RAISE THE STRENGTH OF CRYPTOSYSTEM USING VARIETY EFFECTS

Dr. Mohammed A. Fadhil Al-Husainy
Department of Computer Science, Faculty of Information Technology,
Middle East University, Amman, Jordan

Abstract

Nowadays, images became one of the most types of transmitted data through Internet. Some of these images carry secret information, therefore having an effective cryptosystem for hiding the information inside these images become an urgent need. Many traditional encryption methods are unable to achieve a high degree of protection for information. This paper presents nontraditional method for image encryption through applying substitution and transposition operations in different ways on both key and data. Series of (Linear and Circular), (Left, Right, Up and Down) **rotation** operations on (Bits and Bytes) of key and data are applied to make good confusion effects in the data. Moreover, **XOR** Boolean operations also applied on the key and data to make diffusion effects in the data. These two types of operations will produce a large set of keys. Using this large number of different keys in encrypting image will raise the strength of the encryption system used and achieve a high degree of protection for image. To test the security degree and performance of the encryption system, the system has been applied using different images and analyzing the results key space, key sensitivity, and statistical analysis and other criteria. From these tests, we can conclude that the encryption system can be used effectively to protect digital images.

Keywords: Rotation, XOR, Transposition, Substitution, Information Security

Introduction

Data encryption or cryptography means using a way to protect the information, from unauthorized people, that can be extracted from the data. Cryptography consists of two operations: Encryption operation which transforms the original data to the coded data and Decryption operation which transform back the encrypted data to the original data. In cryptography, key is used to know if a user has permission to deal with information. Only person who knows the key, which is used by the encryption system, can decrypt the coded data and get the information from the original data. The key is the power point of any encryption system. Therefore the strength of any encryption system depends mainly on the key even if the algorithm used known (Petkovic, Jonker, Preface, 2009),(Stevens, et al. 2009),(Kahate, 2008).

Recently, the advancement in technologies of digital and the spread of computer networks give an ability to exchange a lot of digital data over different networks. These data usually contain information which is either private or confidential. Therefore, security techniques become necessary to provide the necessary protection for this information (El-din, H., Ahmed, H., Kalash, H. M., and Farag Allah, O. S.. 2006).

The velocity of development in networks technologies and multimedia allows transmit and share a huge number of digital images through Internet and other networks (Kahate A., 2008). The protection of the multimedia content should be done by using different methods or techniques to achieve a high level of security for this content. These techniques are depending mainly on encryption and they provide either security against piracy, communication security or both. Communication security of digital media (text, images, sound and video) can be implemented by applying of symmetric key encryption. Such data of digital media can be treated as a sequence of binary numbers and all these binary data can be encrypted using an encryption system such as Advanced Encryption Standard (AES) or Data Encryption Standard (DES) (Stinson, 2002). It is very hard to decide what level of security needed to satisfy good protection for the data. It's wise to perform a careful comparison between the cost of the multimedia information to be protected and the cost of the protection technique used itself.

Illegal copying and distribution for digital images enforce us protect these images and make this issue very important in Information technology field (Kahate, 2008), (Arnold, Avez, 1968), (Cheng-Hung, Zhi-Ye, Guo-Shiang, et al. 2011), (Brahim Nini, Chafia Melloul. 2011). Converting an image to another hard understandable image is the core of the work of image encryption techniques (Shujun, Zheng. 2002). Sure, this is a first phase of any cryptography system, where to retrieve the original image from the encrypted image, decryption phase must be applied. Different encryption and decryption techniques are using today for different types of images and applications. One of these techniques was proposed by (Megha Seth, Reader and Kaminee Salam, 2014) uses a selective block encryption technique to achieve the different goals of security such as Availability, Confidentiality, and Integrity. (Quist-Aphetsi Kester, 2013) in his paper used an image encryption technique for selected facial area based on RGB pixel shuffling of m*n size image to provide his technique with two strong points: make it difficult to restore the encrypted image for off-the-shelf software and also make it easy to reconstruct the face back in case the picture or video for the law enforcement agencies. Authors (Nithin N, Anupkumar M Bongale and G. P. Hegde, 2013) presented a block cipher algorithm named FEAL, also called as Japanese Encryption algorithm. FEAL works almost similar to Data Encryption Standard (DES) algorithm, but it is faster than DES.

There are variety of image encryption algorithms available at present, such as Arnold map, Tangram algorithm (Wei, Wei-qi, and Dong-xu. 2000), Baker's transformation (Zhao. 2003), Magic cube transformation (Bao, et al. 2002), and Affine transformation (Zhu, Cao, Hu, et al.. 2003) etc. The secret-key cannot be separated actually from the algorithm in some cryptographic systems. This represents drawback in the needs of the recent cryptographic techniques and make these techniques susceptible to different attacks. Nowadays, most researchers tried to develop new image encryption techniques to overcome the weak point above (Stinson. 2002), (Li, Han, and Zhang. 2002). Another attempt has been presented by (Reddy S. Jyoteeswara Prasad and R. V. S. Sathyanarayana, 2013) focuses on using a binary image as a "keyimage" with the same size as the original image to be encrypted. Keyimage is either 2D grayscale or a 3D color image based on the original 2D (Grayscale image) or 3D (Color) image respectively. Al-Husainy (2012) employed the transposition and substitution operations with XOR and rotation bit operations to produce good diffusion and confusion effects in the encrypted image.

The main goal of the encryption algorithms, that been proposed by authors, is to maximize the distortion in the encrypted data and by using a large number of keys of large size and generated randomly.

Method

Usually, most encryption methods are trying to use one of the two main operations or both: *transposition* and *substitution*. The transposition operation produces a confusion effect in the data while the substitution operation produces a diffusion effect in the data. The idea behind the use of many different kinds of operations in generating keys and encrypting image comes from the truth that the diversity and the large number of keys will add much confusion and difficulties in the face of attackers to break the encrypted image. The encryption system uses different data structures to represent keys and image during the stages of encryption and decryption.

The encryption system represents the keys that are used as a set of two-dimensional square matrices (see Figure 1). Effects performed by the system on the keys can be classified as follow:

- **Confusion Effect:** The system deals with elements of any key as bytes that have been arranged in concentric rings. *Confusion* effects in the key happen by doing two types of rotation operations that used to perform a 90° circular rotation (right or left) for bytes in the cells of each ring in the key.
- **Diffusion Effect:** In addition to the rotation operations on the key, *diffusion* effects in the key through doing XOR bit-operation between bytes of the key and a mask matrix.

The encryption system splits image data into parts to form a set of two-dimensional square matrices have the same size of key (see Figure 1). Effects performed by the system on the image can be classified as follow:

- **Confusion Effect:** The system deals with elements any parts of data as string of bits that have been arranged in rows and columns. *Confusion* effects in the data happen by doing four types of rotation operations (right or left) in rows and (up or down) on columns in any part of data.
- **Diffusion Effect:** In addition to the rotation operations on the parts of data, *diffusion* effects in the data through doing XOR bit-operation between bytes of the keys and the parts of data.

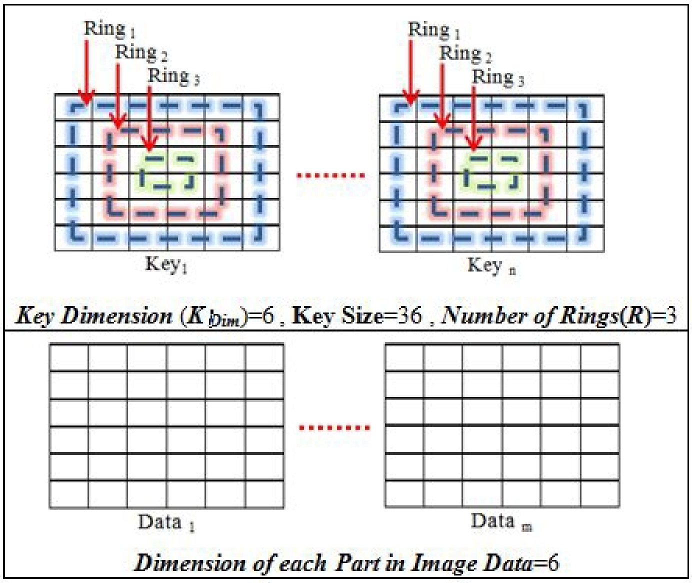

Ring 1
Ring 2
Ring 3
Key1
Ring 1
Ring 2
Ring 3
Key n
Key Dimension (K Dim)=6 , Key Size=36 , Number of Rings(R)=3
Data 1
Data m
Dimension of each Part in Image Data=6

Figure 1: Secret Keys and Parts of Image Data.

To give the reader good understanding for the operations that implemented by the encryption system, brief explanations of some important definitions and terminologies have been listed below:

- **Secret Key (*K*):** a two-dimensional square matrix. Each dimension is (***K_{Dim}***). See (Figure 1) for a simple example.

- **Secret Key Dimension (K_{Dim}):** an even positive integer number which represents the number of bytes in each dimension in ***K***. For example: if K_{Dim}=32, this means that ***K*** consists of (32×32) bytes = 1024 bytes. See Figure 1 for a simple example. It's recommended to choose $K_{Dim} \geq 32$. This will make the key size enough to produce a secure encrypted image.
- **Number of Rings in Secret Key (*R*):** is a number of concentric rings of cells in the secret key ***K***, $R= K_{Dim} /2$. See Figure 1 for a simple example.
- **Mask Matrix (*M*):** a two-dimensional square matrix. Its dimensions are same as secret key dimensions. The encryption system uses this matrix to make a diffusion effect on the key.
- **Original Image (*S*):** A bitmap image such as (.bmp) type is a two-dimensional matrix of pixels. The image has Width, Height, and Palette. The encryption system processes the image file in different way, it treats file of the image as a contiguous series of bytes, where each value of byte is between (0...255) and (1byte = 8bits). The number of bytes in the original image ***S*** (i.e., S_{Length}) equal (Width×Height×Palette)
- **Encrypted Image (*E*):** Like the original image ***S***, but it been generated from ***S*** after finishing the operations that are executing through the encryption stage.
- **Decrypted Image (*D*):** Like the original image ***S***, but it been generated from ***E*** after finishing the operations that are executing through the decryption stage.
- **XOR Boolean Operation:** Boolean operation which uses in the encryption and decryption phases to make changes in the bits of the bytes of the image.
- **Ring rotation (Right and Left) Operations:** Two operations that used to perform a 90° circular rotation (right or left) of bytes in the cells of each ring of the key ***K*** during the encryption and decryption stages.
- **Bit rotation (Right, Left, Down and Up):** bits rotation of the bytes in one row or one column of the data.

Now, the encryption stage of the method has the following steps:

```
Step1: Read (from the user) the dimension $K_{Dim}$ of the secret key K.
Step2: Read (from the user) the necessary number of bytes for the secret key K. The
       number of bytes in K equals $(K_{Dim} \times K_{Dim})$
Step3: Set $R = K_{Dim} / 2$
Step4: Read the bytes of the original image S and store it's bytes as a list SList of two-
       dimensional square matrices of size $(K_{Dim} \times K_{Dim})$. The length of SList (i.e.,
       number of matrices in SList ) is calculated as follow:
       $SList_{Length} = (S_{Length} / (K_{Dim} * K_{Dim}))$
       In other words, SList represents a three-dimensional matrix of size $(SList_{Length} \times$
       $K_{Dim} \times K_{Dim})$
Step5: For w=1 to $SList_{Length}$ do (Step6 to Step11)
```

```
Step6: Doing XOR operation between K and SList: (Diffusion effect in data)
        For i =1 to $K_{Dim}$ do
            For j =1 to $K_{Dim}$ do
                SList[w, i, j] = SList[w, i, j] XOR K[i, j]
Step7: Doing rotation operations on SList: (Confusion effect in data)
        For i =1 to $K_{Dim}$ do
        {
            Row = K[i, i] mod ($K_{Dim}$)
            Column = K[i, $K_{Dim}$ - i] mod ($K_{Dim}$)
            ColumnRound = K[i, $K_{Dim}$ - i] mod ($K_{Dim}$ * 8)
            RowRound = K[i, i] mod ($K_{Dim}$ * 8)
            if ((Row mod 2) == 0)
                Rotate right the bits of the row Row in SList  number of round equal RowRound
            else
                Rotate left the bits of the row Row in SList  number of round equal RowRound
            if ((Column mod 2) == 0)
                Rotate down the bits of the column Column in SList  number of round equal ColumnRound
            else
                Rotate up the bits of the column Column in SList  number of round equal ColumnRound
        }
Step8: Build the mask matrix M from the bytes of K and SList[w]. The values of bytes
       in M come by using the following rule:
        For i =1 to $K_{Dim}$ do
            For j =1 to $K_{Dim}$ do
                If ((SList[w, i, j] mod 2) == 1)
                    M[i, j] = SList[w, i, j]
                Else
                    M[i, j] = K[i, j]

Step9: Build RList list of size R. Each value in RList represents one of the two types of
       rotation operations that will be done on each ring in the key K. (0: rotate right,
       1: rotate left). The values of RList are calculated using the following rule:
        x =1
        For i =1 to $K_{Dim}$ do
            For j =1 to $K_{Dim}$ do
                If (((w mod 2) == 0)) and ((i mod 2) == 0)))
                {
                    RList [x] = (SList[w, i, j] mod 2)
                    x = x +1;
                }
                If (((w mod 2) == 1)) and ((i mod 2) == 1)))
                {
                    RList [x] = (SList[w, i, j] mod 2)
                    x = x +1;
                }
Step10: Doing rotation operations on key K: (Confusion effect in key)
```

```
          For $j$ =1 to $R$ do
              Rotate each ring $j$ in $K$ (either right or left) according to the desired type of rotation in $RList[j]$.
Step11:   Generate new key $K$ by doing XOR operation between $K$ and $M$: (Diffusion effect in key)
          For $i$ =1 to $K_{Dim}$ do
              For $j$ =1 to $K_{Dim}$ do
                  $K[i, j] = K[i, j]$ XOR $M[i, j]$
Step12: Store the bytes of $SList$ in the encrypted image $E$.
```

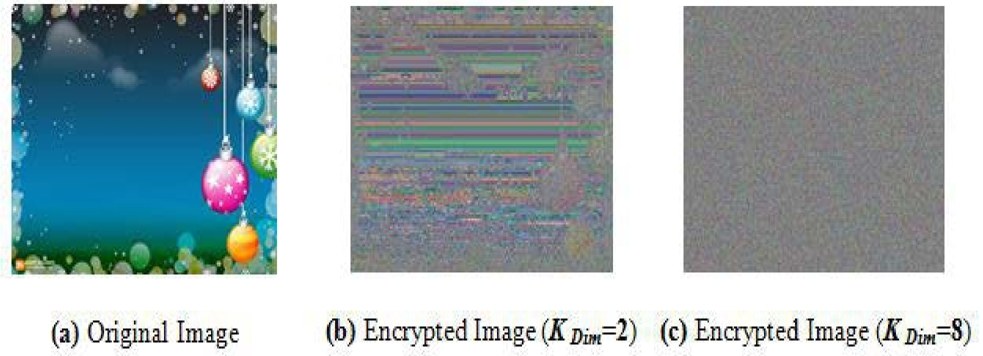
In the decryption stage, the same steps mentioned above are applied on the encrypted image E (but in inverse sequence) to re-generate the original image S.

Results

To test the performance of the proposed encryption system, a set of bitmap images of different sizes has been used in the system. Many measurements such as the key space, key sensitivity, and statistical analysis have been used to check the quality of security of the encryption system.

A. Key space analysis

The number of bits of the key (key space) should be big enough to minimize the feasibility of using the brute-force attack and produce an efficient encryption system. The space of the secret key in the encryption system is ($8 * K_{Dim}^{2}$) bits, where $K_{Dim} \geq 2$. This means that the size of the key can be very large, this will make the guessing of the key by attacker be very hard. Moreover, number of keys that are used by the system will increase whenever the image size increase. To clarify the effect of K_{Dim} value that is chosen to encrypt an image. Figure 2 shows the application of the encryption system using different K_{Dim} values of the secret key K.

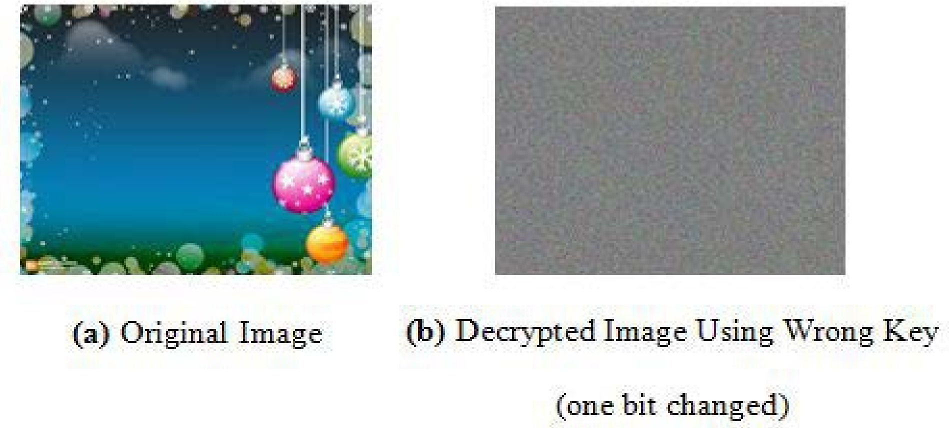


(a) Original Image (b) Encrypted Image (K_{Dim}=2) (c) Encrypted Image (K_{Dim}=8)

Figure 2: (a) Original image. (b, c) Encrypted images using different K_{Dim} of the secret key K.

B. Key sensitivity

The key sensitivity feature of the encryption system can be tested by changing one bit in the key and then use it to decrypt the encrypted image. The decrypted image that is generated using a wrong key is completely different from the decrypted image when we using correct key as it's shown in Figure 3. This means that the encryption system is high sensitive with a small change in the key.

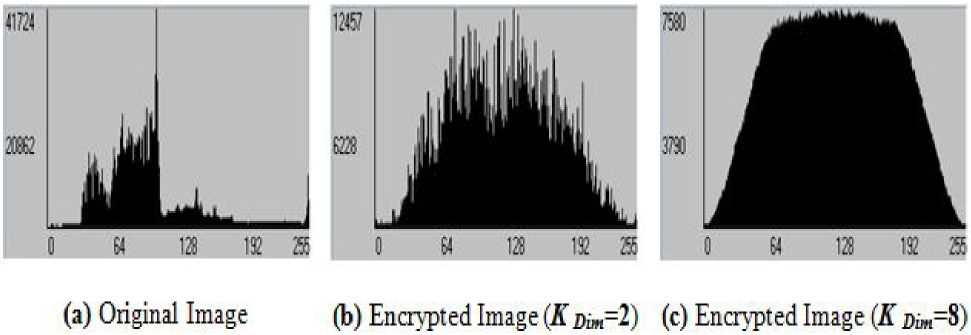


(a) Original Image (b) Decrypted Image Using Wrong Key (one bit changed)

Figure 3: (a) Original image. (b) Decrypted image using wrong key

C. Statistical analysis

The strength of the encryption system against any statistical attack can be proved through histogram. Figure 4 shows the histograms of original and the encrypted image respectively. The histogram of the encrypted image is different from the histogram of the original image. This prevents any unauthorized person which uses the statistical attack from getting any useful information through the histogram.



(a) Original Image (b) Encrypted Image (K_{Dim}=2) (c) Encrypted Image (K_{Dim}=8)

Figure 4: Histogram of: (a) The original image in figure2. (b), (c) The encrypted images in figure2.

Also, number of comparison tests on different images was performed by using the proposed encryption system with Data Encryption Standard

(DSE) and Advanced Encryption Standard (AES). During these tests, measurements such as Signal to Noise Ratio (***SNR***), Peak Signal to Noise Ratio (***PSNR***), Normalized Mean Absolute Error (***NMAE***), and Time of Encryption (***Time***) have been used to check the performance and the quality level of protection that the system can achieve. Table 1 shows some images used in the tests. The recorded results of the experiments have been summarized in Table 2.

Table 1: Some images used in experiments

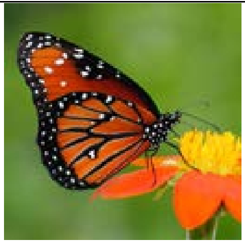
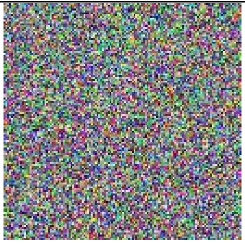
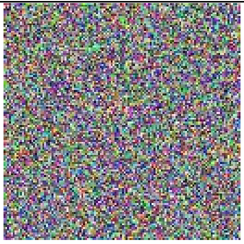
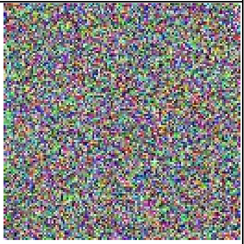
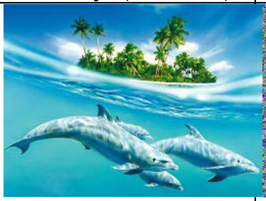
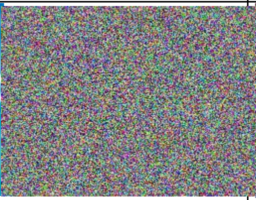
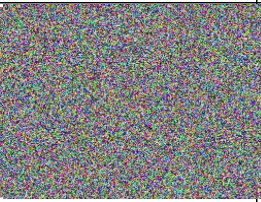
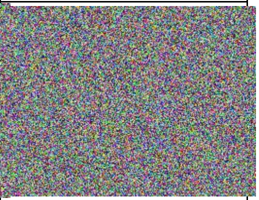
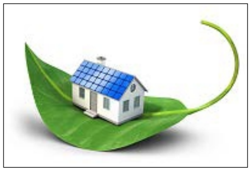
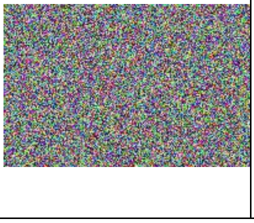
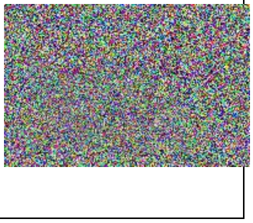
Original Image	***Encryption Method***		
	AES	**DES**	**Proposed Method**
 Butterfly (128×128)			
 Dolphins (256×192)			
 GreenHome(200×132)			

Table 2: Results of experiments

Image	**Measurement**	**Encryption Method**		
		AES	**DES**	**Proposed Method**
Butterfly	$\boldsymbol{SNR}_{db}$	3.11	3.13	3.11
	$\boldsymbol{PSNR}_{db}$	7.31	7.31	7.29
	$\boldsymbol{NMAE}_{\%}$	94.808	95.068	94.546
	$\boldsymbol{Time}_{msec}$	471	484	98
Dolphins	$\boldsymbol{SNR}_{db}$	2.79	2.83	2.85
	$\boldsymbol{PSNR}_{db}$	7.37	7.38	7.39
	$\boldsymbol{NMAE}_{\%}$	62.630	62.452	61.659
	$\boldsymbol{Time}_{msec}$	312	373	247
GreenHome	$\boldsymbol{SNR}_{db}$	0.90	1.01	0.90
	$\boldsymbol{PSNR}_{db}$	5.69	5.75	5.44
	$\boldsymbol{NMAE}_{\%}$	52.400	51.989	51.997
	$\boldsymbol{Time}_{msec}$	324	374	147

Conclusion

Using large number of keys, complex key, and big key space in encrypting image improves the strength of the encryption system and adds extra protection for the images. A series of rotation and XOR operations on the key and image during the encryption phase produced good degree of confusion and diffusion in the encrypted image. The analysis of performance and security of the encryption system showed that the system produces a high degree of protection for image. From the recorded results of the tests, we can find that the image encryption system is good and effective enough to use it in protecting images in variety of domains.

References:

Al-Husainy M. A Novel Encryption Method for Image Security, International Journal of Security and Its Applications, Vol. (6), No. (1), 1-8, 2012.

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

Bao Guan-jun, Ji Shi-ming, and Shen Jian-bin. Magic Cube Transformation and Its Application in Digital Image Encryption. Computer Applications. 22(11):23-25, 2002.

Brahim Nini, Chafia Melloul. Pixel Permutation of a Color Image Based on a Projection from a Rotated View. JDCTA. 5(4):302-312, 2011.

Cheng-Hung Chuang, Zhi-Ye Yen, Guo-Shiang Lin, et al. A Virtual Optical Encryption Software System for Image Security. JCIT. 6(2):357-364, 2011.

El-din, H., Ahmed, H., Kalash, H. M., and Farag Allah, O. S.. 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.

Kahate A. Cryptography and Network Security: Tata-McGraw-Hill, 2nd edition. 2008.

Li Chang-Gang, Han Zheng-Zhi, and Zhang Hao-Ran. Image Encryption Techniques: A Survey. Journal of Computer Research and Development. 39(10): 1317-1324, 2002.

Megha Seth, Reader, Kaminee Salam, Image Encryption and Decryption Using

Selective Block Encryption Technique, International Journal of Computer Science & Engineering Technology (IJCSET), Vol. 5 No. 09 Sep 2014.

Nithin N, Anupkumar M Bongale and G. P. Hegde, “Image Encryption Based on FEAL Algorithm,” vol. 2, March 2013.

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

Quist-Aphetsi Kester, “A Cryptographic Image Encryption Technique for Facial-Blurring of Images,” vol. 3, May 2013.

Reddy S. Jyoteeswara Prasad and R. V. S. Sathyanarayana, “Image Encryption Using Color Key Images,” vol. 2, October 2013.

Shujun, Li., Zheng, X.. 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; 2:708-711.

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; P.55-69. Berlin: Springer, 2009.

Stinson, D.R.. Cryptography Theory and Practice: CRC Press, Inc..; 2002.

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, 1: 601-604, 2000.

Zhao Xue-feng. Digital Image Scrambling Based on the Baker’s Transformation. Journal of Northwest Normal University (Natural Science), 39(2):26-29, 2003.

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, 15(6):711-715, 2003.