Volume 3, No. 1, Jan-Feb 2012

ISSN No. 0976-5697

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

Genetic Algorithm: Tool to Encrypt Image

Dr. Mohammed Abbas Fadhil Al-Husainy
Department of multimedia systems, faculty of science and information technology,
Al-Zaytoonah University of Jordan., Amman, Jordan
dralhusainy@yahoo.com, alhusainy@alzaytoonah.edu.jo

Abstract: Security is an important issue when digital images are transmitted through the internet and cellular phones, as well as being important in encryption of the satellite images, and image encryption is the most useful technique employed for this purpose. Genetic Algorithm (GA) can be regarded as a randomized search procedure that is commonly used to solve the optimization problems. The genetic algorithm uses two reproduction operators - crossover and mutation. Crossover assembles existing genes into new combinations, and mutation produces new genes.In this paper, a new approach of employing the crossover and the mutation operations of the genetic algorithms (GA) to encrypt images was proposed. The results of several experimental, statistical analysis and key sensitivity tests show that the proposed image encryption scheme provides an efficient and secure way for real-time image encryption and transmission.

***Keywords*:**Crossover, Mutation, Information Security, Random, Distortion

I. INTRODUCTION

Because of greater demand in digital signal transmission in recent time, the problem of illegal data access from unauthorized persons becomesneed intelligent and quick solution. Accordingly, the data security has become a critical and imperative issue in multimedia data transmission applications. In order to protect valuable information from undesirable users or against illegal reproduction and modifications, various types of cryptographic schemes are needed. Cryptography offers efficient solutions to protect sensitive information in a large number of applications including personal data security, medical records, network security, internet security, diplomatic and military communications security, etc. through the processes of encryption/decryption.

Cryptography contains two basic processes: one process is when recognizable data, called plain data, is transformed into an unrecognizable form, called cipher data. To transform data in this way is called to encipher the data or encryption. The second process is when the cipher data is transformed back to the original plain data, this is called to decipher, or decrypting the data. To be able to determine if a user is allowed to access information a key is often used. Once a key has been used to encipher information, only someone who knows the correct key candecipher the encrypted data. The key is the foundation of most data encryptions algorithms today. A good encryption algorithm should still be secure even if the algorithm is known[1-5].

Encryption is the process of transforming the information to insure its security. With the huge growth of computer networks and the latest advances in digital technologies, a huge amount of digital data is being exchanged over various types of networks. It is often true that a large part of this information is either confidential or private. As a result, different security techniques have been used to provide the required protection [6].

With the advancements of multimedia and networks technologies, a vast number of digital imagesnow transmitted over Internet and through wireless networks for convenient accessing and sharing[5]. Multimedia security in general is provided by a method or a set of methods used to protect the multimedia content. These methods are heavily based on cryptography and they enable either communication security, or security against piracy (Digital Rights Management and watermarking), or both. Communication security of digital images and textual digital media can be accomplished by means of standard symmetric key cryptography. Such media can be treated as binary sequence and the whole data can be encrypted using a cryptosystem such as Advanced Encryption Standard (AES) or Data Encryption Standard (DES) [7]. In general, when the multimedia data is static (not a real-time streaming) it can treated as a regular binary data and the conventional encryption techniques can be used. Deciding upon what level of security is needed is harder than it looks. To identify an optimal security level, the cost of the multimedia information to be protected and the cost of the protection itself are to be compared carefully.

As a result, protection of digital images against illegal copying and distribution has become an important issue [5,8, 9, 10]. Image encryption techniques try to convert an image to another one that is hard to understand [11]. On the other hand, image decryption retrieves the original image from the encrypted one. There are various image encryption systems to encrypt and decrypt data, and there is no single encryption algorithm satisfies the different image types.

II. RELATED WORKS

At present, there are many available image encryption algorithms such as Arnold map, Tangram algorithm[12], Baker's transformation[13], Magic cube transformation[14], and Affine transformation[15] etc. In some algorithms, the secret-key and algorithm cannot be separated effectively. This does not satisfy the requirements of the modern cryptographic mechanism and are prone to various attacks. In recent years, the image encryption has been developed to overcome above disadvantages as discussed in [7, 16].Conventional encryption algorithms such as DES, AES, IDEA are not suitable for

practicalimage cipher due to some intrinsic features of images such as bulk data capacity, high redundancy,strong correlation among adjacent pixels, etc. It is desirable to develop an efficient image cryptosystem,especially for real-time secure image communication over open networks. To meet this challenge, avariety of image encryption schemes have been proposed. Among them, chaos-based algorithm hassuggested a new and efficient way to deal with the intractable problems of fast and highly secure imageencryption. The fundamental features of chaotic dynamical systems such as ergodicity, mixing property,sensitivity to initial conditions/system parameters, etc. can be considered analogous to some idealcryptographic properties such as confusion, diffusion, balance, avalanche properties, etc.[17-20].

There have been various data encryption techniques [21, 22, 23] on multimedia data proposed in the literature. Genetic Algorithms (GAs) [24] are among such techniques. The genetic algorithm is a search algorithm based on the mechanics of natural selection and natural genetics.

The genetic algorithm uses two reproductionoperators: crossover and mutation. Reproduction give genetic algorithms most of their searching power. To apply a crossoveroperator, parents are paired together. There areseveral different types of crossover operators, and thetypes available depend on what representation is usedfor the individuals. The one-point crossover means that the parentindividuals exchange a random prefix when creatingthe child individuals.The purpose of the mutation operatoris to simulate the effect of transcription errors thatcan happen with a very low probability when a chromosomeis mutated. A standard mutation operatorfor binary strings is bit inversion. Each bit in anindividual has a small chance of mutating into itscomplement, that is, a '0' would mutate into a '1'.

Only few genetic algorithms based encryption have been proposed. Kumar and Rajpal [25] described encryption using the concept of the crossover operator and pseudorandom sequence generator by NLFFSR (Nonlinear Feed Forward Shift Register). The crossover point is decided by the pseudorandom sequence and the fully encrypted data they are able to achieve. Kumar, Rajpal, and Tayal extended this work and used the concept of mutation after encryption. Encrypted data are further hidden inside the stego-image [26].

Husainy proposed Image Encryption using Genetic Algorithm-based Image Encryption using mutation and crossover concept [27].

A. Tragha et al., describe a new symmetrical block ciphering system named ICIGA (Improved Cryptography Inspired by Genetic Algorithms) which generates a session key in a random process. The block sizes and the key lengths are variable and can be fixed by the user at the beginning of ciphering. ICIGA is an enhancement of the system (GIC) "Genetic algorithms Inspired Cryptography" [28, 29].

RasulEnayatifar and Abdul Hanan Abdullah proposed a new method based on a hybrid model composed of a genetic algorithm and a chaotic function for image encryption. In their technique, first a number of encrypted images are constructed using the original image with the help of the chaotic function. In the next stage, these encrypted images are employed as the initial population for starting the operation of the genetic algorithm. Then, the genetic algorithm is used to optimize the encrypted images as much as possible. In the end, the best cipher-image is chosen as the final encryption image [30].

A new approach is suggested in this paper for secure and efficient image encryption. After dividing the image into set of chromosomes (vectors)andapplying crossover and mutation operations on randomly selected genes fromthese chromosomes to generate new encrypted chromosomes that will be used next to construct the encrypted image. Crossover operation also used to randomly reorder the chromosomes of the image to make more confusion in the encrypted image.

III. THE PROPOSED METHOD

The two reproduction operations (crossover and mutation) of the genetic algorithm (GA) are implementing, in different ways, on the source image to get a secure encrypted image.

At the first, a two dimensional bitmap image fileof size (Width×Height×Palette) is treating as a file of bytes (genes) has a ***FileSize***=(***Width*** × ***Height*** × ***Palette***), such that each byte value between (0...255).

To demonstrate the operations that are doing in the encryption phase of the proposed method, we consider have the following simple 2D bitmap image example.

2	10	7	15	32	19
9	64	71	3	15	23
1	12	34	18	5	25
30	11	3	16	27	8
9	30	22	15	7	3
15	6	26	8	30	2
5	11	4	12	27	33
7	18	1	21	2	25

A. Selecting VectorLength:

After selecting a vector (chromosome) length, by the sender of the secret image, such that the proposed encryption method allow to select a ***VectorLength*** between (**2^2**…**2^{32}-1**). The genes (bytes) of each vector are indexing (**0**…***VectorLength*-1**). The selected vector length will be one of three parts which form the *secret key* that is using to encrypt the secret image. (In the above example, weselect***VectorLength*=8**).

B. Calculating the StartCrossoverIndex and StartMutationIndex

Where the reproduction of new individuals,by implementing the crossover and the mutation operations ofGenetic Algorithm (GA), is performthrough selecting genes randomly. And the random numbers sequence, which is generated from any Random NumbersGeneration Algorithm (RNGA),is different and depending on selecting an initial value that is feeding to the algorithm.To generate the random numbers sequence that is needed to do the operations in the proposed encryption method, we can adopt any random numbers generation algorithm.

The proposed method extract two integer values from the bytes of the plainimage to be used next to set the initial value of the random numbers generation algorithm that will be used in

this method to perform the crossover and the mutation operation. These two values are extracted by applying the following two formulas on the bytes of the secure image.

$$StartCrossoverIndex = \left[\frac{[XOR_{i=0}^{FileSize-1}(Byte_i)]}{255} \times (VectorLength - 1)\right] \quad (1)$$

$$StartMutationIndex = \left[\frac{[XNOR_{i=0}^{FileSize-1}(Byte_i)]}{255} \times (VectorLength - 1)\right] \quad (2)$$

Where ***XOR*** and ***XNOR***are the two bitwise logical operations *eXclusive-OR* and *eXclusive-NOR*respectively, that are applying on the bits of all bytes in the image file.

The main point behind selecting these values from the bytes of the source image is to make a dependency between the source image and the random numbers generation algorithm that is using to encrypt the image. Obviously, these two values are different from one image to another.The value of each ***StartCrossoverIndex***and ***StartMutationIndex***is between (**2^2**…**2^{32}-1**). The***StartCrossoverIndex***and ***StartMutationIndex***will be form two parts of the *secret key* that will be used to encrypt the source image next.

The calculated ***StartCrossoverIndex*** and ***StartMutationIndex***for the above example are:

***StartCrossoverIndex*=2**
***StartMutationIndex*=6**

The *secret key* of the proposed encryption method becomes consists of three parts (***VecotrLength***, ***StartCrossoverIndex***, ***StartMutationIndex***), the value of each part of the *secret key* represent by **2^{32}** bits. In the above example the three parts of the *secret key* are (**8, 2, 6**).

C. Segment Source Image into Vectors:

Before starting the implementation of the crossover and the mutation operations on the image, the proposed method segments the image file into number of vectors of length ***VectorLength***. This segmentation operation will produce list of vectors called ***VectorsList***indexed between (**0**…***NoOfVectors*-1**), where:

***NoOfVectors* = (*FileSize*/*VectorLength*)**

	0	1	2	3	4	5	6	7
Vector 0	2	10	7	15	32	19	9	64
Vector 1	71	3	15	23	1	12	34	18
Vector 2	5	25	30	11	3	16	27	8
Vector 3	9	30	22	15	7	3	15	6
Vector 4	26	8	30	2	5	11	4	12
Vector 5	27	33	7	18	1	21	2	25

D. Determine the CrossoverIndex and MutationIndexValues:

Set the ***CrossoverIndex*** and ***MutationIndex***values for each vector. These are done by taking the values of ***StartCrossoverIndex*** and ***StartMutationIndex***and set them as the ***CrossoverIndex*** and ***MutationIndex*** values of the first vector and circularly increment these values from vector to the next vector. When we are doing this operation on the vectors of the above example,itproduces the following indices.

	0	1	2	3	4	5	6	7
Vector 0	2	10	7_c	15	32	19	9_m	64
Vector 1	71	3	15	23_c	1	12	34	18_m
Vector 2	5_m	25	30	11	3_c	16	27	8
Vector 3	9	30_m	22	15	7	3_c	15	6
Vector 4	26	8	30_m	2	5	11	4_c	12
Vector 5	27	33	7	18_m	1	21	2	25_c

We must note here that, the values of genes at ***CrossoverIndex***and ***MutationIndex*** of each vector will remain no change during the encryption operation of the source image because they will be use next in the decryption operation. For the above example, the genes values of Vector 4 at the ***CrossoverIndex*** and the ***MutationIndex*** are ***GeneValue*$_c$**=4 and ***GeneValue*$_m$**=30.

E. Crossover Operation:

Now, we are ready to perform the crossover and the mutation operations. The crossover operation is doing(on the genes of each vectoralone) by selectrandomlytwo genes indices in the vector; and exchanging the values of these genes. This operation is repeating number of times equal (***GeneValue*$_c$**+ ***VectorLength***). For the Vector 0, from the above example, the crossover operation is repeated (7 + 8 = 15 times). Before performing the crossover operation for each vector, the ***CrossoverIndex*** of the vector is set as a new initial value of the random numbers generation algorithm. In Vector 0 of the above example, the ***CrossoverIndex*** is 2.After doing the crossover operation on all vectors of the above example, the vectors become as follow:

	0	1	2	3	4	5	6	7
Vector 0	10	32	7_c	15	19	2	9_m	64
Vector 1	71	34	1	23_c	15	3	12	18_m
Vector 2	5_m	30	1	27	3_c	16	25	8
Vector 3	9	30_m	7	22	15	3_c	15	6
Vector 4	2	5	30_m	12	26	11	4_c	8
Vector 5	7	2	21	18_m	33	1	27	25_c

F. Mutation Operation:

The mutation operation is doing (on the genes of each vector alone) by select randomly geneindex in the vector; and changing the value of this gene by subtracting its value from 255. We must refer here that anyone can choose another operation to change the value of the genes in the mutation operation. This mutation operation is repeating number of times equal (***GeneValue$_m$*** + ***VectorLength***). For the Vector 0 from the above example, the mutation operation is repeated (9 + 8 = 17 times). Before performing the mutation operation for each vector, the ***MutationIndex*** of the vector is set as a new initial value of the random number generation algorithm. In Vector 0 of the above example, the ***MutationIndex*** is 6. After doing the mutation operation on all vectors of the above example, the vectors become as follow:

	0	1	2	3	4	5	6	7
Vector 0	245	223	7_c	15	236	2	9_m	64
Vector 1	184	221	254	23_c	15	3	243	18_m
Vector 2	5_m	30	244	27	3_c	239	230	8
Vector 3	246	30_m	248	233	240	3_c	240	249
Vector 4	253	250	30_m	12	26	11	4_c	8
Vector 5	7	2	21	18_m	33	1	27	25_c

G. Reorder the Vector Sequence:

After complete the crossover and the mutation operations, and to add more confusion in the pixels of the encrypted image,another different implementation of the crossover operation that will be done on the order of the vectors by select randomly two vectors and exchanging their order.This operation is repeating number of times equal (***NoOfVectors***). Before performing this operation, the value (***VecotrLength***+***StartCrossoverIndex***+***StartMutationIndex***) is set as a new initial value of the random numbers generation algorithm. After doing this, we get a new order of the vectors. The order of the vectors of the above example becomes:

	0	1	2	3	4	5	6	7
Vector 5	7	2	21	18_m	33	1	27	25_c
Vector 2	5_m	30	244	27	3_c	239	230	8
Vector 3	246	30_m	248	233	240	3_c	240	249
Vector 1	184	221	254	23_c	15	3	243	18_m
Vector 0	245	223	7_c	15	236	2	9_m	64
Vector 4	253	250	30_m	12	26	11	4_c	8

H. Construct the Encrypted Image:

At the last, restore the new genes of the vectors as 2D bitmap image to construct the encrypted secure image. And save the *secret key*(***VectorLength***, ***StartCrossoverIndex***, ***StartMutationIndex***)that is generated from the proposed encryption method to deliver it to the receiver of the encrypted image. The encrypted image of the above example is shown follow:

7	2	21	18	33	1
27	25	5	30	244	27
3	239	230	8	246	30
248	233	240	3	240	249
184	221	254	23	15	3
243	18	245	223	7	15
236	2	9	64	253	250
30	12	26	11	4	8

When the receiver want to decrypt the encrypted image, he/she feed the three parts of the *secret key* (***VectorLength***, ***StartCrossoverIndex***, ***StartMutationIndex***) to the decryption phase of the proposed method. In the decryption phase, the proposed method does the same steps as in itsencryption phase, but they are doing in reverse order.

IV. EXPERIMENTSAND SECURITY ANALYSIS

To evaluate the proposed encryption method, this method is tested on a number ofbitmap images of type (.bmp) which have different sizes.The required programming codes to implement the proposed method are written using C++ programming language.

Key space analysis, key sensitivity analysis, statistical analysisand Signal to Noise Ratio (SNR) are some of the security teststhat are recommended to be used to test the performance,strength and immunity of encryption methods.

A. Key Space Analysis:

For an effective cryptosystem, the key space should be large enough to make brute-force attack infeasible. The secret key space (***Vector Length***, ***Start Crossover Index***, ***Start Mutation Index***) of the proposed method is (32bits + 32bits + 32bits), this means that the cryptosystem has relatively long number of bits in the secret key. Where the key space of the most well-known secure encryption algorithm AES is 128-bits. Sothis is proof that the proposed cryptosystem is good at resisting brute-force attack.

B. Key Sensitivity:

To evaluate the key sensitivity feature of the proposed method, a one bit change is made in one of the three parts of the secret key and then used it to decrypt the encrypted image. The decrypted image with the wrong key is completely different when it is compared with the decrypted image by using the correct key as shown in Fig.1. It is the conclusion that the proposed encryption method is highly sensitive to the key, even an almost perfect guess of the key does not reveal any information about the plain image.

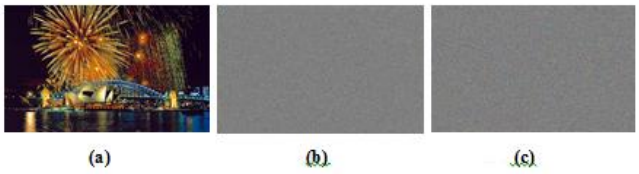

Figure 1: (a) Source image (b) Encrypted image (c) Decrypted image with wrong key

C. Statistical Analysis:

Statistical attack is a commonly used method in cryptanalysis and hence an effective cryptosystem should be robust against any statistical attack. Calculating the histogram and the correlation between the neighbors pixels in the source and in the encrypted image are the statistical analysis to prove the strong of the proposed cryptosystem against any statistical attack.

Fig.2 shows the histograms of the source image in Fig.1 and its encrypted image respectively. It's clear from Fig.2 that the histogram of the encrypted image is completely different from the histogram of the source image and does not provide any useful information to employ statistical attack.

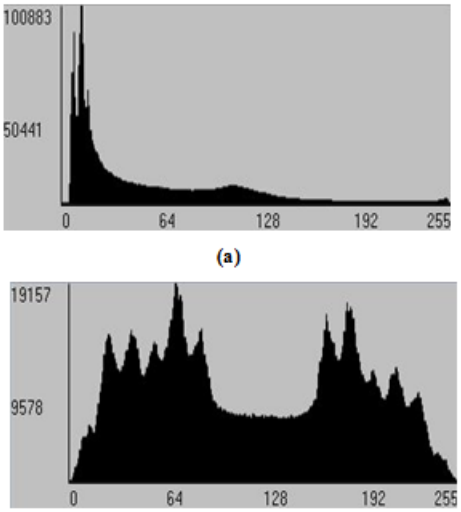

Figure 2:(a) Histogram of the source image in Fig. 1(a)

(b) Histogram of its encrypted image

The correlation coefficient *r*is calculating by using the following formula:

$$r = \frac{\sum_{i=1}^{N}(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{N}(x_i - \overline{x})^2 \times \sum_{i=1}^{N}(y_i - \overline{y})^2}} \quad (3)$$

Where***N*** is the number of pixel pairs,

$$\overline{x} = \frac{1}{N}\sum_{i=1}^{N} x_i$$

And

$$\overline{y} = \frac{1}{N}\sum_{i=1}^{N} y_i$$

The correlation coefficient for horizontal neighbor pixelsof the source image in Fig. 1 is *r*=0.602642 while *r*=0.00108589for its encrypted image. It is clear from these two different values of the correlation coefficient that the strong correlation between neighbor pixels in source image is greatly reduced in the encrypted image. The results of the correlation coefficient for vertical and diagonal neighbor pixels are similar to the horizontal neighbor pixels. The Signal to Noise Ratio (SNR) that is calculated for the above encrypted image in Fig. 1(b) is SNR=1.64494. The SNR is calculated by using the following formula, where *S* and *E* represent the source and the encrypted image respectively:

$$SNR_{db} = \frac{\sum_{i=1}^{width}\sum_{j=1}^{hieght}(E_{ij})^2}{\sum_{i=1}^{width}\sum_{j=1}^{hieght}(E_{ij} - S_{ij})^2} \quad (4)$$

To give more explanation details about the performance of the proposed encryption method and the effect of choosing different length for the ***VectorLength*** parameter on the performance of the method. Table 1 shows the experiments,ofdifferent bitmap images having different sizes (shown in Fig. 3), that are done to encrypt these images by using the proposed encryption method and the recorded results.

Figure 3: Bitmap Images used in the experiments

Table 1: Recorded results of the performance tests

Image	VectorLength	(*SNR*)	Encryption Time (second)	Decryption Time (second)	Correlation *r*
Boat	8	3.5519	6.567	6.614	0.0084541
	32	2.9855	1.887	2.075	0.0088460
	200	2.7947	1.201	1.201	0.0093383
Fireworks	8	2.0746	14.134	13.712	0.0059972
	32	1.8105	4.805	4.103	0.0052834
	200	1.6611	2.995	3.104	0.0010858
Penguins	8	3.6540	9.579	9.282	0.133088
	32	2.3931	2.776	2.309	0.0362672
	200	2.1301	1.264	1.217	0.0085089

From the above recorded results, we note the following points:

a. The values of *SNR*refer to that there is much distortion in the encrypted image, and its increasing when the ***VectorLength*** increase. This means that the encrypted image has good immunity against the Human Visual System (HVS) attack.
b. The encryption and decryption time is decreasing when the ***VectorLength*** increase. This means that when we try to give more security to the encrypted image by maximize the ***VectorLength***, this will not increase the encryption and decryption time.

c. The values of the correlation coefficient of the encrypted image are reducing gradually when the ***VectorLength*** increase. And they areminimized greatly when comparing their valueswith the values of the correlation coefficient of the source image.

V. CONCLUSIONS

A proposed image encryption method was introduced in this paper. The two reproduction operations (crossover and mutation) of GA are usedin the steps of the method to provide good security to the image.Because using long secret key of three parts,theproposed method become effect against the brute-force attack.The visual and the analytical tests that are used through the experiments showed that the proposed method promise to use it in the field of image/data encryption effectively.

VI. REFERENCES

[1] Petkovic, M., Jonker, W. Preface, "Special issue on secure data management," Journal of Computer Security, 17(1), pp.1-3, 2009.

[2] Bernstein, D.J., Chen, T.R., Cheng, C.M., Lange, T. & Yang, B.Y. "ECM on graphics cards". In A. Joux (Ed.), Advances in Cryptology - Eurocrypt 2009 (28th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cologne, Germany, April 26-30, 2009. Proceedings) Vol. 5479. Lecture Notes in Computer Science,pp. 483-501. Berlin: Springer, 2009.

[3] Bernstein, D.J., Lange, T., Peters, C.P. &Tilborg, H.C.A. van. "Explicit bounds for generic decoding algorithms for code-based cryptography". In International Workshop on Coding and Cryptography (WCC 2009, Ullensvang, Norway, May 10-15, 2009. Pre-proceedings). pp. 168-180. Bergen: Selmer Center, University of Bergen, 2009.

[4] Stevens, M., Sotirov, A., Appelbaum, J., Lenstra, A.K., Molnar, D., Osvik, D.A. &Weger, B.M.M. de. "Short chosen-prefix collisions for MD5 and the creation of a 18 rogue CA certificate". In S. Halevi (Ed.), Advances in Cryptology - CRYPTO 2009 (29th Annual International Cryptology Conference, Santa Barbara CA, USA, August 16-20, 2009. Proceedings) Vol. 5677. Lecture Notes in Computer Science. pp. 55-69. Berlin: Springer, 2009.

[5] Kahate A., "CRYPTOGRAPHY AND NETWORK SECURITY", Tata-McGraw-Hill, 2nd edition, 2008.

[6] H. El-din. H. Ahmed, H. M. Kalash, and O. S. Farag Allah, "Encryption quality analysis of the RC5 block cipher algorithm for digital images," Menoufia University, Department of Computer Science and Engineering, Faculty of Electronic Engineering, Menouf-32952, Egypt, 2006.

[7] D.R. Stinson, "Cryptography Theory and Practice," CRC Press, Inc., 2002.

[8] Arnold EA, Avez A, "Ergodic Problems of Classical Mechanics", Benjamin, W. A., New Jersey, Chap. 1, pp.6, 1968.

[9] Cheng-Hung Chuang, Zhi-Ye Yen, Guo-Shiang Lin, et al, "A Virtual Optical Encryption Software System for Image Security", JCIT, vol. 6, no. 2, pp.357-364, 2011.

[10] BrahimNini, ChafiaMelloul, "Pixel Permutation of a Color Image Based on a Projection from a Rotated View", JDCTA, vol. 5, no. 4, pp.302-312, 2011.

[11] Li. Shujun, X. Zheng "Cryptanalysis of a chaotic image encryption method," Inst. of Image Process. Xi'an Jiaotong Univ., Shaanxi, This paper appears in: Circuits and Systems, ISCAS 2002. IEEE International Symposium on Publication Date: 2002, Vol. 2, pp.708-711, 2002.

[12] Wei Ding, Wei-qi Yan, and Dong-xu Qi, "A Novel Digital Hiding Technology Based on Tangram Encryption", IEEE Proceedings of on NEWCAS 2005,and Conways Game", Proceeding of 2000 International Conference on Image Processing, Vol. 1, pp. 601-604, Sept. 2000.

[13] Zhao Xue-feng, "Digital Image Scrambling Based on the Baker's Transformation", Journal of Northwest Normal University (Natural Science), Vol. 39, No. 2, pp. 26-29, Feb. 2003.

[14] Bao Guan-jun, Ji Shi-ming, and ShenJian-bin, "Magic Cube Transformation and Its Application in Digital Image Encryption", Computer Applications, Vol. 22, No. 11, pp. 23-25, Nov. (2002).

[15] Zhu Guibin, Cao Changxiu, Hu Zhongyu, et al., "An Image Scrambling and Encryption Algorithm Based on Affine Transformation", Journal of Computer-Aided Design & Computer Graphics, Vol. 15, No. 6, pp. 711-715, June. 2003.

[16] Li Chang-Gang, Han Zheng-Zhi, and Zhang Hao-Ran, "Image Encryption Techniques: A Survey", Journal of Computer Research and Development, Vol. 39, No. 10, pp. 1317-1324, Oct. 2002.

[17] Scharinger J, "Fast Encryption of Image Data Using Chaotic Kolmogorov Flows", Journal of Electronic Imaging, Vol. 7, No. 2, pp.318-325, 2009.

[18] Behnia S, Akhshani A, Mahmodi H, et al, "A Novel Algorithm for Image Encryption Based on Mixture of Chaotic Maps", Chaos Solutions & Fractals, Vol. 35, No. 2, pp.408-419, 2008.

[19] Patidar V, Pareek NK, Sud KK, "A New Substitution-diffusion Based Image Cipher Using Chaotic Standard and Logistic Maps", Communications in Nonlinear Science and Numerical Simulation, Vol. 14, No. 7, pp.3056-3075, 2009.

[20] Wong KW, Kwok BSH, Yuen CH, "An Efficient Diffusion Approach for Chaos-based Image Encryption" Chaos Solutions & Fractals, Vol. 41, No. 5, pp.2652-2663, 2009.

[21] Douglas, R. Stinson, "Cryptography – Theory and Practice", CRC Press, 1995.

[22] Wenbo M., "Modern Cryptography: Theory and Practice", Publisher: Prentice Hall PTR, Copyright: Hewlett Packard, 2004.

[23] Menzes A. J., Paul, C., Van Dorschot, V., Vanstone, S. A., "Handbook of Applied Cryptography", CRS Press 5th Printing; 2001.

[24] Goldberg D.E., "Genetic algorithms in search optimization & Machine learning", Addison-Wesley, 1989.

[25] Kumar, A. and Rajpal, N. "Application of genetic algorithm in the field of steganography", Journal of Information Technology, 2(1), pp. 12–15, 2004.

[26] Kumar, A., Rajpal, N., and Tayal, A. "New signal security system for multimedia data transmission using genetic algorithms". NCC'05, January 20-28, IIT Kharagpur, pp. 579–583. 2005.

[27] Husainy, M. "Image encryption using genetic algorithm". Information Technology Journal, 5(3), 516–519. 2006.

[28] Tragha, A., Omary, F., and Kriouile, A. "Genetic algorithms inspired cryptography". A.M.S.E Association for the Advancement of Modeling & Simulation Techniques in Enterprises, Series D: Computer Science and Statistics. 2005.

[29] Tragha, A., Omary, F., and Mouloudi, A. "ICIGA: Improved cryptography inspired by genetic algorithms". International Conference on Hybrid Information Technology (ICHIT'06). pp. 335-341, 2006.

[30] RasulEnayatifar a, Abdul Hanan Abdullah, "Image Security via Genetic Algorithm", International Conference on Computer and Software Modeling, pp.221-226, 2011.

Short Bi Data for the Author

Mohammed Abbas Fadhil Al-Husainyreceived the M.Sc. and Ph.D. degrees in 1996 and 2002, respectively. From 1997 to 2002, he was a lecturer in the Department of Computer Science, Al-Hadba University of Mosul. Since 2002 he has been an assistant professor in the Departments: Computer Science and Multimedia Systems, Faculty of Science and Information Technology, Al-Zaytoonah University of Jordan. He lectures in the areas of microprocessors, data structures, algorithm design and analysis, digital design systems, operating systems, cryptography, computer organization, programming languages. His research interests are in the broad field of algorithm design, including multi-media data processing, scheduling algorithms, and cryptography algorithms.