International Conference on Control, Engineering & Information Technology (CEIT'13)
Proceedings Engineering & Technology - Vol.1, pp. 202-206, 2013

A new approach for image encryption combining random itterative permutation and chaotic outputs

Amaria Wael, Hassen seddik, Bouslehi hamdi
CEREP, ESSTT
Tunis, Tunisia
amaria.wael@hotmail.fr
Hamdouchb@gmail.com
CEREP, ESSTT
Tunis, Tunisia

***Abstract*— Encryption systems have gained higher interest for on aright protection and data hiding. Applied for data transmitting, it can be subjected easily to high jacking or possible decryption because of the continue increase of the computer speed. This paper presents a new encryption approach that combines three encryptions techniques: permutation technique, changing intensity pixels combined with encryption technique that use logic operator. Our encryption method is based on using multiple keys by combining iterative random permutations and chaotic encryption by using logistic maps mixed with original image by a logic operator and it's used in changing intensity pixels. This approach is completely reversible with multiple keys to augment perplexity. The presented techniques allow an entire protection against signal modification or theft.**

***Key words*—symmetric Encryption, random Permutation, multiple key, chaotic function.**

I. INTRODUCTION:

Communication security is an application layer technology to protect any transmitted data (speech, image, computer messages) against unwanted communication as well as to guard the information from unauthorized modification while in transit. There are three fundamental methods of secured communication available, namely, cryptography, steganography and watermarking. Among these three methods, the first one, cryptography [1]-[2], deals with the development of techniques for converting information between comprehensible and incomprehensible forms during information transfer. Steganography [3]-[4], is anther technique for hiding and extracting information to be transmitting using a carrier signal.

The last one, watermarking [5]-[6], is a means of developing proper techniques for hiding proprietary information in the perceptual data.

The proposed method of encryption image uses many good keys, using random permutation function combined with a function of changing pixel intensity and another one of encryption (vernam method). Since a large number of keys have used, the security level offered is also high.

Further, the amount of redundant information available in the encrypted image is kept as low as possible, thereby providing fairly high security level against casual observers.

II. THE LOGISTIC FUNCTION:

The logistic map [7] is a basic mapping polynomial, which has chaotic behavior, and it can be obtained by a very simple nonlinear dynamical equation [8].

Recurrence logistics is an example where the recurrence is not linear. This recurrence was popularized by the biologist Robert May in 1976. Its recurrence relation is.

$$\tau(x_n) = x_{n+1} = \lambda\, x_n(1-x_n) \qquad (1)$$

The control parameter "λ" is fixed and chosen so that equation (1) has a chaotic behavior $(3.57 < \lambda < 4)$ [9]. However, if we study the map with a different value of "λ", it shows that it is a trigger for the chaos. Mathematically, the "Logistic map" is written with

"x" is a number between 0 and 1, and represents the initial condition $0 < x_0 < 1$.

"λ" is a positive number [10].

III. ALGORITHM ENCRYPTION BASED ON ITERATION OF THE IMPROVED "LOGISTIC MAP":

After the iteration of the function of "Logistic map" N times, we obtain N value x_n between 0 and 1 and x_0: the initial value and $0 < x_0 < 1$ and λ: control parameter.

In our encryption algorithm we take $x_0 = 0.1777 \in [0, 1]$ and $\lambda = 3.759889 \in [3.57, 4]$, we obtain a chaotic signal and the value generated chaotic sweep the entire range of value between 0 and 1. After 70 iterations, the signal from equation 1 is summarized in (figure 1).

Indeed, the chaotic function "Logistic Map" has several properties, such as frequency and sensitivity to initial conditions (this is a characteristic of all chaotic systems: if we take a different value which is very close to x_0 then the values from the iteration change completely. If we take a different value which is very close to the values of "λ"the iteration changes dramatically: this can be seen by a simulation tool Matlab, in particular by the value of these functions which are completely random. Although they are limited from a few bands, the iterative values never give the impression to converge even after an infinite number of iterations. The change of control parameters (λ) and the initial condition (x_0) by very

close values in order to know the decryption algorithm always gives cryptograms so radically different that it is interesting to use the function in logistic encryption).

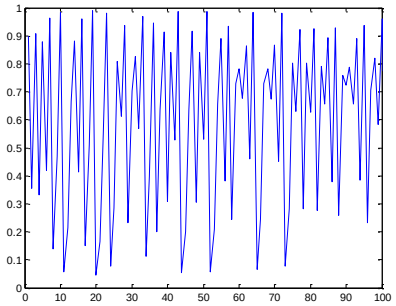

Figure.1: The logistic map outputs after 100 iteration.

IV. PROPOSED APROCH (ENCRYPTION)

This approach is based on three functions: a random permutation function, a feature that has changed is the intensity of the pixels of the image based on the chaotic function and an encryption function "Vernam" also using chaotic function.

A. Permutation function:

➢ Mathematical Definition

Mathematically our permutation function P can be defined by the following equation:

$\forall x \in N$ such as $P(x)=y$ so $\exists\ P^{-1}(y)=x$ with $P \circ P^{-1}(x) = I$ and $P^{-1}(y)-x=\varepsilon$ with $\varepsilon \to 10^{-4}$

➢ Used function:

The permutation has a paramount importance in this approach. It is used to make the system more safe and reliable. An image having pixels swapped does not make sense as opposed to the original image.

We can take "*IM*" the image to encrypt and C is the permutation key that can be string or natural value. If our key is string we convert it to ASCII by using the flowing function:

ASCII: $X \rightarrow Y$

As *X*: string

Y: vector of integers.

C = ASCII (C_1) with $C_1 \in$ X and Y $\in C$

Our function contain "*n"* round each used a key C_i with *n* is the length of our key. In each round the same function that is repeated but with a different key according to the key given thus explained the principle function we can detailed tow round which are as follows:

❖ First round:

$$C = \{C_1, C_2, \ldots\ldots\ldots\ldots, C_n\} \quad (2)$$

With $C_i \in \mathbb{N}^*$ and i$\in$ [1, *n*] and "*n"* is the length of encryption key. The process of permutation is divided into several rounds. The round number is the length of the key "key1".

So we start by the decomposition of our image into many vector whose its length is the value of C_1.

$$IM = V_1^{C_1} V_2^{C_1} \ldots\ldots\ldots\ldots V_n^{C_1} \quad (3)$$

IM: is the image to encrypt.

$V_1^{C_1}$:The first vector of image *IM* of length C_1.

After the decomposition of our image, we permute each bloc by using the permuted function.

$$P(V_1^{C_1} = E_1\, E_2 \ldots\ldots E_{C_1}) = V_{P1}^{C_1} = E_{C_1}\, E_{C_1-1} \ldots\ldots\ldots E_1 \quad (4)$$

$V_{P1}^{C_1}$: The first vector from the image *IM* after permutation.

$$P(V_2^{C_1} = E_1 E_2 \ldots\ldots\ldots\ldots E_{C_1}) = V_{P2}^{C_1} = E_{C_1}\, E_{C_1-1} \ldots\ldots\ldots E_1 \quad (5)$$

$V_{P2}^{C_1}$: The 2nd vector from the image *IM* after permutation.

Finely we have an image permutated by C_1:

$$P_{C_1}(IM) = IM_{(P,C_1)} = V_{P1}^{C_1} V_{P2}^{C_1} \ldots\ldots\ldots\ldots\ldots V_{Pn}^{C_1} \quad (6)$$

$P(IM) = IM_{(P,C_1)}$: Image after the first permutation.

❖ Second round:

$IM_{(P,C_1)}$: is the output of the first round and the input of the second round which start too by the decomposition of our image permuted by C_1 into many blocs whose its length is the value of C_2.

$$IM_{(P,C_1)} = V_{P1}^{C_1} V_{P2}^{C_1} \ldots\ldots V_{Pn}^{C_1} = V_1^{C_2} V_2^{C_2} \ldots\ldots\ldots V_n^{C_2} \quad (7)$$

As the first round after the decomposition of our image, we permute each vector by using the permuted function.

$$P(V_1^{C_2} = E_1 E_2 \ldots\ldots E_{C_2}) = V_{P1}^{C_2} = E_{C_2}\, E_{C_2-1} \ldots\ldots E_1 \quad (8)$$

$$P(V_1^{C_2} = E_1 E_2 \ldots\ldots\ldots E_{C_2}) = V_{P2}^{C_2} = E_{C_2} E_{C_2-1} \ldots\ldots\ldots E_1$$

. (9)

$$P_{C_2}(IM) = IM_{(P,C_2)} = V_{P1}^{C_2} V_{P2}^{C_2} \ldots\ldots\ldots\ldots V_{Pn}^{C_2}$$

Finally we have an image permutated by C_2:

$$P_{C_2}(IM) = IM_{(P,C_2)} = V_{P1}^{C_2} V_{P2}^{C_2} \ldots\ldots\ldots\ldots V_{Pn}^{C_2} \quad (10)$$

When we executed this function we have an image permuted by a random manner so there is not a mathematical model that ensures the passage from permuted image $IM_{(P,C_n)}$ to original image *IM* .

B. Changing pixel intensity:

After permutation performed on the message to be encrypted, which is in our case an image, we change the intensity to complicate the task. This procedure is to change the intensity of

all image pixels by reducing or adding a random value between 0 and *d*/ 2, generated by a logistic function "F".

With $x_0 = 0.1777$ and $\lambda = 3.7598895$

"D" denotes the image dynamic, d = 256 for the case of images coded on 8 bits.

$$F = b_1 b_2 \ldots\ldots\ldots\ldots\ldots\ldots b_n \quad (11)$$

with b_1 : a binary value

$$F = B_1 B_2 \ldots\ldots\ldots\ldots\ldots\ldots B_k \quad (12)$$

With $B_1 = b_1 b_2 \ldots b_7$: the first 7 binary values

$IM(i, j)$ Is the pixel coordinate (i, j) avec $i \in [1..h]$ and with $j \in [1..l]$ such that h is the number of line and l is the number of columns.

And $k = l * k$

If $IM(i, j) < d/2$ then $IM_c(i, j) = IM(i, j) + D_k$ (13)

else $IM_c(i, j) = IM(i, j) - D_k$ (14)

D_k : The value decimal of B_k from the logistic function.

$$D_k = \sum_{i=1}^{n} 2^{i-(k(i-1))} \times b_{i-(k(i-1))} \quad (15)$$

" D_k "is a random value belong to the interval [0, *d*/2].

C. *Encryption:*

For a more secure encryption system, we try to complicate the task. Each pixel intensity found after steps "1" and "2", which represent the permutation and change of intensity, is modified by the XOR function whose input pixel intensity change and a random value generated using a logistic function.

The XOR operator is extremely common as a component in more complex ciphers. By itself, using a constant repeating key, a simple XOR cipher can trivially be broken using frequency analysis. If the content of any message can be guessed or otherwise known then the key can be revealed. Its primary merit is that it is simple to implement, and that the XOR operation is computationally inexpensive. A simple repeating XOR cipher is therefore sometimes used for hiding information in cases where no particular security is required.

$$F = b_1 b_2, \ldots\ldots\ldots\ldots\ldots\ldots, b_n \quad (16)$$

Binarising the image:

$$IM = IM_1, IM_2, \ldots\ldots\ldots\ldots, IM_n \quad (17)$$

$$IM_{cry} = IM \oplus F \quad (18)$$

So we can put the original image and its histogram followed by the encrypted one and its histogram to show the difference between them:

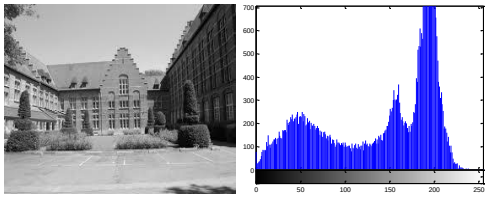

Figure 2. orignal image and its histogram

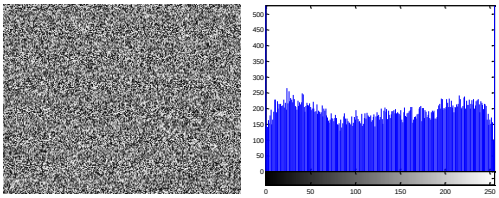

Figure 3. Encrypted image and its equalized Histogram

D. *Complete encryption algorithm :*

In our algorithm, we used three encryption functions, which are presented before so to recapitulate we form a diagram recapitulative to show our full work.

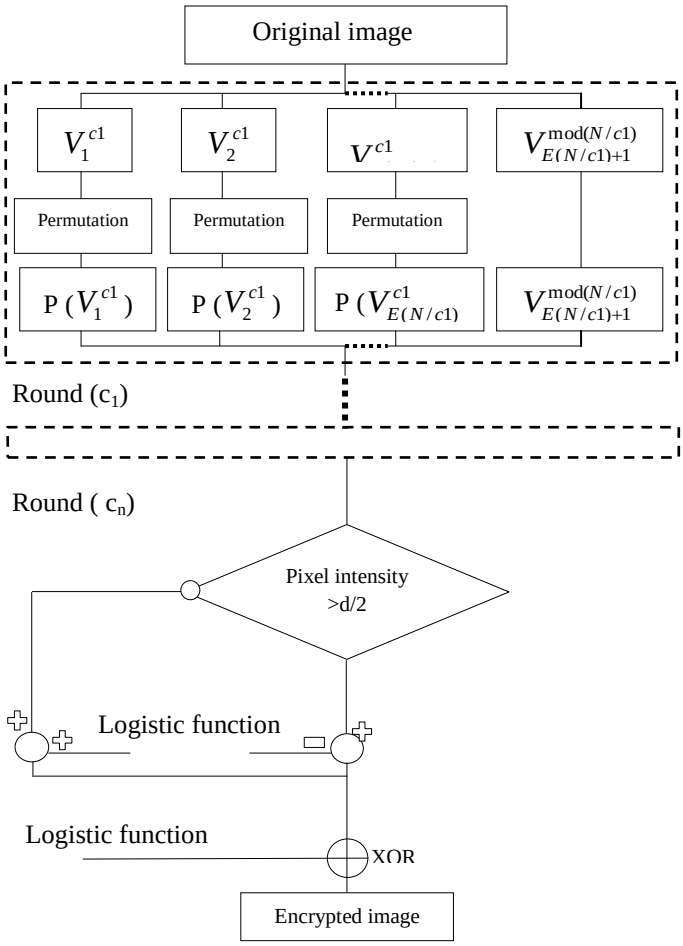

Figure 4. Diagram of a new encryption approach

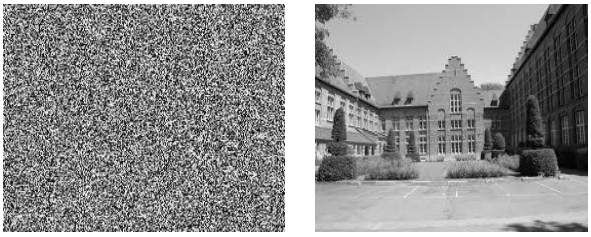

Figure 5. The orignal image and the encrypted one

V. IMAGE DECRYPTION STEPS:

A. *decryption function*

Once the image is encrypted and transmitted just go the opposite way to recover the original image. The last procedure performed on the image during encryption is the XOR function, which has the original image as input whose pixels are switched and having intensities modified and logistics values. This procedure will be the first to perform during decryption by introducing the same parameters that are sent as "key".

$$IM_{cry} = IM_1\ IM_2 \ldots\ldots\ldots\ldots\ldots IM_n \tag{19}$$

$$IM = IM_{cry} \oplus F \tag{20}$$

B. *Changing intensity*

Original intensity of the image are modified, therefore they must be restored by adding (or subtracting) the same value subtracted (or added) during the process of changing intensity. As indicated previously the choice of the operator "+" or "-" is stored in a vector that is sent as a key.

img_bin: is a binary image formed during encryption process.

If img_bin (i ,j)=0 then $IM(i,j) = IM_c(i,j) - D_k$ (21)

Else $IM(i,j) = IM_c(i,j) + D_k$ (22)

V_k : The value decimal derived from the logistic function.

$$D_k = \sum_{i=1}^{n} 2^{i-(k(i-1))} \times b_{i-(k(i-1))} \tag{23}$$

C. *Permutation*

The last operation to be done to recover the original image is permutation one more time.

$$P_{C_n}^{-1}(IM_{(P,C_n)}) = P_{C_n}^{-1}(V_{P_1}^{C_n}\ V_{P_2}^{C_n} \ldots\ldots\ldots\ldots V_{P_n}^{C_n}) \tag{24}$$

$$P_{C_{(n-1)}}^{-1}(IM_{(P,C_{(n\_1)})}) = P_{C_{(n-1)}}^{-1}(V_{P1}^{C_{n-1}}\ V_{P2}^{C_{n-1}} \ldots\ldots V_{Pn}^{C_{n-1}}) = IM_{(P,C_{n-2})}$$
$$\ldots\ldots\ldots\ldots\ldots\ldots \tag{25}$$
$$IM = V_1^{C_1}\ V_2^{C_1} \ldots\ldots\ldots\ldots V_n^{K_1}$$

Once completed, the original image is covered. For a better interpretation of the results, we calculate the difference between the original image and the image decrypted. Losses for this approach is zero, no data is lost.

Figure 6. decrypted image

VII. RESULTS AND INTERPRETATION:

Evaluation tools are used to evaluate the encryption performance so we must quantify its performance and characteristics.

A. *MSE (mean square error):*

We use to quantify the error between the original sequence and the encrypted

$$MSE = \frac{\sum_{i=1}^{n} (I_i - I_i^*)}{n} \tag{26}$$

Such as *n* is the length of the sequence.

*I** and *I* represent respectively the original image and the encrypted image.

B. *PSNR (Peak Signal Noise Ratio):*

It is a function derived from the MSE, it allows complete degradation of image, and it measures offset (in dB) of the original image by contributing to the encrypted image.

$$PSNR = 10Log_{10}\left(\frac{255^2}{MSE}\right) \tag{27}$$

Result of simulation:

PSNR = Inf

C. *correlation:*

A correlation is a number between 0 and 1 which measures the degree of association between two signals. A positive value for the correlation implies a positive association. A negative value for the correlation implies a negative association. In our case we studied the correlation between the original signal and the signal decoded (see figure (2) and figure (6)).

The correlation between the original image and decryption image is equal to 1 deciphered. So it can be concluded from results obtained in this code the correlation between the transmitted image and the received image is perfect, which means that our algorithm is reversible 100%.

D. Test by subtraction:

You can also test our signal by calculating deference between the original signal and the signal is decrypted gives us the following result:

Figure 7: Subtraction result

The previous image shows a high similarity between the two image (original, decrypted) since the deference between them is 0 so we haven't any loss of information this prove that our algorithm reversible 100 %.

VIII. CONCLUSIONS

In this paper we presented a new approach to encryption that contain all the criterion of an encryption algorithm that is robust: the randomness of the function used which is provided by the chaotic function and the function of the permutation, so that complexity of the key in our case it is ensured by the choice of key (value integer or character string), although the choice of encryption methods they have to be conservative and reversible, which is the case in our work

REFERENCES

[1] A. J. Elbirt and C. Paar, "An Instruction-Level Distributed Processor for Symmetric-Key Cryptography," IEEE Trans. Parallel and distributed systems, vol. 16, no. 5, pp. 468-480, May 2005.

[2] W. Stallings, Cryptography and Network Security. Englewood Cliffs, NJ: Prentice Hall, 2003.

[3] E. Besdok, "Hiding information in multispectral spatial images," Int. J. Electron. Commun. (AEU) 59, pp. 15-24, 2005.

[4] S. Trivedi and R. Chandramouli, "Secret Key Estimation in Sequential Steganography," IEEE Trans. Signal Processing, vol. 53, no. 2, pp. 746-757, Feb. 2005.

[5] Y. Wu, "On the Security of an SVD-Based Ownership Watermarking," IEEE Trans. Multimedia, vol. 7, no. 4, pp. 624-627, Aug. 2005.

[6] A. Masoud and A. H. Tewfik, "Geometric Invariance in Image Watermarking," IEEE Trans. Image Processing, vol. 13, no. 2, pp. 145-153, Feb. 2004.

[7] Luo, J.; Shi, H., "Research of Chaos Encryption Algorithm Based on Logistic Mapping", IIH-MSP '06, International Conference on Intelligent Information Hiding and Multimedia Signal Processing, 2006, pp. 381-383, Dec 2006.

[8] Rich Schlesinger," A Cryptography Cource for Non-Mathematicians», the 1st annual conference on information security curriculum development, K Kennesaw , Georgia, October 08-08,2004.

[9] [1]Sudhir Keshari, [2]Dr. S. G. Modani '*Image Encryption Algorithm based on Chaotic Map Lattice and Arnold cat map for Secure Transmission'* [1,2]Department of Electronics and Communication Engineering, Malaviya National Institute of Technology, Jaipur, India. March 2011

[10] Yong Wang, Xiaofeng Liao, Tao Xiang, Kwok-Wo Wong, Degang Yang "Cryptanalysis and improvement on a block cryptosystem based on iteration a chaotic map" Received 14 May 2006; accepted 6 November 2006 .Available online 20 November 2006,Communicated by A.R. Bishop.