ISSN 2395-8618

Cipher Image Damage: An Application of Filters

Rolando Flores-Carapia, Víctor Manuel Silva-García, Benjamín Luna-Benoso and Carlos Rentería-Márquez

***Abstract*—In this paper, color images are encrypted and subsequently damage occlusion is made to the encrypted figures, with different sizes; the intention is to simulate an attack. In this research, two aspects are discussed, namely: the first is to encrypt images with quality; that is, the figures encrypted pass randomness tests proposed in this paper. The second aspect deals with the problem of recovering the encrypted figure information when it has been damaged. To retrieve information from encrypted images, the enryption of images is carried out in two steps: in the first a permutation is applied to the entire image and the second uses the AES cryptosystem with variable permutations. To perform this task an algorithm is used that utilizes the pi number to generate the permutations. To improve the sharpness of the deciphered figures with damage two filters are applied; median and average. To measure the degree of improvement in the damaged images two tests are proposed; the first is the correlation coefficient between adjacent pixels in the horizontal, vertical and diagonal directions. The second is based on the information entropy.**

***Index Terms*—AES with variable permutations, goodness-of-fit test, fourier transform, correlation, entropy of information, median and mean filters.**

I. Introduction

THE purpose of this paper is to analyze the attacks problem on encrypted file images. Such attacks may be in communication or storage, and are important in the context of real-time decisions [1]. That is, when an encrypted message is damaged – by attack or not – and then is decrypted the risk is not knowing the original message and sometimes there is not much time to decide what was the original file, say, less time than to ask for it again. So, the first point is to realize an image encryption with quality. In this sense, there are different image encryption works: there are some recent methods using the Hilbert transform [2], other Chaos [3], [4] and Hyper-chaos [5] or even AES cryptosystem [6] with CBC mode encryption [7], although in this latter case the encryption process is sequential. In the Hilbert transform and Chaos cases, there is the difficulty of not knowing specifically, the size of the keys set. In the case of Hyper-chaos, the keys set is 2^{167} and only the brute force attack is mentioned. The chosen-plaintext attacks as linear [8] or differential [9] are not addressed. In addition, there is a research of color image encryption [10], where the set size of the keys is not specified. Also, there are investigations in optics [11], [12], [13] where the number of keys is also not noted. There is an interesting research in this field [14], however, the original image is modified when the figure is decrypted. On the other hand, there is an important encryption research of color images [15]. This work does not use as proof of the encryption quality any of the proposals in NIST Special Publication 800-22.

As for encrypted damage figures [16], the proposal is to use filters [17] in order to improve the sharpness of images with decrypted damage. This article only uses median and mean filters [17]. It is left for further research to propose other filters using mathematical morphology [18]. Then, for damage encrypted figures two actions that solve this type of problem are proposed, namely: as a first step encrypting images must be such that when these have a failure, it does not appear focused on the decoded figure, see figure 3. Therefore, applying an initial permutation over all image pixels, randomly constructed is recommended. The intention is to spread the figure data, so that the pixels remain randomly distributed. In this sense, when an encrypted figure suffers damage, it does not appear focused on the deciphered image. As a second step, AES with variable permutations is applied over the permutated image. It is shown later that the images encrypted by this method pass the tests of randomness proposed, which are: horizontal, vertical and diagonal correlation [19]; Discrete Fourier Transform (DTF) [20]; entropy; and the proposed “Goodness-of-fit test” [19]. In addition, a sensitivity analysis for the proposed k key is performed; that is, the study of the Correlation coefficient between the figure encrypted with k and $k + 1$, in order to show that no relationship between them is carried out. A filter is used in the figure decrypted with damage, to improve sharpness. Two types of filters are proposed. The first is the median with different sizes of mask and the second is the average. In addition, two instruments to measure the improvement after applying the filters are used. These instruments to measure the quality of the result are the following: the first is the Correlation coefficient between adjacent pixels in the directions: horizontal, vertical and diagonal. The second corresponds to the information Entropy. Surely there are several filters that could do a better job than suggested here. But, this is not the objective of this research which shows the improvement in the decrypted figures when some of the aforementioned filters are applied.

Damage to the encrypted images is carried out by concentric rectangles, see Figure 6, and the fault sizes are 35%, 40% and 45% of the total figure area. Of course, the dimensions of these faults may be higher or lower depending on the “quality” required to decrypt images with damage. However, in general it can be said that for the bigger the faults, the

Manuscript received on May 11, 2015, accepted for publication on July 22, 2015, published on October 15, 2015.

R. Flores-Carapia is with the Department of Security, CIDETEC, Instituto Politécnico Nacional, D.F., 07700, Mexico (e-mail: rfloresca@ipn.mx).

V. M. Silva-García, B. Luna-Benoso, and C. Rentería-Márquez are with the Instituto Politécnico Nacional, Mexico.

http://dx.doi.org/10.17562/PB-52-7 • pp. 67–78

ISSN 2395-8618

TABLE I
A 3×3 MASK

$(x-1, y-1)$	$(x-1, y)$	$(x-1, y+1)$
$(x, y-1)$	(x, y)	$(x, y+1)$
$(x+1, y-1)$	$(x+1, y)$	$(x+1, y+1)$

lower the sharpness of the deciphered image. Furthermore, the damage or noise utilized in this paper is the occlusion noise; other noises are not discussed, for example, additive or multiplicative [21], [22].

It is important to clarify that this article does not use the compression process in the encryption image, because there are some countries whose safety areas do not allow the compression process in encryption images [23]. In other words, the process employed is Encryption → Decryption. The images encrypted in this work appear in many other investigations. These figures are: Peppers, Baboon, Barbara and Lena. A fifth figure is proposed according to the criteria described below.

II. PRELIMINARIES

The discussion begins with the filters, and as mentioned above, only two of them are discussed, namely, the median and the mean or average. It starts with the median filter which is a statistical non-linear type [17]. Two types of masks are used, which are 3×3 and 5×5 elements. In general, it can be said that the filters applied in this research conducted a manipulation in space $(n \times m)$ of image pixels. In the case of the mean filter, it only used the 3×3 size. In both filters the process is as follows: given any figure pixel (x, y) the analysis is made in the neighbor elements. For example, in the particular case of the 3×3 mask the (x, y) pixel has as adjacent pixels all the (x, y) considered in Table 1.

Regarding the median filter the process is as follows: the gray levels of the additive primary colors are arranged red, green and blue for each of the neighbor cells at (x, y) and including the pixel. Denote each of these values, as $M_{r,i}$, $M_{g,i}$ and $M_{b,i}$ for each of the additive primary colors respectively with $i = 1, \ldots, n$; where n is the total number of cells in the mask. In the case of one color figures, the different gray levels are worked. To illustrate the point, as a case study an array of 3×3 is taken, then, for each of the nine cells $(x-1, y-1)$ $(x-1, y)$ $\ldots$ $(x+1, y+1)$ each of the amounts $M_{r,i}$, $M_{g,i}$ and $M_{b,i}$ with $i = 1, \ldots, 9$ are sorted. After the amounts of cells are arranged and the median is denoted as δ; then, the following must comply: δ is greater or equal than the $\lceil n/2 \rceil - 1$ first elements; i.e 50%, and less than the remaining elements. The size of the arrays to be analyzed in this research are 3×3 and 5×5, since for a larger mask the process is slow, although, the quick sort algorithm was used [24].

The damage of the figure does not necessarily achieve greater sharpness. The average filter has equal probability weights; i.e. the same values. The formula is $\sum_{i=1}^{n} M_{c,i} P(M_{c,i})$, where $M_{c,i}$ is the color value c in the cell i, which could be red, green, or blue, and $P(M_{c,i})$ is the probability or relative weight of $M_{c,i}$ value, which for this particular situation is $1/n$. The $P(M_{c,i})$ can be different amounts in different cells, taking into account that $\sum_{i=1}^{n} P(M_{c,i}) = 1$. This filter is classified as linear [17].

The algorithm used for image encryption is AES [25] for the following reasons: because it is a recent symmetric encryption system and as the international standard at this time, it makes this algorithm the most studied in the world. On the other hand, an efficient method for breaking it has not yet been found [26]. It is also noted that AES is a symmetric algorithm [7], which makes it very fast to encrypt information. There are different encryption protocols with AES [7]. In this work the ECB mode is used for the following two reasons: First, it is important to evaluate how variable permutations work in the encryption figures. Second, this mode allows parallelizing the encryption process and thus reduces time.

However, when the ECB protocol is used to encrypt figures that have low randomness degree in their bits and does not apply the variable permutations process, and also uses the same key for all 128 bit blocks of the image, it may be that the encrypted figure could give us information; i.e, the distribution of the different shades of basic colors follows a certain pattern; see Figure 2. This is the reason why an additional element is used in the algorithm, in this case a different permutation in each 128 bits block. This permutation is applied in the first round after the operation x-or rather than at the entrance of the first round as Triple-DES does [27]. It also shows that the keys set of AES cryptosystem can be up to 2^{256} elements.

The "quality" aspect of a figure encryption has to do with the randomness degree in the distribution of the encrypted image bits. In this sense, several methods have been used to measure the degree of randomness [20], although in this research the following are used: Correlation; horizontal, vertical and diagonal; Entropy; Discrete Fourier Transform and a different form to measure the degree of randomness of the bits of an encrypted image is proposed, using a "Goodness-of-fit test" [19]. Transcendental numbers are utilized, which have the characteristic of not being a solution of any polynomial with the form $a_n x^n + a_{(n-1)} x^{(n-1)} + \ldots + a_0$ with $a_i \in \mathbb{Z}$ [28], but also have the property that the decimal point to the right does not follow any regularity, so they are good candidates for use in generating pseudo-random numbers. The pi number is the transcendent to be employed in this article because it has been well studied [29].

The generating of permutations is dependent on the AES key. This is according to the following procedure: if we denote by m the integer that represents the string of 128 bits of the AES key. Then the product $m \times pi$ is also a transcendental number and from it is possible to get the constants that are used to generate permutations based on the following procedure. Given a non-negative integer $l \geq 2$ it is possible to define the set $N_l = \{n \in \mathbb{N} | 0 \leq n \leq l! - 1\}$ and on the other hand, according to the division algorithm of Euclid [30] for

ISSN 2395-8618

all $n \in N_l$, it can be written uniquely as follows:

$$n = C_0(l-1)! + C_1(l-2)! + \ldots + C_{l-2}(1)! + C_{l-1}(0)!.$$

Then, using modular arithmetic it is possible to get the C_i for $i = 1, \ldots, l-1$ in a pseudo-random way. Later it will be shown, how from the expression (1) and using an algorithm there is a way to obtain the pseudo-random permutations: for both situations; the whole image and for each block of 128 bits.

The Entropy is measured according to the formula $-\sum_{x \in X} P_r(x) log_2 P_r(x)$; in Section 5, a more detailed explanation is given. Regarding the color image, each of the primary additives –red, green and blue– is described by one byte; that is, 256 gray levels are sufficient for each of them. Then, if a particular primary color has uniform distribution; i.e, all points are equally likely, the entropy value is 8 [31]. Although, one case in which the Entropy is 8 can be constructed, and the distribution of values is not random. However, in practice this is not so. Thus, values as close as possible to 8 for each of the primary colors, in an encrypted figure are sought.

A statistical test to evaluate the randomness of a bits sequence is formulated by means of a null hypothesis H_0, which states that the bit string is random versus the alternative hypothesis H_a which indicates that this is not so. To accept or reject the null hypothesis a statistical and threshold to define a rejection region is used; so, if the value of the statistical based on the data yields an amount that is in the rejection zone, this implies that the null hypothesis is rejected, otherwise H_0 is accepted.

In any hypothesis test scheme there are two types of errors, namely: type I error and type II error. Type I error is one that is committed when H_0 is true and it is rejected. Type II error is accepting H_0 when this hypothesis is false. The error that is controlled is the type I because it is considered that H_0 is the more important of the two hypotheses. The amount used in this research for the type I error is $\alpha = 0.01$, although $\alpha = 0.001$ can also be used [19].

The probability distributions that are used in the randomness tests are: Chi-square χ^2, the standard normal distribution and Complementary Error Function $erfc(z) = (\frac{2}{\sqrt{pi}}) \int_z^\infty e^{-u^2} du$ [32].

Correlation between two random variables x, y, is performed using the adjacent pixels of an encrypted image for each primary color additive; red, green and blue. The reasoning is this: if the encryption figure has good "quality"; then it is expected that the Correlation coefficient between adjacent pixels horizontal, vertical and diagonal is a number close to zero.

III. Algorithm to Generate Permutations

Suppose for the moment, that in the expression (1) the constants $C_0, C_1, \ldots, C_{l-2}$ are known and based on them the following algorithm is constructed:

Step 0. An array in ascending order is defined as follows: $X[0] = 0$, $X[1] = 1, \ldots, X[l-1] = l-1$.

Step 1. The condition $C_0 < (l)$ is observed; then, $X[C_0]$ is one element of the array in step 0. So, $X[C_0]$ is removed from the arrangement in step 0, and instead is replaced by $X[l-1]$; that is, the last element. If $X[C_0]$ is the last element, then this is replaced for the penultimate element. Note, only two operations are performed; removal and replacement. That is, the other elements of the array remain unchanged.

Step 2. In the same way as in the previous step, the condition $C_1 < (l-1)$ is fulfilled, so, $X[C_1]$ is an array element of step 1. Thus, following the same logic of step 1, $X[C_1]$ is removed and instead is replaced for the last item. Of course, if $X[C_1]$ is the last element, then, the process is the same as step 1.

Step $l-1$. If this process is repeated at the end the following result will appear: $X[C_{l-2}]$ and $X[C_{l-1}] = k$ with $0 \le k \le l-1$. The number $X[C_{l-1}]$ appears automatically since it is the last; that is, $C_{l-1} = 0$. The array of positive integers $X[C_0], X[C_1], \ldots, X[C_{l-2}]$ is a permutation of the array $0, 1, \ldots, l-1$. This procedure is performed in steps $l-1$. Regarding the complexity to perform this algorithm is $\mathcal{O}(l)$, since in each step removal and replacement of an element is carried out, the others remaining unchanged. In this paper two sizes of permutation are generated, namely: the 128 positions and that which permutes the pixels in the entire image. If the figure is about 500,000 pixels, 960 × 540, the size of the problem to be solved is $\mathcal{O}(500,000)$ for this particular case, which means it can be resolved speedily. Actually, the time spent on the construction of the switch from one image of dimensions 960×540 is less than ten milliseconds using software.

Now, it is relevant to show how the constants C_0, $C_1, \ldots, C_{l-2}$ are obtained. Note that it is not important to know the number n, fortunately, because otherwise the integers for the simplest case have a magnitude of $128! - 1 \approx 10^{215}$, and for the more complex case the permutation is over the whole image size, the numbers could be $500,000! - 1$, which is huge.

In this sense, the quantities $(l-1)!, (l-2)!, \ldots$ are only used as marks; that is, it is not necessary to write them with all their digits. Then, the next question to address is: how to choose the pseudo-random values for the C_i for $i = 1, 2, \ldots, l-1$. As mentioned above, this paper used the pi number as follows:

(1) The symmetric cryptosystem key Advanced Encryption Standard-AES, is a string of zeros and ones, which represents a positive integer. Denote this integer as m; then, this paper proposes to multiply m by pi, such that the product is itself a transcendent number. Particularly, in this investigation the symmetric cryptosystem AES-128 is used, although there is the possibility of employing up to 256 bit keys.

(2) After completing the multiplication $l \times pi$, and after the decimal point to the right, one Byte strings are taken which are denoted as: $b_0, b_1, \ldots, b_{126}$: For each plaintext of 128 bits, it will utilize 127 strings of one Byte. It follows that, the sets

ISSN 2395-8618

number of 127 chains of one Byte corresponds to the blocks number of 128-bits that have the image to be ciphered.

Each $C_i = b_i \bmod (128 - i)$, for $i = 0, 1, \ldots, 126$ is defined. Remember, the constant $C_{127} = 0$ since it is the last.

(3) Once the constants C_i, for $i = 0, 1, \ldots, 126$ were calculated for each block of 128 bits the algorithm described above is applied to get the permutations of 128 positions.

When the whole image has to be permutated, the number of positions to exchange, l, can be $500, 000$ or more elements. In such case, the procedure is similar to $l = 128$. But, there are some differences, namely, the size chains a_i is 24 bits, or 3 Bytes. This, since many current images do not exceed 2^{24} bits in the spatial resolution. Furthermore, the three-Byte blocks also represent integers. So, it is proposed to calculate the constants C_i as follows: $C_i = a_i \bmod l - i$, for $i = 0, 1, \ldots, l - 2$ y C_{l-1} = 0.

Sometimes in the image to be encrypted some bytes are subtracted, according to the following criteria: If $24 \times l \bmod 128 \not\equiv 0$ where l is the number of pixels of the figure, then, a minimum amount of Bytes is subtracted, say n, such that $24(l) - 8(n) \bmod 128 \equiv 0$. It is important to note that $8n < 128$ and the $8n$ bits are not encrypted. Once, the $C_0, C_1 \ldots C_{l-2}$ values are known, the π_l permutation over a l elements array is calculated, according to the procedure described previously.

IV. AES Encryption Algorithm with a Variable Permutation

This section explains how the tool developed in the previous section in the process of image encryption is used. There are two steps to encrypt a figure; the first is to generate a permutation of the size image and later apply it over the whole figure. The second step is to use the AES algorithm with a modification, i.e., to utilize a different permutation for each 128-bits block after the x-or operation in the first round. Permutation on the whole image is intended to disperse the information so, when the encrypted figure is damaged, it does not appear in a focused manner in the decrypted figure. Actually, this is the intention of permuting the entire image. On the other hand, the algorithm described in the previous section defines a Bijective function [33], from the integers set to the permutations set, which is denoted as I_m, thus, if I_m is a Bijective function, it follows that it is a one to one function. This is important, since two different sets of constants C_i have associated two different permutations, which means in a general way the resulting block from the $Input \oplus k_1$ operation is modified in a different manner, where the entrance string in the first round is $Input$, and k_1 is the first from the keys schedule.

Another question that may arise is why after the x-or operation? And why in the first round? Regarding the first question, the reason it is not used at the entrance of the first round as with Triple-DES or Triple DES-96 [34], is because some images have areas of the same color; for example, black or white. In this situation a permutation applied to ones or zeros strings does not make any modifications to them. But, when it is used after the x-or operation, this allows modifying the bit strings. Furthermore, why in the first round? Given that the information is mixed in each round, then any changes made in the first round there is more opportunity to scramble information, and at the end of the encryption process the ones and zeros will appear randomly. It is important to explain whether the multiplication of the integer associated to the AES key with pi, somehow affects the communication between two people. Actually, this is irrelevant in a secure communication scheme, such as Public Key Infraestructure (PKI) [35]. The key can be encrypted and transmitted using an asymmetric encryption cryptosystem, for example, ElGamal [36], RSA [37] or Elliptical curve [38]. The receiver can know the key with its private key and compute $l \times pi$. Later, the receiver can compute the variable permutations and the permutation of the whole image.

V. Randomness Analysis of Encrypted Images

As mentioned at the beginning of this work tests of randomness are carried out to find out what is the "quality" of the encrypted images; that is, what is the randomness degree in the colors of encrypted figures. Clearly, when the images are with different shades the process is the same.

A. Correlation, Entropy and Discrete Fourier Transform tests

Randomness analysis of the following tests is started: Correlation in directions; horizontal, vertical and diagonal; Entropy and Discrete Fourier Transform. Also, as mentioned earlier, the encryption of the images is performed without compression, or more specifically lossless information. In any image encryption it is important that the distribution of its bits should be random, in order to avoid bias that might lead to attacks to discover the key or plaintext.

With respect to the Correlation between adjacent pixels of an encrypted image with "quality," it is expected that there is a Correlation coefficient close to zero between adjacent pixels, that is, the linear relationship between them must be very weak [39]. Adjacent pixels are considered in three directions, namely, horizontal, vertical and diagonal.

The process of computing the Correlation between two random variables; x and y, is carried out as follows:

A pixel of the encrypted image is selected randomly. This pixel has a value for red, green and blue which is denoted as x_r, x_g and x_b. After selecting a random pixel, the next pixel is taken in adjacent directions horizontal, vertical or diagonal as appropriate. Similarly, as in the previous case, the adjacent pixel selected has a value for red, green and blue. These amounts are denoted as follows: y_r, y_g and y_b.

So, suppose M pairs of pixels x, y are chosen randomly. Then, it is possible to calculate the Correlations in the three directions for the three primary colors. The formula for calculating the Correlation coefficient in the horizontal

ISSN 2395-8618

direction and for the red color is as follows:

$$r_{h;x_r,y_r} = \frac{\sum_{i=1}^{M}(x_{h;i,r} - \overline{x}_{h,r})(y_{h;i,r} - \overline{y}_{h,r})}{\sqrt{(\sum_{i=1}^{M}(x_{h;i,r} - \overline{x}_{h,r})^2)(\sum_{i=1}^{M}(y_{h;i,r} - \overline{y}_{h,r})^2)}},$$

where $\overline{x}_{h,r}$ and $\overline{y}_{h,r}$ are

$$\overline{x}_{h,r} = \frac{1}{M}\sum_{i=1}^{M} x_{h;i,r} \text{ and } \overline{y}_{h,r} = \frac{1}{M}\sum_{i=1}^{M} y_{h;i,r}.$$

Clearly, the expressions in the vertical and diagonal directions as well as for the green and blue colors are the same. In the case of a mono-color image, the process is the same as a color figure.

In case of the Entropy, the study of pixels dispersion in the images is performed by separating the primary colors at each pixel. When a single color is required, one Byte is necessary to calculate the Entropy, i.e., 256 gray levels. For color figures three Bytes are necessary, one for each basic color. In this vein, it is said that if the distribution of bits is totally random the Entropy is 8. To measure the randomness in strings of zeros and ones in practical cases the proceeding is as follows: When Entropy is near 8 it understands that the string of zeros and ones is random; otherwise it would mean that it is not.

To calculate the Entropy it is assumed that there is a string of pixels. In this regard, it is possible to separate each pixel in the chain into its basic color. Then, suppose that the bits string in the red color is divided into blocks of 8 bits, that is, one Byte; it follows that it has 256 possible values. Frequencies are recorded in a table of 256 classes according to their order of appearance. Therefore, each class is assigned a frequency f_i for $i = 0, 1, \ldots, 255$, so, an estimation of the probabilities for each of the classes is $P[x_i] = \frac{1}{f_i}$, where f_i is the class frequency x_i, $i = 0, 1, \ldots, 255$. In this vein, Entropy, say for red color, is calculated as follows: $H_c = -\sum_{x_i \in X} P_c(x_i) log_2[P_c(x_i)]$, where X is the set of all classes. In a simple manner, from the last expression it can be seen that for the green and blue colors the formulas are the same.

The Discrete Fourier Transform measures the degree of randomness of zeros and ones string, that is, there is no periodicity –repetitive patterns– one followed by another.

The following items appear in the calculation of the test statistical:

- N_0. It is a theoretical amount expected; $\frac{(0.95)n}{2}$, where n is the chain length.
- N_1. It is the number of values below a threshold h, which in turn depends on the string length n.

$f_j = \sum_{k=1}^{n} x_k e^{\frac{2(pi)j(k-1)i}{n}}$. If n is odd, just the last bit string is suppressed. Clearly, f_j has a real and another complex part. The $\|f_j\|$ module is calculated, which is real; later, it is compared with h. If $\|f_j\| < h$ a one is added to N_1 value. Otherwise N_1 remains with the previous value. With the latter dates the quantity

$$d = \frac{N_1 - N_0}{\sqrt{\frac{n(0.95)(0.05)}{4}}}$$

and the test statistical $P - value$ = $erfc(\frac{d}{\sqrt{2}})$; are calculated with $erfc(\frac{d}{\sqrt{2}})$ = $2(1 - \Phi(d))$. The decision rule is: if $P - value$ is less than 0.01 the null hypothesis is rejected, otherwise it is accepted.

B. Randomness Test Proposed

Due to work with images testing randomness based on how the color bits are arranged in an encrypted figure, so the $\chi^2 = \sum_{i=1}^{k}[\frac{(o_i-e_i)^2}{e_i}]$ statistical is used for each primary color. The o_i is the observed value and e_i is the expected value. Using the χ^2 statistical it is possible to quantify the freedom degree that has the distribution of the different shades of colors; red, green and blue.

In all NIST 800-22 tests, to determine the degree of randomness of the bits in a string, this type of proof does not appear, that is, the tone distribution randomness of the basic colors is not measured, so, it is a different situation. In the same way as the tests of 800-22 NIST standard, the proposal proof uses the Goodness-of-fit test, utilizing the statistical, χ^2, which has a probability distribution of Chi-square with $n - 1$ degrees of freedom. The freedom degrees are obtained in the following way: the shades of each color of an image can be represented as a histogram whose abscissa has 256 divisions. Then, the degrees of freedom are 255 [19]. On the other hand, if it is considered that the random variable χ^2 approaches the normal distribution according to the center limit theorem, it follows that the mean and variance of χ^2 statistical are: μ = 255 and $\sigma = \sqrt{2(255)} = 22.5831$ for each basic color.

With this information it is simple to calculate the threshold for a significance level of $\alpha = 0.01$ and $\alpha = 0.001$, considering that both size of significance gray levels are in the right tail of the normal distribution. So, the threshold for significance level $\alpha = 0.01$ is 307.61 and for $\alpha = 0.001$ is 324.78.

Then, the process of making the decision to accept or reject the null hypothesis, according to particular datas is as follows:

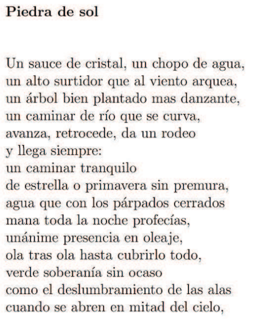
a) The statistical χ^2 = $\sum_{i=1}^{k}[\frac{(o_i-e_i)^2}{e_i}]$ is calculated with specific values, where o_i and e_i are observed and expected values number i.

b) The probability to the right of the value χ^2 is calculated; if this probability is greater or equal to 0.01, then the null hypothesis is accepted, otherwise it is rejected. If the significance level is 0.001 the procedure is similar.

C. Sensitivity Analysis

It is important in image encryption to make the Correlation between two figures encrypted with two very close keys, for example, the difference between these two keys could be one. This Correlation should be close to zero. This means, that no

ISSN 2395-8618



Piedra de sol

Un sauce de cristal, un chopo de agua,
un alto surtidor que al viento arquea,
un árbol bien plantado mas danzante,
un caminar de río que se curva,
avanza, retrocede, da un rodeo
y llega siempre:
un caminar tranquilo
de estrella o primavera sin premura,
agua que con los párpados cerrados
mana toda la noche profecías,
unánime presencia en oleaje,
ola tras ola hasta cubrirlo todo,
verde soberanía sin ocaso
como el deslumbramiento de las alas
cuando se abren en mitad del cielo,

Fig. 1. Type image to be encrypted.

matter the closeness between different keys, the result is that there is no relationship between the two encrypted images.

D. Proposed Image to be Encrypted

In the introduction to this article it was mentioned that a criterion would be presented to choose the figure to be encrypted. This criterion is based on a characteristic of the Goodness-of-fit test that tells us the following: if the tones distribution in each of the three basic colors was totally random $\chi^2 = 0$. In fact, this means that each color histogram is a uniform distribution.

However, when χ^2 has a very large value for each of the primary colors AES with variable permutations should be applied, otherwise the encryption is not efficient, see Figure 2. In addition, there are many relatively small images with χ^2 which can be applied directly to the AES cryptosystem, see Figure 5; i.e. it is not necessary to use a random permutation after the x-or operation in the first round for each 128-bits block of plaintext. Encrypted image passes all the aforementioned tests of randomness and also the proposal. So, in this research, it is proposed to choose an image that has a χ^2 as large as possible for each of the basic colors in order to show that the proposed method is effective for encrypt images.

In this investigation a figure with the following chi-square for the primary colors red, green and blue is used: χ^2_r = 56,638,911.17, $\chi^2_g = 56,555,658.91$ and $\chi^2_b = 55,396,932.18$. This image is of a piece of poetry by Octavio Paz (Mexican poet, 1914–1998) [40]. See Figure 1.

VI. Results Presentation of Encrypted Images

As noted earlier in this article, first the results of the encryption process are shown to verify that the procedure observes the tests of randomness proposals, and also, it is compared with other researches. The images that are ciphered appear in many papers of encryption figures. Such images are five, namely: Peppers, Baboon, Barbara, Lena and the proposed figure. The first four images are presented in Figure 4, and the proposal in Figure 1. The 128-bits key of the AES cryptosystem is written in a hexadecimal system, and it is as follows:

$$k = \texttt{00112233445566778899AABBCCDDEEFF}.$$

In fact, it can be assigned randomly.

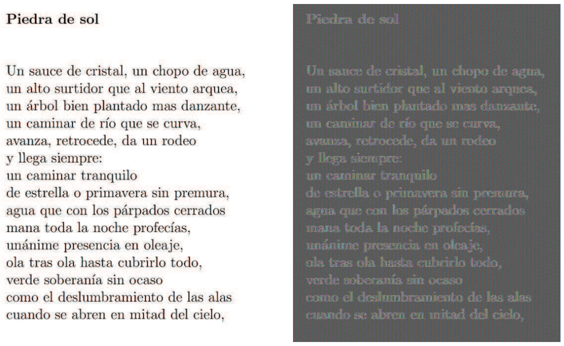
As noted earlier, the k key is associated with a positive integer, which is as follows:

$$l = \texttt{88962710306127702866241727433142015}.$$

Then, multiply the number by pi; that is, the product $l \times pi$ which in turn is a transcendental number. To the right of the decimal point the number of bits needed to cover all the sets of constants used to encrypt the image is taken. The tests: DFT, Goodness-of-fit proposal, Entropy and Correlation coefficient of adjacent pixels in the directions horizontal, vertical and diagonal, also, a sensitivity analysis for the k and $k+1$ key are applied in this section.

A. The Discrete Fourier Transform, Entropy and the Proposed Test

It starts with DFT applied to Figures 4 and 1; later, the test for Goodness-of-fit. In both cases it is encrypted with

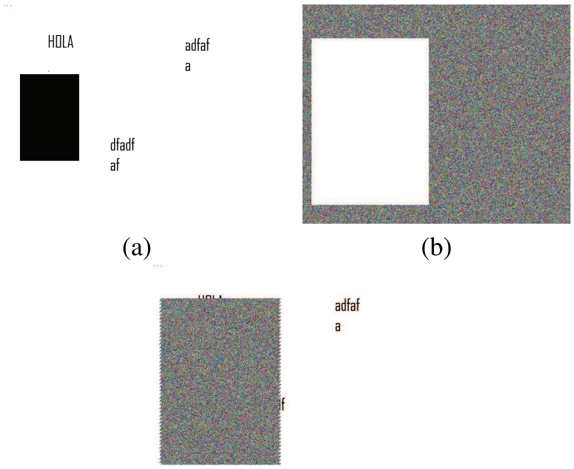


Piedra de sol

Un sauce de cristal, un chopo de agua,
un alto surtidor que al viento arquea,
un árbol bien plantado mas danzante,
un caminar de río que se curva,
avanza, retrocede, da un rodeo
y llega siempre:
un caminar tranquilo
de estrella o primavera sin premura,
agua que con los párpados cerrados
mana toda la noche profecías,
unánime presencia en oleaje,
ola tras ola hasta cubrirlo todo,
verde soberanía sin ocaso
como el deslumbramiento de las alas
cuando se abren en mitad del cielo,

(a) (b)

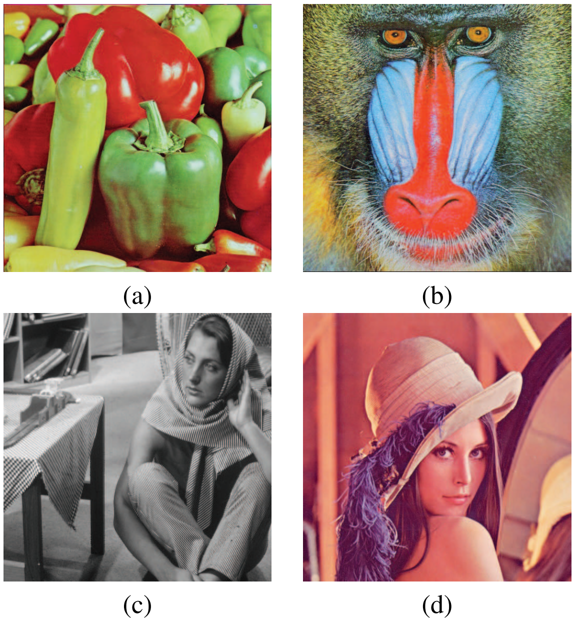
Fig. 2. The original (a) and cipher (b) image without variable permutation.



(a) (b)

(c)

Fig. 3. (a) The original image, (b) the encrypted image with variable permutation and (c) deciphered image with damage.



(a) (b) (c) (d)

Fig. 4. Images: (a) Peppers, (b) Baboon, (c) Barbara and (d) Lena.

the k key. It is determined if the bit strings for each of the colors; red, green and blue approve the randomness criteria for significance level $\alpha = 0.01$.

The test results of the DTF are presented in Table 2 and the Goodness-of-fit test in Table 3.

Regarding the amounts $P - value_r$, $P - value_g$ and $P - value_b$ which are the threshold values for the primary colors; red, green and blue, they appear in Table 3.

Study of the Entropy, for each of the primary colors is conducted separately. As noted in this research earlier, an image is "well encrypted" if the Entropy of each basic color is closer to eight. Table 4 shows the results.

Entropy results presented in Table 4 are better than those shown in other studies [5].

B. Sensitivity Analysis

Concerning the sensitivity, the amounts of the Correlation coefficient between figures encrypted of image (a) Figure 4 are shown, one with the k key and the other with $k + 1$ key. It also clarifies that the way to make this task is by means of a randomly chosen sample as follows: three thousand pairs of pixels (x_i, y_i) are taken, where x_i is a randomly chosen pixel from the encrypted image with k, and y_i is the corresponding pixel in the image encrypted with $k + 1$.

Correlation analysis was performed for the three primary colors; red, green and blue. The results are presented in Table 5.

C. Correlations Between Adjacent Pixels

If an image is "well coded" it is expected that the Correlation coefficient between adjacent pixels in the directions: horizontal, vertical or diagonal, have a Correlation close to 0. This means that there is no linear relationship between adjacent pixels for the same three directions. The Correlations calculation takes a random 3000 pairs sample of pixels and for each basic color in all three directions the analysis is performed.

Images in Figure 4 and Figure 1 are used for this task. The results of the Correlations for the original images are shown in Table 6 and those encrypted in Table 7.

In both tables the values for each primary color are separated. Regarding the notation, the Correlation coefficient is denoted as r, which has two subscripts. The first one indicates the direction: h, v or d and the second is the color: r, g or b. For example, if the diagonal Correlation coefficient is desired for the blue color it is denoted as $r_{d,b}$.

VII. Presentation of Results for Images with Damage

This part carries out the figures analysis with damage, whereas the faults are applied in concentric rectangles; in fact they may be of any other shape, since in the first stage of the encryption process the permutation over the whole image disperses the pixels in a pseudo-random manner. Clearly, when the figure is decrypted with failure the result is less sharp than the original image. This is due to noise which is introduced in the decoded image. The type of noise that is applied in this article is the occlusion; leaving to other research the additive and multiplicative noise.

In this order, two filters are applied, namely: the first is the median with 3×3 and 5×5 masks. The second is the mean, and the objective of both is to reduce the noise; that is, make the decipher images sharper. Also, the object of this work noted above is not to compare filters, but to show that the implementation of some well-known filters can improve the sharpness.

Damage sizes used in this article are shown hereunder: 30%, 40% and 45%. Moreover, the elimination of noise is measured in relation to decoded figures with damage, as the receiver ignores the original information. This research proposes to measure the improvement in sharpness according to the following instruments: the Correlation coefficients in three directions; horizontal, vertical, diagonal and the Entropy.

All these tests are conducted for the three primary colors. Obviously, in the case of images with different gray levels the process is similar. Table 8 shows the results of the decoded image with a fault, and corresponds to image (a) Figure 4. These results for each of the colors are written: red, green, blue and consider different damage sizes: 35%, 40% and 45%.

The mean values are shown in Table 9. These amounts are obtained for the three gray levels of faults mentioned, and for the three basic colors. It is noted that the procedure of encryption and decryption with damage is performed with the image (a) Figure 4, regarding the results of the Correlation coefficient in the three directions: horizontal, vertical, diagonal and for the three primary colors are expressed in Tables 10 and 11. This test applies for the two filters; that is, the median

TABLE II
THE DFT TEST RESULTS APPLIED TO ENCRYPT IMAGES OF FIGURE 1 AND FIGURE 4 WITH k KEY (✓ ACCEPTED, χ REJECTED).

Test name	Significance Label $\alpha = 0.01$	$P - value/Decision$				
		Figure 1	Figure 4			
			(a)	(b)	(c)	(d)
Spectral	Red	0.667/✓	0.308/✓	0.702/✓	0.613/✓	0.769/✓
DTF	Green	0.861/✓	0.023/✓	0.923/✓	0.760/✓	0.183/✓
	Blue	0.504/✓	0.602/✓	0.970/✓	0.531/✓	0.334/✓

TABLE III
THE PROPOSAL TEST RESULTS APPLYING TO ENCRYPT IMAGES OF FIGURE 1 AND FIGURE 4 WITH k KEY (✓ ACCEPTED, χ REJECTED).

Test name	Significance Label $\alpha = 0.01$	$P - value/Decision$				
		Figure 1	Figure 4			
			(a)	(b)	(c)	(d)
Proposal	Red	0.46/✓	0.33/✓	0.46/✓	0.38/✓	0.78/✓
Test	Green	0.04/✓	0.18/✓	0.52/✓	0.48/✓	0.71/✓
	Blue	0.46/✓	0.09/✓	0.21/✓	0.33/✓	0.68/✓

TABLE IV
ENTROPY OF ENCRYPTED IMAGES USING THE k KEY FOR FIGURE 1 AND FIGURE 4.

Entropy	Figure 1	Figure 4			
		(a)	(b)	(c)	(d)
Red	7.99934	7.99929	7.99929	7.99928	7.99934
Green	7.99925	7.99924	7.99930	7.99930	7.99931
Blue	7.99934	7.99921	7.99924	7.99927	7.99932

TABLE V
SENSITIVITY ANALYSIS FOR IMAGE (A) FIGURE 2. USING THE KEYS k KEY AND $k + 1$

Correlation	Results of Sensitivity
	Image (a) figure 2
Red	0.0313
Green	0.0133
Blue	0.0098

and the average. The reasoning here is as follows: Noise reduction is significant as can be seen, when the Correlation as an instrument of measure is used. Also, in general it can be said that for bigger damage, the filters are less efficient, i.e. both measuring instruments may be inaccurate.

In this regard, the size of the fault is important because if the damage is around 80% or 90% it is very difficult with this information to recover the original image.

Figure 6 and Figure 7, where the encrypted images have 40% damage are presented. In Figure 6, the 5×5 median filter for image (a) Figure 4 is applied; and in Figure 7 the average filter for the same image.

VIII. DISCUSSION OF THE RESULTS

The discussion of results is separated into two parts: the first deals with the aspect of image encryption, and the second addresses the problem of encrypted figures with damage.

Regarding the first point, encrypted images pass all the randomness tests proposed in this research. Images were encrypted with the k key, which can be chosen at random. Furthermore, the results of Entropy are better than in other studies.

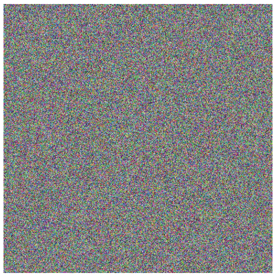


Fig. 5. Image (a) Figure 4 encrypted, using the same key k for each 128 bits block

Sensitivity testing was also performed in order to show that there is no relationship between encrypted images and keys in close proximity.

On the second point, it is shown that when these figures were encrypted and then decoded with damage, they can be improved by using filters. The filters used were: median and average and in both cases the noise decreased. Two measuring instruments to determine the noise reduction were used: Entropy and Correlation coefficient in three directions. In all cases it was possible to reduce noise as illustrated in Tables 8 to 11.

Generally, if the damage is large, say greater than 50%, it can be said that it is very difficult to gather the information of the original image again. Furthermore, the type of damage used was occlusion, leaving for other investigations the analysis of additive and multiplicative noise.

IX. CONCLUSIONS

This research has presented a different way of encrypting color images consisting of two steps, namely in the first a permutation is applied over all image pixels and in the second the AES cryptosystem with variable permutations is used. To show that the encrypted images are of “quality” four tests were applied. The result was that all suggested images that were encrypted passed the randomness tests, and in some cases the

TABLE VI
The Correlation coefficient results in directions; horizontal, vertical, and diagonal for colors red, green and blue for original images Figure 1 and Figure 4.

color	Correlation Coefficient	Figure 1	Figure 4			
			(a)	(b)	(c)	(d)
Red	$Horizontal$	0.60	0.99	0.86	0.89	0.97
	$Vertical$	0.73	0.99	0.77	0.95	0.98
	$Diagonal$	0.50	0.98	0.73	0.88	0.96
Green	$Horizontal$	0.63	0.98	0.90	0.90	0.97
	$Vertical$	0.80	0.98	0.85	0.95	0.98
	$Diagonal$	0.49	0.96	0.84	0.88	0.96
Blue	$Horizontal$	0.60	0.97	0.92	0.89	0.95
	$Vertical$	0.77	0.97	0.87	0.96	0.96
	$Diagonal$	0.47	0.96	0.85	0.88	0.93

TABLE VII
The Correlations results in directions: horizontal, vertical and diagonal for colors red, green and blue, for cipher images with k of Figure 1 and Figure 4.

color	Correlation Coefficient	Figure 1	Figure 4			
			(a)	(b)	(c)	(d)
Red	$Horizontal$	0.031	0.033	0.009	0.004	0.000
	$Vertical$	0.003	0.016	0.041	0.061	0.011
	$Diagonal$	0.004	0.002	0.003	0.038	0.010
Green	$Horizontal$	0.002	0.004	0.006	0.016	0.008
	$Vertical$	0.002	0.002	0.037	0.019	0.026
	$Diagonal$	0.001	0.019	0.014	0.019	0.020
Blue	$Horizontal$	0.005	0.012	0.010	0.040	0.022
	$Vertical$	0.013	0.008	0.009	0.005	0.040
	$Diagonal$	0.014	0.008	0.000	0.003	0.015

TABLE VIII
Entropy results of image (a) Figure 4 with several damage sizes, using median filter

Test name	Size damage Figure 4	Entropy of deciphered image with damage	Entropy with median filter 3 × 3	Entropy with median filter 5 × 5
Red color Entropy	35%	7.851	7.655	7.614
	40%	7.874	7.677	7.627
	45%	7.895	7.694	7.641
Green color Entropy	35%	7.756	7.287	7.258
	40%	7.795	7.279	7.244
	45%	7.829	7.272	7.228
Blue color Entropy	35%	7.649	7.185	7.105
	40%	7.703	7.221	7.113
	45%	7.751	7.261	7.122

TABLE IX
Entropy values of image (a) Figure 4 with several damage sizes, using the average filter

Test name	Size damage Figure 4	Entropy of deciphered image with damage	Entropy with average filter
Red color Entropy	35%	7.851	7.551
	40%	7.874	7.499
	45%	7.895	7.438
Green color Entropy	35%	7.756	7.068
	40%	7.795	7.011
	45%	7.829	7.959
Blue color Entropy	35%	7.649	7.100
	40%	7.703	7.058
	45%	7.751	7.017

ISSN 2395-8618

TABLE X
ENTROPY RESULTS OF IMAGE (A) FIGURE 4 WITH SEVERAL DAMAGE SIZES, USING MEDIAN FILTER

Test name	Direction	Size damage Figure 4	Correlation of decoded image with damage	Correlation with median filter $3 \times x3$	Correlation with median filter 5×5
Red color Correlation	Horizontal	35%	0.392	0.957	0.964
		40%	0.311	0.931	0.956
		45%	0.284	0.922	0.941
	Vertical	35%	0.374	0.961	0.955
		40%	0.339	0.927	0.954
		45%	0.258	0.913	0.946
	Diagonal	35%	0.380	0.938	0.934
		40%	0.321	0.926	0.931
		45%	0.253	0.896	0.913
Green color Correlation	Horizontal	35%	0.217	0.938	0.906
		40%	0.190	0.904	0.894
		45%	0.155	0.888	0.885
	Vertical	35%	0.255	0.947	0.878
		40%	0.227	0.929	0.871
		45%	0.191	0.910	0.882
	Diagonal	35%	0.237	0.919	0.830
		40%	0.207	0.880	0.819
		45%	0.163	0.871	0.812
Blue color Correlation	Horizontal	35%	0.162	0.877	0.939
		40%	0.141	0.842	0.892
		45%	0.128	0.786	0.909
	Vertical	35%	0.185	0.892	0.910
		40%	0.152	0.847	0.905
		45%	0.107	0.824	0.874
	Diagonal	35%	0.148	0.860	0.853
		40%	0.113	0.796	0.848
		45%	0.120	0.722	0.825

TABLE XI
CORRELATION VALUES OF IMAGE (A) FIGURE 4 WITH SEVERAL DAMAGE SIZES, WITH AVERAGE FILTER

Test name	Direction	Size damage Figure 4	Correlation of decoded image with damage	Correlation with average filter
Red color Correlation	Horizontal	35%	0.392	0.957
		40%	0.311	0.918
		45%	0.284	0.900
	Vertical	35%	0.374	0.921
		40%	0.339	0.911
		45%	0.258	0.893
	Diagonal	35%	0.380	0.897
		40%	0.321	0.884
		45%	0.253	0.852
Green color Correlation	Horizontal	35%	0.217	0.929
		40%	0.190	0.841
		45%	0.155	0.816
	Vertical	35%	0.255	0.875
		40%	0.227	0.866
		45%	0.191	0.846
	Diagonal	35%	0.237	0.820
		40%	0.207	0.722
		45%	0.163	0.753
Blue color Correlation	Horizontal	35%	0.162	0.845
		40%	0.141	0.807
		45%	0.128	0.787
	Vertical	35%	0.185	0.831
		40%	0.152	0.816
		45%	0.107	0.782
	Diagonal	35%	0.148	0.772
		40%	0.113	0.749
		45%	0.120	0.687

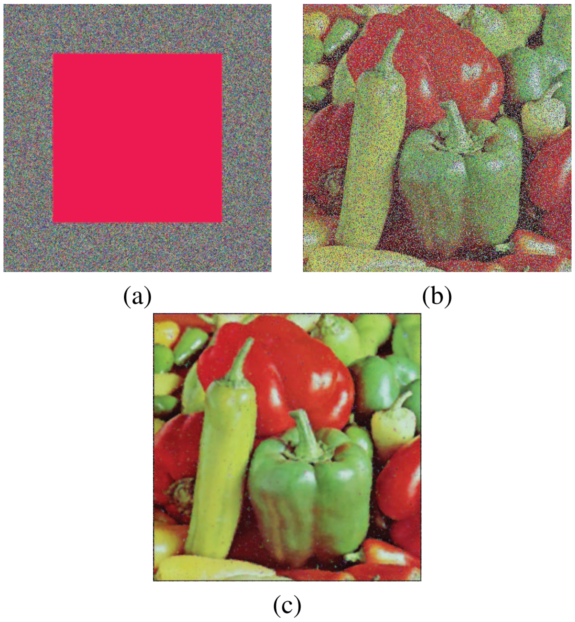


Fig. 6. (a) Ciphered Figure 5 with 40% of damage, (b) deciphered image(a) and (c) median 5 × 5 filter applied to (b).

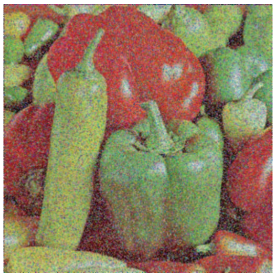


Fig. 7. Image (b) Figure 6 deciphered with 40% damage and average filter applied

results were better than others. Two popular filters were used to improve the deciphered figures with faults. The measuring instruments to determine the degree of improvement in the decoded images show that there is a reduction in noise, i.e. the Entropy amount was reduced and the Correlations coefficient value increased. Finally, the software was developed in C ++ and the time encryption of figures proposed in this research was around 85 milliseconds, and an Intel Core i7 processor was used.

Acknowledgments

The authors would like to thank the Instituto Politécnico Nacional (Secretaría Académica, COFAA, SIP, CIDETEC, ESCOM and ESFM), the CONACyT, and SNI for their financial support to develop this work.

References

[1] S. Baruah, V. Bonifaci, G. D'Angelo, H. Li, A. Marchetti-Spaccamela, N. Megow, and L. Stougie, "Scheduling real-time mixed-criticality jobs," *IEEE Transactions on Computers*, vol. 61, no. 8, pp. 1140–1152, 2012.

[2] L. Xuemei, X. Tong, and L. Dai, "A novel scheme reality preserving image encryption," in *2011 Third International Conference Measuring Technology and Mechatronics Automation*. IEEE, 2011, pp. 218–221.

[3] J. Li and L. Gan, "Study on chaotic cryptosystem for digital image encryption," in *2011 Third International Conference Measuring Technology and Mechatronics Automation*. IEEE, 2011, pp. 426–430.

[4] C. Fu, J. Chen, H. Zou, W. Meng, Y. Zhan, and Y. Yu, "A chaos-based digital image encryption scheme with an improved diffusion strategy," *Optics Express*, vol. 20, no. 3, pp. 2363–78, 2012.

[5] H. Zhu, C. Zhao, and X. Zhang, "A novel image encryption-compression scheme using hyper-chaos and chinese remainder theorem," *Signal Processing: Image Communication*, vol. 28, no. 6, pp. 670–680, 2013.

[6] "Advanced encryption standard (AES)," 2001, FIPS PUB 197.

[7] D. R. Stinson, *Cryptography: Theory and Practice*. Chapman & Hall/CRC Press, 2005.

[8] M. Matsui, "Linear cryptanalysis method for DES cipher," *Lecture Notes in Computer Science*, vol. 765, pp. 386–397, 1994.

[9] E. Biham and A. Shamir, "Differential cryptanalysis of the full 16-round DES," *Lecture Notes in Computer Science*, vol. 740, pp. 487–496, 1993.

[10] S. Keshari and S. G. Modani, "Color image encryption scheme based on 4-weighted fractional fourier transform," *Journal of Electronic Imaging*, vol. 21, no. 3, pp. 033 018–1–6, 2012.

[11] W. Chen, X. Chen, and C. Sheppard, "Optical color-image encryption and synthesis using coherent diffractive imaging in the fresnel domain," *Optics Express*, vol. 20, no. 4, pp. 3853–65, 2012.

[12] P. Refregier and B. Javidi, "Optical image encryption based on input plane and fourier plane random encoding," *Optical Letters*, vol. 20, no. 7, pp. 767–769, 1995.

[13] W. Chen and X. Chen, "Double random phase encoding using phase reservation and compression," *Journal of Optics*, vol. 16, no. 2, pp. 025 402–7, 2014.

[14] W. Chen, B. Javidi, and X. Chen, "Advances in optical security system," *Advances in Optics and Photonics*, vol. 6, no. 2, pp. 120–155, 2014.

[15] Y. Wu, G. Yang, H. Jin, and J. P. Noonan, "Image encryption using the two-dimensional logistic chaotic map," *Journal of Electronic Imaging*, vol. 21, no. 1, p. 013014, 2012.

[16] V. M. Silva-García, R. Flores-Carapia, C. Rentería-Márquez, B. Luna-Benoso, C. A. Jiménez-Vázquez, and M. D. González Ramírez, "Cipher image damage and decisions in real time," *Journal of Electronic Imaging*, vol. 24, no. 1, pp. 013 012–1–13, 2015.

[17] R. Gonzalez and R. Woods, *Digital Image Processing*. Prentice Hall, 2008.

[18] P. Soille, *Morphological Image Analysis*. Springer-Verlag, 2004.

[19] R. Wolpe and R. Myers, *Probability and Statistics for Engineers and Scientists*. Prentice Hall, 2007.

[20] A. Rukhin, J. Soto, J. Nechvatal, M. Smid, E. Barker, S. Leigh, M. Levenson, M. Vangel, D. Banks, A. Heckert, J. Dray, and S. Vo, *A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications*. NIST 800-22, 2010.

[21] J. Peters, D. Janzing, and B. Scholkopf, "Causal inference on discrete data using additive noise models," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 33, no. 12, pp. 2436–2450, 2011.

[22] Y. Yao and S.-S. Chen, "Multiplicative noise enhances spatial reciprocity," *Physica A: Statistical Mechanics and its Applications*, vol. 413, no. 1, pp. 432–437, 2014.

[23] "Prácticas comerciales – requisitos que deben observarse para la conservación de mensajes de datos," 2002, Nom-151. Norma Oficial Mexicana NOM-151-SCFI-2002.

[24] P. Dymora, M. Mazurek, and D. Strzalka, "Long-range dependencies in quick-sort algorithm," *Przeglad Elektrotechniczny*, vol. 90, no. 1, pp. 149–152, 2014.

[25] J. Daemen and V. Rijmen, "AES proposal: Rijndael," 1998.

[26] D. A. Osvik, A. Shamir, and E. Tromer, "Cache attacks and countermeasures: The case of AES," *Lecture Notes in Computer Science*, vol. 3860, pp. 1–20, 2006.

[27] "Data encryption standard (DES)," 1999, FIPS PUB 46-3.

[28] S. Michael, *Calculus: cálculo infinitesimal*. Barcelona, Spain: Reverte, 1993.

[29] "Jaohxv. pi world," http://ja0hxv.calico.jp/pai/estart.html, accessed: 2010-09-30.

[30] J. Gallian, *Contemporary abstract algebra*. Brooks/Cole, 2011.

[31] E. Shannon, "A mathematical theory of communication," *Bell Systems Technical Journal*, vol. 27, no. 3, pp. 379–423, 1948.

ISSN 2395-8618

[32] M. Abramowitz and I. Stegun, *Handbook of Mathematical Functions*. Washington: National Bureau of Standards, 1968.

[33] T. M. Apostol, *Análisis Matemático*. Barcelona: Reverté, 1994.

[34] V. M. Silva-García, R. Flores-Carapia, C. Rentería-Márquez, and B. Luna-Benoso, "Triple-DES-96 cryptographic system," *International Journal of Contemporary Mathematical Sciences*, vol. 8, no. 19, pp. 925–934, 2013.

[35] A. Gómez, *Enciclopedia de la Seguridad Informática*. México: Alfaomega, 2007.

[36] T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Transaction on Information Theory*, vol. 31, no. 4, pp. 469–472, 1985.

[37] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public key cryptosystems," *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, 1978.

[38] R. Azarderakhsh and A. Reyhani-Masoleh, "Parallel and high-speed computations of elliptic curve crytography using hybrid-double multipliers," *Transactions on Parallel and Distributed Systems*, vol. 26, no. 6, pp. 1668–1677, 2015.

[39] D. Jay, *Probabilidad y Estadística: para ingeniería y ciencias*. International Thompson, 2005.

[40] J. D. Argüelles, *Antología general de la poesía mexicana*. México: Océano, 2012.