Data Hiding by LSB Substitution Using Genetic Optimal Key-Permutation

Marghny Mohamed[1], Fadwa Al-Afari[2] and Mohamed Bamatraf[3]
[1]Faculty of Information and Computers, Assiut University, Egypt
[2, 3] Faculty of Engineering, Hadhramout University of Science and Technology, Yemen

Abstract: *The least significant bit (LSB) embedding method is one of the most commonly used techniques; it targets the LSB's of the host image to hide the data. This paper deals with three main steganography challenges (i.e. capacity, imperceptibility, and security). This is achieved by hybrid data hiding scheme incorporates LSB technique with a key-permutation method. The paper also proposes an optimal key permutation method using genetic algorithms for best key selection. Both normal and optimized methods are tested with standard images, varying both data size as well as key space. Final experimental results show decrement in computation time when increasing number of keys, at the same time system security improves.*

Keywords: *Steganography; LSB substitution; Cryptography; Genetic Algorithm.*

Received May 18, 2009; Accepted March 10, 2010

1. Introduction

The popularity of the Internet offers a great convenience to the transmission of a large amount of data over networks. Some of them may be secret information which is candidate to unauthorized access. In order to keep the unauthorized user away, variety of techniques have been proposed, data encryption and data hiding are two main methods in data security.

Data encryption uses a certain algorithm to transform data into cipher text only the user that has keys can decrypt the secret data from the cipher texts. For any unauthorized user who does not have a key, the ciphertext will look like nothing but streams of meaningless code. Although data encryption is a good way to secure data, it still has some weaknesses. The appearance of cipher texts would give unauthorized user an impulse to recover them. Moreover, the unauthorized users might even simply destroy the ciphertext out of range when they have trouble recovering them so that the legal receivers cannot get the data in time. That is the reason why data hiding has been researched recently.

Data hiding techniques embed the important data into multimedia data such as images, videos or sounds. Digital images are considered good cover carriers because of their insensitivity to human visual system. Watermarking and steganography are two major branches of information hiding technology. Each has its own specific characteristics.

The first branch is used to embed a distinguishable symbol, e.g., a signature or a trademark, into host signals to authorize the ownership of the signals. Here the size of the symbol usually small, ranging from one bit to thousands of bits to represent the symbol. Watermarking focuses on maintaining high robustness against attacks. It must ensure that the embedded information can be successfully detected or extracted from the watermarked signals, even if confronted with attacks such as filtering, resampling, lossy compression, etc.

The second branch (i.e. Steganography) is the art of covered or hidden writing. The word Steganography comes from the Greek words steganos and graphia, which together means "hiding writing" [10]. The purpose of steganography is covert communication to hide a message from a third party. This differs from cryptography, which is intended to make a message unreadable by a third party but does not hide the existence of the secret communication. Although steganography is separate and distinct from cryptography, there are many analogies between the two, and some authors categorize steganography as a form of cryptography since hidden communication is a form of secret writing. The digital steganography process has three basic components [11]: 1) the data to be hidden (secret data), 2) the cover file (cover-carrier), in which the secret data are to be embedded, and 3) the resulting stego-file (stego-carrier).

In the literature, many techniques for data hiding have been proposed [2-5]. One of the common techniques is based on manipulation the least significant bit (LSB) plans. A LSB substitution method replaces some LSB of the cover-image with the secret data [1,6-8,13,14].

Wang et al. [6] proposed to embed secret messages in the moderately significant bit of the cover-image. A genetic algorithm is developed to find an optimal substitution matrix for the embedding of the secret messages. They also proposed the use of local pixel adjustment process (LPAP) to improve the image quality of the stego-image.

Recently, Wang et al. [8] proposed a novel method to embed data inside the host image. The method based on simple LSB substitution data hiding. They also developed the optimal *k* LSB substitution method to solve the problem when *k* is large.

Chang et al [7] proposed a method of finding the optimal LSB in image hiding by dynamic programming strategy. The proposed method finds the optimal LSB substitution that Wang [8] found of approximate OLSB, the method reduces the computation time too.

X. Li et al [13] proposed a novel steganographic method based on JPEG and Practical Swarm Optimization algorithm. His method inspired from the optimal LSB substitution approach used by Wang [8], such a substitution strategy could be used in spatial domain, and thus applied to transform domain.

In this paper, a method have been proposed based on LSB substitution. To prevent illicit access of the data and obtain better embedding results, a key-permutation method with an optimal LSB substitution method is presented.

A random key is generated and then distributed to the communication parties.

Before embedding the data into the LSB of the cover image, it is represented with the help of the key (encrypted) at the sending end; an opposite operation is then performed at the receiving end to reveal the secret data. Optimization of the key is another phase of the proposed model. It is achieved by selecting best embedding results for a set of all possible keys using genetic algorithms are proposed method.

The rest of this paper is organized as follows. The concepts of image hiding by LSB substitution are presented in section 2. In section 3, the proposed method is described, the proposed key permutation method is introduced, and then the optimal substitution of the LSB by using key permutation method is demonstrated. Experimental results with a brief discussion are given in section 4. Finally conclusions are presented in section 5.

2. Data Hiding By Simple Lsb Substitution

In this section, the general operations of data hiding by simple LSB substitution method are described. Let C be the original 8-bit grayscale cover-image of $M_c \times N_c$ pixels, represented as

$$C = \{ x_{ij} \,|\, 0 \le i \le M_c,\ 0 \le j \le N_c,\ x_{ij} \in \{0,1,2,\ldots, 255\} \}, \qquad (1)$$

where M is the n-bit secret message which can be represented by

$$M = \{ m_i \,|\, 0 \le i < n,\ m_i \in \{0,1\} \}. \qquad (2)$$

Suppose that the n-bit secret message M is to be embedded into the k-rightmost LSBs of the cover-image C. Firstly, the secret message M is rearranged to form a k-bit virtual image M', which can be represented as

$$M' = \{ m'_i \,|\, 0 \le i < n',\ m'_i \in \{0,1,\ldots, 2^k-1\} \}, \qquad (3)$$

where $n' = M_c \times N_c$. The mapping between the n-bit secret message M= $\{ m_i \}$ and the embedded message $M' = \{ m'_i \}$ can be defined as follows:

$$m'_i = \sum_{j=0}^{K-1} \mathrm{m}_i \times \mathrm{k} + \mathrm{j} \times 2^{k-1-j} \ . \qquad (4)$$

Secondly, a subset of n' pixels $\{x_1, x_2, \ldots, x_n\}$ is chosen from the cover-image C in a greed upon sequence. The embedding process is completed by replacing the k LSBs of x_i by m'_i . Mathematically, the pixel value x_i of the chosen pixel for storing the k-bit message m'_i is modified to form the stego-pixel x'_i as follows:

$$X'_i = x_i - x_i \bmod 2^k + m'_i \ . \qquad (5)$$

In the extraction process, given the stego-image S, the embedded messages can be extracted without referring to the original cover-image. Using the same sequence as in the embedding process, the set of pixels $\{x'_1, x'_2, \ldots, x'_n, \}$ storing the secret message bits are selected from the stego-image. The k LSBs of the selected pixels are extracted and lined up to reconstruct the secret message bits. Mathematically, the embedded message bits m'_i can be recovered by

$$m'_i = x'_i \bmod 2^k. \qquad (6)$$

As the method described above has the same problems with many steganograpic schemes. The quality of the stego-image produced by simple LSB substitution may be not acceptable. It means that the method degrades the image quality and probably attracts unauthorized attention. Once he/she notices the stego-image, he/she can simply extract and analyze the LSB to get the secret message. To solve these problems, a key permutation technique is integrated with an optimal LSB substitution method to improve the security of the model and quality of the stego-image.

3. The Proposed Method

In this section, a key permutation method is introduced; we will first look at how the key permutation method works. Then, we shall demonstrate the optimal substitution of the LSB by using key permutation method.

3.1. Key Permutation Method

As shown in section 2. The cover image C, and the secret message M are defined. C and M are rearranged to form block-bits (blk) getting C'' and M'' respectively. where

$$C'' = \{ c''_i \mid 0 \le i \le 2^{blk}\text{-}1 \mid c''_i \in \{0,1,2,\ldots, 2^{blk}\text{-}1\}\} \quad (7)$$

$$M'' = \{ m''_i \mid 0 \le i \le 2^{blk}\text{-}1 \mid m''_i \in \{0,1,2,\ldots, 2^{blk}\text{-}1\}\} \quad (8)$$

Mathematically, the ciphering process will be obtained by performing bitwise XOR operating $\oplus$ to each block of the C'' with M'' as follow:

$cipher_i = c''_i \oplus m''_i, \quad 1 \le i \le length$ of (M) in blk (M_{blk}),

then

cipher= $\{cipher_i \mid 1 \le i \le length$ of M'' in $blk \mid cipher_i \in \{0,1,2,\ldots, 2^{blk}\text{ -}1\}\ \}$,

where

$$cipher_i = c''_i \oplus m''_i \quad (9)$$

3.1.1. Key generation

All possible permutations of the blk-bit key is generated, as

$$(key_{blk} = \{e_1,e_2,e_3,\ldots,e_n\}), \quad (10)$$

Where $n= 2^{blk}$, and e_i is the ith element of the key, i is the index of the ith element in the key, where each key is of size *blk*.

Before the sender embeds the data into the k-LSBs of the cover image C'',the method utilizes a sequential search in order to locate and return the positions all of elements in the key sequence representing the binary of the ciphered secret data plain-text characters as follows.

$$Position_i = locate(cipher_i, key_{blk}),$$

where

$$Position = \{ position_i \mid 1 \le i \le length\ (M'') \mid position_i \in \{0,1,2,\ldots, 2^{blk}\text{ -}1\}\ \}. \quad (11)$$

Finally, the embedding process is completed by replacing the k-LSBs of C'' by the position getting the stego-image S.

3.1.2. Secret message recovery

The following steps must be followed in order at the receiving end:

- Position extraction: The position data is extracted from the k-LSBs of the stego-image S by

$$Position = extract(stego\text{-}image,k) \quad (12)$$

- Cipher data retrieval: here the ciphered data ill be obtained from the key$_{blk}$ by according to its position

$$cipher = \text{key}_{\text{blk}}\ (position). \quad (13)$$

- Deciphering: M'' will be obtained by xoring the ciphered data with the C'', as follows,

$$M'' = cipher \otimes C''. \quad (14)$$

- Original secret message reconstruction: now M will be reconstructed by rearranging the M'' from blk-bit to its original form.

$$M = \text{map}(M''). \quad (15)$$

We evaluate the image quality of the proposed method in term of the peak signal-to-noise ratio (PSNR), which is commonly employed in image processing researches. The PSNR is estimated in decibel (dB) are defined as:

$$PSNR = 10*log_{10}\left(\frac{255}{MSE}^{2}\right), \quad (16)$$

there, *MSE* denotes the mean square error, which is defined as:

$$MSE = \frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n}(x_{ij} - y_{ij}),^{2} \quad (17)$$

here: x_{ij} denotes the original pixel value, and y_{ij} denotes the processed pixel value, and m and n denoted the width and height of the image respectively.

3.2. The optimal LSB Substitution: Genetic Algorithm

The data hiding technique discussed in previous section works well if the secret data is embedded in the k ŁSBs of the host image, where k is less than or equal to 3. However, it works poorly if k is greater than 3 because the number of all possible keys permutations will grow exponentially as k increases. For example, assume that k=4, there are a total of $(2^4)!$ =16! possible of key permutations (about 20,000 billions of key permutations) that can be utilized to embed data. To obtain the optimal embedding result, the simplest method is to calculate the PSNR for each substitution, and select the one having the maximum PSNR as the optimal result. Hence, it is very impractical and time consuming for us to compute the PSNR for each permutation. A genetic algorithm is thus developed to solve this problem, where GA is a randomized search procedure that is commonly used to solve the optimization problems. A solution in the problem domain corresponds to an individual in a GA, which is represented by a chromosome containing many genes. An objective function called the fitness function is used to evaluate the quality of each chromosome. In general, GA is mainly comprised of the following three operators, namely, (1) reproduction, (2) crossover, and (3) mutation. Reproduction retains the current chromosome's genes, crossover assembles existing genes into new combinations, and mutation produces new genes. The procedure of GA is started by specifying an initial population in the first generation, and during each next generation, the individuals in the population undergo the activities of reproduction, crossover and mutation, to produce their offspring. Then a fitness function is applied to each offspring to determine its quality. The individuals with high quality will survive and form the population of the next generation. The process will repeat for many times until a predefined requirement is satisfied, or a constant number of iterations is exceeded. In this study, a chromosome G in GA consisting of 2^k genes can be described by a key permutation as

$$G = g_0 g_1 \ldots g_{2^K - 1}, \qquad (18)$$

where g_0 represents the first element of the key, g_1 represents the second element of the key, and so on. For example suppose k=3, then the chromosome length are 2^3 =8 and they can be represented as:
G = 5 2 4 1 7 0 6 3.
The genetic operators crossover and mutation, are defined as follows.

- Crossover: Given two chromosomes G1 and G2, new chromosomes given by the crossover operators are G1', G2' have all possible permutations of the parents, i.e., no repetition of any elements.

- Mutation: Given a chromosome $G = g_0 g_1 \ldots g_{2^K - 1}$, two genes of G are selected randomly and their values are replaced with each other. For example, assume that G=01234567 and the two randomly selected genes are 1 and 5, then the mutated chromosome is G=05234167.

Finally, the fitness function F is defined in this study as the mean square error MSE, as described above in equation 17.

4. Experimental Results

To evaluate the effectiveness of the proposed method, three experiments were conducted. In the first experiment, the method is applied on two standard 8-bits per pixel, gray scale cover images, *"baboon"* and *"lena"*, each with the size 512×512 pixels as shown in Figure 1.

For optimal key selection in the proposed method; our experiments used the following genetic parameters selection. The population size is 10, cross over ratio is 0.8, mutation ratio is 0.2 and the number of generations used to find the optimal key is 100. And can be further improved these parameters to get better results.

(a) baboon (b) Lena

Figure1. Cover image.

Figure 2. Test image used as a set of secret messages.

The secret messages have different size of a gray scale image "tiff" as shown in Fig. 2. Theses images of sizes 512×256 pixels for 4-LSB insertion, 384× 256 pixels for 3-LSB, 256×256 for 2-LSB pixels and 256×128 pixels for 1-LSB, respectively.
The results of embedding the secret messages into the cover images are listed in Table 1.

The PSNR is used in this work to evaluate the image quality as described in Equation 16.

Table 1. The results of embedding the secret images into the cover images.

Cover images	k	LSB	Optimal GA	My-LSB	My-Optimal GA
baboon	1	51.1415	51.1415	51.1380	51.1723
	2	44.0205	44.7440	44.0526	44.2475
	3	37.8642	38.7295	-	-
	4	31.3307	-	31.4595	32.5326
Lena	1	51.1299	51.1524	51.1471	51.1681
	2	44.0216	44.7638	44.0656	44.3714
	3	37.8626	38.7242	-	-
	4	31.2818	-	31.4258	32.2161

4.1. Discussion of experiment 1

From the results shown in Table 1, the experiment compares the embedding results obtained by the simple substitution method in column labeled *My-LSB* and the GA approach in column labeled *My-Optimal GA* respectively, when k=1,2 and 4-LSBs insertion. The results show the difference between the two methods, indicates that the quality of the stego-image is improved when the GA approach is utilized.

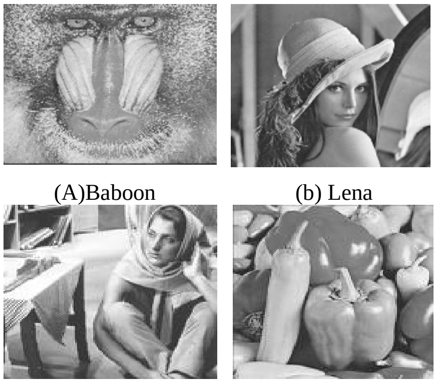
The column labeled *LSB* is the simple LSB substitution method and the column labeled *Optimal GA* is the optimal LSB substitution method proposed in [8].

From the results listed in Table 1, we notice that the values of improvement for the quality of stego-images in range between 0.021 and 1.0731, these values are significant according to [8].

4.2. Experiment 2

In the second experiment, we evaluate the effect of increasing the number of keys on the proposed method. The results of this experiment are shown in Table 2, the experiment used the following dataset:

- Cover images: Fig. 3.
  - Lena 131×131 pixels –grayscale (17160 bits)
  - Baboon 131×131 pixels –grayscale (17160 bits)
  - Barbara 131×131 pixels –grayscale (17160 bits)
  - pepper 131×131 pixels –grayscale (17160 bits)
- Secret messages: random data of the following:
  - File Secret1 of size 65× 33 bytes (17160 bits) for the 1-LSB insertion
  - File Secret2 of size 66× 65 bytes (17160 bits) for 2-LSB insertion.
  - File Secret4 of size 131× 65 bytes (17030bits) for 4-LSB insertion.



(A)Baboon (b) Lena

(C) Barbara (d) Pepper

Figure 3. Cover image.

Cover images	K	LSB	Cipher	Optimal GA Key=1	Optimal GA Key=5	Optimal GA Key=10
Lena	1	51.1647	51.1323	51.1505	51.2046	51.2511
	2	44.3986	44.0916	44.2077	44.2724	44.3082
	4	32.4312	31.5180	32.2298	32.4150	32.5619
Baboon	1	51.1969	51.1810	51.1810	51.2025	51.2386
	2	44.3482	44.1166	44.2016	44.2805	44.3255
	4	32.5423	31.5118	32.3736	32.5330	32.7748
Barbara	1	51.1257	51.1166	51.1662	51.1928	51.2247
	2	44.4064	44.1489	44.2188	44.2623	44.3216
	4	32.5996	31.8362	32.0646	32.2925	32.4537
Pepper	1	51.1606	51.0780	51.2056	51.2262	51.2583
	2	44.3600	44.0885	44.3016	44.3359	44.3812
	4	32.4441	31.6173	32.2150	32.3490	32.4915

Table 2. The results of embedding the secret messages into the cover images with optimal embedding.

4.3. Discussion of experiment 2

From the results listed in Table 2, we can notice that, PSNR is increasing with increasing of the number of key permutations, for key=1, 5 and 10 respectively, and different LSBs insertion (k=1,2 and 4). That means the quality of stego-image improved by applied the optimal key permutation. Fig. 4 clearly shows the stego-image quality *(PSNR)* increases with number of key increase.

From Table 2 the column labeled *cipher* is the PSNR of the encrypted secret data before the key-permutation method was applied to it.

4.4. Experiment 3

The goal of this experiment is to show the effect of increasing the number of key permutations of the proposed method on the computation time. To conduct this experiment we used the 8-bits per pixel gray scale cover image *"lena"*, with the size 131×131 pixels. The secret messages are random data files of different size:

- File *Secret1* of size 65× 33 bytes (17160 bits) for the 1-LSB insertion
- File *Secret2* of size 66× 65 bytes (17160 bits) for 2-LSB insertion.
- File *Secret4* of size 131× 65 bytes (17030bits) for 4-LSB insertion.

Table 3 shows the average value of the computation time, over ten runs, for the 1-LSB, 2-LSB, 4-LSB insertion. And Fig. 5 shows these results.

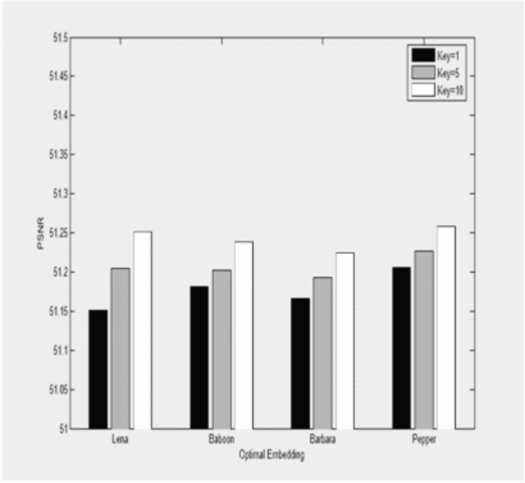


Figure 4. The results of optimal embedding secret1.txt to the 1-LBS of the cover images with different key numbers.

4.5. Discussion of experiment 3

The computation time decreases by the number of key increase until reaches to some point, where the computation time will have small changes with the change in the number of keys. This is due to reduce the search space of the genetic algorithm process will get fast results compared to huge search spaces.

It means:

> By increasing the number of key permutation, the Computation time decreases and the System security increases.

Table 3. The average time over 10-runs for different keys number 1,2,…10.

Keys no.		1-LSB	2-LSB	4-LSB
Key=1	Time (Second)	105.4991	120.1741	473.7810
	PSNR	51.1505	44.2077	32.1906
Key=2	Time (Second)	68.8122	66.0454	354.5630
	PSNR	51.1882	44.2134	32.2916
Key=3	Time (Second)	41.8217	47.1014	204.9530
	PSNR	51.1688	44.2353	32.3298
Key=4	Time (Second)	32.1593	41.0517	165.539
	PSNR	51.2005	44.2313	32.3741
Key=5	Time (Second)	27.6000	30.3281	127.3408
	PSNR	51.2046	44.2724	32.3783
Key=6	Time (Second)	27.5826	27.5265	125.4404
	PSNR	51.2036	44.3101	32.4838
Key=7	Time (Second)	24.7391	25.6921	121.6624
	PSNR	51.2046	44.3113	32.4923
Key=8	Time (Second)	22.6343	24.5029	105.953
	PSNR	51.2226	44.3080	32.5357
Key=9	Time (Second)	21.8937	23.0389	111.5062
	PSNR	51.2056	44.3213	32.5490
Key=10	Time (Second)	19.8483	22.0172	100.6626
	PSNR	51.2511	44.3211	32.5752

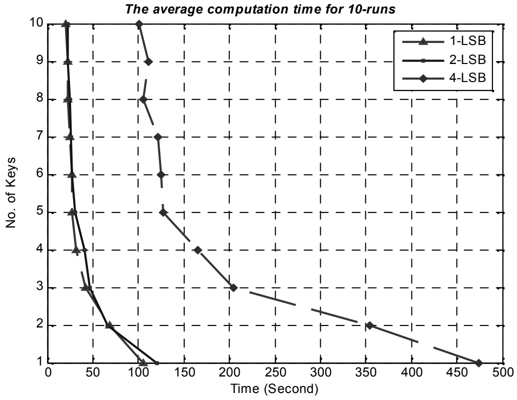


Figure 5. The average computation time over 10-runs.

5. Conclusion

Steganography, a branch of information hiding technology, aims to protect important data in transmission. Message capacity and stego-image quality are two important criteria in evaluating a steganogarphic method. The basic concept of the proposed method is by simple LSB substitution. To increase the system performance (capacity, security), the method of optimal LSB substitution is presented in corporation with key permutation. We integrate our scheme with a genetic algorithm to solve the problem of hiding important data in the rightmost k LSBs of the cover-image when k is large, which may involve a huge computation time to find the optimal result. Our experimental results have shown that the proposed method provides good image quality and large message capacity as well as increase in the system immunity.

References

[1] Bender, W., Morimoto, N. and Lu, A., "Techniques for data hiding", *IBM Syst. J.*, vol. 35, no. 3/4, pp. 313–336, 1996.

[2] Chan, C.K. and Cheng, L.M., "Hiding data in images by simple LSB substitution", *Pattern Recognition*, vol. 37, no. 3, pp. 469–474, 2004.

[3] Chang, C.C., Hsiaob, J.Y. and Chan, C.S., "Finding optimal least-significant-bit substitution in image hiding by dynamic programming strategy", *Pattern Recognition*, vol. 36, pp. 1583 – 1595, 2003.

[4] Chang, C.C. and Tseng, H.W., " Data hiding in images by hybrid LSB substitution ", *3rd International Conference on Multimedia and Ubiquitous Engineering*, art. no. 5318917, pp. 360-363, 2009.

[5] Chen, T.S., Chang, C.C. and Hwang, M.S., "A virtual image cryptosystem based upon vector quantization", *IEEE Trans. Image Process*. vol. 7, no. 10, pp. 1485–1488, 1998.

[6] Chung, K.L., Shen, C.H. and Chang, L.C., "A novel SVD- and VQ-based image hiding scheme", *Pattern Recognition Lett*., vol. 22, no. 9, pp. 1051–1058, 2001.

[7] Johnson, N.F. and Jajodia, S., "Exploring Steganography: Seeing the Unseen", *IEEE Computer Journal*, vol. 31, no.2, pp. 26-34, 1998.

[8] Katzenbeisser, S. and Petitcolas, F.A.P., *Information Hiding Techniques for Steganography and Digital Watermarking*, Artech house, Inc., 2000.

[9] Krutz, R.D., Consulting Editor, *Hiding in plain sight: Steganography and the Art of Covert communication*, Wiley Publishing, Inc., 2003.

[10] Li, X. and Wang, J., "A steganographic method based upon JPEG and particle swarm optimization algorithm", *Information Sciences*, vol. 177, no. 15, pp. 3099–3109, 2007.

[11] Marvel, L.M., Boncelet, C.G. and Retter, C.T. (1999), "Spread spectrum image steganography", *IEEE Trans. Image Process*., vol. 8, no. 8, pp. 1075–1083.

[12] Schneier, B., *Applied Cryptography Second Edition: Protocols, Algorithms, and Source Code in C*, Wiley Computer Publishing, John Wiley & Sons, Inc., 1996.

[13] Wang, R.Z., Lin, C.F. and Lin, J.C., "Hiding data in images by optimal moderately significant-bit replacement", *IEE Electron. Lett*., vol. 36, no. 25, pp. 2069–2070, 2000.

[14] Wang, R.Z., Lin, C.F. and Lin, J.C., "Image hiding by optimal LSB substitution and genetic algorithm", *Pattern Recognition*, vol. 34, no. 3, pp. 671–683, 2001.

Marghny Mohamed is a member in Dept. of Computer Science, Faculty of Computers and Information Science, Asyut University, Asyut, Egypt., date of birth: June 1965, received the PhD degree in computer science from the University of Kyushu, Japan, in 2001, and the MS from Asyut university in computer science, in 1993 and BS degrees in Mathematics from Asyut University, Egypt, in 1988. He is an assistant professor in the Department of Computer Science, University of Asyut and Vice-President for Community Services and Environmental Affairs of the Faculty of Computers and Information Science. His Fields of research are Data Mining, Text Mining, Information Retrieval, Web Mining, Machine Learning, Natural Language Processing, Pattern Recognition, Neural Networks, Evolutionary Computation, Fuzzy Systems. Dr. Marghny is a member of the Egyptian mathematical society and Egyptian syndicate of scientific professions., he is a member of some research projects in Asyut university, Egypt. Data Mining.

Fadwa Al-Afari received her Bachelor Science degree in computer science in 2003 at Hadramout University for \Science and Technology in Yemen and received her Master in 2009 at Assiut University –Egypt in steganography which is hiding data in any digital media, steganography is part of data security. She has worked teacher in Computer Engineering department in faculty of Engineering and Petroleum in Hadramout University and she is curreny a Ph.D. student in networks and data security at Assiut University –Egypt. Her interest subjects are networks, data security, data mining and image processing.

Mohammed Bamatraf is currently an assistant professor at Hadhramout University of Science and technology, Yemen, Faculty of Engineering, Department of Computer Engineering, teaching several computer science subjects. He received his B.Sc (computer Science) from Poona University, India, his M.sc (computer Science) from Osmania University, India, and his PhD from Assiut University, Egypt. His Doctoral thesis was about modified data mining techniques and its application in medical diagnosis and intrusion detection. He published several papers in local as well as international conferences. His research areas of interest includes: data and network security, medical informatics, data mining and machine learning, and bioinformatics. His research activities are currently focused on the application of Bioinformatics and Machine Learning data and network security.