International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2016): 79.57 | Impact Factor (2015): 6.391

Watermarking and Encryption to Protect Multimedia Information

Rishi Batra, Abhishek Goyal, Sheetal Sharma

Abstract: ***As the information technology is developing, frauds are also increasing simultaneously; daily new techniques are coming to improve the security of information, i.e. steganography, encryptions and security of transmit signals. In this paper I present a scheme to secure multimedia information. The aim is to develop a way for increasing security of the transmitted signals. First the encrypted algorithm which is being based on the chaotic iteration of the logistic is improved. Simultaneously my method is presented which improve the efficiency of composition.***

1. Introduction

Affectability of the initials several algorithms are being proposed. The primary calculation utilizes the emphasis of the strategic guide to encode a message proposed by Batista. From that point forward, numerous different calculations are being proposed.

Xiang Tao which who proposed the new algorithm of chaotic encoding, Tao divided message into 64 bits and then uses the logistic function in order to code the message

I propose the improved algorithm which uses the different keys, by these improvements which further increases the security and the encryption process.

My aim is to combine two different techniques in order to secure multimedia information, watermarking and encryption. At first we use to water mark the information by inserting a message, then after we crypt the total message.

2. Presentation of Encryption – Watermarking Scheme

Following steps are used to present encryption algorithm (figure 1)

At the early we attempted chaotic algorithm for encryption and decryption of Xiang Tao to make it stronger against cryptanalysis attacks.

At second stage we made an algorithm to put a cover message in a cover image

At third stage we developed an algorithm at extracts the message inserted into the cover image.

2.1 Presentation of Xiang Tao Algorithm

Recently an algorithm was proposed by Xiang Tao based on logistic map which can recapitulate the algorithm in these steps

Step 1: We repeat logistic map present in Equation (1)

$$r(x_n) = x_n + 1 = \propto x_n (1 - x_n);\ x_n \in [0, 1] \quad (1)$$

$$\propto = 3.999996.$$

We take N = 70, we note ω the result of logistic map, repetition as given in the following Equation (2):

We will write ω in binary form.

$$\omega = r^{N}_{0}(x) \quad (2)$$

Step 2: We divide the binary order into blocks; each block is created by 8 bits (Equation 3)

$$m = p0, p1, p2 \cdots pl - 1, pl, \cdots p2l - 1, p2l \cdots (3)$$

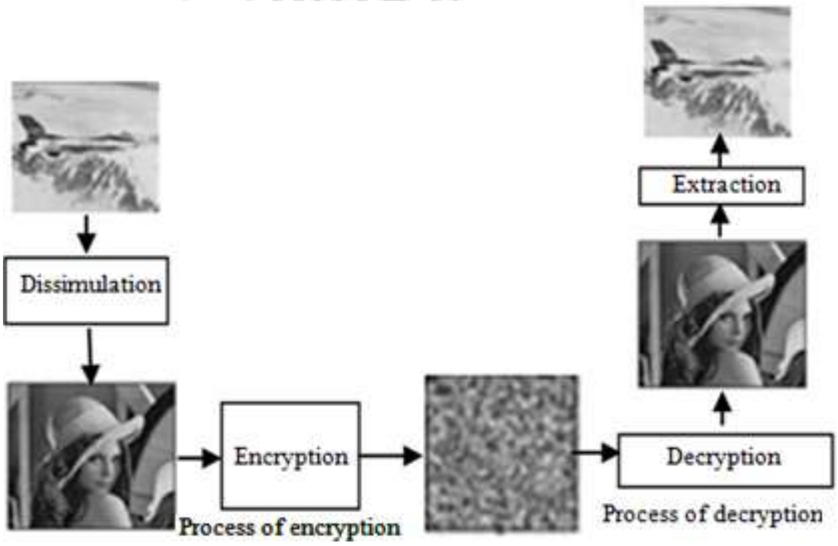

Figure 1: Encryption-watermarking system

Step 3: The representation of binary of 'x' is given by this method: (Equations. 4, 5, 6 and 7)

With, $x \in [0, 1]$, $b_j(x) \in [0,1]$

$$x = 0.b_1(x)\, b_2(x) \ldots b_i(x) \qquad (4)$$

$$b_i(x) = \sum_{r=1}^{2^i - 1} (-1)^{r-1} \Theta(r/2^i)(x) \qquad (5)$$

$$\theta_t = \begin{cases} 0 & si \quad x < t \\ 1 & si \quad x \geq t \end{cases} \qquad (6)$$

$$B_i^n = \{b_i(\tau^n(x))\}_{n=0}^{\infty} \qquad (7)$$

$$A_j = B_i^1 B_i^2 \ldots\ldots\ldots B_i^{64} \qquad A_j' = B_i^{65} B_i^{66} \ldots\ldots\ldots B_i^{70}$$

We take, $i = 3$, $x = 0.b_1(x)\, b_2(x)\, b_3(x)$ and we repeat 70 times. D_j is the decimal value of A_j', we will use this value to repeat the logistic map successively.

Step 4: We permute P_j and D_j by left cyclic shift.

Step 5: We apply XOR function between A *j* and P′ *j* (Equation 8)

$$Cj = P'j \oplus Aj \qquad (8)$$

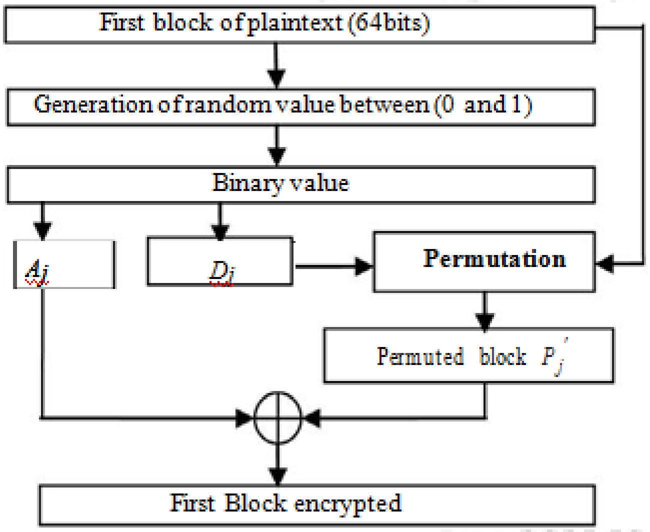

Figure 2: Farah at all algorithm

I divide C_i on 8 bits blocs.

$c_j, c_{j+1} \ldots, c_{j+7}$ and $p_j, p_{j+1}, \ldots p_{j+7}$ present respectively the blocks of plaintext.

We noticed by the reference of author [5] that the key and plaintext are not dependent on each other (figure 2), this error is corrected. This idea is to find the error between the key and the plaintext.

As clearly stated that when the key and plaintext are not dependent, the cryptosystem will be broken easily by using a specific plaintext method, so the idea is to mix the key of the encryption algorithm with the message, we noticed that the key is dependent only on message, that's why it is difficult to search for the key.

Figure 2 depicts different steps taken for encryption of the plaintext. D_j will be changed it will directly depend di on plaintext.

It was noted that the Tao algorithm is broken by Farah at all [6] .

Farah at all encryption algorithms is made up of two parts: To crypt the 1st block of the plaintext, another for the other block.

2.2 Encryption of the first block plaintext

70 different values generated are made by two blocks. One B_J 64-bit block and the other B_j' of 6 bits.

We note:
D_j : the decimal value of the last six bits of the 70 values generate random.

The result of D_j right permutation of P_j, gives P_j' (Equation 9).

$$C_j = A_j + P'j \qquad (9)$$

To encrypt the first block text Farah considers

$yy = rand$ (1, 70) as one key which allows to generate 70 random(unpredictable) values between and 1 from which we obtain A_1 and D_1 (A_1 trained (formed) by 64 bits and D_1 of 6 dits). In other words the generation of these values will be mask. It does not depend on the iterations of the logistic map (does not depend on the initial condition and on the parameter of control of the logistic map)

Thus key becomes (a, w, yy).

3. Proposed Algorithm

Encryption algorithm proposed present two parts; an encryption algorithm to the first text block (the same algorithm of Farah at all [6]) and an encryption algorithm of other blocks (Figure 3)

- **Encryption of the other blocks plaintext**

My algorithm is summarized by the following steps:
The steps 1, 2 and 3 are the same as the encryption algorithm of Xiang Tao in reference [4].

Step 4: we choose a new initial value x_0 and μ_2 value of control parameter of the logistic map. The logistic map is the result of this iteration (Equation 10).

$$g_2 = \tau^{64}(x_0) \qquad (10)$$

Step 5: g_2 is the initial value and μ_2 the parameter control of the logistic map**.**
We iterated the logistic map (m . n) times.

Step 6: The binary representation of (m . n) value is given by method (see Equations 4, 5 and 6) and z (1, $m.n$) is a vector line containing binary values.

Step 7: When z (i, j) =1 we permute P_j by D_j (see Equation 11) left cyclic bits. (See Figure 1) Otherwise unchanged.

D_1: Decimal value of 70 values generate by logistic map.

D_2: Decimal value of the first six bit of A_j.

D_3: Decimal value of the last six bit of A_j.

Each block to A_j is divided into two groups of eight bits.

We calculate the decimal value of each group; D_4 is the sum of these values.

$$D_j = (D_1 \; D_2 + D_2 \; D_3 + D_3 \; D_1) \bmod 64 \quad (11)$$

Step 8: We applied *XOR* function between A_J and $P_j{}'$. (Equation 8)

Step 9: Each block to C_j is divided into groups of 8 bits. We calculate the decimal value of each group. We note D_5 the sum of these values. After we calculate D^* by using (Equation 12)

$$D_j = (D_j \; D_5 + D_5 \; D_4 + D_4 \; D_j) \bmod 64 \quad (12)$$

If all blocs of plaintext are encrypted the encryption is finished, other side ω is calculated (Equation 13) and becomes the initial condition of the next block and returns to step 2.

$$\omega = r^{D^*}(\omega) \quad (13)$$

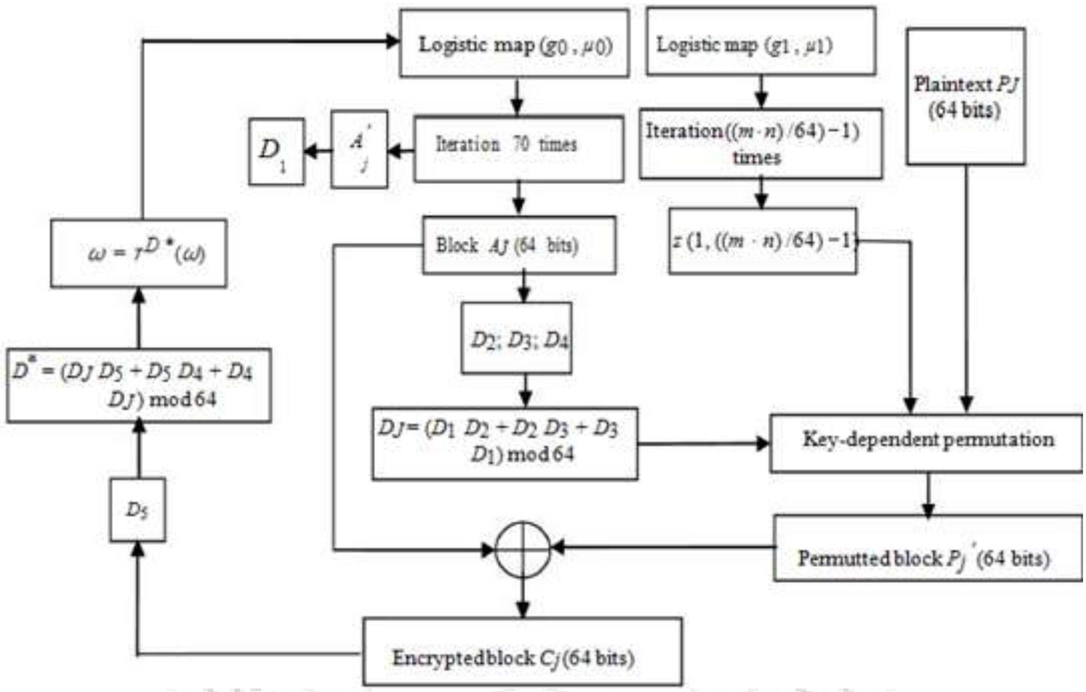

Figure 3: Proposed algorithm

From the above algorithm analysis, the attacker must possess Logistic map parameters g_0, μ_0, g_1, μ_1 and the random values *yy*, which are totally 5 keys. The encryption algorithm improves the security and the robustness of images. The decryption algorithm process can be seen as the inverse of the encryption process. (Figures 4, 5, 6, 7)

Figure 4: Image original

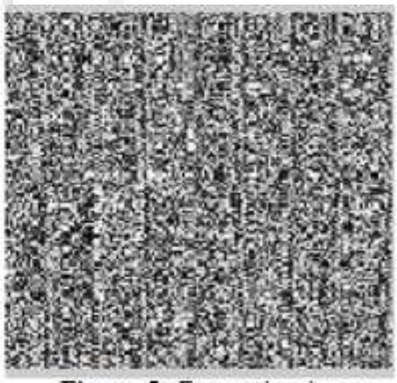

Figure 5: Encryption image

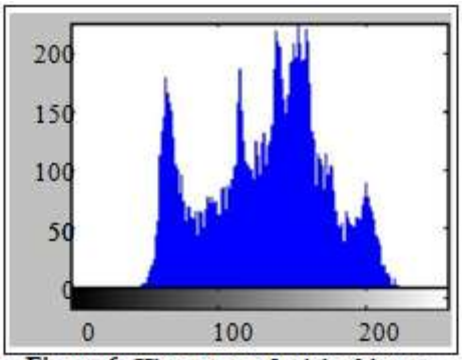

Figure 6: Histogram of original image

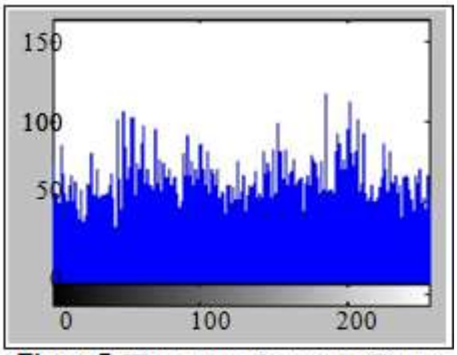

Figure 7: Histogram of encrypted image

3.1 Insertion of mark in DCT (Discrete Cosine Transform) domain

The dissimilation of information must be made in the frequency domain because of robustness of spatial domain.

In this area the data is inserted into very important location in the cover image that allows the protection against attacks such as compression.

1) Choose the appropriate size of inserted image (36.36 is used in our simulation)
2) Converting gray level image to binary message.
3) Divide the message into 8.8 block
4) Apply DCT to each block.
5) We chose two blocks to insert information in the medium frequency
6) Insertion of the secret message with modification of selected DCT coefficients.
7) Apply the DCT inverse on each block.

Let (u_1, v_1) and (u_2, v_2) are two indices chosen for the dissimilation which must belong to the medium frequency band and has the same coefficients in the standard quantification JEPEG to ensure that our message is inserted into the appropriate location i.e. protect it without being lost during compression or other type of attack.

Let the coefficients of the DCT transformed $(8 * 8)$ of the image carrier with the message. In the algorithm dissimilation we chose arbitrarily the two boxes and insert binary message because the last sound within the medium frequency band and presents same coefficients in the standard quantification JPEG.

3.1.1 Extraction of the message

The algorithm of extraction of the message apply a DCT $(8 * 8)$ in the image watermarked then conducted a test on the two boxes and arbitrarily selected in the algorithm to remove dissimilation binary message.

The image inserted is shown in Figure 8.

Figure 8: Inserted image

3.2 Decryption

To get the initial image we must decrypt it with the inverse process of encryption using this equation:

$$P_j' = A_j \oplus C_j \qquad (14)$$

To analyze the results obtained, it is important to develop tools to measure the error between decrypted and original image. Among these methods we find histogram analysis, key sensitivity and the degree of correlation between two images.

3.3 Histogram Analysis

A histogram is a graph representing the distribution of a continuous variable. So we can notice clearly the difference between the histogram of Lena before encryption (figure 9) and the histogram of the encrypted image of Lena shown in Figure 10, where pixels are distributed uniformly, which can resist against attacks.

So, we can say that this algorithm is efficient because it provides additional security to the hiding message which makes chaotic cryptography essential for any secure transmission of confidential information.

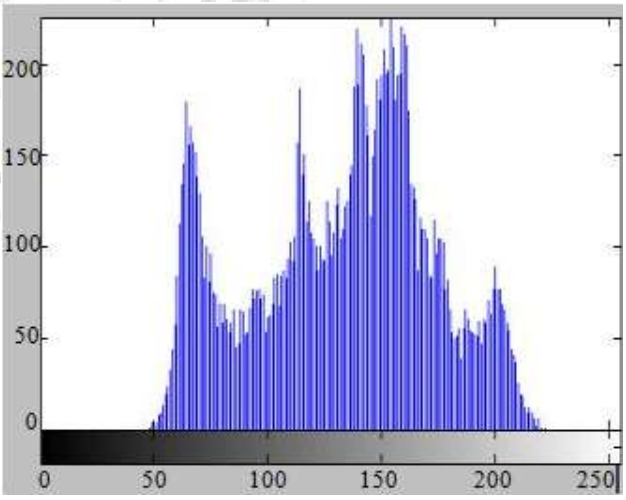

Figure 9: Histogram of original image

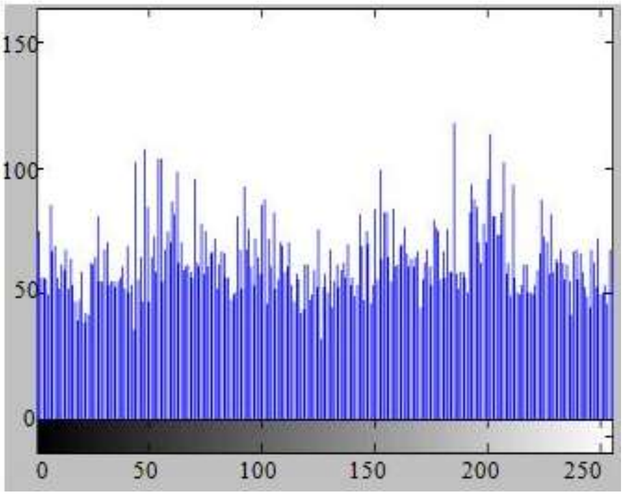

Figure 10: Histogram of encrypted image

Also, we use this technique to prove the friability of our work by comparing the original plain-image and the extracted decrypted one.

As shown in Figure 12 that represents the histogram of the decrypted image there is no difference between the two histograms.

This means that the image is decrypted without any perturbation and deterioration of quality. (figures 11, 12).

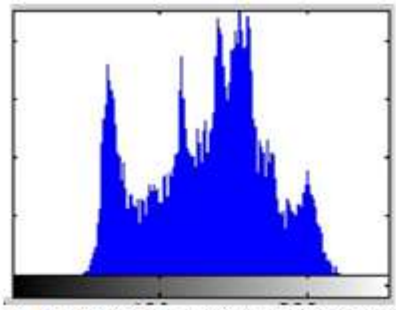

Figure 11: Histogram of the original image

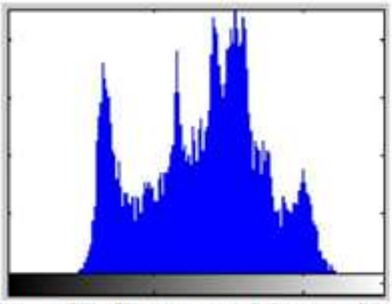

Figure 12: Histogram of decrypted image

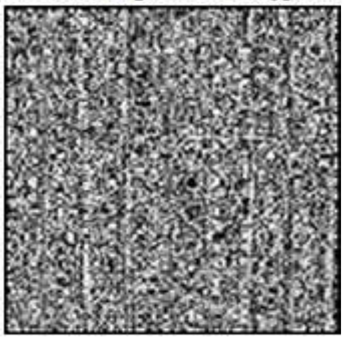

Figure 13: Encrypted Lena μ_0 = 3.9999995 g_0 = 0.778 μ_1= 3.88888888 g_1= 0.666

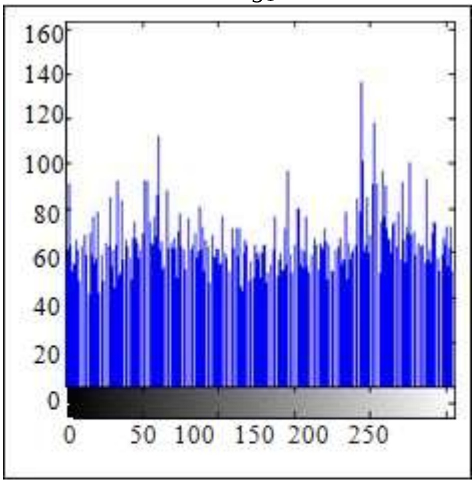

Figure 14: Histogram of Encrypted Lena μ_0 = 3.9999995 g_0 = 0.778 μ_1= 3.88888888 g_1= 0.666

3.4 Key sensitivity

To test key sensitivity, we encrypt the image with the values mentioned above and we decrypt it with different values. The result of this test shows that it is impossible to decrypt the image, if we make a small change to the initial condition "g" or the value "μ" of the logistic map. These results are illustrated in Figures 13,14,15,16.

3.5 The Correlation between two adjacent pixels

For an ordinary image each pixel is highly correlated with its adjacent pixels in the horizontal or vertical direction. An encryption algorithm should produce images ideal encrypted with the correlation between adjacent pixels is negligible. (figures 17, 18) For Lena μ_0 = 3

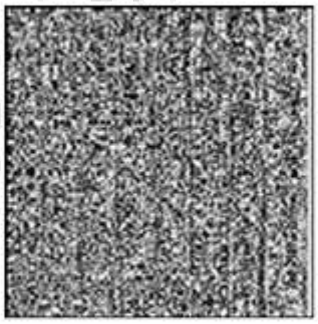

Figure 15: Encrypted 9999996 g_0 = 0.778 μ_1= 3.88888888 g_1 = 0.666

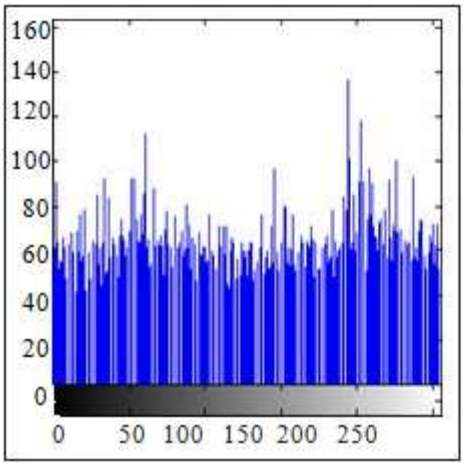

Figure 16. Histogram of Encrypted Lena $\mu_0 = 3.9999996$ $g_0 = 0.778$ $\mu_1 = 3.88888888$ $g_1 = 0.666$

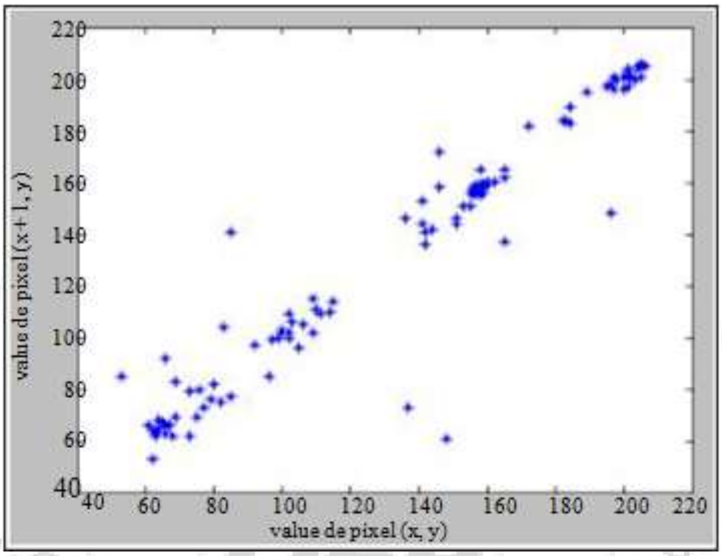

Figure 17: The correlation between two adjacent pixels vertically for the original image

An ordinary image, each pixel is highly correlated with its adjacent pixels in the horizontal or vertical direction. An ideal encryption algorithm should produce images whose numerical correlation between adjacent pixels is negligible.

The following figures show the correlation tests between adjacent pixels horizontally or vertically of the original image and the encrypted one.

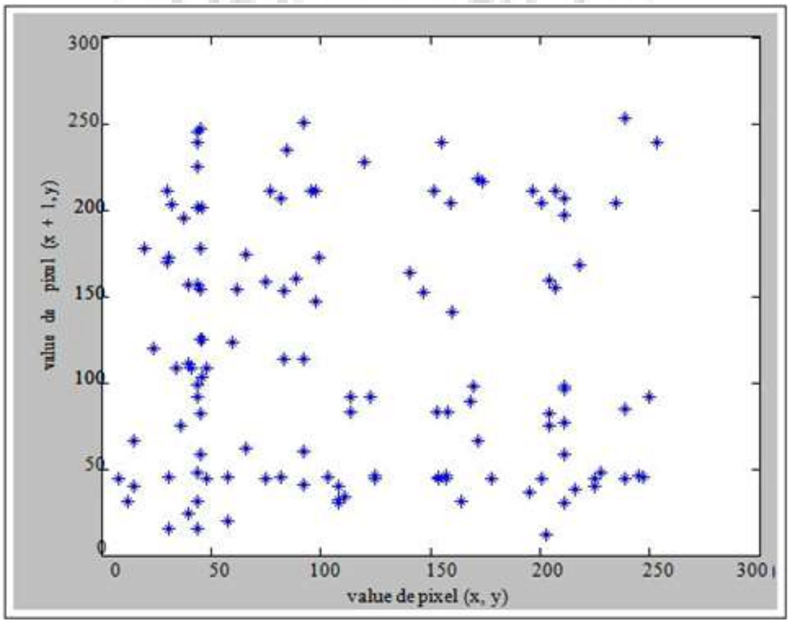

Figure 18: The correlation between two adjacent pixels horizontally for the original image

It is clear that the correlation between adjacent pixels is very small in the encrypted image. This means that the proposed algorithm is efficient because the pixels of the image are distributed uniformly. And for this reason the image will be protected against attacks.

4. Conclusion

This article provides a vision of crypto-watermarking approach. First I improved an encryption algorithm based on chaotic iteration of the Logistics map. Then, an algorithm is developed to watermark information by using DCT method to insert message.

My experimental study showed that the improved encryption algorithm is sensitive to the key and the initial condition which expresses the resistance of the latter.

References

[1] Batista, MS. (1998). Cryptography with chaos. *Phys Lett*.
[2] Yang, T. (2004). A Survery of chaotic secure communication systems. *Int J conput*, 2 (2) 81-130.
[3] Ben Farah, M., Kachouri, A., Samet, M. (2008). Secure chaotic communication system based on active sliding mode synchronization, *In*: Prod GEI, Tunisia.
[4] Xiang, T., Liao, X., Tang, G., Chen, Y., Wong, K. (2006). A novel block cryptosystem based on iterating a chaortic map, *Phys. Lett*.
[5] Wang Y, Liao X, Xiang T, Wong K, Yang D. (2007). Cryptanalysis and improvement on a block cryptosystem based on iteration a chaotic map, *Phys Lett*.
[6] Ben Farah, M., Kachouri, A., Samet, M. (2011). Inprovement of cryptosystem based on iterating chaotic map, *Journal of Non Linear science and numerical simulation*, 16 (6) 2543-2553.
[7] Neil F. Johsen, Stefan, C., Katazenbeisser. (2000). A survey of steganographic techniques, Artech House books, ISBN 1-58053-035-4. Available on amazon .com. jannury.
[8] Tribhuwan, K.T., Vikas, S. (2007). An Improved and Robust DCT based Digital Image Watermarking Scheme. *International Journal of Computer Applications* (0975 – 8887), 3 (1), June.
[9] Zhongwei He, Wei Lu, Wei SunJiwu Huang. Digital image splicing detection based on Markov features in DCT and DWT domain, *Pattern Recognition*, 45 (12) 4292–4299, December.
[10] Zhenxing Qian, Wenwen Wang, Tong Qiao. (2012). An Edge Detection Method in DCT Domain, *Procedia Engineering*, 27, p. 344–348.