Steganography: Applying and Evaluating Two Algorithms for Embedding Audio Data in an Image

Khaled Nasser ElSayed
Computer Science Department, Umm Al-Qura University

***Abstract*—Information transmission is increasing with grow of using WEB. So, information security has become very important. Security of data and information is the major task for scientists and political and military people. One of the most secure methods is embedding data (steganography) in different media like text, audio, digital images. This paper presents two experiments in steganography of digital audio data file. It applies empirically, two algorithms in steganography in images through random insertion of digital audio data using bytes and pixels in image files. Finally, it evaluates both experiments, in order to enhance security of transmitted data.**

Keywords—Steganography; Encryption and Decryption; Data and Information Security; Data Hiding; Images; Data Communication

I. Introduction

Nowadays, data and information have become the most important hot issue. Technology of transmission and communication of data and information between sites in the same country or overseas are done through LANs, WANs, or WEB networks. This process is done through leased lines, microwave, or satellites. From academic wise, information security is the science that cares and searches in the theories and strategies of providing information protection against violation of unauthorized peoples. From technology wise, information security is tools and procedures needed to ensure (grant) information protection from inside and outside dangers. From law wise, information security is the process and studies performed to protect the security and integrity of data and information against its violations.

Data hiding or embedding refers to the nearly invisible embedding of information within a host data set such as text, image, or video [1], [2]. In steganographic applications, the hidden data are a secret message whose mere presence within the host data set should be undetectable; a classical example is that of a prisoner communicating with the outside world under the supervision of a prison warden. In this context, the data hiding represents a useful alternative to the construction of a hypermedia document, which may be less convenient to manipulate [3].

Steganography is the art and science of hiding information by embedding messages within other, seemingly harmless message. Steganography means "covering writing" in Greek. As the goal of steganography is to hide the presence of a message and to create a covert channel, it can be seen as the complement of cryptography whose goal is to hide the content of the message [4].

Next, section II, will emphasizes on related research work, while, section III, will present the techniques used in information security, like cryptography and steganography. Section IV will emphasize steganography methods, while section V and VI will present and evaluate two experiments.

II. Related Work

In recent works [5][6][7], it has been shown that digital data can be effectively hidden in an image so as to satisfy the criteria that the degradation to the host image is imperceptible and it should be possible to recover the hidden under a variety of attack. The main idea is to view the data hiding problem as a communication with channel side information [8] [9].

Steganography can be used in a lot of useful applications. For example copyright control of materials, to enhance the robustness of an image search engines and smart identity cards where the details of individuals are embedded in their photographs [10]. Steganography may be classified as pure, symmetric and asymmetric. While pure steganography does not need any exchange of information, symmetric and asymmetric need to exchange of keys prior sending the messages. Steganography is highly dependent on the type of media being used to hide the information. Medium being commonly used include text, images, audio files, and network protocols used in network transmissions [11].

Chandramouli and Memon [12] developed the most common method used to hide the message which involved the usage of Least Significant Bit (LSB). They apply the filtering masking and transformation on the cover media.

Abdullatif and Shukur [13] proposed a blind color image steganography method that embeds secret message by spraying theme on the blocks in the high order bits in color channel such as blue. However it also depends on the constant sequence spread spectrum method to survive loss compression image like JPG.

Atawneh and et al [14] presented common approaches and tools that are used in digital image steganography. It is shown mathematically and graphically. The differences between steganography, cryptography and watermarking technique are discussed. The authors also highlighted the current steganography tools and demonstrate how the secret information is embedded into image through the tools.

Ameen and et al [15] presented two methods for destroying steganography content in an image that are the overwriting and the de-noising method. The overwriting method is a random data that can be written again over steganographic images

while the de-noising method uses two kinds of destruction techniques that are filtering and discrete wavelet techniques. These two methods have been simulated and evaluated over two types of hiding techniques that are Least Significant Bit LSB technique and Discrete Cosine Transform DCT technique.

Hamid and et al. [16] presented the use of an image file as a carrier and the taxonomy of current steganographic techniques. The authors analyzed and discussed steganography techniques for their ability in information hiding and the robustness to different image processing attacks. They also briefly discussed steganoanalysis which is the science of attacking steganography.

The proposed work emphasizes on information protection against unauthorized persons while passing through networks. It presents cryptography and steganography algorithms. Then it presents the process of hiding of a message (digital audio data file) in an image file (cover images) using random insertion techniques through applying two experiments: insertion using byte level and insertion using pixel level. Finally, it gives evaluation for both applied algorithms.

III. Information Security Techniques

Our problem is distinguishing between important and the most important information, and, thus protecting information against violation. Data and information are used by all, individuals, companies, organizations, and countries.

Information security techniques are the procedures, tools, and products used to protect or at least decrease danger and violation of information, networks, information systems and their databases. There are many security tools already have been used in information environment like identity-passwords and fire walls, cryptography, intrusion detections, and anti-virus systems.

A. Cryptography

Cryptography is the transformation of data and information to unclear and non- understood code (looks has no means) to prevent unauthorized access of information. While, decryption is getting (extracting) the original information from the encrypted one. In the time being, cryptography gets more attention in information security field. This is because cryptography is the most important security techniques to provide secretly, integrity, and availability of information. In general, cryptography, and its application, specially, electronic signature, is the only way for grantee the responsibility over electronic nets.

Nowadays, internet is largest multimedia for information transmission. Keys are used in data encryption and decryption, and are based on complicated mathematical formulas (algorithms).

1) Symmetric cryptography (secret key): Where, both sender and receiver use the same secret key in message encryption and decryption they agree on using a pass phrase. The pass phrase can use capital, small, and other characters. The cryptography software transfers the pass phrase to binary number and adding other symbols to increase its length. The resulted binary number constitutes the cryptography key of the message. After receiving the encrypted message, the receiver uses the same phrase to retrieve the original message. The problem of this type is the unsafe distribution of the secret key.

2) Un-Symmetric cryptography (general key): comes due to the unsafe distribution of key. It uses two related key instead of one key; public key and private key. The private key is known only by the sender and used to encryption and decryption of the message. While, the general key is known by multiple user and used in decryption to retrieve the original message that was encrypted using the private key. The owner of the private key can retrieve the message using the general key. Although, this message is better than the symmetric one, it is not away from violation.

B. Steganography

Steganography can be done through embedding or inserting (hiding) messages in text, voice, or image file. To perform that correctly, data integrity should be the same after applying steganography. Data can't be protected by making it single block, it should be fragmented into several blocks during execution. These blocks should be secured against modification through attack. Also, we should predict that those blocks could be distorted so symbol corrections should be used.

1) Steganography in text: Where there are three theories. The first theory is, Line-Shift Coding, which code text lines vertically. The problem of this theory is that most of text symbol coding is visible to the reader and pixels between texts could be measured manually or automatically. The second theory is, Word-Shift Coding, which depends on coding document through moving horizontal positions of words of a text. This method is less visibly to the reader. The third theory is, Feature Coding, which depends on random distribution of text which makes violating is too difficult.

2) Steganography in an audio: Where too factors should be considered: Digital representation of audio and signal transmission. Digital representation has two major features : simple quantization method, where quality degree of digital audio is represented in 16 bits by Windows Audio-Visual(WAV) and Audio Interchange File Format(AIFF), and temporal sampling rate, which is in some ranges. Two theories are used: (1) Low-bit Encoding and (2) Phase Coding.

There is four media for signal transmission: (1) Digital End-to-end Environment: if audio file is copied directly with no change from machine to another, it will be sent through this environment. (2)Increased/Decreased Re-sampling Environment: signal is re-encoded to the lowest or the highest coding rate. This environment is suitable for steganography. (3)Analog Transmission and Re-sampling: this environment is used when signal is transformed to analog system. (4) Over Air Sampling: this environment when signal is over air from microphone and is loaded on media for transmission.

3) Steganography in an image: Which inserts the data file (hided message) inside the image file (cover image)? The message could be in normal text or encrypted text or even another image. Fig. 1, presents the general scheme of

steganography in an image at the sender and the receiver. The problem now is how to insert the message in the cover image. Next section will emphasizes these methods.

IV. Steganography Methods

There are some methods used to hide information in digital images could be applied on different image files with different level of success.

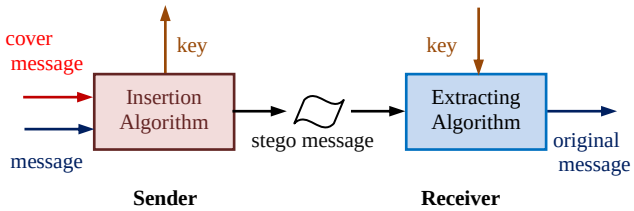

Fig. 1. General scheme of Steganography in images

A. Least Significant bit insertions

Embedding information in an image file by insertion in least significant bits is a famous and simple method, but it is sensitive for any manipulation of images, even it is was simple.

To insert information in least significant bit of an image bytes encoded in 24 bits, 3 bits are stored in each image pixel. For example, in an image of 768×1024 pixels, we can insert 2,359,295 digits of information (294,912 bytes or characters). If the inserted information was compressed before insertion, more characters could be inserted. The stego image that includes the inserted data will look as it will before insertion. We can insert information in the least significant bit and the following bit, without any change detected in the image viewing.

As example, to insert the character A (10000011) in the following 3 pixels (assuming no compression) (9 bytes) :

(00100111 11101001 11001000)
(00100111 11001000 11101001)
(11001000 00100111 11101001)

After insertion, the resulted 3 pixels will be as follow :

(00100111 11101000 11001000)
(00100110 11001000 11101000)
(11001001 00100111 11101001)

B. Algorithms and Transformation

Steganography using least significant bits is an easy and fast method, but it is too sensitive for any little change done in the image due to manipulation, processing, or lossless compression.

JPEG-JSTEG is a steganography method that integrates compressing algorithm with information steganography. It generates stego image by JPEG methodology from inputs of lossless cover image and message to be hided. JPEG software is modified to accept one digit steganography and the output file is TFIF standard. The TFIF standard consists of lossless and lossless parts. Software contains the message and the cover image by JPEG algorithm to generate steganography image in lossless JPEG.

JPEG image uses discrete cosine transform (DCT) to perform compression. This transform is lossless compression, because computing of cosine value exactly is not possible, and replicated computation that uses low precession numbers results in circulation errors in the final results. Difference ranges between the original given values and the extracted values depend on the applied method in transformation computation (DCT).

Beside DCT, fast Fourier Transform could be used to manipulate images and wavelet transform. This method maintains image in quality degree higher than the tools that depends only on a least significant bits. At using extra coding for shapes, we should compensate between message size and insertion. If the message has a small size, it could be inserted many times, but large size message could be inserted once, because of the huge part of the image occupied by the message.

V. Experiment 1: Random Insertion Using Bytes

This method transforms both image and message into two arrays of bytes, then it generates random positions in bytes range of image, where number of positions is equal to number of message bytes or characters (when size of message array is less than or equal to image array). Then, message bytes are inserted in the positions randomly generated in image bytes. Finally, those positions are kept in a key array.

A. Insertion Algorithm

Fig. 2, presents the algorithm used in random insertion using bytes. Assume that the bytes array shown in Fig. 3-a, resulted from transforming certain message into bytes. Also, I want to insert that message bytes array in the image bytes array shown in Fig. 3-b. So, I used a random function that generates random numbers (like those listed in Fig. 3-c).

This list of numbers represented in the key array, will be the numbers of positions in the image bytes array, where the message bytes will be inserted. Also, their number is equal to the message bytes array size. Those positions will be stored in the key array. The result of inserting the message bytes array (3-a) in the random positions (3-c) of image bytes array (3-b) is shown in Fig. 3-d (bold bytes represents the inserted bytes).

B. Results and Evaluation

Steganography in images using bytes algorithm emphasized the following results:

- Random generating lets discovering insertion positions more difficult than other steganography methods.
- Distortion looks as hashed points in the image, and it can't depend on image properties like brightness or sharpness or others.
- Distortion is too much because message insertion is done randomly not selectively for less important positions as in least significant method.
- Distortion could be reduced by reducing inserted digits in image bytes, as example, by inserting half a byte from message bytes in one image byte (or more or less).

- When the message is of huge size and the image is somehow of a small size, this method is not applicable.

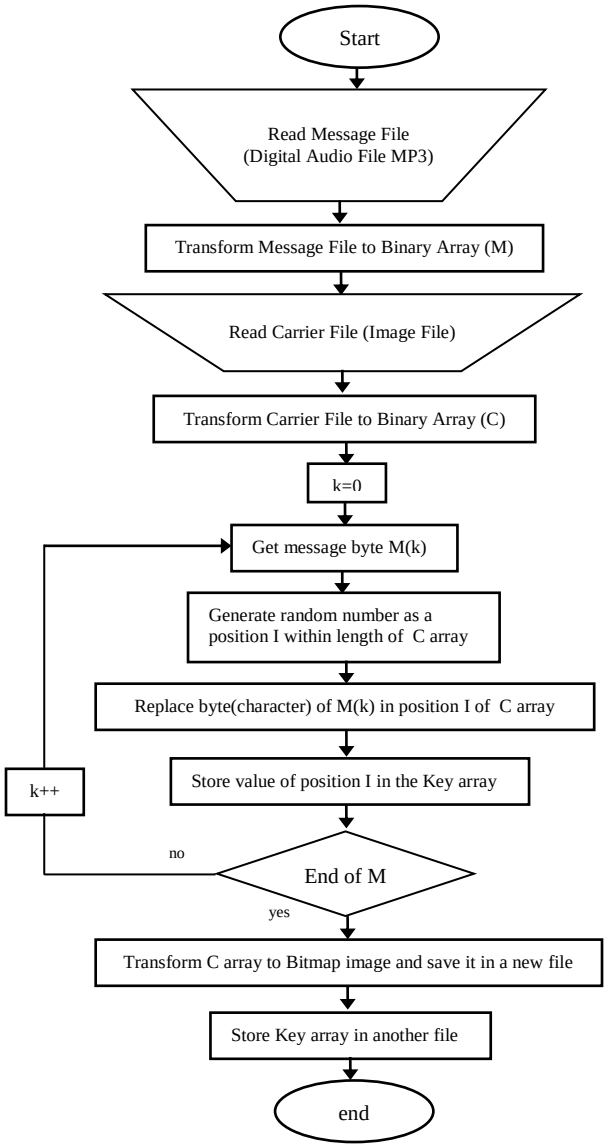

Fig. 2. Algorithm of steganography in images using bytes

a. message bytes array

1	00010101
2	01011010
3	11101001
4	00100101
5	00101010
6	01010010

b. image bytes array

1	00001111	11	11111111	21	00000111
2	00000111	12	00001111	22	11000000
3	11000011	13	11100000	23	00000111
4	11111100	14	11111111	24	11011000
5	00111111	15	11110000	25	00000000
6	00011111	16	00000111	26	00011111
7	11111101	17	11110011	27	11110001
8	01110111	18	11100010	28	11111000
9	11111110	19	11111100	29	00111111
10	00111111	20	01111111	30	00000000

c. position randomly generated

(8, 3, 19, 20, 12, 28)

d. resulted stego image bytes array

1	00001111	11	11111111	21	00000111
2	00000111	12	**00101010**	22	11000000
3	**01011010**	13	11100000	23	00000111
4	11111100	14	11111111	24	11011000
5	00111111	15	11110000	25	00000000
6	00011111	16	00000111	26	00011111
7	11111101	17	11110011	27	11110001
8	**00010101**	18	11100010	28	**01010010**
9	11111110	19	**11101001**	29	00111111
10	00111111	20	**00100101**	30	00000000

Fig. 3. steps and results of steganography using bytes(continued)

VI. Experiment 2: Random Insertion Using Pixels

In the method of random insertion using pixels, image is transformed to pixels array, each pixel is represented in 3 bytes, and message is transformed to a byte array. Then, random positions are generated in image pixels rang, with number of positions equal to message bytes (when the size of message bytes array is less or equal to the size of image pixels array). Each message byte is inserted in a position of image pixels, where positions are generated randomly and kept in the key array.

A. Insertion of a Byte in a Pixel

Assume that we have the byte (00101010) shown in Fig. 4-a, and we want to insert it in a certain pixel, shown in Fig. 4-b. The resulted pixels after insertion are shown in Fig. 4-c.

a. the message byte to be inserted.

(00101010)

b. the original pixels of an image.

(01010100	10101110	10111111)
R	G	B
84	174	191

c. pixels of image after insertion

(01010**001**	10101**010**	10111**10**)
R	G	B
81	170	190

Fig. 4. Inserting a byte in certain pixel

We can notice that last 3 bits in the first two bytes and the last 2 bits in the third byte contain message byte value that will result in simple change rate in each color of pixels colors. These bits are the least significant bits. By this method, the random insertion using pixels will be done. Distortion could be decreased if we insert each message byte in three pixels. We can see Fig. 5, which presents insertion of a message byte in 3 pixels.

Notice the change occurs only in the first digit in the first 8 bytes. It is clear that the distortion rate at inserting 1 byte in 3 pixels will be less than the distortion happen at inserting 1 byte in 1 pixel. But, the size of a message to be inserted in certain image should be of smaller size. We can use a moderate solution by inserting 2 message bytes in 3 pixels.

a. the message byte to be inserted.

(00101010)

b. the original pixels of an image.

01000101	10001111	10110010
01010111	11101110	10111001
01010000	10111110	00001111

c. pixels of the image after insertion

0100010**0**	1000111**0**	1011001**1**
0101011**0**	1110111**1**	1011100**0**
0101000**1**	1011111**0**	00001111

Fig. 5. Inserting a byte in certain pixel

B. Random Insertion Algorithm

Assume that the bytes array shown in Fig. 6-a, resulted from transforming certain message into the byte. And, we want to insert that message bytes array in the image pixels shown in Fig. 6-b.

To insert the message bytes array listed in Fig. 6-a in the image pixels in Fig. 6-b, we use a random function that generates random numbers (like those listed in Fig. 6-c). This list of numbers represents the key array, that will be the positions numbers in the image bytes array, where the message bytes will be inserted, and their number is equal to the message bytes array size. Those positions will be stored in the key array. The result of inserting the message bytes array (6-a) in the random positions (6-c) of image pixels (6-b) is shown in Fig. 6-d (bold digits represents the inserted bytes). Fig. 7, presents the algorithm used in steganography in image using pixels.

a. message bytes array

1	00010101
2	01011010
3	11101001

b. image bytes array

1	00001111	11111111	00000111
2	00000111	00001111	11000000
3	11000011	11100000	00000111
4	11111100	11111111	11011000
5	00111111	11110000	00000000
6	00011111	00000111	00011111
7	11111101	11110011	11110001
8	01110111	11100010	11111000
9	11111110	11111100	00111111
10	00111111	01111111	00000000

c. position randomly generated

(3,6,10)

d. resulted stego image bytes array

1	00001111	11111111	00000111
2	00000111	00001111	11000000
3	11000**000**	11100**101**	00000**01**
4	11111100	11111111	11011000
5	00111111	11110000	00000000
6	00011**010**	00000**110**	00011**10**
7	11111101	11110011	11110001
8	01110111	11100010	11111000
9	11111110	11111100	00111111
10	00111**111**	01111**010**	00000**01**

Fig. 6. Steps and results of steganography using pixels (continued)

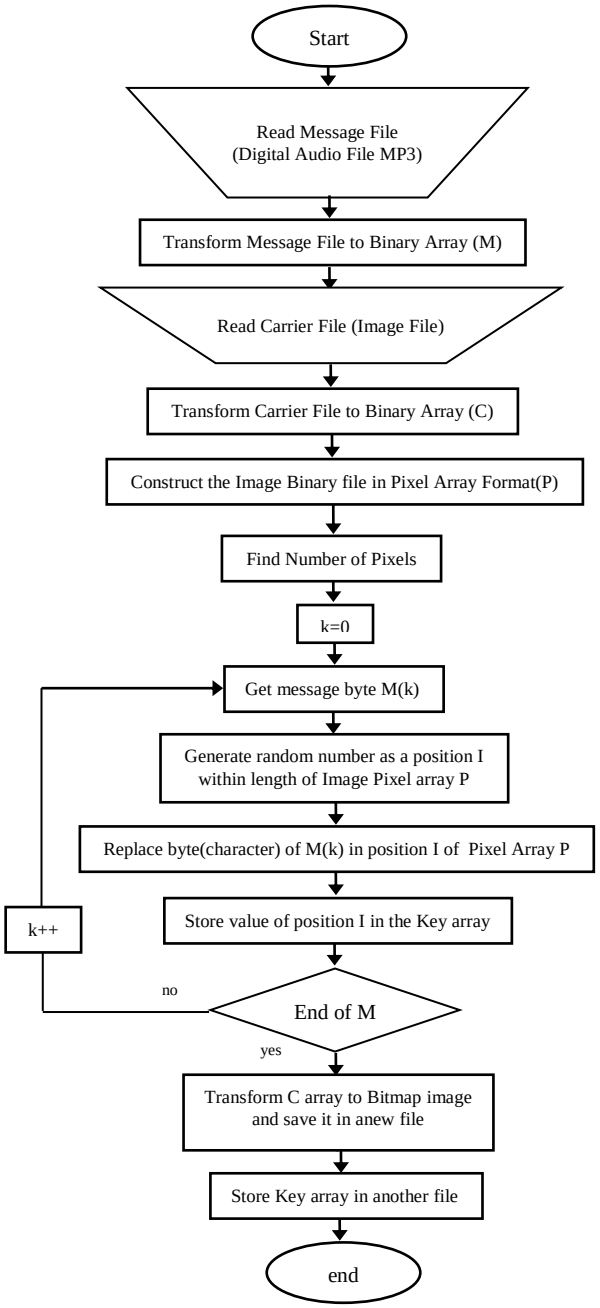

Fig. 7. Algorithm of steganography in images using pixels

C. Results and Evaluation

Steganography in images using pixels algorithm emphasized the following results:

- Distortion will be very small or even null, because of using least significant bits.
- Random generation of hiding position make discovering them is very difficult in contrast with other methods of steganography, where message could be discovered once the message used is known.

VII. Conclusions

This paper presented two algorithms in steganography in images through random insertion (hiding) of data using bytes and pixels. Newly, generating function of random values were built specially for steganography. It was impossible to find out message data, in contrast with other methods. Random generating lets discovering insertion positions more difficult than other steganography methods.

In the method of random insertion using bytes, both image and message were converted into two arrays of bytes, then it generated random positions in bytes range of image, where number of positions is equal to number of message bytes or characters. Then, message bytes were inserted in the positions randomly generated in image bytes. Finally, those positions were kept in a key array. Distortion was too much because message insertion is done randomly not selectively for less important positions as in least significant method. Distortion could be reduced by reducing inserted digits in image bytes, as example, by inserting half a byte from message bytes in one image byte (or more or less).

While, in the method of random insertion using pixels, image was transformed to pixels array, each pixel was represented in 3 bytes, and message was transformed to byte array. Then, random positions were generated in image pixels rang with number of positions equal to message bytes. Each message byte was inserted in a position of image pixels, where positions were generated randomly and kept in the key array. steganography in images using pixels algorithm emphasized that distortion would be very small or even null, because of using least significant bits.

Those experiments were applied to improve data transmission security. So, future work will evaluate some different insertion techniques and evaluate using different cover messages, in order to minimize distortion in the hided messages.

Refrences

[1] W. Bender, D. Gruhl and N. Morimoto, "Techniques for Data Hiding", IBM System Journal, Vol. 35, 1996.

[2] M. D. Swanson, M. Kobayashi and A. H. Tewfik, "Multimedia Data-Embedding and Watermarking Strategies", Proc. IEEE, Vol. 86, No. 12, pp. 1064-1087, June 1986.

[3] P. Moulin and M. Kivanc Mihcak, "A Framework for Evaluating the Data-Hiding Capacity of Image Sources", IEEE Int. Conf. On Image Processing, Vancouver, Canada, Oct. 2000.

[4] C. Cachin, "An information-theoretic model for steganography" in Information Hiding, 2nd Int. Workshop (D.Aucsmith, ed.) vol. 1525 of Lecture Notes in Computer Sciences, pp. 306-318, 1988.

[5] B. Chen and G. W. Wornnell, "Quantization index modulation: A class of provably good methods for digital watermarking and information embedding", IEEE Trans. On Info. Theory, vol. 47, no. 4, pp. 1423-1443, May 2001.

[6] K. Solanki, N. Jacobsen, S. Chandrasekaran, U. Madhow and B. S. Manjunath, "High-volume data hiding in images: Introducing perceptual criteria into quantization based embedding", ICASSP, May 2002.

[7] N. Jacobsen, K. Solanki, S. Chandrasekaran, U. Madhow and B. S. Manjunath,"Image adaptive high volume data hiding based on scalar quantization", IEEE Military Comm. Conf. (MILCOM), Oct.2002.

[8] M. H. Costa, "Writing on the dirty paper", IEEE Trans. On Info. Theory, vol. 29, no. 3, pp. 439-441, May 1983.

[9] P. Moulin and J. A. O'Sullivan, "Information-theoretic analysis of Information hiding", IEEE Trans. On Info. Theory, vol. 49, no. 3, pp. 563-593, May 2003.

[10] Y. Yunus, S. Ab Rahman, J. Ibrahim, " Steganography: A Review of Information Security Research and Development in Muslim World", American Journal of Engineering Research (AJER), Volume-02, Issue-11, pp-122-128, 2013.

[11] S. Mahajan, A. Singh, "A Review of Methods and Approach for Secure Steganography", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 10, pp-67-70, 2012.

[12] R. Chandramouli, N. Memon, "Analysis of LSB Based Image Steganography Techniques", in proceeding of IEEE ICIP, 2001.

[13] F. Abdullatif , A. W. Shukur, "Blind Color Image Steganography in Spatial Domain", Ibn Al- Haitham J. For Pure & Appl. Sci. Vol.24 (1), 2011.

[14] S. Atawneh, A. Almomani1, and P. Sumari, "Steganography in Digital Images: Common Approaches and Tools," IEEE Technical Review, Vol 30, Issue 4, 2013.

[15] S. Y. Ameen and M. R. Al-Badrany, "Optimal Image Steganography Content Destruction Techniques", Proceedings of the 2013 International Conference on Systems, Control, Signal Processing and Informatics, 2013.

[16] N. Hamid, A. Yahya, A. Badlishah, D. Najim and L. Kanaan, "Steganography in image files: A survey", Australian Journal of Basic and Applied Sciences, 7(1): 35-55, 2013.

AUTHOR PROFILE

The Author is Dr. Eng. Khaled N. ElSayed. He was born in Cairo, Egypt 9 Oct. 1963. He has got his PhD of computers and systems fromFaculty of Engineering, Ain Shams University, Cairo, Egypt, 1996. He has worked as an associate professor of computer science, in Umm-AlQura Uni. in Makkah, Saudi Arabia since 2006. Artificial Intelligence is his major. His interest research is Distant Education, E-Learning, and Agent.

Dr. Khaled N. ElSayed translated the 4th edition of "Fundamentals of Database Systems", Ramez Elmasei and Shamkant B. Navathe, Addison Wesley, fourth edition, 2004, published by King Saud University, Riyadh, Saudi Arabia, 2009. He is also the author several books in programming in C & C++, Data structures in C& C++, Computer and Society, Database Design and Artificial Intelligence.