Indonesian Journal of Electrical Engineering and Computer Science
Vol. 11, No. 2, August 2018, pp. 496~503
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v11.i2.pp496-503

An Efficient Schema of A Special Permutation Inside of Each Pixel of an Image for Its Encryption

Hana Ali-Pacha[1], Naima Hadj-Said[2], Adda Ali-Pacha[3], Mustafa Mamat[4], Mohamad Afendee Mohamed[5]
[1,2,3]Laboratory of Coding and Information Security, University of the Sciences and the Technology of Oran - Mohamed Boudiaf, USTO-MB, Po Box 505 El M'Naouer Oran 31000, Algeria
[4,5]Faculty of Informatics and Computing, Universiti Sultan Zainal Abidin, Terengganu, Malaysia

Article Info

Article history:

Received Feb 7, 2018
Revised Apr 12, 2018
Accepted Apr 21, 2018

Keywords:

Cryptography
Permutation
Feistel
Entropy
Pixel Correlations
NPCR
UACI

ABSTRACT

The development of communications and digital transmissions have pushed the data encryption to grow quickly to protect the information, against any hacking or digital plagiarisms. Many encryption algorithms are available on the Internet, but it's still illegal to use a number of them. Therefore, the search for new the encryption algorithms is still current. In this work, we will provide a preprocessing of the securisation of the data, which will significantly enhance the crypto-systems. Firstly, we divide the pixel into two blocks of 4 bits, a left block that contains the most significant bit and another a right block which contains the least significant bits and to permute them mutually. Then make another permutation for each of group. This pretreatment is very effective, it is fast and is easy to implement and, only consumes little resource.

Corresponding Author:

Mohamad Afendee Mohamed,
Faculty of Informatics and Computing,
Universiti Sultan Zainal Abidin,
Terengganu, Malaysia.
Email: mafendee@unisza.edu.my

1. INTRODUCTION

With the advent of the Internet and computer communication networks (local area networks, metropolitan area networks and wide area networks) and the use of satellite links, the new industrial revolution in the computing and telecommunications has led to storage and the transmission of large amounts of confidential data and a growing concern to protect their access. Many secure access techniques have been proposed namely cryptography via substitution and permutation, steganography via data embedding and biometric. Cryptography is needed to have the non-intelligible data except to the audience desired, and although there are already many algorithms available [1], still research for other encryption systems is still relevant [2, 3].

Cryptography can be divided into public key and secret key cryptosystem [4]. As it name applied, public-key system makes use of different key for encryption and decryption as opposed to secret-key system which employ the same key. Modern secret-key system can be grouped into two, those that emerged from the concept introduced by Feistel network such as Data Encryption Standard (DES) and those which are not [5].

Image security is an important subject and similar to text, it can be addressed using encryption [6]. However, there are other characteristics of image that must be considered and not be found in text. Things such as image quality and relationship between neighboring pixels plays important role in adding security features to an image.

In this studies, we proposed bit scrambling within each pixel prior to encryption. The idea is based on Feistel network which was known to be able to increase the security of the data. This process is simple and efficient; it is based on intra-block permutations of pixels of an image.

This paper is organized as follows: Section 2 description of the Feistel network. In Section 3, we propose our approach of encryption based on the Feistel network with a block is equal one pixel. In Section 4, we analyze the results and we valid the proposal cipher. Section 5 concludes this paper.

2. FEISTEL CIPHER

Horst Feistel introduced symmetric cipher structure commonly used in the construction of many modern block ciphers including the Data Encryption Standard (DES) which later be adopted as a standard. The structure consists of encryption and decryption function of which its module is similar, and for some even identical and only require a reversal of the key schedule.

A Feistel network is an iterated cipher with an internal function called a round function. Let say we have a message M split into blocks of certain size each is divided into two halves L_0, and R_0 and K_0, K_1… K_n be the sub-keys for the rounds 0, 1, …, n respectively. Following to Figure 1, for each round i, we compute $L_{i+1}=R_i$ and $R_{i+1}= L_i \oplus F(R_i, K_i)$ to produce the ciphertext (R_{n+1}, L_{n+1}) after n rounds. Likewise, the decryption of a ciphertext (R_{n+1}, L_{n+1}) is achieved by computing $R_i = L_{i+1}$ and $L_i =R_{i+1}\oplus F(L_{i+1}, K_i)$ for i= n, n-1… 1, 0 to obtain the original (L_0, R_0).

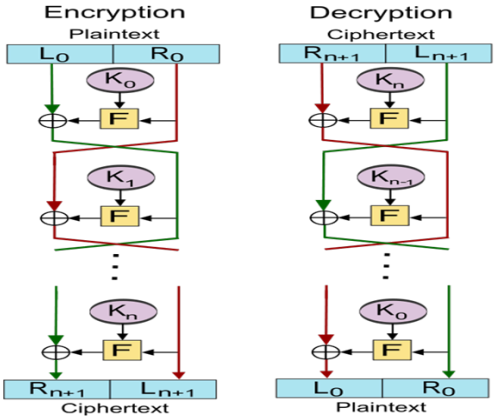

Figure 1. Feistel Network

This network has inspired our proposed technique by taking as a block a single pixel and divide into two parts of 4 bits for each for further processing.

3. PROPOSED SCRAMBLING SYSTEM

In The plaintext is denoted as M, is the message to be encrypted [7, 8]. The file may be originated from some text, image, voice, or video. One the lowest level, M is just a se-quence of binary representation, which can be transmitted or stored. In case of a pixel of an image, it is associated to a value less than 256 gray level (1 byte). Therefore, the plaintext is a finite series of bytes (8 bits), and each byte is in the form shown in Figure 2.

Figure 2. Pixel of Plaintext

3.1 Intra-Block Permutation of Pixels (IBPP)

Permutation is about rearranging objects (P1, P2, .., P8) among each other. A permutation of n elements is a bijective function that forms a group structure and thus object is always recoverable by some inverses [9, 11]. The idea of our technique is to introduce the Intra-block Permutation of Pixels (IBPP) which consists of two steps. Firstly, a mutually permutation of these two blocks is done: blue to red and vice versa [3] such that the output is now shown in Figure 3.

Figure 3. Inter-block permutation

Next, we permute the positions within each block of the new pixel permuted. To clarify on how it works, consider a permutation function $\sigma = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 3 & 4 & 1 & 2 \end{pmatrix}$ which can be defined as: $\sigma(1) = 3$; $\sigma(2)=4$; $\sigma(3)=1$; $\sigma(4)=2$. The result of this operation is shown in Figure 4.

Figure 4. Intra-block permutation

If we use the operation above within our encryption function, therefore to recover the original message, an inverse function σ^{-1} called decryption need to be applied and can be shown as in Figure 5.

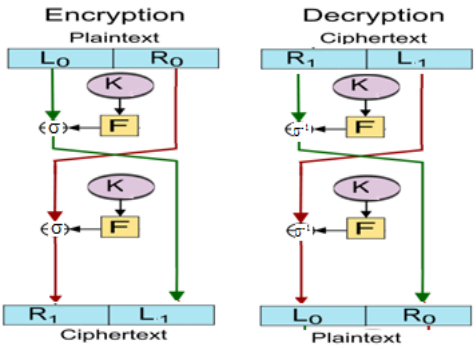

Figure 5. Proposed Specific Permutation

Let F be the function of permutation σ in encryption part and, F is the inverse function of permutation σ in decryption part with L_0, L_1, R_0 and R_1 are parts of pixel of 4 bits for each one. Then, the proposed block cipher is implemented under the operating encryption CBC Mode.

3.2 Inverse Permutation of Pixels PBIP

The decryption is the process to produce a pixel back to the original plaintext. We note that, the inverse permutation PBIP (IPBIP) is equals PBIP. In other words, $PBIP^{-1}$ = PBIP such that PBIP(PBIP(half-Pixel)) = half-Pixel. For any permutation σ, its inverse is denoted as σ^{-1}, therefore, applying a function σ followed by σ^{-1}, or otherwise is equivalent to applying the identity permutation [9, 11]. Consider $\sigma^{-1} = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 3 & 4 & 1 & 2 \end{pmatrix}$.

For 4 bits, we can represent them by a 4-tuple of distinct elements and constitutes an ordered list without possible repetition, that is to say, in which the order of the elements is taken into account (if the we

interchange two elements of the list, there is a different list and an element cannot occur more than once). Table 1 shows the 24 different permutations.

Table 1. Permutation of 4 elements

σ(1)σ(2)σ(3)σ(4).			
1234	2134	3124	**4123**
1243	**2143**	**3142**	4132
1324	2314	3214	4213
1342	**2341**	3241	4231
1432	2431	**3412**	**4312**
1423	**2413**	**3421**	**4321**

In what follows, we consider only the permutations that satisfy the following assumptions:

$\sigma(1)=1$ et $\sigma(i)=j \ \forall \ i,j = 2, 3, 4$
$\sigma(1)\neq 1$ et $\sigma(i)\neq i \ \forall \ i,j = 2, 3, 4$

They are 15 permutations validate for our application. The identity permutation, was duplicated for having 16 permutations, and we placed them in a table of correspondence.

3.3 Fields of the Encryption Key

With this pretreatment, there will be more than 6 bits in the field of the encryption key. Two bits to determine the nature of the permutation according to Table 2 and four bits to determine the type of the permutation according to Table 3 (the reading of matrix is: line + column).

Table 2. Nature of the Permutation

Bits	Nature of the Permutation
00	Without permutation
01	Permutation of the right block only
10	Permutation of the left block only
11	Permutation of the two blocks

Table 3. Type of the Permutation

	00	01	10	11
00	1234	1234	1243	1324
01	1342	1432	1423	2143
10	2341	2413	3142	3412
11	3421	4123	4312	4321

4. RESULTS AND INTERPRETATIONS

To measure the performance of our proposed technique without any encryption system, we have used images of 512 * 512 pixels size, and each pixel is coded as 8 bits. The application was simulated on a PC HP ProBook 4520 running Windows 7 Professional 64 bits with intel® Core™ i5 CPU M 480 @ 2.67GHZ (4 CPUs),-2.7GHz processor, 4096MB RAM memory and ATI Mobility Radeon HD 6370 graphics card. We take the adding encryption key as 111110, i.e. Permutation of the two blocks and 4312 ($\sigma(1)= 4, \sigma(2)= 3, \sigma(3)= 1, \sigma(4)= 2$).

4.1 Histogram of Images

A histogram of an image is a discrete function that maps the number of pixel for each color intensity simply by counting the number of pixel having certain intensity in the image [8]. Therefore, it can be displayed as probability density function.

Figure 6-8, it is observable that the histogram of plaintext image has changed tremendously from its corresponding ciphertext image. Moreover, the uniformity of the histogram of encrypted image hardening the task of statistical extraction of original pixels from the plaintext image.

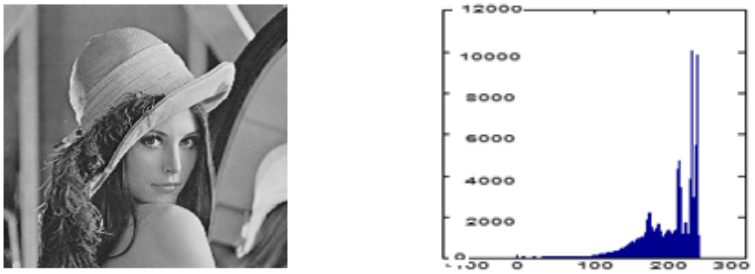

Figure 6a. Image plaintext of Lena and her histogram

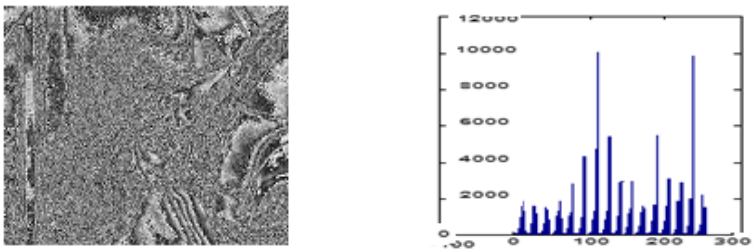

Figure 6b. Image encrypted of Lena and her histogram

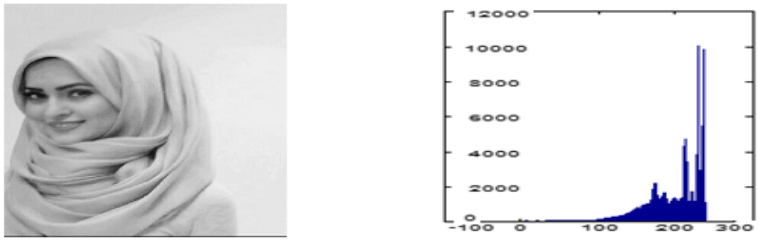

Figure 7a. Image plaintext of Hedjab and her histogram

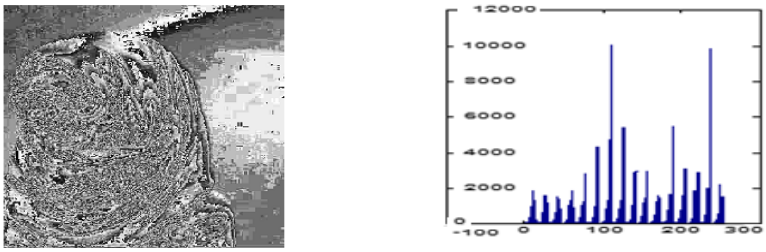

Figure 7b. Image encrypted of Hedjab and her histogram

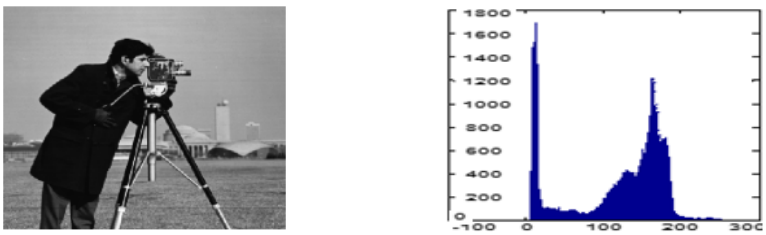

Figure 8a. Image plaintext of cameraman and her histogram

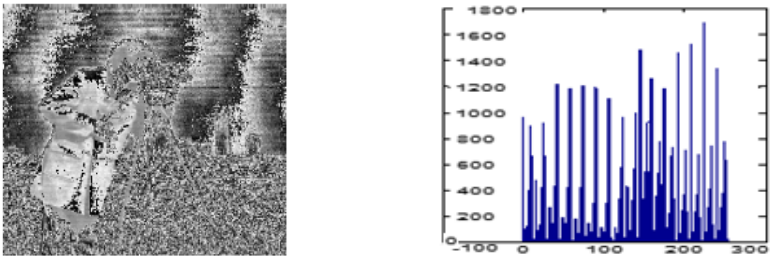

Figure 8b. Image encrypted of cameraman and her histogram

4.2 Correlation between two adjacent pixels

To determine statistically the correlation between two random variables is similar to studying the strength of the bond that can exist between these variables via linear regression. As an example, the correlation coefficient between two sets having the same number of elements, X=(x_1, …, x_n) and Y=(y_1, …, y_n) of each of the two series is obtainable via linear correlation coefficient of Bravais-Pearson given by $Coef(X,Y) = \frac{cov(X,Y)}{\sqrt{D(X)}.\sqrt{DY}}$ [8], whereas the covariance between X and Y is given by $cov(X,Y) = \frac{1}{N}\sum_{i=1}^{N}\big((X_i - E(X)\big).(Y_i - E(Y)))$. The average of X is $E(X) = \frac{1}{N}\sum_{i=1}^{N} X_i$ whereas the average of Y is $E(Y) = \frac{1}{N}\sum_{i=1}^{N} Y_i$. The standard deviation of X is $D(X) = \frac{1}{N}\sum_{i=1}^{N}(X_i - E(X))^2$ whereas the standard deviation of Y is $D(Y) = \frac{1}{N}\sum_{i=1}^{N}(Y_i - E(Y))^2$.

The correlation coefficient takes the value between -1 and 1. Technically, any values in the this range determine the degree of linear dependencies of these two variables. If the value of the coefficient close to -1 and 1, the highly correlated they are as opposed to lowly correlated when it is approaching 0.

We randomly selected 1000 pairs of two adjacent pixels from both encrypted and plaintext image for the purpose of testing. Figure 9 shows the correlation of horizontal pixels, correlation of vertical pixels, correlation of diagonal pixels for plaintext image of Lena.

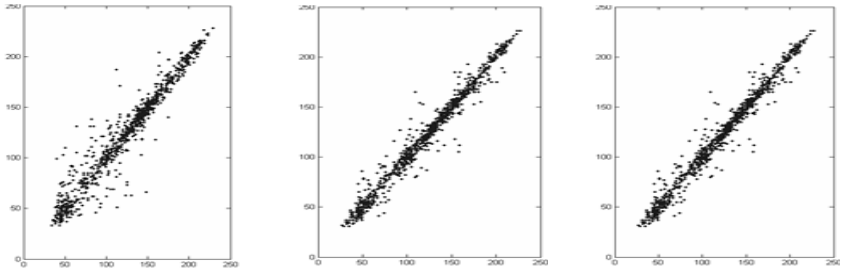

Figure 9. Distribution of the adjacent pixels plaintext image of Lena

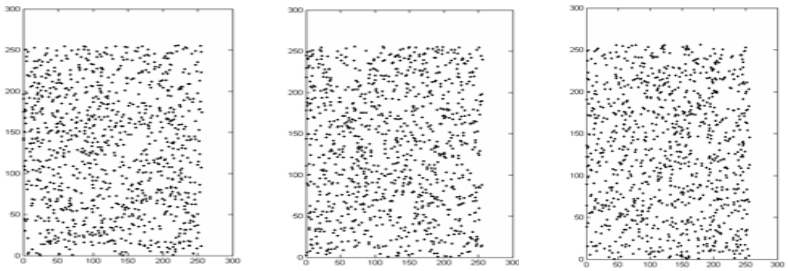

Figure 10. Distribution of the adjacent pixels for encrypted image of Lena

It is observed from Figure 10 that, the adjacent pixels are highly correlated on the encrypted image and thus the encryption has created a major difficulties in retrieving information. In addition, the

autocorrelation's coefficients which is close to 1 for plaintext images and 0 for ciphertext image proved the proper functioning of our proposed technique.

4.3 Calculation of the entropy

The average amount of information [8] associated with each symbol without memory source is defined as the mathematical expectation (denoted by E {.}). Specific information provided by the observation of each of the possible symbols $\{s_1,...,S_n\}$ called the entropy (in bits) is given by $H(S) = \sum_{i=0}^{2^N-1} P(M_i) \, . \, log_2 (\frac{1}{P(M_i)})$, where $P(M_i)$ represents the probability of symbol M_i.

It is noted from Table 4, the comparisons between the entropies of plaintext images and their encrypted, that the entropy of the encrypted images is greater than the entropy of the images plaintext, showing that, the encryption creates a high level of disorder. The uniformity of the encrypted image histogram indicates that the gray levels occurs almost at the same number of times and consequently the entropy is drive higher up. Consequently for each pixel, the entropy must be close to the theoretical 8 bits and this is shown in Table 4.

Table 4. Comparison of correlation coefficients and entropy between the plaintext images and their encrypted image

	Correlation Coefficient of the		Entropy	
Picture	Image Plaintext	Ciphertext Image	Plaintext Image	Ciphertext Image
lena	0.9719	0.0114	7.4455	7.7502
cameraman	0.9335	-0.0175	7.0097	7.0311
peppers	0.9913	-0.0092	6.9769	7.0901
coins	0.9749	-0.0116	6.3071	6.6185
football	0.9454	-0.0110	5.6760	6.7058
mandrill	0.8675	0.0152	6.9010	7.3579
clown	0.9711	0.0096	7.3406	7.5205
barbara	0.8954	-0.0087	7.6100	7.6321
hidjab	0.9381	-0.0095	6.6865	7.3478
Emir Abdelkader	0.9781	0.0107	7.0332	7.0960

4.4 Difference between the original and the permuted images

To evaluate the strength of image encryption algorithms/ciphers with respect to differential attacks we use two measurements, the Number of Pixels Change Rate (NPCR) and the Unified Average Changing Intensity (UACI) [12]. The purpose is to test the difference between the original image P_1 and the permuted one C_1.

If D is a matrix with the same size as images P_1 and C_1, $D(i,j)$ is determined as $D(i,j) = \begin{cases} 1 & if\ P_i(i,j) \neq C_1(i,j) \\ 0 & else \end{cases}$, and thus $NPCR = \frac{\sum_{i=0}^{M-1}\sum_{j=0}^{N-1} D(i,j)}{MxN} x100\%$, where M and N are the width and height of P_1 and C_1 respectively whereas the $UACI = \frac{1}{MxN}\sum_{i=0}^{M-1}\sum_{j=0}^{N-1}\frac{/P_1(i,j)-C_1(i,j)/}{255} x100\%$. Conventionally, a high NPCR/UACI score is usually interpreted as better resistance to differential attacks. The measurements of the NPCR and UACI obtained on Lenna image are 99.5945% 33.4253% respectively, and thus it is proven that our technique resists to the differential attacks.

5. CONCLUSION

To test the effectiveness of this pretreatment, we rely upon the four commanly used indicators namely image histogram, correlation between two adjacent pixels, entropy, and NPCR/UACI measurements. All four gave a good appreciation for adding this pretreatment to a cryptographic algorithms

REFERENCES

[1] Agrawal M. Cryptography: A Survey. *IETE Technical Review*. 1999; 16(3-4): 287-296.
[2] Denning DE. The Future of Cryptography. Georgetown University Accessible at www.guru.cosc.geortown.edu denning/crypto. 1996.
[3] Schneier B. Applied Cryptography-Protocols, Algorithms and Source Code in C. New York, Second Edition. John Wiley & Sounds, Inc. 1996.
[4] Mohamed MA. A Survey on Elliptic Curve Cryptography. *Applied Mathematical Sciences*. 2014; 8(154):7665-7691.

[5] Dayananda LN, Senthil KK. (2017). Survey on cryptographic block cipher methods to solve the security issues. *International Journal of Engineering and Technology*. 2017; 9(4): 3115-3129.
[6] Kumari M, Gupta S, Sardana P. *3D Research*. 2017; 8: 37.
[7] Rotman J. A First Course in Abstract Algebra. Third Edition, University of Illinois at Urbana-Champaign, Upper Saddle River, New Jersey. Prentice Hall. 2008.
[8] Chen G, Mao Y, Chui CK. A symmetric image encryption scheme based on 3D chaotic cat maps. *Chaos, Solitons and Fractals*. 2014; 21:749–761.
[9] W Feng, W Bao. A new technology of remote sensing image fusion. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*. 2012; 10(3): 551-556.
[10] Scherk J. Algebra: A Computational Introduction. University of Toronto. Chapman and Hall/CRC. 2009.
[11] Wilson MC. Random and exhaustive generation of permutations and cycles. *Annals of Combinatorics*. 2009; 12(4): 509–520.
[12] Y Wu, J P Noonan, and S Agaian. NPCR and UACI Randomness Tests for Image Encryption. *Cyber Journals: Multidisciplinary Journals in Science and Technology, Journal of Selected Areas in Telecommunications*. 2011; 31-38.