*International Journal of Advanced Technology and Engineering Exploration, Vol 8(75)*
*ISSN (Print): 2394-5443   ISSN (Online): 2394-7454*
*http://dx.doi.org/10.19101/IJATEE.2020.762150*

**Research Article**

# The six-dos transposition cipher based on the rubik's cube

**Hana Ali-Pacha[1], Naima Hadj-Said[1], Adda Ali-Pacha[1*], Mohamad Afendee Mohamed[2*] and Mustafa Mamat[2]**

Laboratory of Coding and Security of Information, University of Sciences and Technology of Oran –Mohamed Boudiaf, USTO-MB, PoBox 1505 El M'Naouer Oran 31000 Algeria[1]
Faculty of Informatics and Computing, Universiti Sultan Zainal Abidin, Terengganu, Malaysia[2]

Received: 17-December-2020; Revised: 15-February-2021; Accepted: 18-February-2021

## Abstract

***A transposition cipher is a method of encryption by which the positions held by units of plaintext are shifted according to a regular function so that the ciphertext constitutes a permutation of the plaintext. That is, the order of the units is changed (the plaintext is reordered). Mathematically a bijective function is used on the characters' positions to encrypt and an inverse function to decrypt. Knowing that the vertical and horizontal rings of the Hungarian cube can move left and skilfully in a circular manner, as well as any face of the cube. We have been inspired by this Hungarian cube (Rubik's cube) to realize our image encryption system that is a kind of diffusion cipher or a cipher transposition. We have baptized it: Six-Dos Transposition. The implementation is simple and we have improved the security of the encryption system by eliminating the linearity effect of the coefficient of the adjacent pixels. The length of the encryption key of the cryptosystem that uses Six-Dos Transposition is increased by 63 bits for proposal 1 which is a single Six-Dos transposition to encrypt the main image and, it increased by 129 bits for proposal 3 which uses two Six-Dos transpositions to encrypt the main image. We encrypt each sub-image with the same Six-Dos transposition, and at the end, we encrypt the main image with a special Six-Dos transposition.***

## Keywords

***Cryptography, Rubik's cube, Permutation, Diffusion, Transposition.***

## 1.Introduction

Internet-based communication has become an important part of our daily lives [1]. It expands its usefulness into businesses and commercials such as banking transactions, patients monitoring, and military operations. Data security is at the heart of this commission that supports secure and safe passage of data traversal. Particularly, the storing and transmitting of data has been blessed with the emergence of technology known as encryption or cryptography [2−4] that offers an acceptable level of security.As the sophistication of attacks increase, researches towards improving existing algorithms, or even constructing a new breed of algorithms further takes place to produce more robust algorithms that can withstand all known attacks [5−11]. On what concern us, we proposed in this paper is a diffusion method inspired by the Hungarian cube. The idea is to take the movements of the cube in a projection in a plane, and the plane is the only one representing an image.

The diffusion method is responsible for 'diffusing' each plaintext symbol over the entire ciphertext. A slight modification in the plaintext must result in a significant modification in the ciphertext. This principle is often realized by the permutation function aims to conceal the redundancy by distributing the influence of a bit of key on all the cipher. The Six-back transposition that we have proposed is a permutation function, it realizes the diffusion of all the pixels in such a way that the image is suitably scrambled.

## 2.Literature review

### 2.1Rubik's cube diffusion

The Rubik's Cube (*Figure 1*) is a puzzler invented by Ernő Rubik in 1974, spread rapidly throughout the world in the 1980s.

*Author for correspondence

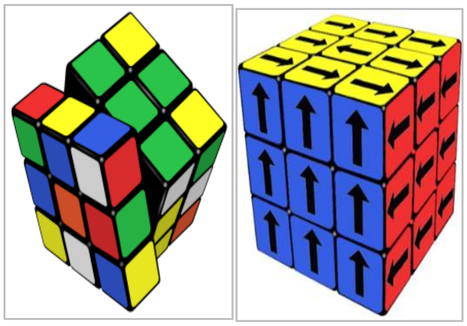

**Figure 1** Cube de rubik

We have been inspired by this Rubik's cube [12] to realize transposition encryption. As seen in the following figure, the vertical and horizontal rings can move left and skilfully circular, as well as any face of the cube.

## 3.Methodology

The idea of Six-Dos Transposition Cipher is to take the movements of the cube in a projection in a plane, and the plane is the only one representing an image.

- A shift of alternate lines: a left shift of b positions for odd lines and a right shift of d positions for even lines, *Figure 2a*.

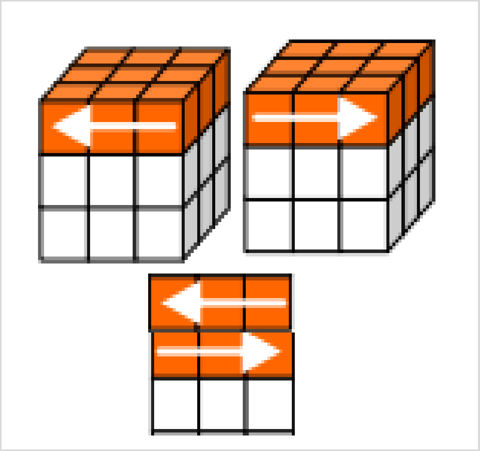

**Figure 2(a)** Shifting alternate lines

- A shift of the alternating columns: a displacement at the top of a position for the odd columns and a displacement at the bottom of c positions for the even columns, *Figure 2b*.

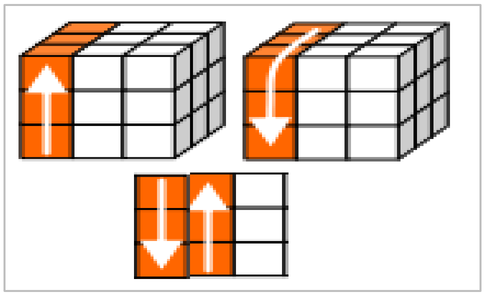

**Figure 2(b)** Shifting alternate columns

- A rotation of the image on itself, otherwise the rows of the matrix become columns, and vice versa, *Figure 2c*.

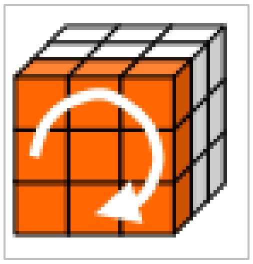

**Figure 2(c)** Rotation of the image on itself

In the rotation of the image on itself as shown in *Figure 2d*, we consider only these four cases (positions). Which can be coded as follows:

- (normal image, was shifted with a rotation of 0° is encoded by rot = 00,
- the image shifted with a rotation of 90° is coded by rot = 01,
- the image shifted with a rotation of 180° is coded by rot = 11,
- and an image shifted with a rotation of 270° is coded by rot = 10.

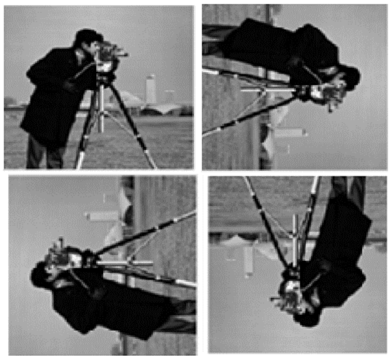

**Figure 2(d)** Rotation of the cameraman on itself

- We have multiplied a, b, c d, by w, as an accelerator to amplify the interference see equations (equation 7), (equation 8), (equation 9), and (equation10).

## 3.1Operating principle

We realized the diffusion of our crypto-system in four steps as follows *Figure 2e*:

1. A rotation of the image on itself
2. Shifting Alternate Lines
3. Shifting Alternate Columns
4. The inverse of the rotation

Or (it depends on the encryption key),

1. A rotation of the image on itself
2. Shifting Alternate Columns
3. Shifting Alternate Lines
4. The inverse of the rotation

| Rot | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| Inv-Rot | 00 | 10 | 11 | 01 |

We note to calculate the inverse of the rotation, you just need to invert the positions of the bits of the rotation.

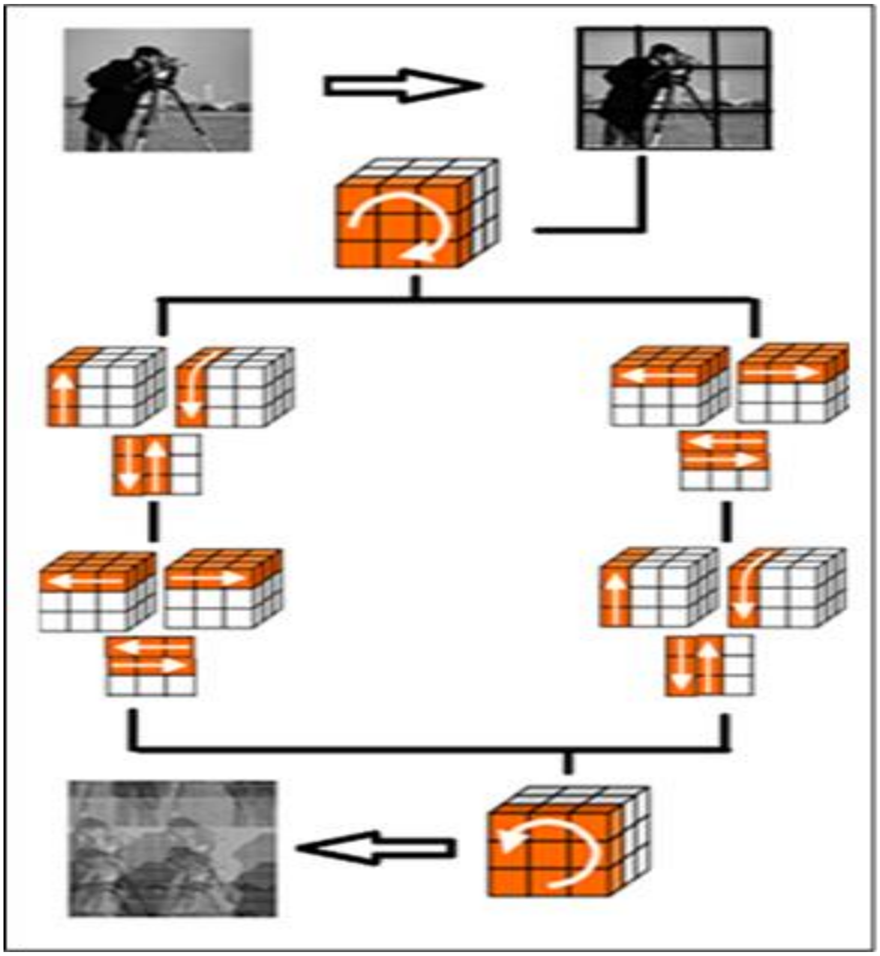

**Figure 2(e)** Block diagram of our approach

## 3.2Evaluating a cryptosystem

The proposed cryptosystem will be verified for some properties for validation purposes. We take into two abbreviations CCPI and CCEI representing the Coefficient of correlation of the original image and the Coefficient of correlation of the encrypted image.

We apply the value (a, b, c, d) = (224, 232, 86, 168) for the images of Lena and Cameraman as in *Figure 3a*, to be encrypted into *Figure 3b*. As for the correlation of the adjacent pixels [13, 14], from the perspective of probability and statistics, the correlation between two random variables represents the strength of the bond that exists between them. The searched link is an affine relationship, and it is the linear regression. Consider calculating the correlation coefficient between two sets: X ($x_1$, …, $x_n$) and Y ($y_1$, …, $y_n$) having the same length. The correlation measurement can be acquired by calculating the linear correlation coefficient of Bravais-Pearson [15] such as the following:

$$Coef(X,Y) = \frac{Cov(X,Y)}{\sqrt{D(X)}\sqrt{D(Y)}} \quad (1)$$

Covariance between x and y is given as follows:

$$Cov(X,Y) = \frac{1}{N}\sum_{i=1}^{N}((X_i - E(X)) * (Y_i - E(Y))) \quad (2)$$

The average of X is:

$$E(X) = \frac{1}{N}\sum_{i=1}^{N} X_i \quad (3)$$

The average of Y is:

$$E(Y) = \frac{1}{N}\sum_{i=1}^{N} Y_i \quad (4)$$

The standard deviation of X is:

$$D(X) = \frac{1}{N}\sum_{i=1}^{N}(X_i - E(X))^2 \quad (5)$$

The standard deviation of Y is:

$$D(Y) = \frac{1}{N}\sum_{i=1}^{N}(Y_i - E(Y))^2 \quad (6)$$

**Figure 3(a)** Original image of Lena and cameraman

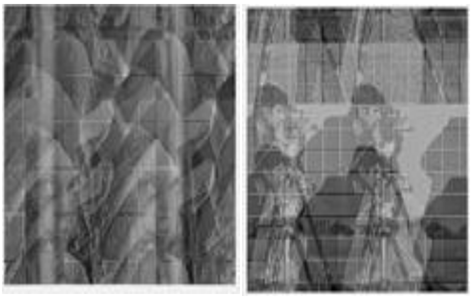

**Figure 3(b)** Encrypted image of Lena and cameraman

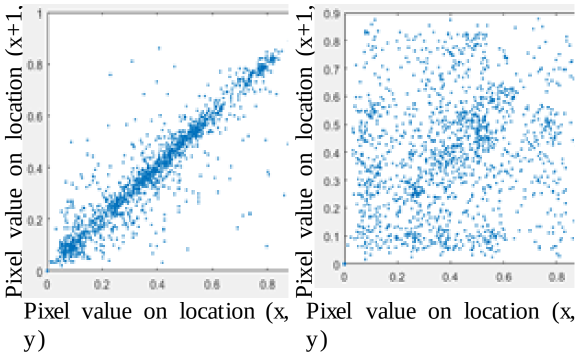

**Figure 4(a)** Spatial representation of horizontally adjacent pixels: Lena image (left) and encrypted image (right)

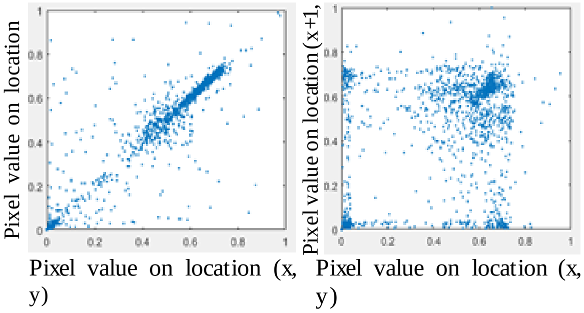

**Figure 4(b)** Spatial representation of horizontally adjacent pixels: cameraman image (left) and her the encrypted image

The correlation coefficient is bounded by -1 and 1. The degree of linear dependence between the two variables can be obtained from the given intermediate values. The closer the coefficient to the boundary (-1 and 1), the correlation between variables is said to be strong, denoted as highly correlated. A correlation equals 0 shows the variables are uncorrelated. To test the correlation coefficient, we selected 2000 pairs of two adjacent pixels (x, y), from both encrypted and original images.

These two *Figures 4a* and *4b* show the correlation factor between two horizontally adjacent pixels from original and encrypted images. It is observable that the neighbouring pixels from Lena image (diffusion with the values (a, b, c, d) = (224, 232, 86, 168) experience a high correlation (coefficient = 0.9843), whereas it encrypted counterpart has low correlation (coefficient = 0.1111). Similarly, the Cameraman image (diffusion with the values (a, b, c, d) = (224, 232, 86, 168) experience a strong correlation (coefficient = 0.9840), while in the encrypted version has low correlation (coefficient = 0.0027).

Encrypted images that come with a low correlation between two neighboring pixels are known to be difficult to be cryptanalysis. Moreover, from the original image, we can have several lines to fit this cloud of points but one that possesses a remarkable property of a line representable by Y = aX + b (a linear correlation).

## 4.Result and discussion

Consider the original image and encrypted image of Lena and the Cameraman. Consider our image as a matrix. A special permutation is performed in three steps concerning the columns and concerning the rows of the matrix as follows:

a. Compared to the columns: For odd columns, move to the right of (a) positions: + a. For even columns, move to the left of (c) positions: -c

b. Compared to the Lines: For odd lines, move to the right of (b) positions: + b. For even paired lines, move to the left of (d) positions:-d.

c. Compared to the rotation of the image on itself: rot=00, rot=01, rot=10, rot=11.

Let the data be (a, b, c, d) = (713, 482, 129, 503). One takes rot = 00.

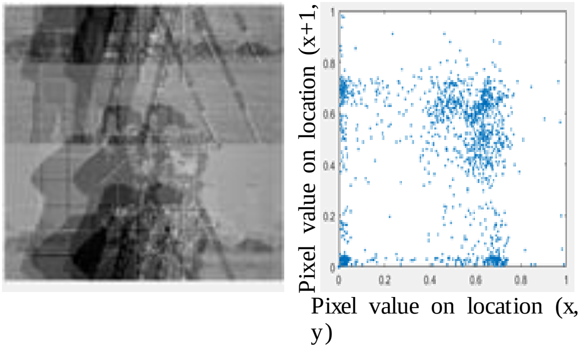

**Figure 5(a)** Cipher image of "cameraman" and its Spatial Representation of Horizontally Adjacent Pixels

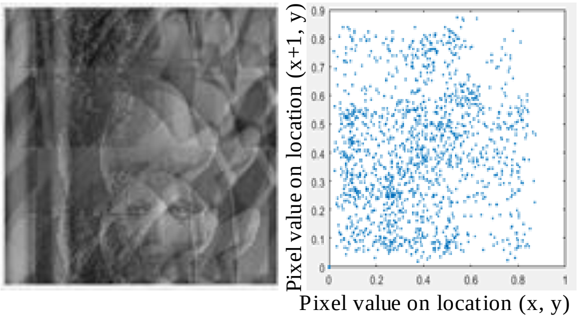

**Figure 5(b)** Cipher image of "Lena" and its Spatial Representation of Horizontally Adjacent Pixels

Based on *Figures 5a* and *5b*, we see from *Table 1* that, the correlation between neighboring pixels in the plaintext and encrypted images are respectively 0.9840 and 0.5129 for the cameraman, and 0.9843 and 0.2294 for Lena. The fact is, the low correlation in the encrypted image hardened the attack against the proposed cryptosystem.

**Table 1** Encryption With (a, b, c, d) = (713, 482, 129, 503)

| Image | CCPI | CCEI |
|---|---|---|
| Cameraman | 0.9840 | 0.5129 |
| Lena | 0.9843 | 0.2294 |

### 4.1Acceleration factor

The diffusion is accelerated with the use of an accelerator factor w. This number w will multiply by all the values a, b, c, and d. In other words, we use the values $a^*$, $b^*$, $c^*$, and $d^*$ as encryption values:

$$a^* = a * w \; mod(256) \qquad (7)$$

$$b^* = b * w \; mod(256) \qquad (8)$$

$$c^* = c * w \; mod(256) \qquad (9)$$

$$d^* = d * w \; mod(256) \qquad (10)$$

We will see the influence of the accelerator factor on encryption.

Let have another example, the following values (a, b, c, d) = (123, 131, 57, 92). For different values of w, we have the following *Figures 6(a)* to *6(m)* represent two images: encryption image (left) and spatial representation of horizontally adjacent pixels (right), the correlation coefficient between horizontally adjacent pixels is given, and in several cases does not reflate the well-known coefficient of Bravais-Pearson in linear dependence.

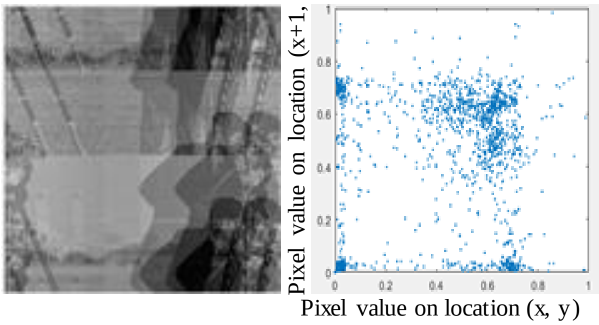

**Figure 6(a)** Cipher image of "Cameraman" and its spatial representation of horizontally adjacent pixels, W=1, CCEI= 0.5173

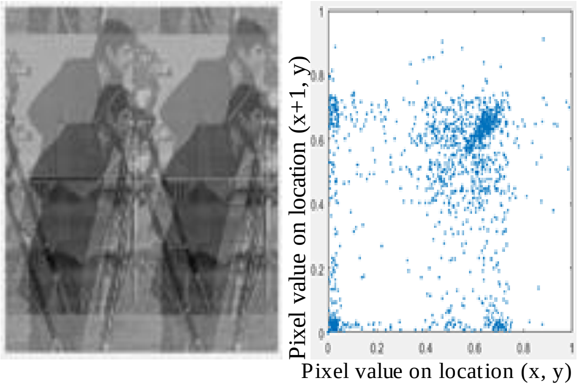

**Figure 6(b)** Cipher image of "Cameraman" and its spatial representation of horizontally adjacent pixels, W=4, CCEI= 0.0062

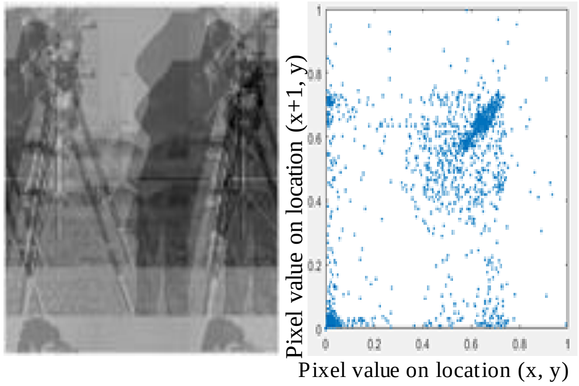

**Figure 6 (c)** Cipher image of "Cameraman" and its spatial representation of horizontally adjacent pixels, W=13, CCEI= 0.1666

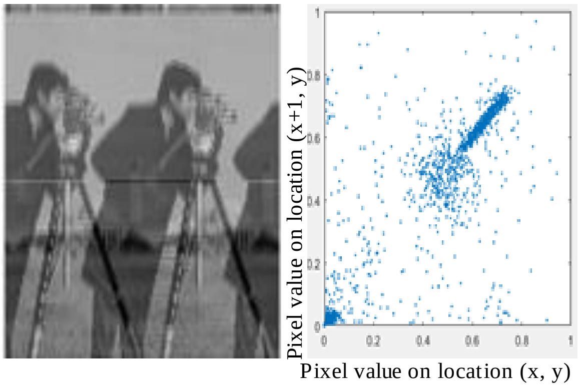

**Figure 6(d)** Cipher image of "Cameraman" and its spatial representation of horizontally adjacent pixels, W=27, CCEI= 0.0277

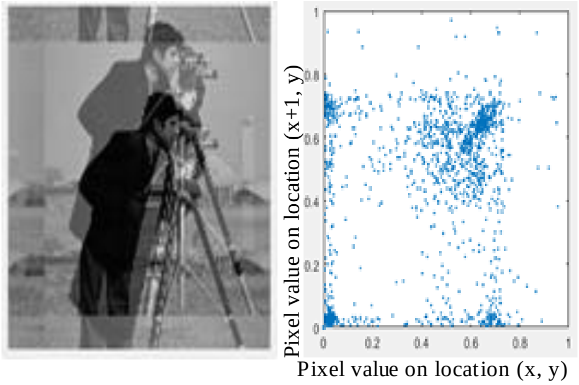

**Figure 6(e)** Cipher image of "Cameraman" and its spatial representation of horizontally adjacent pixels, W=31, CCEI= 0.9068

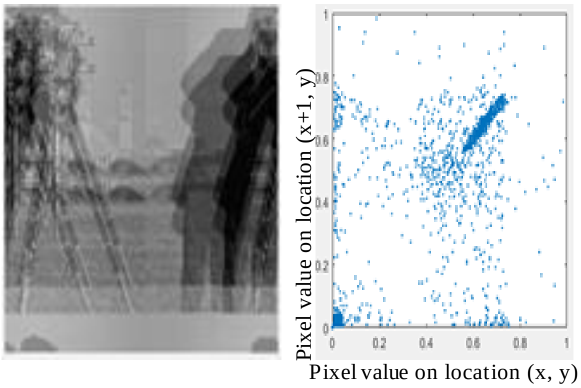

**Figure 6(f)** Cipher image of "Cameraman" and its spatial representation of horizontally adjacent pixels, W=71, CCEI= 0.4968

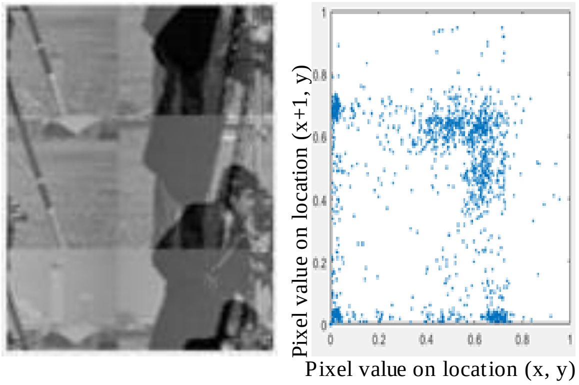

**Figure 6(g)** Cipher image of "Cameraman" and its spatial representation of horizontally adjacent pixels, W=93, CCEI= 0.8526

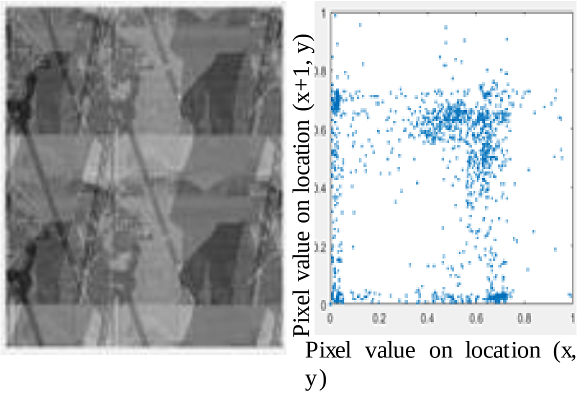

**Figure 6(h)** Cipher image of "Cameraman" and spatial representation of horizontally adjacent pixels, W=96, CCEI= 0.0042

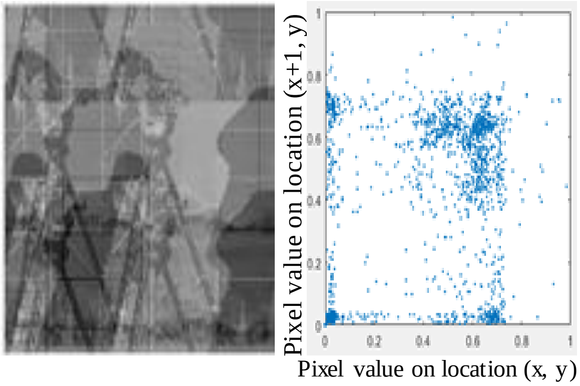

**Figure 6(i)** Cipher image of "Cameraman" and its spatial representation of horizontally adjacent pixels, W=127, CCEI= 0.1423

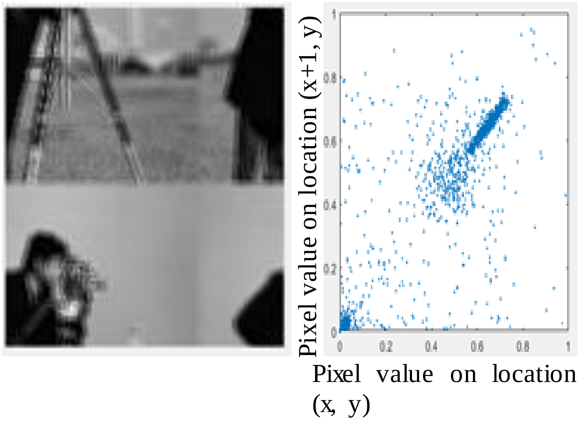

**Figure 6(j)** Cipher image of "Cameraman" and its spatial representation of horizontally adjacent pixels, W=155, CCEI= 0.8353

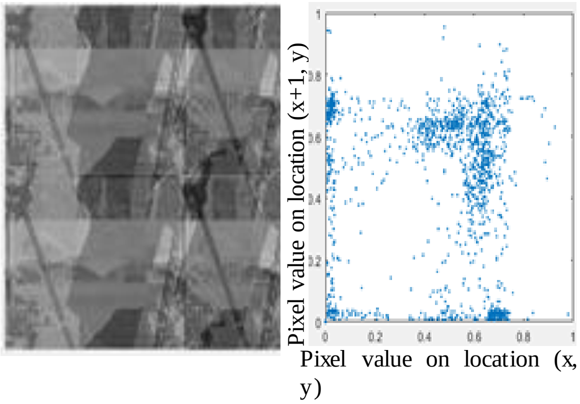

**Figure 6(k)** Cipher image of "Cameraman" and its spatial representation of horizontally adjacent pixels, W=160, CCEI= 0.0053

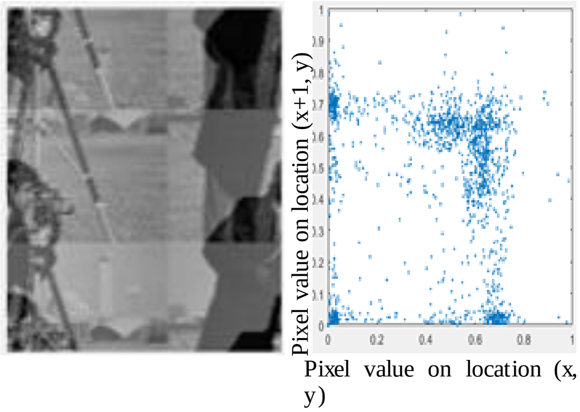

**Figure 6(l)** Cipher image of "Cameraman" and its spatial representation of horizontally adjacent pixels, W=163, CCEI= 0.8522

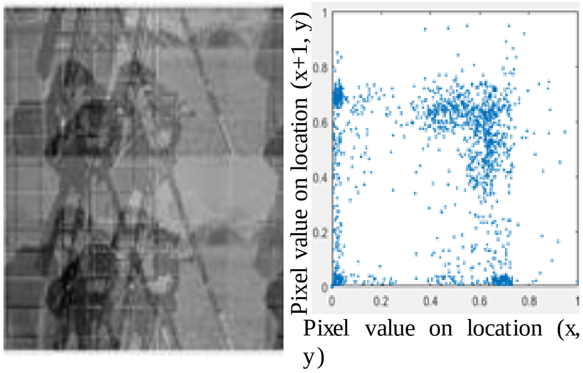

**Figure 6(m)** Cipher image of "Cameraman" and its spatial representation of horizontally adjacent pixels, W=231, CCEI= 0.3497

*Table 2* shows as the influence of the accelerator factor in the correlation coefficient between horizontally adjacent pixels of cameraman image for the values (a, b, c, d) = (123, 131, 57, 92), but in several cases, does not reflate the well-known coefficient of Bravais-Pearson in linear dependence.

**Table 2** Correlation coefficient between horizontally adjacent pixels for different values of w for encrypted cameraman image with (a, b, c, d) = (123, 131, 57, 92)

| W | CCEI | W | CCEI |
|---|---|---|---|
| 0 | 0.9840 | … | … |
| 1 | 0.5713 | 96 | 0.0042 |
| 2 | 0.0964 | 112 | 0.0161 |
| 4 | 0.0062 | 184 | 0.0666 |
| 12 | -0.0517 | 228 | 0.0107 |
| 58 | 0.0167 | 252 | 0.0061 |

The curve of *Figure 7* represents the 256 values of the accelerator factor w (wi = i, such as i = 0 to 255) for the encryption of the image of the cameraman.

The *Figures* in *Figure 8(a)* to *8(g)* represent two images: encryption image (left) and spatial representation of horizontally adjacent pixels (right), the correlation coefficient between horizontally adjacent pixels is given, and in several cases does not reflate the well-known coefficient of Bravais-Pearson in linear dependence.

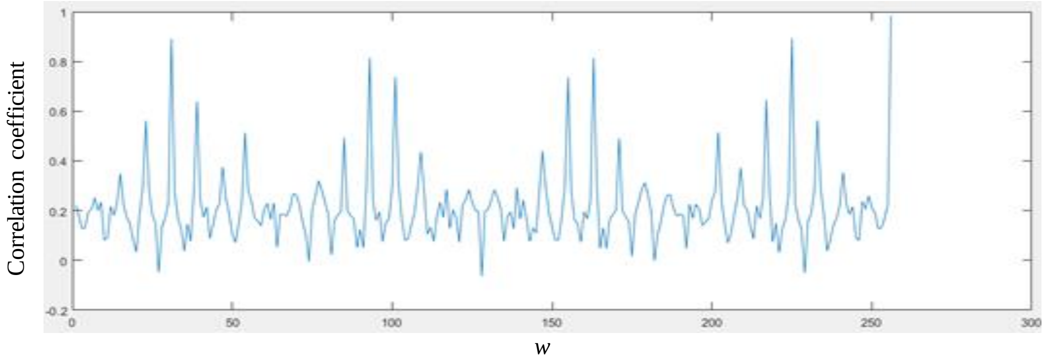

**Figure 7** The correlation coefficient between horizontally adjacent pixels for different values of w for encrypted cameraman image with (a, b, c, d) = (123, 131, 57, 92)

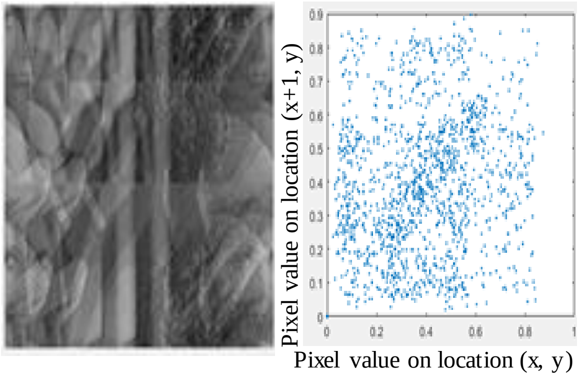

**Figure 8(a)** Cipher image of "Lena" and its spatial representation of horizontally adjacent pixels, W=1, CCEI= 0.2234

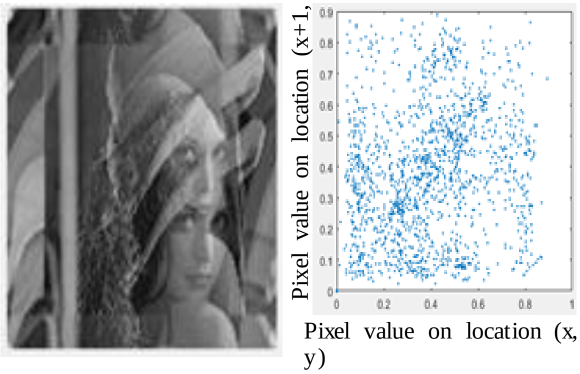

**Figure 8(b)** Cipher image of "Lena" and its spatial representation of horizontally adjacent pixels, W=31, CCEI= 0.8906

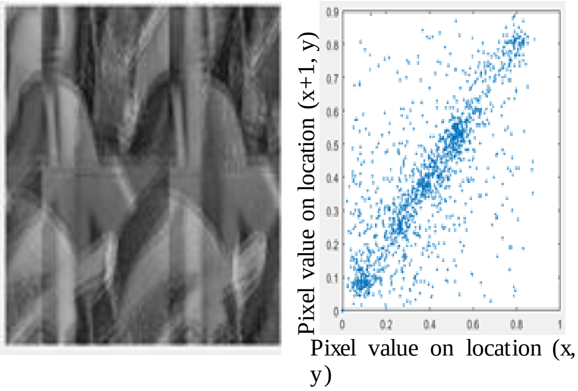

**Figure 8(c)** Cipher image of "Llena" and spatial representation of horizontally adjacent pixels, W=74, CCEI= 0.0044

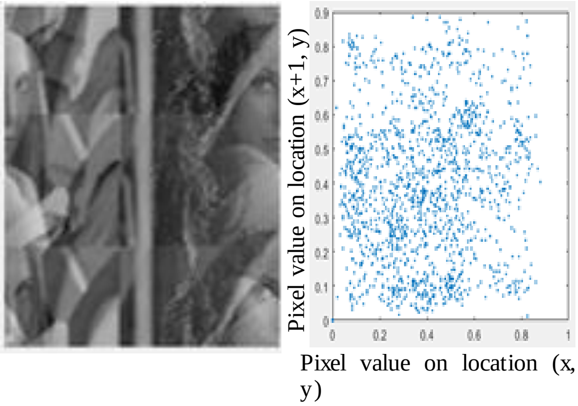

**Figure 8(d)** Cipher image of "Lena" and its Spatial representation of horizontally adjacent pixels, W=93, CCEI= 0.8159

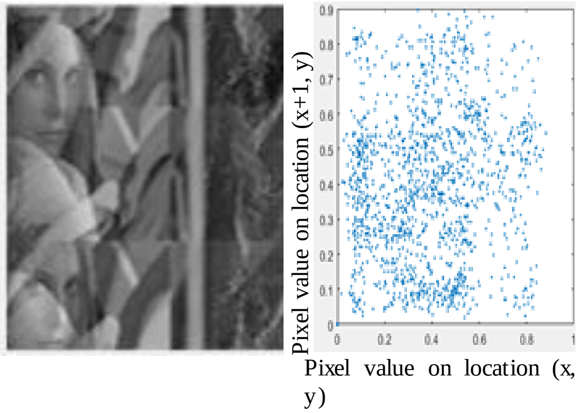

**Figure 8(e)** Cipher image of "Lena" and its Spatial representation of horizontally adjacent pixels, W=163, CCEI= 0.8142

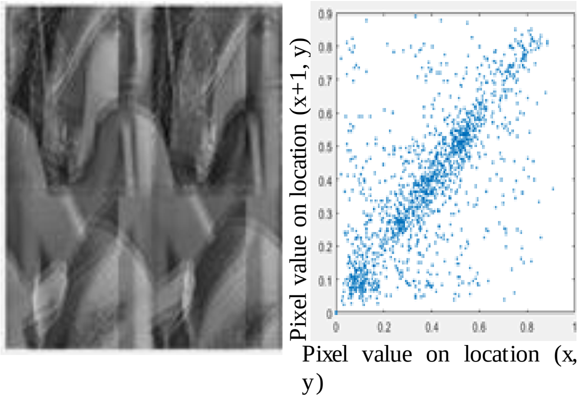

**Figure 8(f)**Cipher image of "Lena" and its Spatial Representation of Horizontally Adjacent Pixels, W=182, CCEI= 0.0018

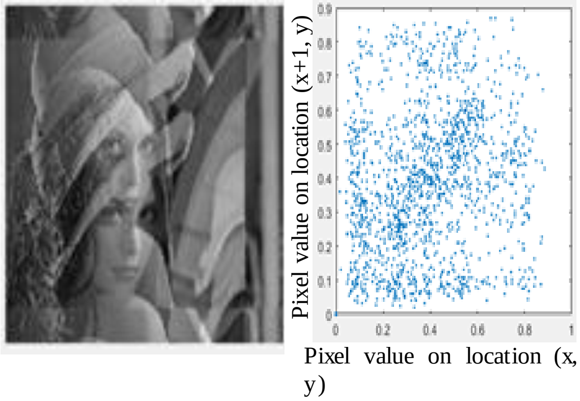

**Figure 8(g)** Cipher image of "Lena" and its Spatial representation of horizontally adjacent pixels, W=225, CCEI= 0.8948

*Table 3* shows the influence of the accelerator factor in the correlation coefficient between horizontally adjacent pixels of cameraman image for the values (a, b, c, d) = (123, 131, 57, 92), but in several cases, does not reflate the well-known coefficient of Bravais-Pearson in linear dependence. Let have the following values (a, b, c, d) = (713, 482, 129, 503). For different values of w, we have:

The curve of *Figure 9* represents the 256 values of the accelerator factor w (wi = i, such as i = 0 to 255) for the encryption of the image of Cameraman.

The *Figures 10(a)* to *10(f)* represent two images: encryption image (left) and spatial representation of horizontally adjacent pixels (right), the correlation coefficient between horizontally adjacent pixels is given, and in several cases does not reflate the well-known coefficient of Bravais-Pearson in linear dependence.

**Table 3** Correlation coefficient between horizontally adjacent pixels for different values of w for encrypted lena image with (a, b, c, d) = (123, 131, 57, 92)

| W | CCEI | W | CCEI |
|---|---|---|---|
| 0 | 0.9843 | … | … |
| 1 | 0.2234 | 165 | 0.0468 |
| 4 | 0.1306 | 175 | 0.0162 |
| 20 | 0.0334 | 182 | -0.0018 |
| 74 | -0.0044 | 246 | 0.0800 |
| 97 | 0.0753 | 251 | 0.1900 |

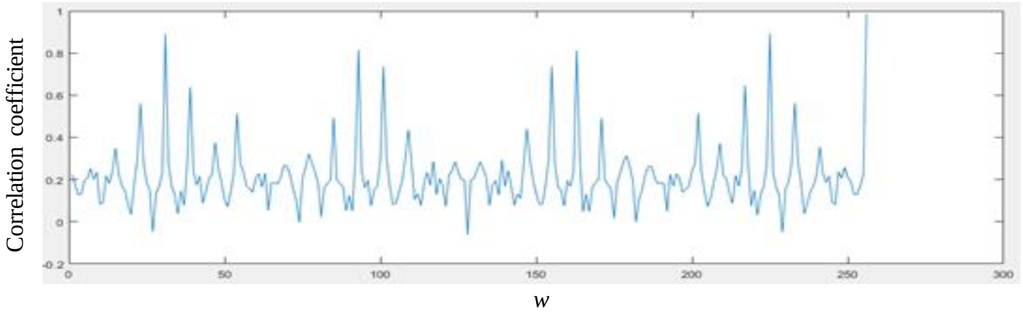

**Figure 9** The correlation coefficient between horizontally adjacent pixels for different values of w for encrypted Lena image with (a, b, c, d) = (123, 131, 57, 92)

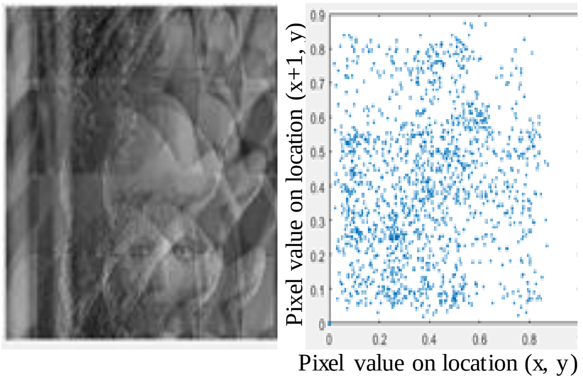

**Figure 10(a)** Cipher image of "Lena" and its spatial representation of horizontally adjacent pixels, W=1, CCEI= 0.2294

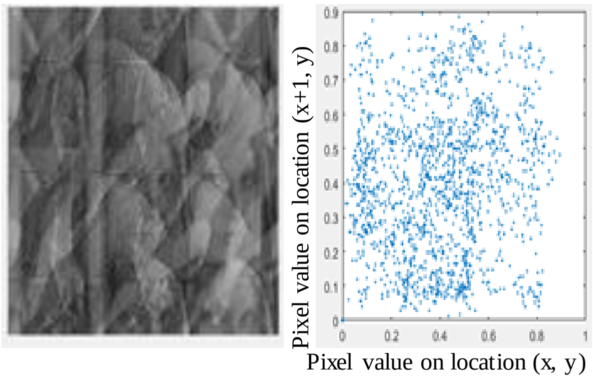

**Figure 10(b)** Cipher image of "Lena" and its spatial representation of horizontally adjacent pixels, W=2, CCEI= 0.1429

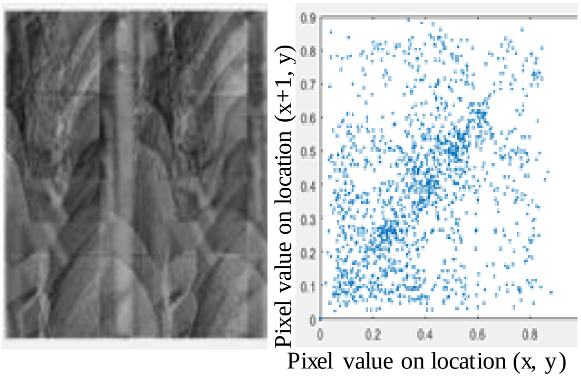

**Figure 10 (c)** Cipher image of "lena" and its spatial representation of horizontally adjacent pixels, W=3, CCEI= 0.0722

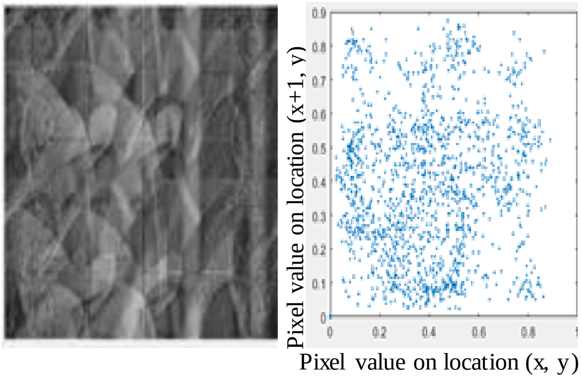

**Figure 10 (d)** Cipher image of "Lena" and its spatial representation of horizontally adjacent pixels, W=5, CCEI= 0.1736

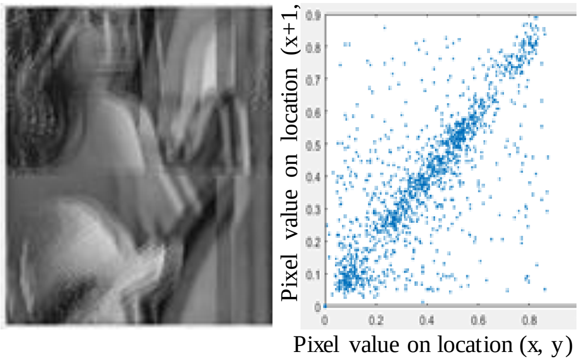

**Figure 10(e)** Cipher image of "Lena" and its spatial representation of horizontally adjacent pixels, W=7, CCEI= 0.2998

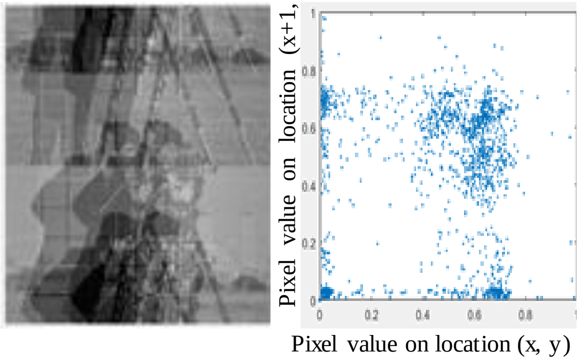

**Figure 10(f)** Cipher image of "Cameraman" and its spatial representation of horizontally adjacent pixels, W=1, CCEI= 0.5129

The curves of *Figure 11* and *Figure 12* represent the 256 values of the accelerator factor w (wi = i, such as i = 0 to 255) respectively for the encryption of the images of Lena and the cameraman.

*Figure 11* for Lena image and *Table 4* and *Figure 12* cameraman image, show the influence of the accelerator factor in the correlation coefficient between horizontally adjacent pixels for the values (a, b, c, d) = (713, 482, 129, 503), but in several cases, does not reflate the well-known coefficient of Bravais-Pearson in linear dependence.

*Figure 13* shows the influence of the accelerator factor in the correlation coefficient between horizontally adjacent pixels for the values (a, b, c, d) = (123, 131, 57, 92), between two encrypted images of Lena and Cameraman.

*Figure 14* shows the influence of the accelerator factor in the correlation coefficient between horizontally adjacent pixels for cipher image of "Lena for two different values the values: and of (a, b, c, d) = (123, 131, 57, 92) and of (a, b, c, d) = (713, 482, 129, 503).

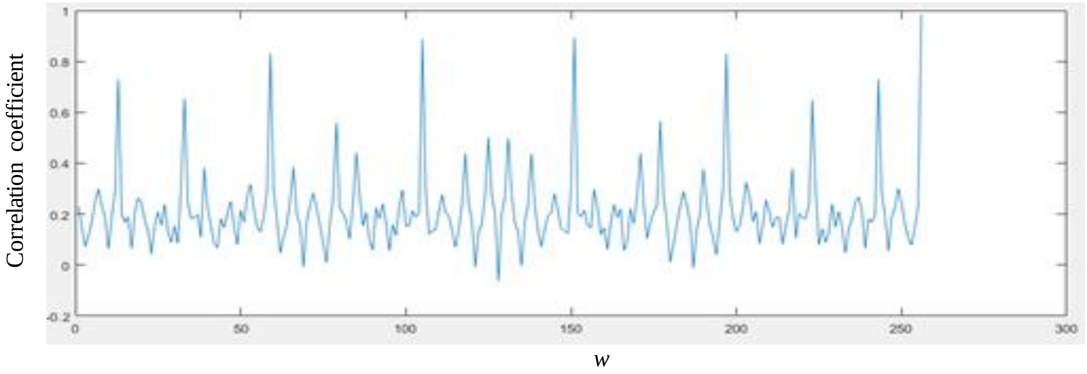

**Figure 11** The correlation coefficient between horizontally adjacent pixels for different values of w for encrypted Lena image with (a, b, c, d) = (713, 482, 129, 503)

**Table 4** Correlation coefficient between horizontally adjacent pixels for different values of W for encrypted cameraman image with (a, b, c, d) = (713, 482, 129, 503)

| W | CCEI | W | CCEI |
|---|---|---|---|
| 0 | 0.9840 | … | … |
| 1 | 0.5129 | 174 | 0.0083 |
| 4 | 0.0307 | 200 | 0.0072 |
| 22 | 0.0091 | 206 | -0.0040 |
| 30 | -0.0014 | 256 | 0.9840 |

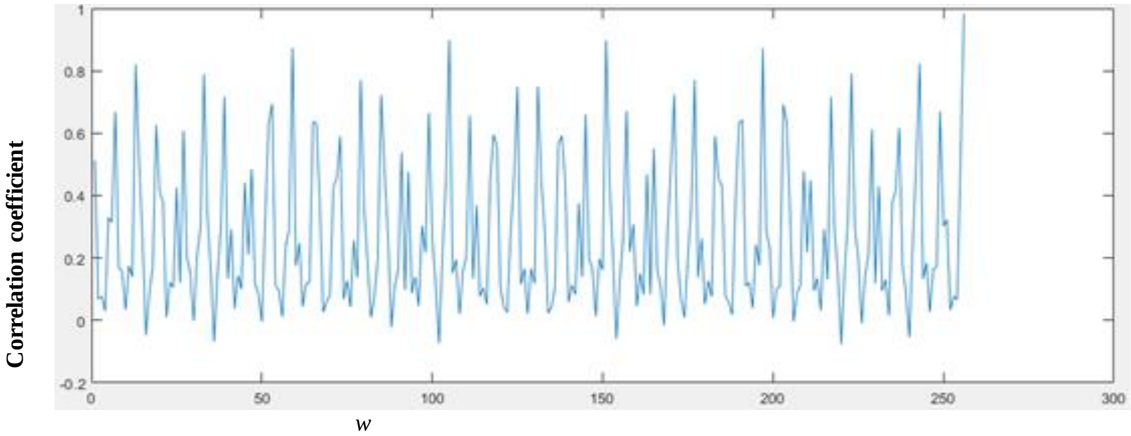

**Figure 12** The correlation coefficient between horizontally adjacent pixels for different values of w for encrypted cameraman image with (a, b, c, d) = (713, 482, 129, 503)

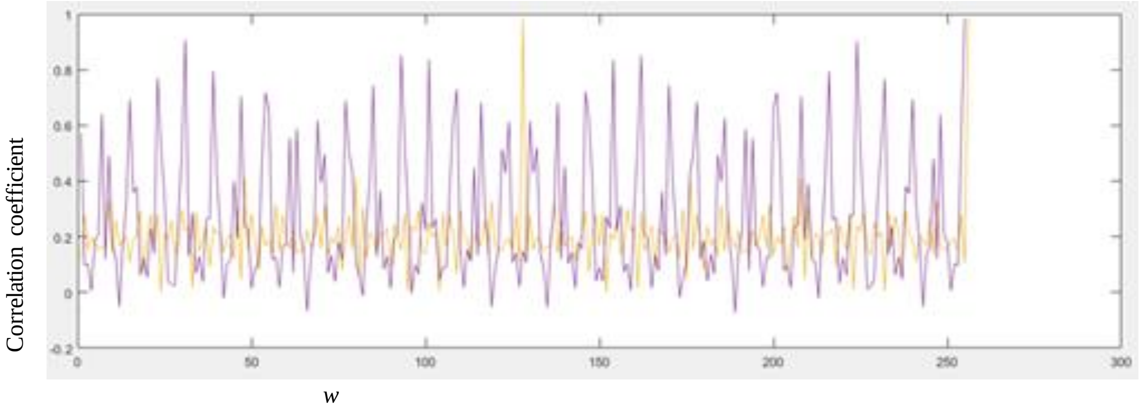

**Figure 13** The correlation coefficient between horizontally adjacent pixels for different values of w of cipher image of Lena and Cameraman with (a, b, c, d) = (123, 131, 57, 92) respectively

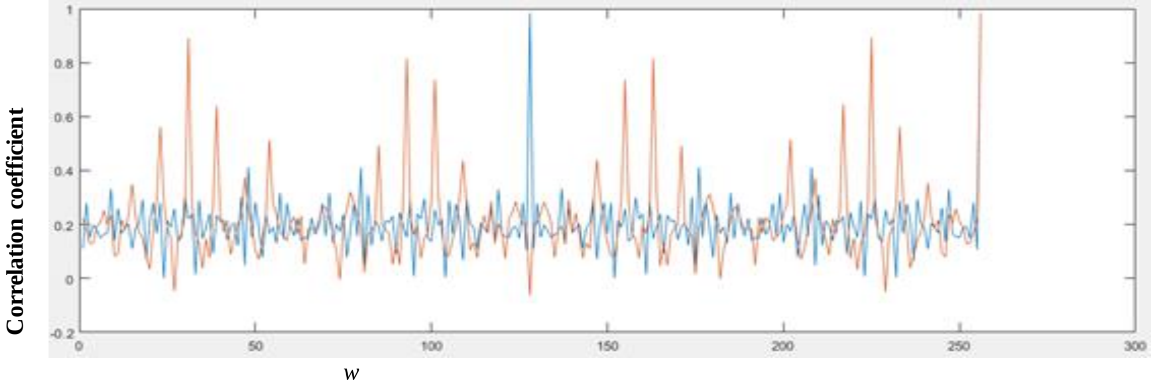

**Figure 14** Correlation coefficient between horizontally adjacent pixels for different values of w for cipher image of Lena and cameraman with (a, b, c, d) = (123, 131, 57, 92) and (a, b, c, d) = (713, 482, 129, 503)

We have obtained that the coefficient of correlation is equal to 0.98 and hence the image is scrambled.

We can note in several cases in our implementation, we have got undesirable linear correlation coefficient values:

A correlation equal to “0”: Figures (*Figure 6(d), Figure 6(c), Figure 6(f), Figure 10(e)*).

A correlation equal near to “1”: Figures (*Figure 6g, Figure 6(j), Figure 6(l), Figure 6(b), Figure 6(d), Figure 6€, Figure 10(f)*).

Then, it is always possible to calculate a correlation coefficient [15], but such a coefficient does not always manage to account for the relationship that exists between the variables studied. Indeed, it assumes that we try to judge the existence of a linear relationship between our variables. It is therefore not suitable for judging correlations that are not linear. It also loses its interest when the studied data are very heterogeneous since it represents an average relationship and that we know that the average does not always have a meaning, especially if the data distribution is multi-modal. If the two variables are independent [15], then their correlation is 0. However, the converse is false because the correlation coefficient only indicates a linear dependence. Other phenomena, for example, can be correlated exponentially, or in the form of power (see two-variable statistical series in elementary mathematics).

### 4.2Improvement of six-dos transposition

We can use several Six-Dos transpositions or just only one. For example (Refer to *Figures. 15(a) to 15(c)*): We have Proposal 1, which is a single Six-Dos transposition to encrypt the image.

P1) Do the Six-Dos transposition for all image, with the values (a, b, c, d, w) = (123, 131, 57, 92, 1).

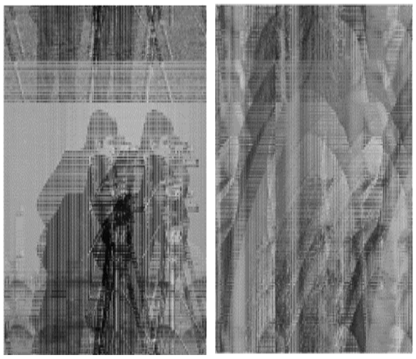

**Figure 15(a)** P1 for cameraman and Lena

An obvious improvement of our Six-Dos transposition is to divide the main image into $4^n$ sub-images, for each sub-image, it has $2^{8-n}$ pixels in row and $2^{8-n}$ pixels in columns.

For n = 1, we will have 4 sub-images, and each sub-image contains only 16324 pixels, 128 pixels in a row, and 128 pixels in columns.

For n = 2, will have 16 sub-images, and each sub-image contains only 4096 pixels, 64 pixels in a row, and 64 pixels in columns.

For n = 3, we will have 64 sub-images, and each sub-image contains only 1024 pixels, thirty-two pixels in a row, and thirty-two pixels in columns.

For n = 4, will have 256 sub-images, and each sub-image contains only 256 pixels, sixteen pixels in row and sixteen pixels in columns.

For n = 5, will have 1024 sub-images, and each sub-image contains only 64 pixels, eight pixels in a row, and eight pixels in columns.

For n = 6, will have 4096 sub-images, and each sub-image contains only 16 pixels, four pixels in row and four in columns, this case is not interesting.

Proposal 2: We can use ($4^n$+1) times Six-Dos transpositions to encrypt the main image by steps. We encrypt each sub-image with a special Six-Dos transposition, and at the end, we encrypt the main image with a special Six-Dos transposition.

Proposal 3: We can use two Six-Dos transpositions to encrypt the main image by steps. We encrypt each sub-image with the same Six-Dos transposition, and at the end, we encrypt the main image with a special Six-Dos transposition.

For example, we divide the image into four sub-images ($4^n$, n=1), make the six-Dos transposition with the same data on the four sub-images separately, with the values (a, b, c, d, w) = (123, 131, 57, 92, 1).

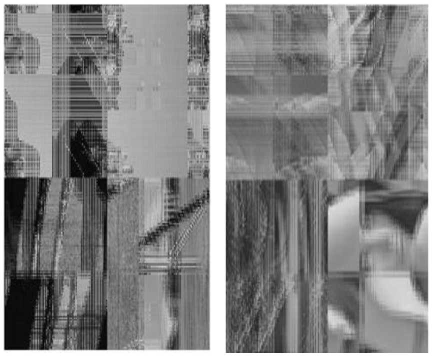

**Figure 15(b)** P2 for cameraman and Lena

For another example, we divide the main image into sixteen sub-images, make the six-Dos transposition with the same data on the sixteen sub-images separately and make the Six-Dos transposition for the resulting image.

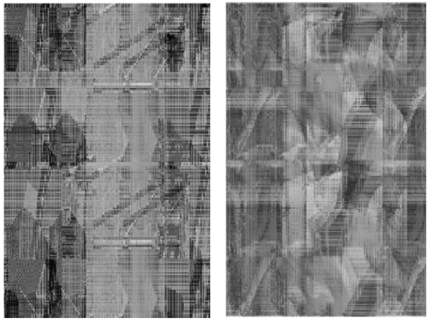

**Figure 15(c)** P3 for cameraman and Lena

The encryption system with two times Six-Dos transpositions works well.

## 4.3Key-space

For this cryptosystem and the simple proposal 1 (P1), the secret key field is made up of 7 fields as follows, with the following parameters:

**Table 5a** Encryption key field for proposal 1

| A | B | C | D | W | R | K |
|---|---|---|---|---|---|---|

1) Consider the values a, b, c, d of the order of $10^4$ ($10^3 \approx 2^{10}$, and $10 \approx 2^3$, hence $10^4 \approx 2^{13}$), 4 * 13 = 52 bits.
2) Scalar W that varies between 1 and 256 →8 bits.
3) R: 2-bit rotation of the image on itself (00, 01, 10, 11).
4) K=C/L: Beginning by the Columns or beginning by the Lines. We adopt: if K=1, we beginning with the columns, if K=0, we beginning with the lines.

We will have a key of (52 + 8 + 2 + 1 =) 63 bits.

In this case, we have a key-space of around $\mathbf{2^{63}}$. The keys-pace is large enough to make a brute force attack impossible.

**For Proposal 2:** If we use the division of the image to be encrypted in power (n) of 4, n = 1, 2, .., 6, we encrypt each sub-image with a special Six-Dos transposition, and at the end, we encrypt all the main image with a special Six-Dos transposition, then we can use the encryption key in order of ; ($4^n$+1) 63 bits which is very huge, this proposal is not practical.

**Table 5b** Encryption key field for proposal 2

| N | Number of Sub-image | Dimension of Sub-image | Key in bits | Key-space |
|---|---|---|---|---|
| 1 | 4 | 128*128 | 315 | $2^{315}$ |
| 2 | 16 | 64*64 | 1071 | $2^{1071}$ |
| 3 | 64 | 32*32 | 4096 | $2^{4095}$ |
| 4 | 256 | 16*16 | 16191 | $2^{16191}$ |
| 5 | 1024 | 8*8 | 64575 | $2^{64575}$ |
| 6 | 4096 | 4*4 | 258111 | $2^{258111}$ |

**For Proposal 3:** For this proposal, we use only two Six-Dos transpositions to encrypt the main image by steps. We encrypt each sub-image with the same Six-Dos transposition, and at the end, we encrypt the main image with a special Six-Dos transposition.

**Table 5c** Encryption key field for proposal 3

| $A_1$ | $B_1$ | $C_1$ | $D_1$ | $W_1$ | $R_1$ | $K_1$ | $A_2$ | $B_2$ | $C_2$ | $D_2$ | $W_2$ | $R_2$ | $K_2$ | N |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|

N can take the values 1, 2, 3, 4, 5, and 6, so it is coded on 3 bits. In this proposal, we have the encryption key in order of (2* 63 +3) bits= 129 bits. For this proposal, we have a key-space of around $\mathbf{2^{129}}$. The key-space is large enough to make a brute force attack impossible. The length of the key is greater than the well-known Advanced Encryption Standard (AES) published in 2001 uses key sizes of 128 bits.

## 5.Conclusion

We have inspired by the Hungarian cube (Rubik's cube) to realize our image encryption system that is a kind of diffusion cipher or a cipher transposition. We have baptized it: Six-Dos Transposition. It works well and, it has broken the concept of linearity correlation between adjacent pixels. We can conclude that our new Six-Dos transposition is secret key encryption and, as 63 bits length key for proposal 1 which is a single Six-Dos transposition to encrypt the main image, the key-space is of the order of $2^{63}$, which is more than to the well-known Data

Encryption Standard (DES) algorithm $2^{56}$. For this proposal three, we use only two Six-Dos transpositions to encrypt the main image. We encrypt each sub-image with the same Six-Dos transposition, and at the end, we encrypt all the main image with a special Six-Dos transposition, in this case, we have 129 bits length key, the key-space is of the order of $2^{129}$, which is more than to the Advanced Encryption Standard (AES) which is $2^{128}$. The length of the encryption key of the cryptosystem that uses Six-Dos Transposition is increased by 63 bits for proposal 1 which is a single Six-Dos transposition to encrypt the main image and, it increased by 129 bits for proposal 3 which uses two Six-Dos transpositions to encrypt the main image. We encrypt each sub-image with the same Six-Dos transposition, and at the end, we encrypt the main image with a special Six-Dos transposition.

## Acknowledgment

The authors would like to express gratitude to the Center for Research Excellence & Incubation Management, Universiti Sultan Zainal Abidin for supporting this research.

## Conflicts of interest

The authors have no conflicts of interest to declare.

## References

[1] B. Schneier. Applied cryptography. In Protocols, Algorithms and Source Code in C, John Wiley & Sons, Inc, New York, Second Edition, 1996.

[2] Nimbhorkar SU, Malik LG. A survey on elliptic curve cryptography (ECC). International Journal of Advanced Studies in Computers, Science and Engineering. 2012; 1(1):1-5.

[3] Ali-Pacha H, Hadj-Said N, Ali-Pacha A, Mamat M, Mohamed MA. An efficient schema of a special permutation inside of each pixel of an image for its encryption. Indonesian Journal of Electrical Engineering and Computer Science (IJEECS). 2018; 11(2):496-503.

[4] Sambas A, Vaidyanathan S, Mamat M, Mohamed MA, Sanjaya WM. A new chaotic system with a pear-shaped equilibrium and its circuit simulation. International Journal of Electrical and Computer Engineering. 2018; 8(6):4951-8.

[5] Wulandari GS, Rismawan W, Saadah S. Differential evolution for the cryptanalysis of transposition cipher. In 3rd international conference on information and communication technology (ICoICT) 2015 (pp. 45-8). IEEE.

[6] Toemeh R, Arumugam S. Breaking transposition cipher with genetic algorithm. Elektronika ir Elektrotechnika. 2007; 79(7):75-8.

[7] Lee RB, Shi ZJ, Yin YL, Rivest RL, Robshaw MJ. On permutation operations in cipher design. In international conference on information technology: coding and computing, 2004. Proceedings. ITCC 2004(pp. 569-77). IEEE.

[8] Yousif IA. Proposed a permutation and substitution methods of serpent block cipher. Ibn AL-Haitham Journal For Pure and Applied Science. 2019; 32(2):131-44.

[9] Khan M, Rasheed A. Permutation-based special linear transforms with application in quantum image encryption algorithm. Quantum Information Processing. 2019; 18(10):1-21.

[10] Wang X, Zhao H. Fast image encryption algorithm based on parallel permutation-and-diffusion strategy. Multimedia Tools and Applications. 2020:1-20.

[11] Janardhanan SV, Sanjeeva P. Bogdanov map-based permuted double image encryption. Anais Da Academia Brasileira de Ciências. 2020; 92(2):1-14.

[12] Davies JW, Morris AO. The schur multiplier of the generalized symmetric group. Journal of the London Mathematical Society. 1974; 2(4):615-20.

[13] Chen G, Mao Y, Chui CK. A symmetric image encryption scheme based on 3D chaotic cat maps. Chaos, Solitons & Fractals. 2004; 21(3):749-61.

[14] Vaidyanathan S, Sambas A, Mohamed MA, Mamat M, Sanjaya WM. A new hyperchaotic hyperjerk system with three nonlinear terms, its synchronization and circuit simulation. International Journal of Engineering and Technology. 2018; 7(3):1585-92.

[15] Joyner D. Adventures in group theory: Rubik's Cube, Merlin's machine, and other mathematical toys. JHU Press; 2008.

**Hana ALI-Pacha,** is a PhD student in Telecommunication option Cryptography and Data Security, University of Sciences and Oran (U.S.T.O) Algeria. He received the diploma of Master II of Cryptography and Data Security in 2016. His research interests are in the cryptography, wireless networks and systems security.
Email: hana.alipacha@univ-usto.dz

**Hadj-Said Naima** was born in Algeria. She received the engineering degree in telecommunications from the Institute of Telecommunication of Oran - Algeria (ITO) in 1986, and the magister degree from ITO in (1992) and a PhD from the University of Sciences and Technology of Oran- Algeria (USTO) in 2005. Now, she is a Professor (teacher/researcher) at the computer sciences Department of University of Sciences and Technology of Oran (USTO). Her interest research are in the area of Digital Communications, and cryptography, his research interests are coding, cryptography and security.
Email: naima.hadjsaid@univ-usto.dz

**Adda ALI-Pacha** was born in Algeria. He received the engineering degree in telecommunications from the Institute of Telecommunication of Oran - Algeria (ITO) in 1986; also, he got university degrees in mathematics in 1986 and a magister in signal processing in November 1993. Later he obtained a Ph.D. in safety data in 2004. He worked in the telecommunications administration (PTT Oran) in the position of the head of telephone traffic for two years (1986 -1988), since 1989, he is at the University of Sciences and Oran (U.S.T.O) Algeria, Technology of as a Professor (teacher/researcher) in the Electronics Institute. The Telecommunication domains are his favorite interest fields' research, his research interests are coding, cryptography and security, FPGA.
Email: a.alipacha@gmail.com

**Mohamad Afendee Mohamed** received his PhD in Mathematical Cryptography and currently serves as a lecturer at the Universiti Sultan Zainal Abidin. His research interests include both theoretical and application issues within the domains of Information Security, and Mobile and Wireless Networking.
Email: mafendee@unisza.edu.my

**Mustafa Mamat** is currently a Professor of Computational and Applied Mathematics at Universiti Sultan Zainal Abidin (UniSZA), Malaysia since 2013. He obtained his Ph.D from UMT in 2007 specialization in optimization field. To date, he has successfully supervised more than 70 postgraduate students and published more than 260 research papers in various international journals and conferences. His research interest includes unconstrained optimization such as hybrid conjugate gradient methods, three term methods, Quasi-Newton methods and chaotic systems. Currently, he is an Editor in Chief for Malaysian Journal of Computing and Applied Mathematics (a UniSZA journal in applied science) and an editor for Indonesian Journal of Science and Technology. Prof Dr. Mustafa is also a Visiting Professor at Universiti Tun Hussien Onn Malaysia (2014-2019), Visiting Professor at Universitas Kanjuruhan, Malang Indonesia (April 2016 until March 2018), Visiting Professor at Universitas Muhammadiyah Ponorogo, Indonesia (August 2016 until July 2018).
Email: must@unisza.edu.my