Efficient Image Chaotic Encryption Algorithm with No Propagation Error

Abir Awad and Dounia Awad

Many chaos-based encryption methods have been presented and discussed in the last two decades, but very few of them are suitable to secure transmission on noisy channels or respect the standard of the National Institute of Standards and Technology (NIST). This paper tackles the problem and presents a novel chaos-based cryptosystem for secure transmitted images. The proposed cryptosystem overcomes the drawbacks of existing chaotic algorithms such as the Socek, Xiang, Yang, and Wong methods. It takes advantage of the increasingly complex behavior of perturbed chaotic signals. The perturbing orbit technique improves the dynamic statistical properties of generated chaotic sequences, permits the proposed algorithm reaching higher performance, and avoids the problem of error propagation. Finally, many standard tools, such as NIST tests, are used to quantify the security level of the proposed cryptosystem, and experimental results prove that the suggested cryptosystem has a high security level, lower correlation coefficients, and improved entropy.

Keywords: Chaos-based cryptosystem, NIST, perturbation technique.

Manuscript received Mar. 5, 2010; revised June 10, 2010; accepted June 25, 2010.

This work has been carried out within the framework of the research project "Apport du Chaos dans la Sécurité des Systèmes Communicants Optiques et Mobiles (ACSCOM)" supported by ANRT.

Abir Awad (phone: +33 2 43594909, email: awad@esiea-ouest.fr) is with the Operational Cryptology and Virology Laboratory (C +V)^O, ESIEA-OUEST, Laval, France.

Dounia Awad (email: dounia_awad@hotmail.com) is with the Department of Informatique, Lebanese University, Tripoli, Lebanon.

doi:10.4218/etrij.10.1510.0063

I. Introduction

Chaos has sensitivity to initial conditions and system parameters (ergodicity and mixing), which are analogous to the confusion and diffusion properties of a good cryptosystem.

In recent years, a large amount of work using digital chaotic systems to construct cryptosystems has been done [1]-[4]. Basically, a number of very different approaches to the use of chaos can be found in the literature [5]-[9].

In order to be used in all applications, chaotic sequences must seem absolutely random and have good cryptographic properties. Many studies on chaotic maps have been undertaken [10], [11]. In [12], we studied and improved some existing techniques used to generate chaotic signals with desired statistical properties and comply with National Institute of Standards and Technology (NIST) statistical tests. Indeed, to obtain better dynamical statistical properties and avoid the degradation caused by the digital chaotic system working in a 2^N finite state, a perturbation technique is used.

It is well known that images are different from texts in many aspects, such as high redundancy and correlation. In most natural images, the value of any given pixel can be reasonably predicted from the values of its neighbors. Many researchers have proposed schemes with combinational permutation techniques [13]-[16].

In this paper, we propose an algorithm based on two chaotic permutation methods: the cyclic shift bit permutation method and a bit permutation method. The former can be a permutation of bits, bytes, or a set of bytes, and the latter is applied on 8 bits whose positions are also controlled by chaos.

The proposed algorithm is an enhancement of the enhanced 1-D chaotic key-based algorithm for image encryption proposed by Socek [7] and the cryptosystems proposed by

Xiang [8], Yang [9], Lian [15], and Wong [16]. The algorithm proposed by Xiang has two remaining problems: the binary sequence used for substitution leaks the trajectory of the chaotic map for easy cryptanalysis, and the encryption speed is still slow compared to conventional cryptosystems. The encryption of a symbol requires 320 to 383 iterations (Table 1 in [8]).

To overcome the drawbacks mentioned above, a new scheme of a block cryptosystem with output feedback (OFB) was proposed [9].

In their algorithms, Socek and Yang propose perturbing the chaotic values with the encrypted data [7], [9]. The perturbation that they propose is not efficient because each encrypted block depends on all the previous encrypted ones. If an error occurs in the encrypted image transmitted on a noisy channel, we will obtain random errors in the decrypted image. Consequently, it is better to use an external perturbation which is independent of the encrypted data, as we did in our algorithm.

The same conclusion can be applied to the Lian [15] and Wong [16] algorithms. In these methods, the pixel value mixing depends on the value of the previously processed pixel. The diffusion effect is injected by adding the current pixel value with the previous permuted pixel. This diffusion method is also not efficient because it helps the error propagation phenomenon. Thus, if a transmission error occurs in the encrypted image, we obtain random errors in the decrypted image.

The paper is organized as follows. Section II briefly introduces the original schemes proposed by Socek [7], Xiang [8], and Yang [9]. Section III describes the proposed algorithm. Section IV explains the decryption process. Section V introduces the perturbed generator used. The simulation results and security analysis are given in section VI. In section VII, we examine the problem of error propagation. The last section concludes this paper.

II. Overview of Two Existing Algorithms

1. Socek Algorithm

The encryption algorithm in Fig. 1 transforms an image I using an SP-network generated by a piecewise linear chaotic map (PWLCM) and a 128 bit secret key. The algorithm performs r rounds of an SP-network on each pixel. The next iteration of the chaotic map is perturbed using the previous cipher block.

The permutation is made on the 8 bits of each block made up of 4 bytes. In other words, we use a permutation of degree 8 to add diffusion to the system. Actually, the fastest way to achieve this is by using a table lookup approach.

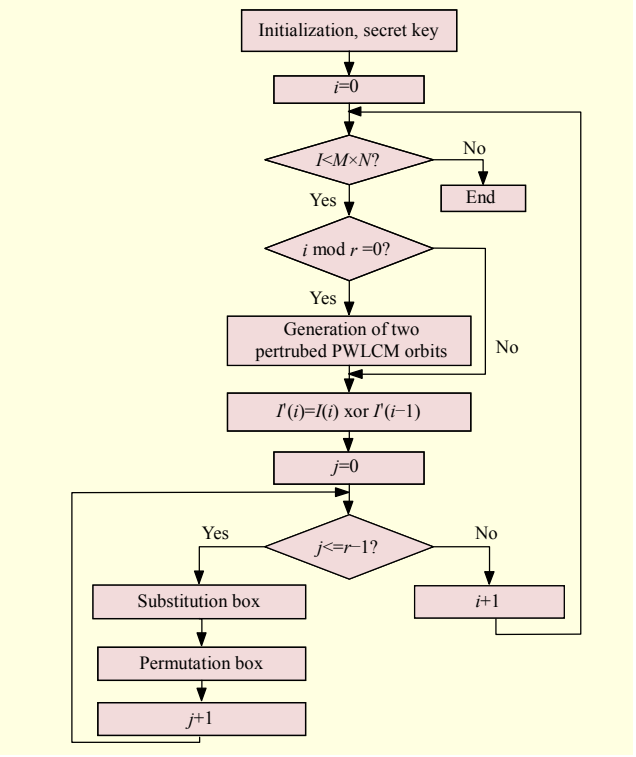


Fig. 1. Socek encryption algorithm.

This approach is fast, but the memory requirements are considerably high. A number of permutation methods have been proposed [7], [17]-[19].

Among these, the Socek method [7] is the most attractive. It is fast and has good cryptographic properties.

2. Xiang and Yang Algorithm

The proposed scheme is described below, and an illustration is given in Fig. 2.

The steps of the Xiang encryption algorithm are as follows:

Step 1. The logistic map is iterated 70 times.

Step 2. The binary sequences A_j supplied by all the third bits of the chaotic values must be obtained.

Step 3. An integer D_j is computed as the decimal value of a part of the chaotic value bits.

Step 4. The key dependent permutation method [8], [9] is used. This method permutes the block with left cyclic shift D_j bits as illustrated in Fig. 3.

Step 5. A bit xor operation is used to mask the permuted data with the binary sequence A_j.

Step 6. The value D_j will be used to iterate the logistic map successively after the current block has been encrypted.

The key dependent permutation is controlled by the chaotic value. The permutation is then different for different message

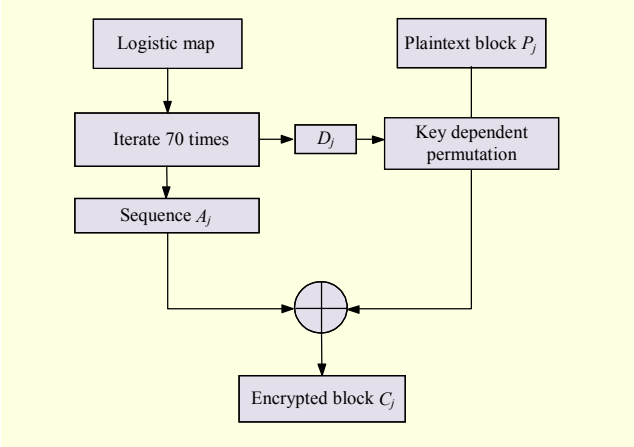


Fig. 2. Xiang encryption algorithm.

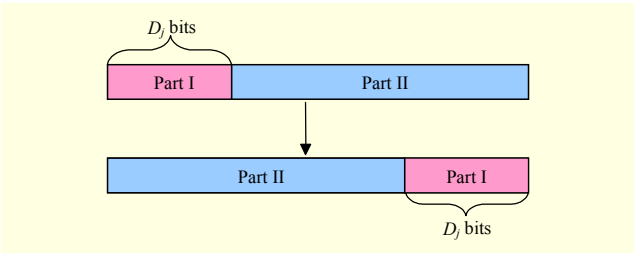


Fig. 3. Xiang permutation method.

blocks. This permutation can take the form of pixel position permutation or bit permutation.

However, this algorithm is not secure. The binary sequence A_j leaks the trajectory of the chaotic map for easy cryptanalysis, and the encryption speed is slow. The number of iterations required for the encryption of a symbol is both large and random.

Later in his paper, Yang [9] proposed using OFB to overcome this problem. He generates the binary sequence A_j using the cipher image.

The use of the encrypted blocks to perturb the chaotic orbits, proposed by Socek and Yang in their algorithms, is not efficient. Perturbation methods cause propagation of errors in the decrypted images when a bit error occurs in the transmitted encrypted image. Consequently, they are not suitable to transmission on a noisy channel. In [20], we proposed an improvement of Socek's algorithm using a different manner to perturb the chaotic orbit. However, the encrypted images cannot pass all NIST tests. In the next section of this paper, we propose a new algorithm that is secure and better suited to transmission than these algorithms.

III. Proposed Encryption Algorithm

In this section, we present the proposed algorithm (Fig. 4) for

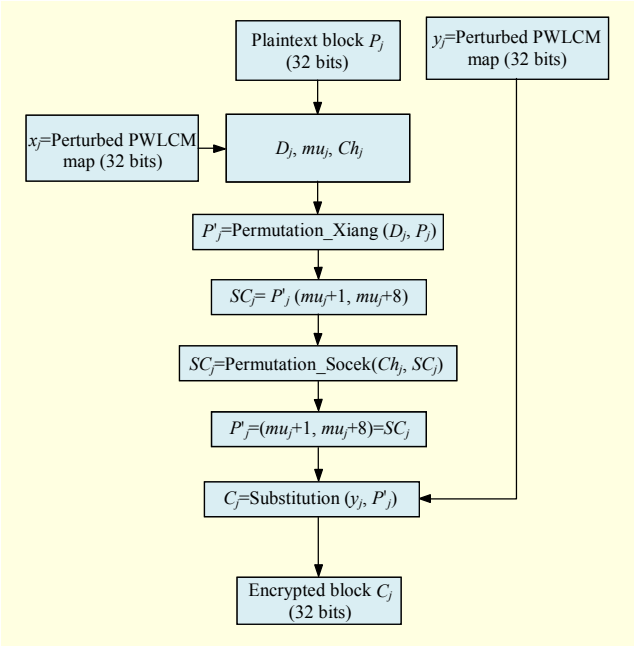


Fig. 4. Proposed encryption algorithm.

image encryption that we implemented with Matlab.

Let I be an $M \times N$ image with a byte pixel value. A block cipher is an encryption scheme which breaks up the plaintext messages into blocks of fixed length (32 bits or b=4 bytes) and encrypts one block at a time. A block value is denoted by P_j, $0 \leq j < M \times N/b$.

The characteristics and steps of the proposed encryption algorithm are:

Step 1. The key size is 128 bits.

Step 2. The PWLCM currently used is replaced with a perturbed PWLCM to improve statistical properties.

Step 3. In fact, the chaotic value is generated on 32 bits, and then this value is used to give the tree parameters D_j, mu_j, and Ch_j as

$$D_j = \mathrm{mod}(x_j, 31), \tag{1}$$

$$mu_j = \mathrm{mod}(x_j / 2^7, 25), \tag{2}$$

$$Ch_j = \mathrm{mod}(x_j / 2^{16}, 8!). \tag{3}$$

The first parameter D_j is used to control Xiang permutation. The second parameter mu_j indicates the position of the 8 bits considered to be permuted by the Socek method. The last one, Ch_j, is used to control the Socek permutation method.

Step 4. The permutation box adds diffusion to the system in two steps. First, the bits of each block are permuted with left cyclic shift D_j bits according to the approach illustrated in Fig. 5. Then these bits are permuted by the Socek method. The

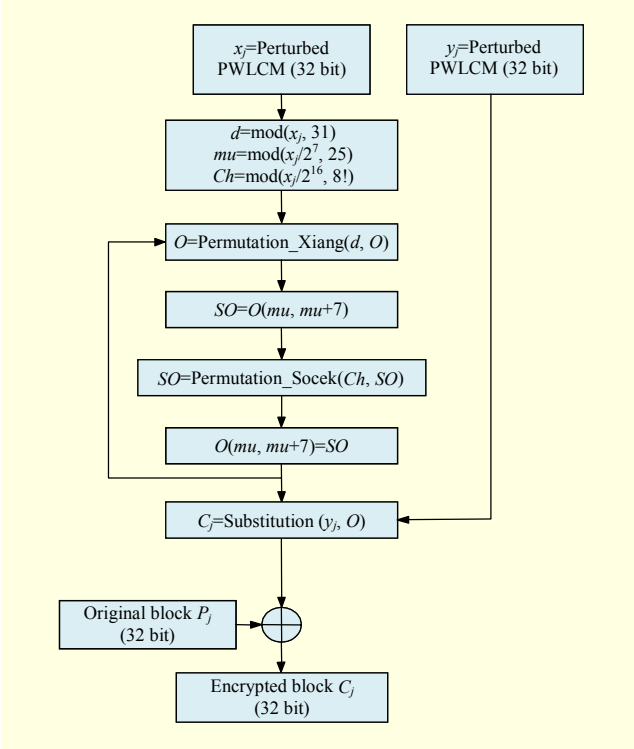


Fig. 5. Proposed encryption algorithm with OFB operation mode.

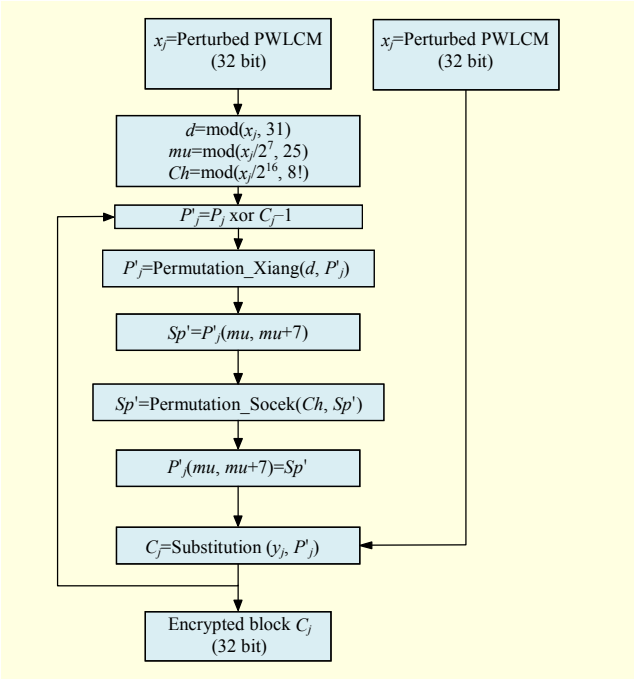


Fig. 6. Proposed encryption algorithm with CBC operation mode.

last one permutes only 8 bits of the block. These bits are chosen by the chaotic value, and this permutation is also controlled by the chaotic map.

Step 5. Another perturbed chaotic map is used to control the substitution box (S-box). The S-box used is the classic chaotic masking technique. The following manipulation (4) is applied.

Step 6. Substitution $(y_j, P'_j) = y_j \oplus P'_j$, (4)

where y_j and P'_j are two blocks of 4 bytes.

In order to disturb the high correlation among adjacent pixels, we propose a scheme that includes two permutation methods. These methods, developed by Xiang and Socek, are chaotic. They are applied to a block of 4 bytes. The first permutation can be a bit permutation or a pixel permutation method, and the second one permutes 8 bits whose positions are given by the chaotic value *mu*.

Our algorithm does not permit the propagation of errors which result from the consideration that the permutation of each block is independent of the other block of the image. The proposed permutation techniques decrease the correlation in the encrypted image and avoid some of the inconveniences of the existing permutation methods, such as the Lian [15] and Wong [16] techniques. Therefore, the proposed algorithm uses a perturbed chaotic map with good dynamic properties that we will explain in the next section. The used perturbation technique does not only enhance the characteristics of the chaotic map, as we proved in [12], but also helps to avoid the propagation of errors in the decrypted image. In fact, the results obtained by Socek [7] and Yang [9] are not suitable to the transmission on a corrupted channel. In their algorithms, they use a perturbation technique of the chaotic map using the encrypted data. Then, if a transmission error occurs in the ciphered image, random errors occur in the decrypted image.

Our algorithm overcomes the drawbacks of the above mentioned algorithms, and some experimental results will be drawn in section VI to prove the robustness and security of the proposed algorithm.

Figures 5 and 6 show the encryption algorithms with OFB and cipher block chaining (CBC) operation modes.

IV. Decryption Process

The decryption algorithm depends on the cipher mode used. For the OFB mode, CFB mode, and counter-mode encryption, the decryption algorithm is the same as that of the encryption. However, for the CBC mode, it differs slightly from the encryption algorithm. To decrypt an encrypted image, we need to perform the inverse transformations (Fig. 7).

In the inverse Socek method, the bits are rearranged according to the array indices (8-$p(i)$) instead of $p(i)$ used in the encryption process. Therefore, we need to reverse the order of the substitution and bit permutation methods. Then, we use the inverse methods to decrypt the image.

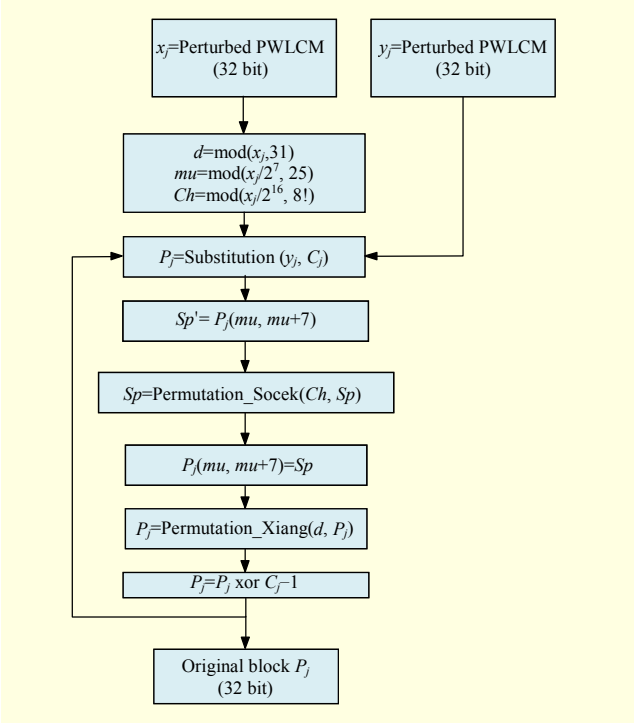


Fig. 7. Proposed decryption algorithm with CBC operation mode.

V. Perturbed PWLCM

A PWLCM is a map composed of multiple linear segments.

$$x(n) = F[x(n-1)] = \begin{cases} x(n-1)\times\dfrac{1}{p}, & \text{if } 0 \le x(n-1) < p, \\ \left[x(n-1)-p\right]\times\dfrac{1}{0.5-p}, & \text{if } p \le x(n-1) < 0.5, \\ F[1-x(n-1)], & \text{if } 0.5 \le x(n-1) < 1, \end{cases} \quad (5)$$

where the positive control parameters are $p \in (0; 0.5)$ and $x(i) \in (0; 1)$. Since digital chaotic iterations are constrained in a discrete space with 2^N elements, it is obvious that every chaotic orbit will eventually be periodic and in a cycle with a limited length not greater than 2^N [21], [22]. Generally, each digital chaotic orbit includes two connected parts: $x_1, x_2, \ldots, x_l$ and $x_l, x_{l+1}, \ldots, x_{l+n}$, which are respectively called "transient branch" and "cycle." Accordingly, l and $n+1$ are respectively called "transient length" and "cycle period," and $l+n$ is called "orbit length."

To improve the dynamic statistical properties of generated chaotic sequences, a perturbation-based algorithm is used. The cycle length is expanded, and consequently, good statistical properties are reached. Many perturbation techniques have been proposed. For example, Socek [7] and Yang [9] use a perturbation-based algorithm. The orbits are perturbed by the encrypted blocks. Their algorithms are very secure, but a bit transmission error causes a random number of erroneous bits in the decrypted image. In this paper, we use another perturbation technique using a maximal length linear feedback shift register (LFSR), which is a suitable candidate for perturbing the signal generator [21].

Here, for computing precision N, each x can be described as

$$x(n) = 0.x_1(n)x_2(n)\ldots x_i(n)\ldots x_N(n) \qquad x_i(n) \in \{0,1\}, \; i = 1, 2, \ldots, N. \quad (6)$$

The perturbing bit sequence can be generated every n clock by

$$Q^+_{k-1}(n) = Q_k(n) = g_0 Q_0(n) \oplus g_1 Q_1(n) \oplus \ldots \oplus g_{k-1} Q_{k-1}(n) \qquad n = 0, 1, 2, \ldots, \quad (7)$$

where $\oplus$ represents 'exclusive or,' $g = [g_0\, g_1 \cdots g_{k-1}]$ is the tap sequence of the primitive polynomial generator, and $Q_0\, Q_1 \ldots Q_{k-1}$ are the initial register values of which at least one is non zero.

The perturbation begins at n= 0 and then occurs periodically every Δiterations (Δ is a positive integer) with $n = l\times\Delta$, l=1, 2, …. The perturbed sequence is given by

$$x_i(n) = \begin{cases} F[x_i(n-1)], & 1 \le i \le N-k, \\ F[x_i(n-1)] \oplus Q_{N-i}(n), & N-k+1 \le i \le N, \end{cases} \quad (8)$$

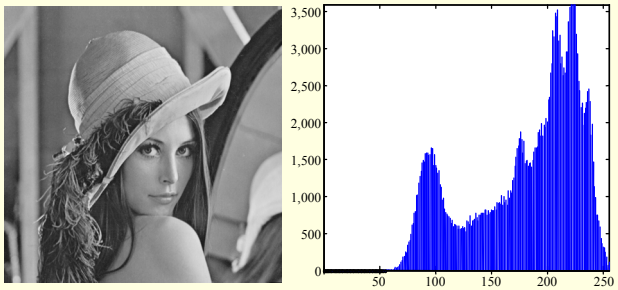
where $F[x_i(n)]$ represents the i-th bit of $F[x(n)]$. The perturbation is applied on the last k bits of $F[x(n)]$. When $n \neq l\times\Delta$, no perturbation occurs, so $x(n) = F[x(n-1)]$.

The lower boundary of the system cycle length is given by (9) (see appendix):

$$T_{\min} = \Delta\times\left(2^k - 1\right). \quad (9)$$

VI. Simulation Results and Security Analysis

Some experimental results are given in this section to demonstrate the efficiency of our scheme. The plain image 'Lena' in 512×512 format and the corresponding histogram are shown in Fig. 8. We also perform some tests on the colored Lena image (see Fig. 9(a)). In addition, grayscale and colored Mandrill images are used to prove the efficiency of the algorithm. The performance of our algorithm is then proved through several indicators: the Pearson's correlation coefficient r, number of pixels change rate (NPCR), unified average changing intensity (UACI), histograms, plaintext and key sensitivity, entropy information H, and NIST statistical tests.



(a) (b)

Fig. 8. (a) Lena image and (b) its histogram.

(a) (b)

Fig. 9. Colored (a) Lena and (b) Mandrill images.

1. Efficiency of Perturbed PWLCM

This subsection presents an experimental comparison of the original PWLCM and the perturbed PWLCM. Both chaotic maps are then used to control the Socek bit-permutation method. Then, we show the correlation between the original and the obtained permuted images. To do this test, we used the original colored images of Lena and Mandrill in 512×512×3 format (Fig. 9).

To quantify the dependence between two images, Pearson's correlation coefficient is commonly used. Given by (13), this coefficient is obtained by dividing the covariance between the two images (12) by the product of their standard deviations as in (10) and (11). E in (10) is the expected value operator. $P_1(i,j)$ and $C_1(i, j)$ are pixels gray values of the original and the permuted images, respectively.

$$E(x) = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} P_1(i,j) , \tag{10}$$

$$D(P_1) = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} [P_1(i,j) - E(P_1(i,j))]^2 , \tag{11}$$

$$\mathrm{cov}(P_1, C_1) = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} [P_1(i,j) - E(P_1(i,j))][C_1(i,j) - E(C_1(i,j))], \tag{12}$$

Table 1. Mean values of correlation coefficients of intra-component of original and permuted images.

Correlation	Lena image	Mandrill image	Permuted image using PWLCM to control Socek method		Permuted Lena image using perturbed PWLCM to control Socek method	
			Lena	Mandrill	Lena	Mandrill
Red (R) component Correlation	0.0642	0.1911	0.0057	0.0171	0.0035	0.0155
Green (G) component Correlation	0.0426	0.0883	0.0033	0.0066	0.0025	0.0055
Blue (B) component Correlation	0.0360	0.0948	0.0051	0.0152	0.0046	0.0138
Mean value	0.0476	0.1247	0.0047	0.0130	0.0035	0.0116

Table 2. Inter-components correlation coefficients of original and permuted images.

Correlation	Lena image	Mandrill image	Permuted image using PWLCM to control Socek method		Permuted image using perturbed PWLCM to control Socek method	
			Lena	Mandrill	Lena	Mandrill
Correlation between R and G	0.8786	0.3565	0.1330	0.1280	0.0615	0.0703
Correlation between G and B	0.9106	0.8074	0.0954	0.0684	0.0381	0.0591
Correlation between B and R	0.6764	0.1237	0.0463	0.0161	0.0120	0.0088

$$r_{P_1C_1} = \frac{\mathrm{cov}(P_1, C_1)}{\sqrt{D(P_1)}\sqrt{D(C_1)}}. \tag{13}$$

Tables 1 and 2 give the correlation coefficients of intra-components and inter-components of original and permuted images. It can be seen that the use of chaotic maps reduces significantly the intra-component correlation coefficients which are already low. However, the proposed perturbed PWLCM associated with the Socek method presents the lower average correlation coefficient compared to that obtained with a normal PWLCM. Results in Table 2 are similar to those in Table 1, and the same conclusions can then be formulated.

2. Histograms of Original and Encrypted Image

The encrypted image of Lena and its histogram are shown in

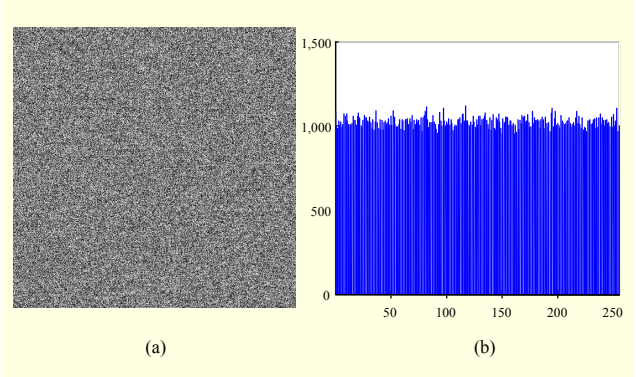


Fig. 10. (a) Encrypted image of Lena and (b) its histogram.

Fig. 10. As we can see, the histogram of the ciphered image is fairly uniform and is significantly different from that of the original image as shown in Fig. 8(b).

3. Comparison between Original and Encrypted Image

Common measures like correlation, NPCR, and UACI are used to test the difference between the original image, P_1, and the encrypted one, C_1.

We calculate the correlation coefficient r of the original and encrypted images by using (10)-(13).

As mentioned, NPCR stands for the number of the pixel change rate. Then, if D is a matrix with the same size as images P_1 and C_1, $D(i,j)$ is determined as

$$D(i,j)=\begin{cases}1 & \text{if } P_1(i,j)\neq C_1(i,j),\\ 0 & \text{else.}\end{cases} \tag{14}$$

NPCR is defined by

$$\text{NPCR}=\frac{\sum_{i=0}^{M-1}\sum_{j=0}^{N-1}D(i,j)}{M\times N}\times 100, \tag{15}$$

where M and N are the width and height of P_1 and C_1.

The UACI measures the average intensity of differences between the plain image and the ciphered image. UACI is defined by

$$\text{UACI}=\frac{1}{M\times N}\sum_{i=0}^{M-1}\sum_{j=0}^{N-1}\frac{\left|P_1(i,j)-C_1(i,j)\right|}{255}\times 100. \tag{16}$$

In Table 3, we summarize the correlation, NPCR, and UACI obtained between the original image and the encrypted ones. It is shown that we have obtained a low correlation between the original and the encrypted images. The NPCR and UACI are high enough to confirm that the two images are very different. The high difference between the original and the encrypted images and the randomness of the encrypted images prove that the algorithm is secure against the cipher text only attack.

Table 3. Correlation, NPCR, and UACI between original image and encrypted ones.

	Correlation	NPCR	UACI
Lena image	–0.0022	99.6246	29.9932
Mandrill image	–0.0028	99.6235	33.0823

4. Key Sensitivity

An encryption scheme has to be key-sensitive, meaning that a tiny change in the key will cause a significant change in the output. In order to demonstrate key sensitivity, the following experiments have been performed with a slightly different key.

Figure 10(a) shows the encrypted image of Lena with the following key: alpha=0.35899926, beta=0.25899926, x_0=0.7239, and y_0=0.5672. Here, alpha and beta are the control parameters of the PWLCMs, and x_0 and y_0 are the initial conditions of these maps. We encrypt the same image using the slightly changed key as follows: alpha=0.35899927. We obtain a figure similar to Fig. 10(a).

Table 4 shows the difference between the two ciphered images of Lena. Similar results are obtained for Mandrill image. As we can see here, our algorithm is quite sensitive to the key. The two obtained encrypted images are very different and resemble random data. Our algorithm has a long key of 128 bits. Because it is very sensitive to the key, it is secure against the brute force attack.

Table 4. Correlation, NPCR, and UACI between two ciphered images encrypted with slightly different keys.

	Correlation	NPCR	UACI
Lena image	0.0029	99.6128	33.4420

5. Plaintext Sensitivity

For the test of sensitivity on small plain image changes, we used two plain images of Lena different by only one bit. The obtained encrypted images are identical only in 22%. This result demonstrates that the cipher is sensitive to small changes in the original image. Then, we can conclude that the algorithm resists the plaintext attack and differential attack.

6. Correlation of Adjacent Pixels

Statistical analysis conducted on a large amount of images shows an average of 8 to 16 adjacent pixels are correlated in

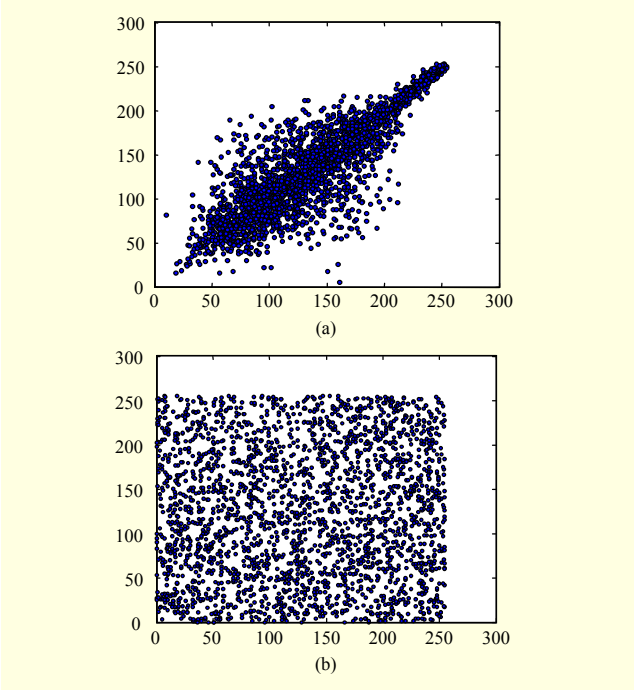


Fig. 11. Correlation distributions of two horizontally adjacent pixels (a) in the original image and (b) in the ciphered image.

Table 5. Correlation coefficients of adjacent pixels.

Model	Original image		Ciphered image	
	Lena	Mandrill	Lena	Mandrill
Horizontal	0.9829	0.9203	0.0377	0.0282
Vertical	0.9907	0.8631	0.0107	–0.0022
Diagonal	0.9722	0.8494	0.0119	–0.0193

each image.

To test the correlation between horizontally, vertically, and diagonally adjacent pixels from the image, we calculate the correlation coefficient of a sequence of adjacent pixels by using the following formulas (10)-(13).

Figure 11 shows the correlation distributions of two horizontally adjacent pixels in the original and the ciphered images of Lena. In Table 5, we show the correlation coefficients of the Lena and Mandrill images.

7. Information Entropy Analysis

Entropy is a statistical measure of randomness that can be used to characterize the texture of an image. It is well known that the entropy $H(m)$ of a message source m can be calculated as

Table 6. Entropy value for images encrypted with different algorithms.

Algorithm	Original image	Xiang algorithm	Proposed image
Entropy	7.3479	7.9950	7.9993

$$H(m)=\sum_{i=0}^{2^N-1} p(m_i)\log_2\frac{1}{p(m_i)}, \quad (17)$$

where $p(m_i)$ represents the probability of message m_i, N=8.

When an image is encrypted, its entropy should ideally be 8. If the entropy value is lower than this, a certain degree of predictability is introduced which threatens the security of the encrypted image. In Table 6, we show the entropy of the original image, the one encrypted by using the Xiang algorithm [8] and the proposed algorithm. The values obtained are very close to the theoretical value 8, and the entropy found using our algorithm is better than the value obtained with the Xiang algorithm. This means that information leakage in the encryption process is negligible, and the encryption system is secure against the entropy attack. Similar results are obtained using Mandrill image.

8. NIST Statistical Tests

Among the numerous standard tests for pseudo-randomness, a convincing way to show the randomness of the produced sequences is to confront them with the NIST statistical tests. The NIST statistical test suite [23] is a statistical package consisting of 188 tests that were developed to test the

Table 7. NIST statistical test for 100 encrypted images by enhanced Socek and proposed algorithms.

Statistical test	Improved Socek algorithm	Proposed algorithm
Frequency	93	100
Block frequency	99	98
Runs	97	97
Longest run	97	97
Rank	100	98
Discrete Fourier Transform	99	97
Cumulative sums 1	94	100
Approximate entropy	98	98
Universal	99	97
Serial 1	99	98
Linear complexity	98	100
Overlapping templates	99	98

randomness of arbitrary long binary sequences produced by either hardware-based or software-based cryptographic random or pseudorandom number generators. These tests focus on a variety of different types of non-randomness that could exist in a sequence.

To verify our results, we use the above test suite to test the randomness of a sequence made of 100 encrypted images of length 512×512=2,097,152 bits. We tested sequences given by the improved Socek algorithm that we proposed in [20] and by the algorithm we are proposing here. In Table 7, we show the results for a number of tests. The sequences passed all the other tests. These results are not shown here. Note that the 100 encrypted images were generated with randomly selected secret keys.

VII. Propagation Error

A bit error is the substitution of a '0' bit for a '1' bit, or vice versa. These errors are generated by the transmission channel as a consequence of interference and noise. The error propagation phenomenon implies that errors in the encrypted text produce errors in the decrypted plaintext. Therefore, it is important that the decrypting process be able to recover from bit errors in the ciphertext.

In this section, we examine the problem of error propagation in two cipher block modes of operation, CBC and OFB, using the Lena image. Similar results are obtained for the Mandrill image. As we can see in Table 8, in CBC mode, all bit positions that contain bit errors in a cipher text block will produce an random bit error in the same decrypted block and a specific bit error in another one. The other bit positions are not affected. For the OFB mode, bit errors within a ciphertext block do not affect the decryption of any other block.

Table 8. Effects of bit errors using proposed algorithm in cipher block modes of operation OFB and CBC.

Erroneous blocks in ciphered image	Number of erroneous blocks in deciphered image		Erroneous blocks in deciphered image	
	OFB mode	CBC mode	OFB mode	CBC mode
(1, 1)	1	2	(1, 1)	(1, 1), (1, 4)
(50, 100)	1	2	(50, 100)	(50, 100), (50, 104)
(405, 238)	1	2	(405, 238)	(405, 238), (405, 241)

Table 9. Effects of bit errors using Socek algorithm in CBC operation mode.

Erroneous blocks in the ciphered image	Number of erroneous blocks in the deciphered image
(1, 1)	527
(50, 100)	1250
(405, 238)	228

The results obtained for Socek, Yang, Lian, and Wong algorithms are not compliant with the recommendations exposed in [24]. For example, Table 9 gives the effects of bit error using Socek algorithm.

In fact, in their algorithms, they use a perturbation technique of the chaotic map or a diffusion method using the encrypted data. Then, if a transmission error occurs in the ciphered image, there are random errors in the decrypted image. However, in our algorithm, we perturb the chaotic value with an LFSR, and the permuted blocks are independent. As a result, we manage to avoid the propagation error in the decrypted image.

VIII. Conclusion

In this paper, a new chaos-based cryptosystem is proposed. Our cryptosystem is based on the original Socek algorithm, as well as the algorithms developed by Xiang, Yang, and Wong, but unlike previous algorithms, ours produces cryptograms suitable for transmission on insecure and noisy channels.

Furthermore, the introduction of the perturbation technique has expanded the length of the chaotic orbit cycle and enhanced the dynamic statistical properties of the generated chaotic sequences. The obtained results of uniformity, key sensitivity, plaintext sensitivity, correlation, entropy, and NIST statistical tests prove the robustness and the high security level of the proposed cryptosystem.

Appendix

Theoretical Analysis of Expanded Cycle Length

Assume that the system has entered a period T state after n_0 iterations, that is, $x_i(n+T)=x_i(n)$ $\left(\text{for } n > n_0; 1 \le i \le N\right)$ and $n_1 = l_1 \times \Delta > n_0$ (l_1 is a positive integer), then $x_i(n_1+T)=x_i(n_1)$ for $1 \le i \le N$. If $T \ne l \times \Delta$ (l is a positive integer), the above equation implies $F[x_i(n_1-1+T)] = F[x_i(n_1-1)] \oplus Q_{N-i}(l_1)$ (for $N-k+1 \le i \le N$). Since period T is defined as $F[x_i(n_1-1+T)] = F[x_i(n_1-1)]$ (for $1 \le i \le N$), thus, $Q_{N-i}(l_1)=0$ (for $N-k+1 \le i \le N$). Because the initial sequences $Q_0, Q_1, \ldots, Q_{k-1}$ are not all zeros, the previous case will not occur. This implies that we only have $T = l \times \Delta$, which means $F[x_i(n_1-1+T)] \oplus Q_{N-i}(l+l_1) = F[x_i(n_1-1)] \oplus Q_{N-i}(l_1)$ (for $N-k+1 \le i \le N$). As a result, we find $Q_{N-i}(l+l_1) = Q_{N-i}(l_1)$ (for $N-k+1 \le i \le N$). This implies $l = \sigma\left(2^k-1\right)$, where σ is a positive integer.

Therefore, the system cycle length is given by $T = \sigma \times \Delta \times (2^k - 1)$, and $T_{\min} = \Delta \times (2^L - 1)$ is the lower bound of the system cycle length.

References

[1] A. Riaz and M. Ali, "Chaotic Communications, Their Applications and Advantages over Traditional Methods of Communication," *IEEE Commun. Syst., Networks Digital Signal Process.*, July 2008, pp. 21-24.

[2] G. Millérioux, J.M. Amigo, and J. Daafouz, "A Connection between Chaotic and Conventional Cryptography," *IEEE Trans. Circuits Syst.*, vol. 55, no. 6, July 2008, pp. 1695-1703.

[3] L. Kocarev, "Chaos Based Cryptography: A Brief Overview," *IEEE Trans. Circuits Syst. Mag.*, vol. 1, no. 3, 2001, pp. 6-21.

[4] G. Alvarez and S. Li, "Some Basic Cryptographic Requirements for Chaos Based Cryptosystems," *Int. J. Bifurcation Chaos*, vol. 16, no. 8, 2006, pp. 2129-2151.

[5] T. Yang, C.W. Wu, and L.O. Chua, "Cryptography Based on Chaotic Systems," *IEEE Trans. Circuits Syst.*, vol. 44, no. 5, Feb. 1997, pp. 469-472.

[6] G. Jakimoski and L. Kocarev, "Chaos and Cryptography: Block Encryption Ciphers Based on Chaotic Maps," *IEEE Trans. Circuits Syst.*, vol. 48, no. 2, Feb. 2001, pp. 163-169.

[7] D. Socek et al., "Enhanced 1-D Chaotic Key Based Algorithm for Image Encryption," *IEEE Security Privacy for Emerging Areas in Commun. Networks*, 2005, pp. 406-407.

[8] T. Xiang et al., "A Novel Block Cryptosystem Based on Iterating a Chaotic Map," *Phys. Lett. A*, vol. 349, 2006, pp. 109-115.

[9] D. Yang et al., "A Novel Block Cryptosystem Based on Iterating Map with Output Feed-Back," *Chaos, Solitons and Fractals*, vol. 41, 2009, pp. 505-510.

[10] S. El Assad and C. Vladeanu, "Digital Chaotic Codec for DS-CDMA Communication Systems," *Lebanese Sci. J.*, vol. 7, no. 2, 2006, pp. 55-71.

[11] L. Kocarev et al., "Discrete Chaos I: Theory," *IEEE Trans. Circuits Syst. Mag.*, vol. 53, no. 6, June 2006, pp.1300-1309.

[12] A. Awad et al., "Comparative Study of 1-D Chaotic Generators for Digital Data Encryption," *IAENG Int. J. Comput. Sci.*, vol. 35, no. 4, 2008, pp. 483-488.

[13] D. Xiao, X. Liao, and P. Wei, "Analysis and Improvement of a Chaos-Based Image Encryption Algorithm," *Chaos, Solitons & Fractals*, vol. 40, no. 5, 2009, pp. 2191-2199.

[14] M. Ali B. Younes and A. Jantan, "An Image Encryption Approach Using a Combination of Permutation Techniques Followed by Encryption," *IAENG Int. J. Comput. Sci. Network Security*, vol. 8, no. 4, 2008, pp. 191-197.

[15] S.G. Lian, J. Sun, and Z. Wang, "A Block Cipher Based on a Suitable Use of Chaotic Standard Map," *Chaos, Solitons and Fractals*, vol. 26, no. 1, 2005, pp. 117-129.

[16] K.W. Wong, B.S.H. Kwok, and W.S. Law, "A Fast Image Encryption Scheme Based on Chaotic Standard Map," *Phys. Lett. A*, vol. 372, no. 15, 2008, pp. 2645-2652.

[17] Z. Shi and R. Lee, "Bit Permutation Instructions for Accelerating Software Cryptography," *IEEE Application-Specific Syst. Architectures Processors*, 2000, pp. 138-148.

[18] R.B. Lee, Z. Shi, and X. Yang, "Efficient Permutation Instructions for Fast Software Cryptography," *IEEE Micro*, vol. 21, no. 6, 2001, pp. 56-69.

[19] Y. Hilewitz, Z.J. Shi, and R.B. Lee, "Comparing Fast Implementations of Bit Permutation Instruction," *IEEE Signals Syst., Comput.*, vol. 2, 2004, pp. 1856-1863.

[20] A. Awad, S.E. Assad, and D. Carragata, "A Robust Cryptosystem Based Chaos for Secure Data," *IEEE Int. Symp. Image/Video Commun. over Fixed Mobile Networks*, Bilbao, Spain 2008.

[21] S. Tao, W. Ruli, and Y. Yixun, "Perturbance Based Algorithm to Expand Cycle Length of Chaotic Key Stream," *IEEE Electron. Lett.*, vol. 34, no. 9, 1998, pp. 873-874.

[22] S. Li et al., "On the Security of a Chaotic Encryption Scheme: Problems with Computerized Chaos in Finite Computing Precision," *Comput. Phys. Commun.*, vol. 153, no. 1, 2003 pp. 52-58.

[23] A. Rukin et al., "A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications," *NIST Special Publication*, pp. 800-822 (with revisions dated May 15, 2001).

[24] M. Dworkin, "Recommendation for Block Cipher Modes of Operation. Methods and Techniques. Computers Security," Computer Security Division, Nat. Inst. Standards Technol., Gaithersburg, MD 20899-8930, 2001.

Abir Awad received her BS in electrical and electronic engineering from Lebanese University, Tripoli, Lebanon, her MS in science and technology from the University of Technology of Compiègne, France, in 2005 and 2006, respectively. She received her PhD in electronics from the University of Nantes, France, in 2009. She is now with the Operational Cryptology and Virology Laboratory at Ecole Supérieure d'Informatique, Électronique, Automatique Ouest.

Dounia Awad received a diploma in computer science from Lebanese University, Tripoli, Lebanon, in 2009, and is currently enrolled in a master's degree research program in computer science at Lebanese University.