of implementations are shown in Tables 5 and 6. As shown our method needs less time to perform hiding and retrieving process compared to LSB method. When using the proposed method MSE was always zero and PSNR was always infinite, which mean that there is no change in original image after hide the message.

Table 5: Short messages implementation

Message length (Byte)	Proposed method		LSB	
	Hiding time(sec.)	Retrieving time(sec.)	Hiding time(sec.)	Retrieving time(sec.)
16	0.014000	0.000001	0.047000	0.034000
51	0.044000	0.000001	0.044000	0.034000
61	0.056000	0.000001	0.044000	0.035000
68	0.059000	0.000001	0.044000	0.034000
81	0.071000	0.000001	0.045000	0.033000
100	0.090000	0.000001	0.045000	0.034000

Table 6: Long Messages Implementation

Message length (Byte)	**Proposed method**		**LSB**	
	Hiding time(sec.)	Retrieving time(sec.)	Hiding time(sec.)	Retrieving time(sec.)
1600	1.396000	0.008000	0.048000	0.036000
3200	2.769000	0.027000	0.050000	0.037000
6400	5.560000	0.087000	0.050000	0.041000
12800	11.019000	0.318000	0.056000	0.050000
25600	21.669000	0.667000	0.065000	0.050000
51200	44.621000	1.417000	0.084000	0.097000

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The proposed method provides high safety procedures, because it is very difficult to guess the tag file of the hidden message in the cover image. To obtain extra security we can encrypt the tag file with any know encryption technique.

5. GENERAL COMPARISON

Comparing our method with LSB methods we can list the following points:

1- In term of Steganalysis "which used to detect messages from stego images produced by steganography" [23], LSB methods hidden data can be extracted easily. On the other hand, our using algorithm there is no direct relation between cover image, tag file, and hidden message, so it is very hard to analysis the hidden message which makes the proposed algorithm more secure.

2- Cover image quality using other techniques specially LSB; degraded the cover image quality which may used as indication of the existence of hidden message. But with our method image quality will not be affected at all.

3- Existing methods must take care of cover image size compared to hidden message, so that cover image must be very large compared to stego massage. However, our method can hide any message size with the same cover image.

6. CONCLUSIONS

In literature there are many methods have been implemented to hide any secret message inside an image. The existing methods have problems with cover image coloring, size, and quality. In this paper a new method of Steganography was implemented and tested. The proposed method designed to overcome traditional Steganography methods problems, especially the LSB stego methods. Compared with traditional LSB technique, experimental results showed that our method faster and has high accuracy with zero MSE and infinite PSNR, which means no image distortion. In addition, using multi location with Tag file makes hacking process impossible. The proposed method can be implemented for both Gray and color images.

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.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[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.

[10]Gaurav Bhatnagar, Balasubramanian Raman ," A new robust reference watermarking scheme based on DWT -SVD", 0920-5489/$ –see front matter © 2008 Elsevier B.V. All rights reserved. doi:10.1016/j.csi.2008.09.031.

[11] C. De Vleeschouwer, J. F. Delaigle, a nd B. Macq, "Invisibility and application functionalities in perceptual watermarking an overview," Proceedings of the IEEE. vol. 90, pp. 64-77, 2002.

[12] Prof. Ziad A.A. Alqadi, Prof. Mohammed K. Abu Zalata, Ghazi M. Qaryouti, Comparative Analysis of Color Image Steganography, *IJCSMC, Vol. 5, Issue. 11, November 2016, pg.37 – 43*.

[13] Dr. Ashraf Abu-Ein, Prof. Ziad A.A Alqadi, Dr. Jihad Nader, A TECHNIQUE OF HIDING SECRETE TEXT IN WAVE FILE, International Journal of Computer Applications · October 2016. DOI: 10.5120/ijca2016911732.

[14] K. Matrouk, A. A. Hasanat and H. Alashalary, Prof.Ziad Al-Qadi and Prof. Hasan Al-Shalabi, "Speech fingerprint to identify isolated word person", World Appl. Sci. J.,vol. 31, no. 10, pp. 1767-1771, 2014.

[15] Rafael C. Gonzalez, and Richard E. Woods, Digital Image Processing, Prentice Hall, second edition, 2002.

[16] Cox, I. J., Miller, M. L., Bloom, J. A., Fridrich, J., Kalker, T., Digital watermarking and steganography(2nd. ed.). New York: Morgan Kaufmann, 2008.

[17] Cheddad, A., Condell, J., Curran, K., & Kevitt, P. Mc., Digital image steganography: survey and analyses of current methods signal processing. (pp. 727-752), Vol. 90, Issue 3, 2010.

[18] Katzenbeisser, S., &Petitcolas, F. A., Information hiding techniques for steganography and digital watermarking. (pp. 43-78). London: Artech House, 2000.

[19] Roque, J. J., & Minguet, J. M., SLSB: Improving the steganographic algorithm LSB. Proceedings The Ibero-American Congress on Information Security (CIBSI). (pp. 398-408), 2009.

[20] Cvejic, N., Seppanen T., Increasing robustness of LSB audio steganography by reduced distortion LSB coding. Proceedings ITCC 2004 International Conference on Information Technology: Coding and Computing, 2004.

[21] Mazen Abu Zaher, Modified Least Significant Bit (MLSB), Computer and Information Science Vol. 4, No. 1; January, 2011.

[22] Bandyopadhyay, S. K., Bhattacharyya, D. , Ganguly, D. , Mukherjeel, S. & Das, P., A tutorial review on steganography. International Conference on Contemporary Computing, 2008.

[23] Tanmoy Sarkar, and Sugata Sanyal, "Steganalysis: Detecting LSB Steganographic Techniques ", Cornell University Library, May 2014.

Journal of Theoretical and Applied Information Technology
15th July 2018. Vol.96. No 13

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 1: LSB Results For Short Messages

Message length (Byte)	LSB			
	MSE	PSNR	Hiding time(sec.)	Retrieving time(sec.)
16	0.0023	171.5483	0.047000	0.034000
51	0.0075	159.6936	0.044000	0.034000
61	0.0100	156.9268	0.044000	0.035000
68	0.0118	155.1881	0.044000	0.034000
81	0.0130	154.2309	0.045000	0.033000
100	0.0166	151.8106	0.045000	0.034000

Table 2: LSB Results For Long Messages

Message length (Byte)	LSB			
	MSE	PSNR	Hiding time(sec.)	Retrieving time(sec.)
1600	0.2582	124.3661	0.048000	0.036000
3200	0.5119	117.5207	0.050000	0.037000
6400	0.9336	111.5123	0.050000	0.041000
12800	0.9336	111.5123	0.056000	0.050000
25600	0.9336	111.5123	0.065000	0.050000
51200	0.9336	111.5123	0.084000	0.097000

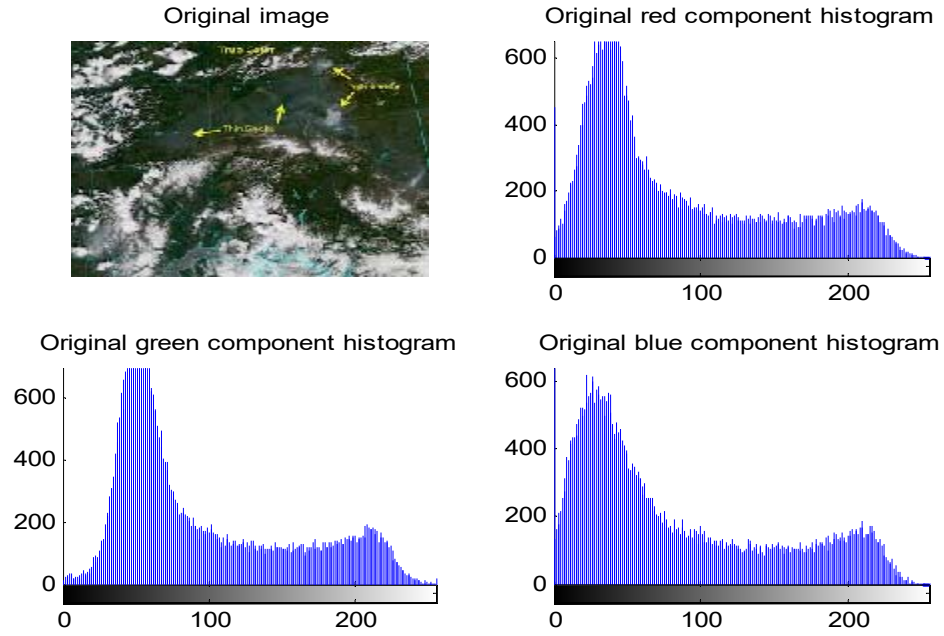

Figure 7: Original Image

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

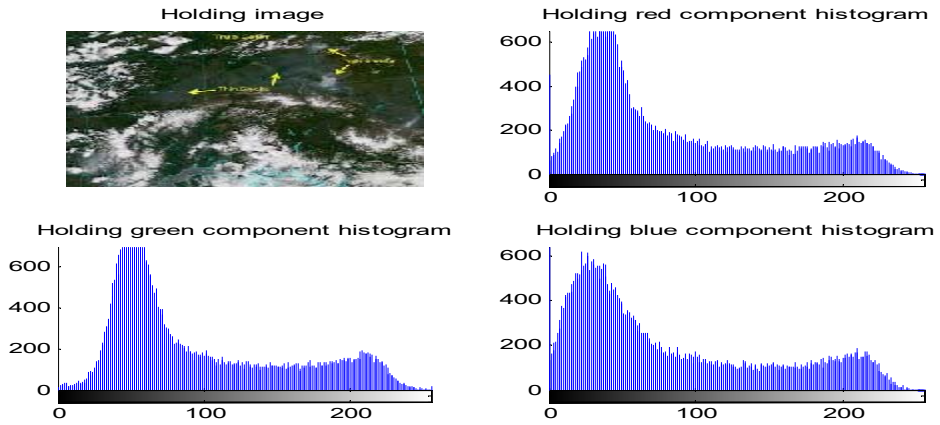

Figure 8: Image After Inserting 16 Bytes Message

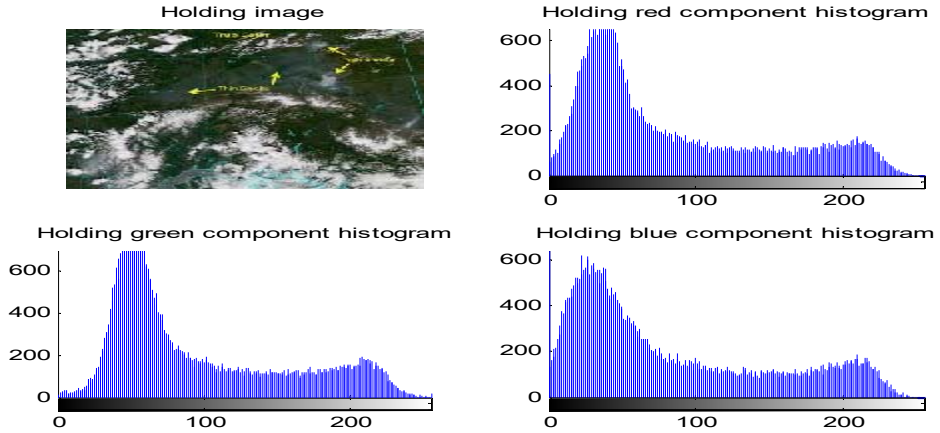

Figure 9: Image After Inserting 51200 Bytes Message