Optics & Laser Technology 64 (2014) 145–155

Contents lists available at ScienceDirect

Optics & Laser Technology

journal homepage: www.elsevier.com/locate/optlastec

Fast encryption of RGB color digital images using a tweakable cellular automaton based schema

Faraoun Kamel Mohamed*

Computer Sciences Department, Djilalli Liabbes University, Sidi Bel Abbès, Algeria

ARTICLE INFO

Article history:
Received 27 January 2014
Received in revised form
4 March 2014
Accepted 15 May 2014
Available online 10 June 2014

Keywords:
Reversible cellular automata
Images encryption
Tweakable pseudorandom permutations

ABSTRACT

We propose a new tweakable construction of block-enciphers using second-order reversible cellular automata, and we apply it to encipher RGB-colored images. The proposed construction permits a parallel encryption of the image content by extending the standard definition of a block cipher to take into account a supplementary parameter used as a tweak (nonce) to control the behavior of the cipher from one region of the image to the other, and hence avoid the necessity to use slow sequential encryption's operating modes. The proposed construction defines a flexible pseudorandom permutation that can be used with efficacy to solve the electronic code book problem without the need to a specific sequential mode. Obtained results from various experiments show that the proposed schema achieves high security and execution performances, and enables an interesting mode of selective area decryption due to the parallel character of the approach.

1. Introduction

The extensive need for fast and secure image's representation, transmission and storage schemas has become more and more important and crucial, due to the huge expansion of images and multimedia use in current nowadays applications, and especially because digital images contain generally private information associated with financial, medical or personal interest. Digital images, and specially colored ones, have some specific characteristics that make them different from traditional binary data types, and hence their encryption using classical standards cannot achieve the best efficacy and performances. High pixels correlation and blocks redundancy are specific characteristics of digital images that need dedicated encryption algorithms to provide better speed and security performances.

Many images encryption schemas have been proposed in the literature using different models and techniques to handle the image's particularities described above. Chaos-based encryption schemas use a confusion/diffusion principle to ensure resistance of the ciphered images against known-plaintext and chosen-plaintext attacks [1–6], and conjointly, cellular automata (CA) theory has also been successfully and widely used to build robust images cryptosystems by exploiting the CA's randomness properties and the capacity to exhibit complex and unpredictable behavior. Since the first work proposed by Wolfram [7] to build a stream cipher using elementary one-dimensional cellular automaton evolved with the transition rule 30, many approaches emerged using different classes and models of CAs. Works in [8–10] propose variants of CA-based stream ciphers for image encryption using combination of several rules to generate pseudo-random numbers sequences and combining them to the target image using the Vernam model. But even if stream ciphers are generally considered to be the fastest class of cryptosystems, they are vulnerable to the known-plaintext attacks unless specific mechanism of key randomization is used, by enabling the use of a different key for each different enciphering operation.

The block ciphering class of cryptosystems has also been addressed using cellular automata theory, and many related schemas have been proposed using the paradigm of reversible cellular automata (RCA). When using a block ciphering schema, the plain-data is considered as a sequence of fixed length blocks, and the enciphering process is performed according to a specific and predefined operation mode such as CBC, CTR or OCB. Each block is ciphered independently, and the result is used as input to encipher the next block in an iterated way. Block ciphers are generally resistant to known-plaintext and chosen plaint-text attacks, and permit to deal perfectly with the redundant nature of digital images since the same blocks are never encrypted in the same way. However, they are generally sequential and iterative (except for the CTR mode that acts like a stream cipher), and are as a result very slow with respect to stream ciphers. Different CA-based block ciphers have been already proposed [11–14] but they

* Tel.: +213 775323650.
E-mail address: kamel_mh@yahoo.fr

http://dx.doi.org/10.1016/j.optlastec.2014.05.012
0030-3992/

generally use a specific operation mode to handle the block encryption's enchainment that is completely different than standardized operation modes. Unfortunately, existing CA-based cryptosystems are almost all sequential and as a result, the parallel implicit nature of CAs is not effectively exploited.

In the present work, we propose a new RCA-based block cipher that has the useful property to be tweakable. Using second-order cellular automatons, a tweakable pseudorandom permutation is constructed, and then used in a parallelizable encryption schema that enciphers each block of a given plain image independently from the others. The proposed system is shown experimentally to be robust against both known-plaintext and chosen-plaintext attacks, unlike stream based CAs approaches, and it provides the advantages of full parallelization and selective data decryption. The special tweak (nonce) is included as a parameter in the construction of the block cipher to handle the electronic code book problem, so that two blocks having the same content are never encrypted in the same way. This technique avoids the need for block dependency of standard block operating modes, and allows a coherent parallelization of the encryption.

The remaining of the paper is organized as follows: in Section 2, a theoretic background about cellular automata and second-order reversible class is first presented. Section 3 details the construction of the proposed cipher, and exposes the parallel image's encryption model. Security analysis and encryption performances results are presented in Sections 4 and 5, and conclusions are finally drawn in Section 6.

2. Elementary and reversible cellular automata

A Cellular Automata consist of a number of cells arranged in a regular lattice; each cell has its own state that can change in a discrete time step. States of the whole CA's cells are updated synchronously using a local transition rule that define each new cell's state using its old state, and the states of the corresponding neighbors. The neighbors are a specific selection of cells relatively chosen with respect to a given cell's position that can be defined for each cell i using a radius r on the lattice. This will give $n=2r+1$ different neighbors including the cell i itself. The boundaries cells of the lattice are concatenated together in a cyclic form to deal with the finite size automatons.

Formally, if we define the state of a cell i at the time t with q_i^t, its state on time $t+1$ will depend only on the states of the corresponding neighborhood at the time t, by applying a transition rule that defines the way states are updated. If the neighborhood radius is r, and only two cell states are defined, the length of each transition rule is then 2^{2r+1} bit, and the number of possible rules is $2^{2^{2r+1}}$. For one dimensional binary CAs, a transition rule is generally coded using the integer value of the corresponding binary representation. In the present work, we consider one-dimensional binary CAs with radius $r=3$, so that we have 2^{128} possible rule.

Unlike elementary cellular automata, a reversible cellular automaton is a specific case of CA in which every configuration has a unique predecessor. That is, RCAs are constructed in such a way that the state of each cell prior to an update can be determined uniquely from the updated states of all the cells. Several methods are known to construct cellular automata rules that are reversible. The second-order cellular automaton method invented by Toffoli and Margolus [15], in which the update rule combines states from two previous steps of the automaton permit to turn any one-dimensional binary rule into a reversible one using the fact that the state of a cell at time t depends not only on its neighborhood at time $t-1$, but also on its state at time $t-2$. This is achieved by combining the ith cell state at time t with the state of the same cell in time $t-2$ using the xor operator.

If we define the configuration state of a given CA at each time step t by C^t, we can build a second-order RCA based on any elementary CA using the following equation:

$$C^t = F(C^{t-1}) \oplus C^{t-2} \tag{1}$$

when the map "F" denotes the global evolution map of the used basic CA. The defined RCA can then be reversed trivially using the following equation:

$$C^{t-2} = F(C^{t-1}) \oplus C^t \tag{2}$$

Second-order RCAs defined using Eq. (2) are always reversible even if the basic used CA defined by the map F is not. We can so construct as much RCAs as possible existing CAs. Reversibility is performed using the same transition rule in both directions, raising qualitatively the same behavior of one-order CAs as pointed by Wolfram [16], which makes the use of such defined RCAs very appropriate for cryptosystems building.

Instead of using one initial configuration like standard one-dimensional CA, two initial configurations are used to evolve a second-order RCA. Starting from two configurations C^{-1} and C^0 it gives after n time step tow configurations C^{n-1} and C^n. By running the RCA backward starting from C^{n-1} and C^n as initial configurations, we can recover the two configurations C^{-1} and C^0 after exactly n iteration using the same transition rule and the same principle of combining with the $(t-2)$th state at each time step t. Security of RCA-based cryptosystems is assured by the impossibility to reconstruct initial conditions pair from any given pair of consecutive configurations without the knowledge of the transition rule used initially.

3. The proposed encryption schema

The proposed block cipher belongs to the symmetric category, so the same secret key K is used by both encryption and decryption process. The key length is 128 bit, in order to ensure a sufficiently large key space robust against exhaustive key search attacks. We propose the construction of a tweakable block cipher defined by a pseudorandom permutation that take a key, a tweak (nonce) and a 256 bit data block as input to produce a ciphered block of the same size as output. We start by detailing the construction of the cipher, and then we present the general parallel image encryption schema. In the following, the terms block ciphers and pseudorandom permutations are synonyms.

3.1. Construction of a tweakable cipher using second-order CAs

Let us first give some basic definitions about permutations and pseudorandom permutations. A function Φ defined on the set of all binary strings of length n into the same set $\Phi:\{0,1\}^n \to \{0,1\}^n$ is said to be a *permutation* if and only if it is a bijection (i.e. Φ^{-1} exist and is efficiently computable).

A family of permutation Φ_k defined by

$$\Phi_k : \mathrm{K}\ x\{0,1\}^n \to \{0,1\}^n$$
$$(k,x) \to y = F(k,x) \tag{3}$$

is said to be a pseudo-random permutation family if it cannot be distinguished from a truly random permutation selected randomly from the set of all permutations on functions domain for any value of k [17]. Given the output of Φ_k and the output of a truly random function, no polynomial algorithm that can distinguish between the two outputs must exit. A pseudorandom permutation family can be considered as a collection of pseudorandom permutations, where a specific permutation may be chosen using a key. In the following, we use the term PRP to refer to any pseudorandom

permutation family Φ_k. Formally, a PRP is said to secure if the advantage of any distinguishing algorithm from a truly random permutation is negligible.

Pseudorandom permutations have been largely studied and analyzed to be used for cryptographic purposes. Almost all block ciphers can be considered as PRPs such as the standards DES or AES. Many PRP's construction algorithms have been proposed in the literature, where the most known and used is the standardized Lubby–Rackoff construction using Feistel networks [18]. Furthermore, a tweakable PRP takes a supplementary parameter as input in addition to the secret key and the plain block, when these parameters permit to control the value of the outputted ciphered block without affecting the secret key [19]. The security of a tweakable cipher always depends on the secret key, and the knowledge of the tweak by an attacker does not provide any additional information about the plain data. In the following, we define a new construction of tweakable PRPs using second-order reversible cellular automate, and we show in later sections that it can achieve very promising performances competitive to those of the standard ones.

Let us define the function Φ that transforms an input plaintext block B_i of 256 bit size into a ciphered block CB_i of same size. We extend the definition of a PRP to take three parameters as input instead of two in the standard definition. The function Φ takes a key K (the secret key of encryption), a tweak n_i (specific for each block B_i) and the block B_i to produce the ciphered block CB_i. If we note by $\{0,1\}^n$ the set of possible binary stings with size n, the function Φ can be defined by

$$\Phi : \{0,1\}^{128} x \{0,1\}^{32} x \{0,1\}^{256} \rightarrow \{0,1\}^{256}$$
$$(K, n_i, B_i) \rightarrow CB_i = F(K, n_i, B_i) \quad (4)$$

The second parameter n_i is introduced to prevent the ECB encryption problem, such that the same plain blocks are never ciphered in the same way since n_is are different and specific to each block B_i and never repeat for the same key. The parameter n_i is named tweak of the block cipher and can be defined by a nonce or a random initialization vector. We have considered in the present work that the value of n_i is simply the order i of the block B_i represented on 32 bit, so that we can deal with 2^{32} different blocks of the same plain images, and be able to encipher images of size $32*2^{32}=2^{37}$ byte, which is largely sufficient to encipher large color digital images.

As explained in Section 2, second order RCAs start evolving from two different configurations: an initial configuration C^0 and a pre-initial configuration C^{-1}, to give after m consecutive iteration two corresponding configurations C^{m-1} and C^m. This mechanism is used to build the proposed PRP that acts like the following: first, the plain block B_i is split into two 128 bit sub-blocks BL_i and BH_i (standing for the low and high order parts of B_i). The two sub-blocks then combine each one using a xor with a sub-key Sk_i derived by altering the key K using the nonce n_i. This alteration is performed in a simple but effective way illustrated in Fig. 1. After the xor combination, the resulting pair of configurations (the two sub-blocks) undergoes five rounds of 8 iterations each. At the end of each round, the two resulting configurations are exchanged and used as input of the next round. During each round, the input configurations are evolved using a different transition rule. The first, third and fifth rounds are performed using the master key K as transition rule, when the second and fourth rounds use the sub-key Sk_i as transition rule. Totally, 40 iterations are performed, and at the last, the resulting configuration C^{39} and C^{40} are combined newly with the sub-key Sk_i using a xor operator, then concatenated to form the final ciphered block CB_i.

Since second-order RCA's initial conditions are very sensitive to variations, only 1 bit modification of the key is sufficient to produce a completely different evolution behavior. This fact ensures that even if the sub-key derivation process is trivial, it ensures the modification of at least 1 bit of the key K when using any nonce n_i and by the way all sub-keys Sk_i are different and will produce a completely different behavior when used to encipher different plaintext blocks B_i that have the same content.

Decryption using the proposed schema is performed using exactly the same steps and the same parameters. If the input of the function is the key K, the nonce n_i and the ciphered block CB_i, the output will be automatically the plain block B_i. This can be considered as a great advantage, since this will permit to use the same hardware circuits for both encryption and decryption if hardware implementation is used and the same programming code of software implementation holds.

The proposed PRP is experimentally shown to be indistinguishable from a random permutation, and very sensitive to small variations of each parameter B_i, n_i and the key K. These two properties make it enough secure and suitable for cryptographic applications. Analysis of the schema with corresponding experiments and results is presented in the next sections.

3.2. The parallel images encryption schema

Unlike most existing image encryption schemas, the proposed approach is completely parallel and there is no need for multiple sequential iterations to ensure a complete avalanche criterion satisfaction. Input color or gray-scale plain image is considered as a set of independents 256 bit blocks that can be ciphered independently from one another using the proposed PRP. Blocks can be ciphered by groups or independently depending on the parallelism level of the used platform, and the number of used processors or threads.

The input image to be enciphered is first decomposed in equal length 256 bit blocks $B_0,\ldots,B_L$, when a padding schema can be used if the size of the image is not multiple of 256. Each block B_i is transmitted to the PRP Φ with the secret key K and its corresponding range i used as a tweak. The output of Φ will be the corresponding ciphered block $CB_i=\Phi(K,i,B_i)$. After all plain blocks are processed, corresponding ciphered block $CB_0,\ldots,CB_L$ are combined using the same ranges to form the final ciphered image. Since enciphering of each block is independent from the others, it is clear that the task can be fully parallelized. Fig. 2 illustrates the proposed image enciphering process. When the goal is to encipher high resolution color digital images, we propose to use a specific way to convert a plain color image into a set of contiguous 256 bit plain blocks. Color images are represented by three distinct channels of colors: the red channel, the green channel and the blue channel. Instead of ciphering each channel apart, a channel mixing step is used to mix data from different channels and hence provide further confusion aspect in the resulting ciphered image. Each pixel is represented in a given channel by 1 byte, so we collect them one by one from one channel to the other using the following mixing mechanism: the first byte is the blue one followed by the green and the red bytes respectively, and then we return to take the second blue byte followed by the second green and red bytes. This collection mechanism is continued until an array of bytes containing the entire image's data is constructed, and finally this resulting array is segmented into contiguous 256 bit blocks (8 byte per block). This mechanism has demonstrated great enhancement of the ciphering performances with respect to directly fragmenting the image into plain blocks.

Since the proposed PRP is self-invertible ($\Phi=\Phi^{-1}$), the decryption process is performed in the same way. The enciphered image is inputted to the same system using the same key, and the output will be imperatively the original plain image after rearranging the pixels by an inverse channel mixing mechanism.

It is important to note that plain blocks having the same content can never be ciphered in the same way since the tweak

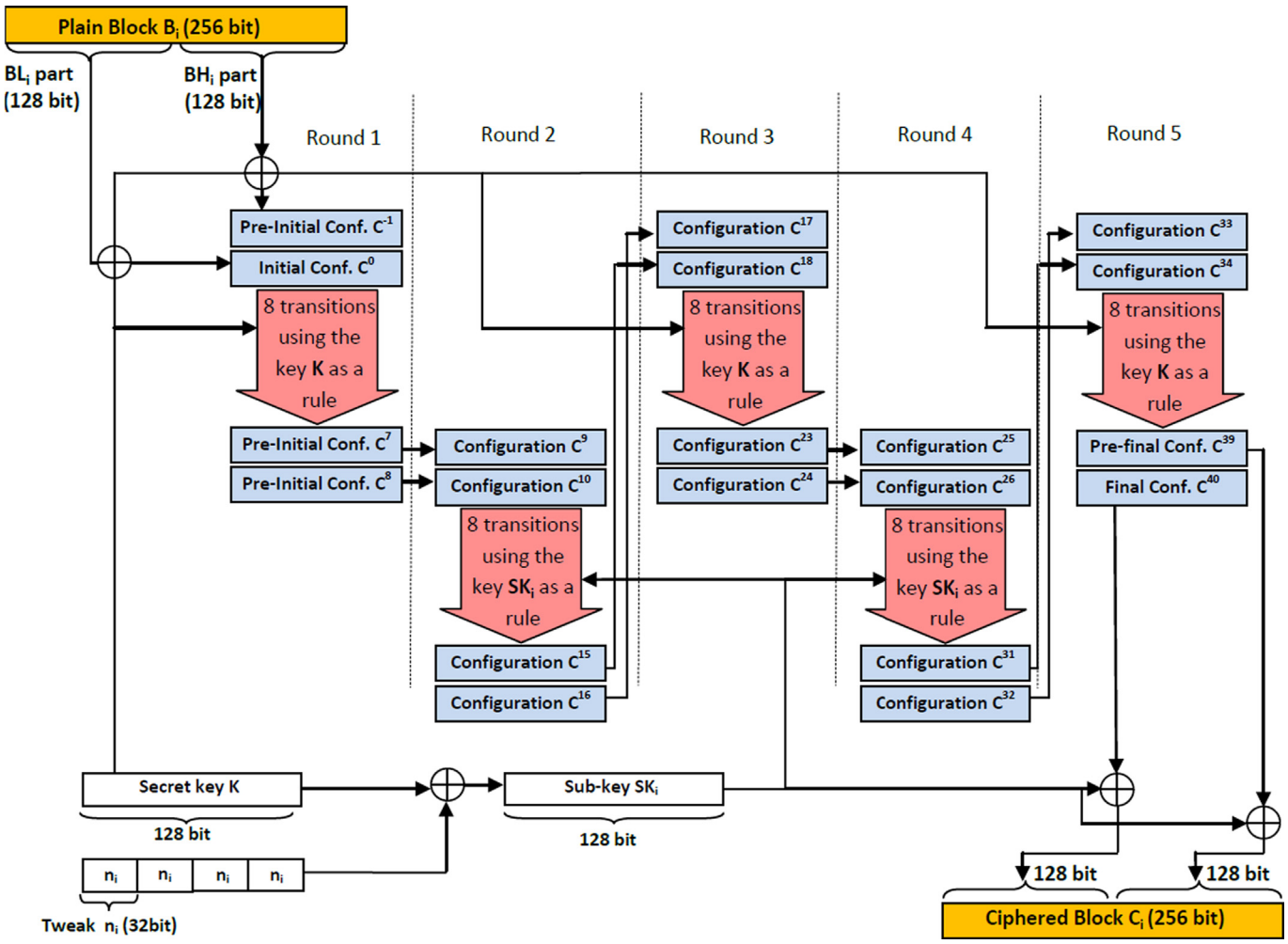

Fig. 1. Proposed RCA's based tweakable cipher for encrypting/decrypting a single 256 bit block.

is different and never repeated for the same image of size lower than 2^{37} byte. The introduction of the tweak in the proposed PRP is provided to solve the ECB enciphering problem and remove the need for sequential block enchaining or iterative confusion/diffusion that are time consuming. The security and efficacy of the proposed schema are based upon the extreme sensitivity of the proposed PRP to any small variation of the nonce value, which is proven experimentally in the following section.

4. Security analysis and experimental results

In this section, several experiments and tests are presented to evaluate the security and robustness of the proposed approach. We first analyze the security aspects of the proposed tweakable block cipher, with respect to the randomness and sensitivity criterions. Then, we evaluate the security of the global image's enciphering schema using several statistical tests, including its robustness against major cryptanalysis attacks classes.

4.1. Analysis of the cipher's performances

As stated in Section 3.1, a secure block cipher must satisfy two main criterions: to be indistinguishable from a random permutation, and to be very sensitive to small variations of its inputs. The former criterion is shown experimentally by evaluating the pseudo-randomness degrees of the output during enciphering of plain images, when the later criterion is demonstrated by evaluating the sensitivity of output with respect to small variations of the three inputs: plain block, nonce and the secret key.

The sensitivity to the cipher's plain blocks variation can be evaluated using the strict avalanche criterion [23] that abstracts the non-linearity of the cipher. A block cipher defined by the function Φ satisfies the strict avalanche criterion if the distribution of bit-difference between two ciphered blocks that correspond to two plain blocks differing only on 1 bit, must follow a binomial distribution B(½,n). Mathematically, if H denotes the Hamming distance between two binary blocks, then this criterion is described by the following:

$$\forall x, y \in \{0,1\}^{256} : H(x,y) = 1 \Rightarrow H(\Phi(x), \Phi(y)) \approx \mathrm{B}\left(\frac{1}{2}, n\right) \quad (5)$$

The criterion can be verified by measuring the amount of proximity between the experimental distribution computed for the block cipher using a sufficiently large samples set, and the theoretic binomial distribution using the χ^2 goodness-of-fit tests that will express the degree of non-linearity of the block cipher defined by the function A given cipher is highly non-linear if the value of χ^2 is close to 0, when such property is less verified for high values of χ^2. We measured an experimental distribution of the set of Hamming distances obtained using a set of randomly selected plain blocks, ciphered each time with fixed keys and tweaks. The obtained experimental distribution is illustrated in Fig. 3 in comparison with the expected theoretic curve. The χ^2 obtained

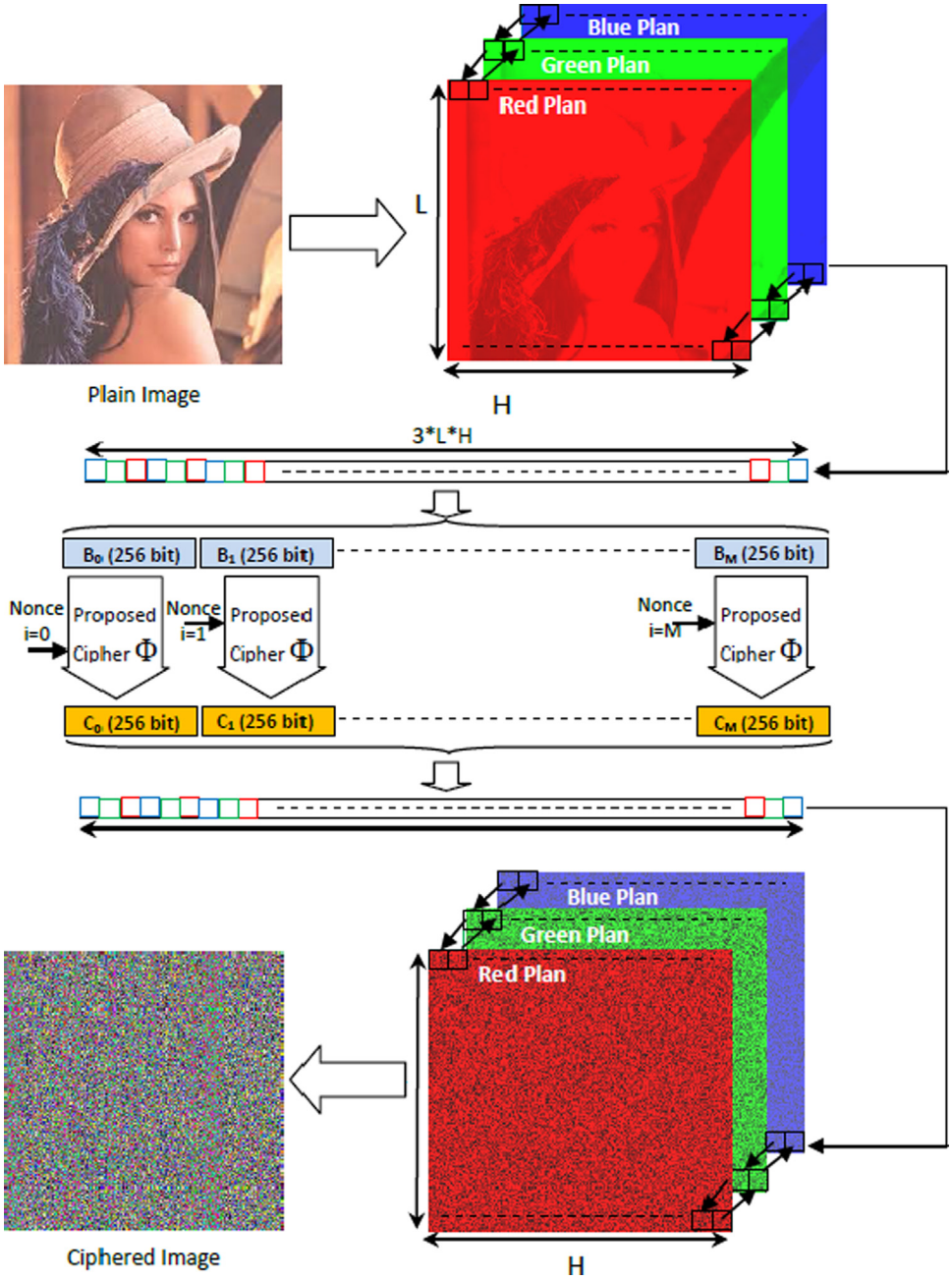

Fig. 2. Illustration of then color image encryption schema.

value for this experiment is equal to 0.02548, which imply that the Hamming distribution is exactly flowing the distribution B(½,256), and hence the avalanche criterion is satisfied by the proposed cipher.

Sensitivity to the key variation is measured by comparing the cipher's outputs (in term of percentage of different bits) when using a fixed plain block and a fixed tweak values with the 256 possible different one-bit-modified copies of a given key, and taking the averaged result on 10^6 such experiments that are performed using a set of 10^6 randomly generated keys. Same experiment is performed to measure the sensitivity to the tweak with a fixed plain block and a fixed key (using 10^6 randomly generated tweak, and measuring bit differences with the 32 possible one-bit-modified copies). Obtained results for these experiments are illustrated in Fig. 4(a) and (b). It is clear that averaged bit difference is always close to the optimal value (50%) for the three inputs which proof that the cipher's output is very sensitive to any elementary 1 bit modification of any input. This characteristic ensures the robustness of the proposed schema against both linear and differential cryptanalysis, and guarantees that avalanche criterion is perfectly satisfied for all cipher's inputs.

4.2. Security analysis of the proposed image's parallel cryptosystem

In order to evaluate performances and security of the proposed image's cryptosystem, different statistical tests and measurements are performed using three 1024×1024 color images: Lena, Boat and a specific image depicting a text (image of a document) are illustrated respectively in Fig. 5(a), (b) and (c). Corresponding ciphered images using a 128 bit random key are presented in Fig. 5(e), (f) and (g). The image of the text in Fig. 5(c) has been enciphered in the gray-scale mode, since it contains only black and white pixels. It is clear that even with quasi uniform aspect of the text image, the resulting ciphered image is completely confused

and randomized. The following sections illustrate different obtained results with respect to several security aspects.

4.2.1. Information entropy and image correlation

According to Shannon's theory, information entropy is one of the main randomness measurements of information. High entropy values express a high degree of randomness and for any message coded on m bit, the upper bound of the entropy is m. Since color images are coded on 24 bit/pixel (8 bits for each color channel), the optimal entropy value for each color is 8, and the entropy of ideally random image should be very close to this bound. The entropy is calculated using the following formula:

$$H = -\sum_{i=0}^{2m-1} p_i \mathrm{Log}_2(p_i) \tag{6}$$

when p_i is the probability distribution of different color components values of an image (from 0 to 255 for the red, green and the blue each) that can be approximated by their frequencies.

Table 1 illustrates different entropy values obtained for plain and ciphered images with respect to the three different channels of colors. The results are also compared to those obtained by two existing CA-based and chaos-based cryptosystems proposed in [20] and [21]. It is clear that ciphered images have near to optimal entropies, and as a result, provides good randomness properties, that prevent any statistical cryptanalysis attacks to gain useful information from the ciphered image.

Another important statistical test that permits to show the high quality of the diffusion and confusion properties of the proposed cryptosystem is the correlation measure among ciphered images pixels. Since digital images have generally redundant content, they present a strong correlation between adjacent pixels unlike ciphered images that should have a near to zero correlation to avoid any possible information deduction that leads to a possible statistical attack. To perform a pixel's correlation test on an image, a set of 20,000 random pairs of adjacent pixel is chosen (in vertical, diagonal and horizontal directions) and the correlation coefficient is then calculated and plotted in a correlation diagram using the formula stated in [20]. As an example, Fig. 6 illustrates the correlation distribution of horizontally, vertically and diagonally adjacent pixels for the plain and ciphered versions of the Lena image. Table 2 lists the corresponding correlation values calculated for the three used images Boat, Lena and the Text image.

4.2.2. Histogram analysis

The information given by an image histogram represents the statistical distribution of pixels values. Enciphered images must be similar to random ones and lead to a pseudo-uniform distribution (uniform histogram), unlike plain images that have irregular distributions depending on the image content. We can see from Fig. 7 that histogram of the three ciphered images is uniform with respect to the three RGB color: red, green and blue channels. Consequently, no statistical attack can reveal any information about the plain image without knowledge of the secret key. Note that the Text image (c) contains only black and white pixels and by the way the corresponding histogram represents only the number of black and white pixels, while the corresponding ciphered image is a gray-scale random one according to its histogram.

4.2.3. Sensitivity to key variations

An important security aspect of any cryptosystems is to be resistant against differential and linear attacks. Such aspect is satisfied if the encryption result is very sensitive to small elementary variations of the used secret key. To evaluate the key sensitivity degree of the proposed approach, the following experiment is performed: for a given plain image, enciphering using a random key K is first performed to obtain a reference ciphered image. Then, 128 one-bit modifications are performed on each of the 128 different bits of K followed by enciphering of the plain image using the resulting modified key. We can then compute the

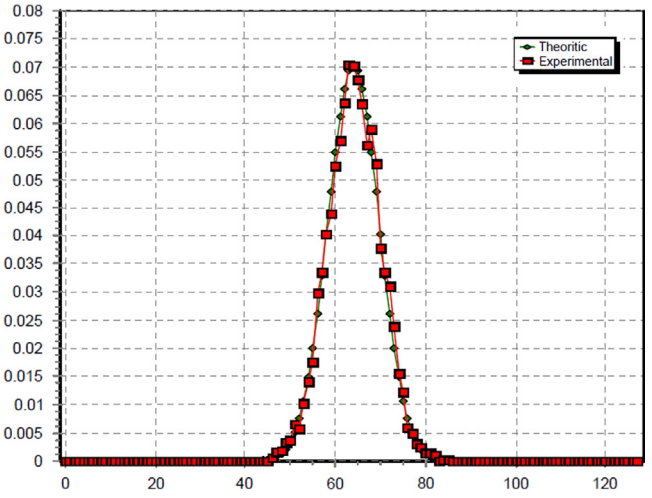

Fig. 3. Theoretic vs. experimental curve of the cipher's output distribution.

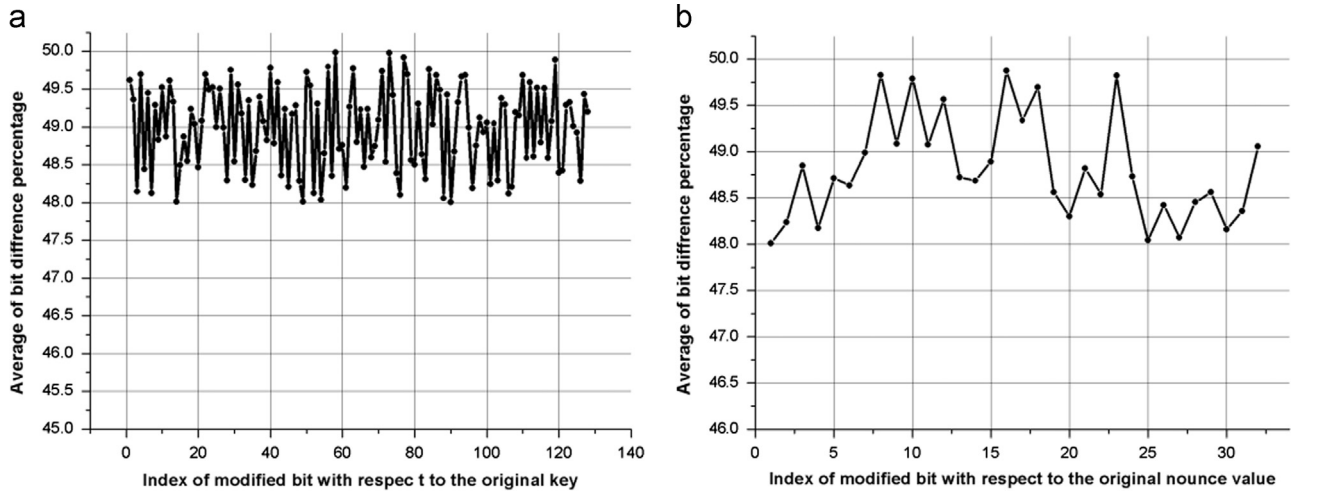

Fig. 4. Sensitivity of the proposed cipher to elementary input's changes: (a) key sensitivity and (b) tweak sensitivity.

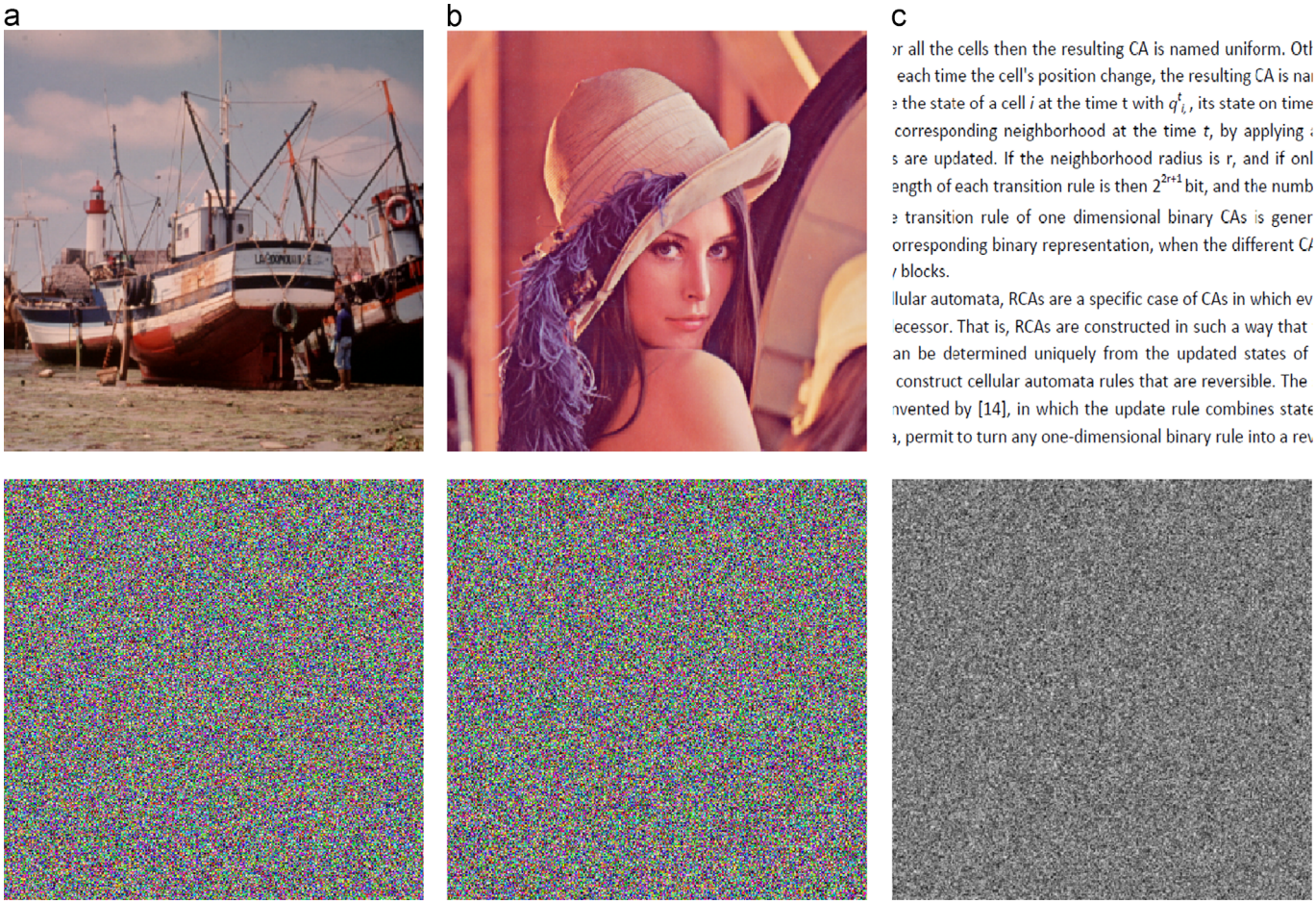

Fig. 5. Images used for analyzing the cryptosystem: (a) Boat, (b) Lena and (c) Text, with corresponding ciphered images (e), (f) and (g) respectively.

Table 1
Entropy of plain/cipher images for the proposed and existing cryptosystems.

Image	Plain image			Ciphered image			Ref. [20] (ciphered)	Ref. [21] (ciphered)
	Red	Green	Blue	Red	Green	Blue		
Boat	7.2363	7.1025	7.0587	7.9996	7.9999	7.9989	7.9368	7.9997
Lena	7.1425	7.1782	7.1325	7.9899	7.9997	7.9979	7.9643	7.9957
Text	7.0725	7.0914	7.0102	7.9987	7.9889	7.9899	7.9487	7.9961

percentage of difference between the resulting ciphered image and the reference ciphered image obtained using the original key K. Computation of the difference percentage can be performed (on 512×512 images) using the following equation:

$$\text{diff} = \left(\frac{1}{512*512} \sum_{i=1}^{512} \sum_{j=1}^{512} sg(C[i,j] - C'[i,j]) \right) * 100 \quad (7)$$

when C is the reference ciphered image, C′ is the resulting ciphered image (using the modified key), and *sg* is the function defined by

$$sg(x) = \begin{cases} 1 & \text{if } x \neq 0 \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

This experiment was performed using 1000 different random keys, and the averaged results are illustrated by Fig. 8(a). We can easily note that difference rates are very high for each bit position modification and so the enciphering key is very sensitive to modifications. Another way to show key sensitivity is to compare the deciphered image using a wrong key (that differs only on 1 bit to the correct one) with the correct deciphered key and compute the percentage of difference between the correct and the wrong decrypted images. Results of these experiments using Lena image are presented in Fig. 8(b) and proof that decryption is also very sensitive to small key variations. We note that in both encryption and decryption sensitivity's experiments, the difference percentage is higher than 99%; hence the content of the whole enciphered image changed is only 1 bit of the flipped key. Such behavior demonstrates the high non-linearity of the cipher, and implies that even knowledge of any part of the key will not give any advantage to an attacker since no useful information about the plain image can be derived.

4.2.4. Statistical distribution of the cipher's output

A good encryption scheme should have excellent performance in resisting statistical attack. To achieve such property, the generated ciphered data from any given plain image having any statistical distribution must be indistinguishable from a random sequence to insure security against chosen plain-text attacks. The

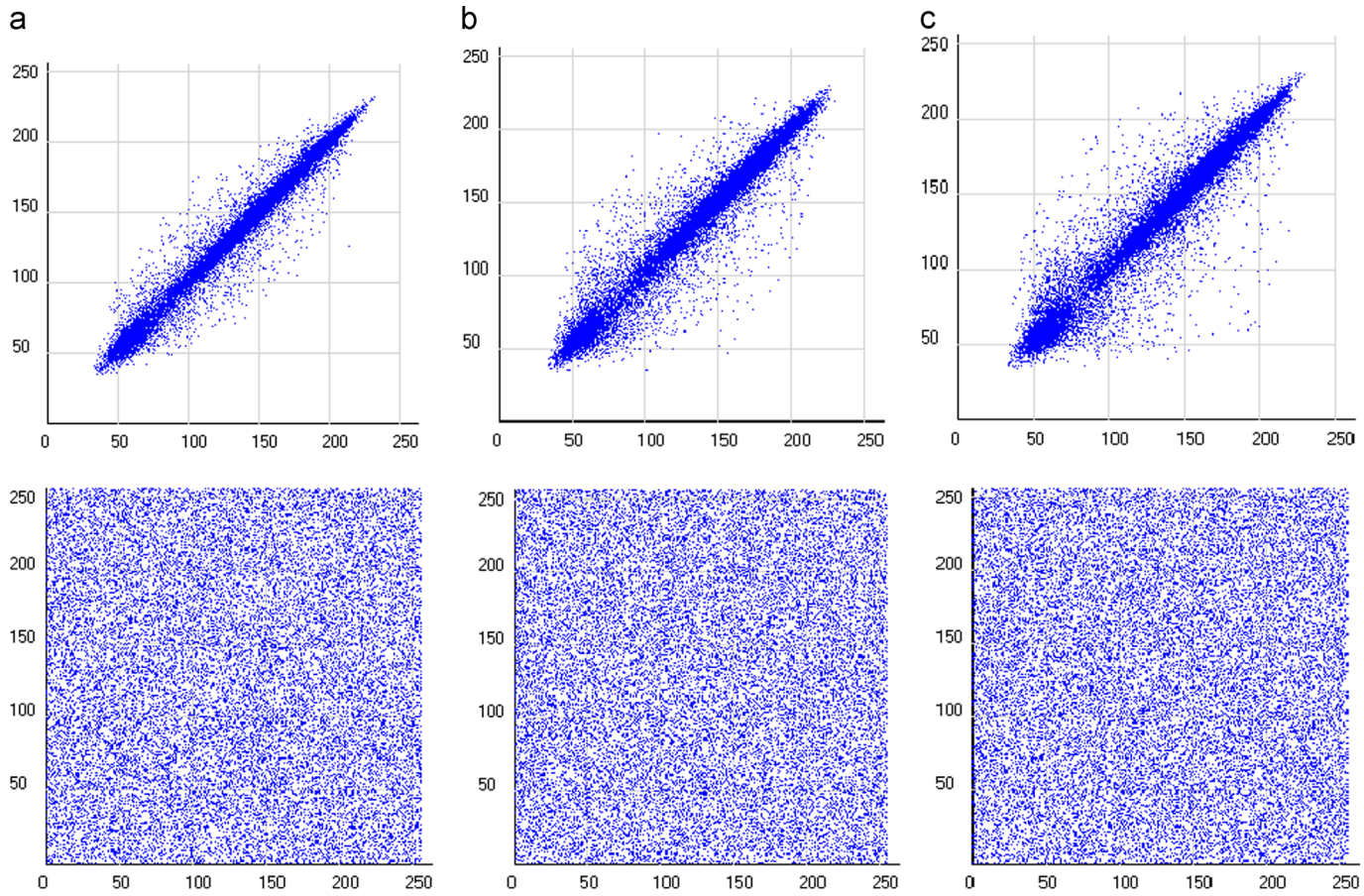

Fig. 6. Correlation distribution for plain/ciphered Lena image: (a) horizontal, (b) vertical and (c) diagonal.

Table 2
Correlation coefficients of adjacent pixels of different images.

Image	Plain image			Ciphered image		
	Horizontal	Vertical	Diagonal	Horizontal	Vertical	Diagonal
Boat	0.9789	0.9721	0.9489	0.0012	0.0031	−0.0041
Lena	0.9701	0.9791	0.9502	0.0039	0.0023	0.0069
Text	0.9752	0.9808	0.9531	0.0031	−0.0060	0.0027

indistinguishability of the sequences produced by the proposed cryptosystem is shown experimentally by generating a number of different ciphered images corresponding to a set of plain images with different types and sizes. Each obtained ciphered stream is analyzed using the Diehard statistical Tests batteries [22] to evaluate the corresponding randomness degree. The averaged *P*-value for each test is taken over 200 different generated ciphered images, and the results are reported in Table 3. It is clear from the obtained results that the cryptosystem's outputs pass all statistical tests provided by Diehard suits, and by the way mean that ciphered images cannot be distinguishable from uniform random sequences and security of the cryptosystem is then assured against any statistical chosen plain text attacks.

5. Performances analysis and comparison

As mentioned above, the main advantage of the proposed parallel approach is to permit very high encryption/decryption rates with respect to existing sequential models due to parallelized nature of the schema. We have implemented the cryptosystem using MMX assembly instruction on a Delphi 6 programming environment and experimenting using an i7-2600 3.40 GHz platform. Multi-threading model is used to exploit the inherent parallelism by decomposing the plain images in different sets of blocks that are ciphered independently by different threads. The resulting performances outperform almost all sequential approaches, like illustrated by results reported in Table 4 comparing the proposed schema to chaotic confusion/diffusion approach [23], existing CA-based approach [23], block-based AES in CBC and CTR operating modes and the A5/1 algorithm.

We note that obtained encryption/decryption rates depend on the used platform and the number of possible threads and processors. Fig. 9 illustrates the evolution of encryption/decryption time with respect to the number of used threads for a given 3000×3000 image. Upper bounds of the encryption speed are only limited by the platform characteristics and the number of possible computation units. If hardware implementation is used, an additional level of parallelism can be exploited since cellular automatons can run asynchronously for each block which will lead to a further enhancement of the overall encryption/decryption performances.

6. Conclusions

The present paper presents a new parallelizable image encryption schema using a specific cellular automata-based tweakable block cipher. Enciphering/deciphering of any color plain image can be run in parallel using multiple threads/processors without affecting the security and coherence of the cryptosystem. The parallelism has been made possible by defining an extended

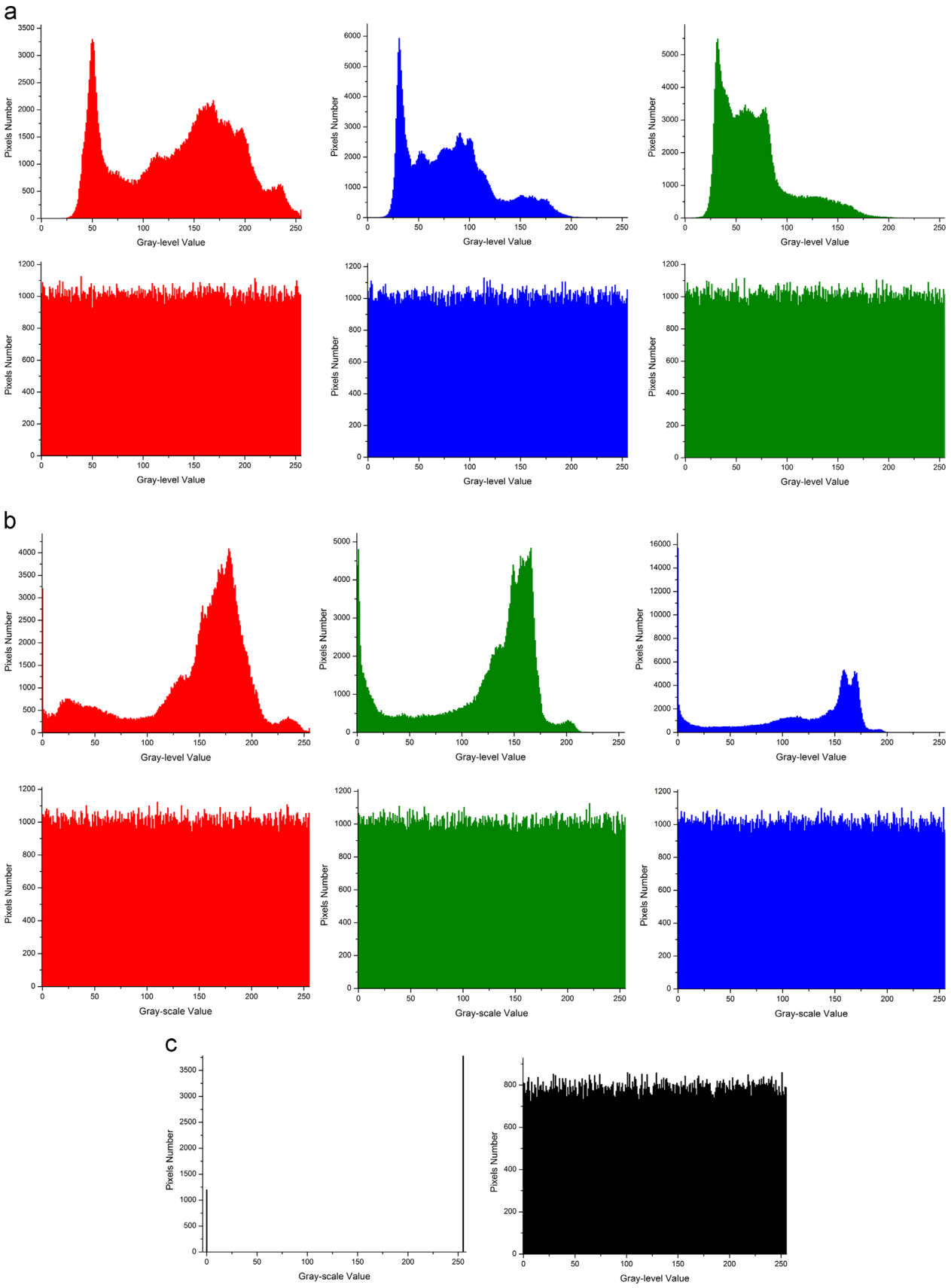

Fig. 7. Histograms of plain/ciphered images: (a) Lena, (b) Boat and (c) Text.

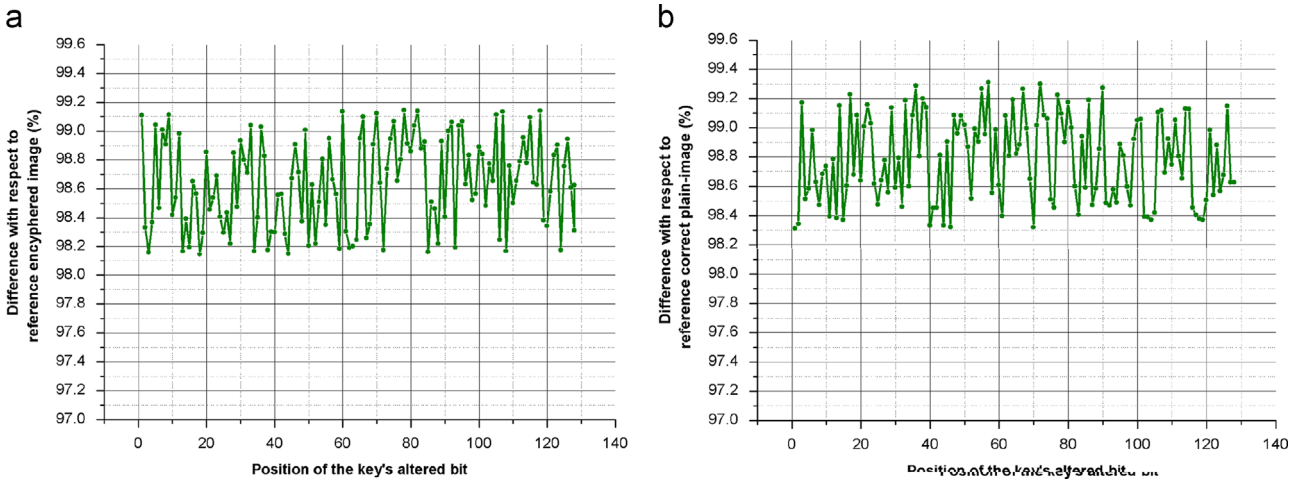

Fig. 8. Key sensitivity to 1-bit modifications: (a) encryption sensitivity; (b) decryption sensitivity.

Table 3
Averaged results obtained with Diehard Test using 200 different ciphered images.

Test name	Averaged *P*-value	Result
Birthday spacing	0.815606	Pass
Overlapping permutation	0.956312	Pass
Rank test 31 × 31	0.798521	Pass
Rank test 32 × 32	0.634508	Pass
Monkey DNA	0.685215	Pass
Count-the-1's test for specific bytes	0.501248	Pass
Parking lot	0.254896	Pass
Minimum distance	0.425142	Pass
Random sphere	0.398745	Pass
The squeeze test	0.781711	Pass
Overlapping sums	0.5012483	Pass
The runs—up test, down test	0.921252, 0.425681	Pass
Craps—no of wins, throws/game	0.892014, 0.592042	Pass
Rank test	0.915874	Pass
Monkey 20 bit per word	0.878520	Pass
Monkey OPSO, OQSO	0.953324	Pass

Table 4
Encryption time performances comparison for different image sizes.

Plain-image size	Encryption time (in ms)					
	Ref. [20]	Ref. [22]	AES(CBC)	AES(CTR)	A5/1	**Proposed**
512 × 512	1785	3521	1135	963	1236	758
1024 × 1024	7154	13968	4568	3901	4423	3097
2048 × 2048	14521	28324	9321	7796	8924	6491

tweakable definition of pseudorandom permutation using reversible second-order automata mechanism. The PRP can be applied independently on each image block such that encryption of any block relay only on its content and its index in the image used as ciphering tweak without the need for anterior enciphered block information. The security of the proposed approach is induced by the dynamical and chaotic behavior of RCAs, and their high sensitivity to small initial conditions and evolution key variations.

The two main advantages of the proposed approach are first the parallel mode of operation for encryption and decryption that permits to achieve high performances when using multi-processor platforms, and second, the selective area deciphering such that any specific image's area can be deciphered without knowledge of the full ciphered image. The proposed approach is very robust to data deterioration or noisy transmission, since any ciphered data corruption will affect only the corrupted block without influencing the decryption result of prior or posterior blocks. Obtained results

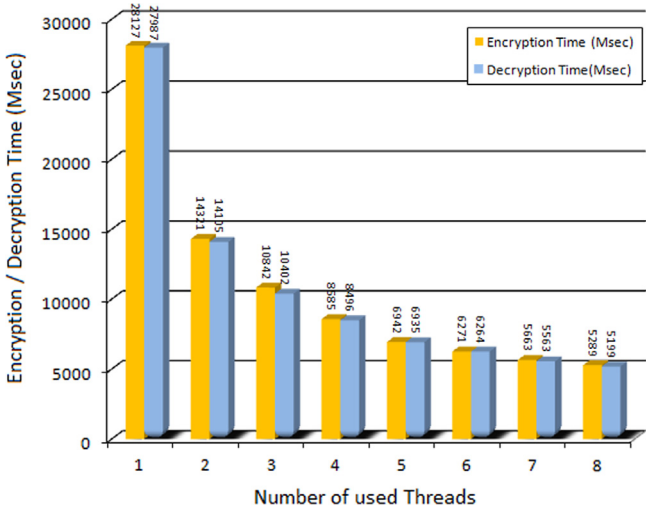

Fig. 9. Encryption/decryption time with respect to the number of threads (on an 8 processors machine).

show the robustness and high performance degree of the proposed schema even with a non-optimized code. We assume that better performances can be achieved if hardware implementation is used.

References

[1] Wang X, Zhao J, Liu H. A new image encryption algorithm based on chaos. Opt Commun 2012;285:562–6.
[2] Francois M, Grosges T, Barchiesi D, Erra R. A new image encryption scheme based on a chaotic function. Signal Process: Image Commun 2012;27:249–59.
[3] Kanso A, Ghebleh M. A novel image encryption algorithm based on a 3D chaotic map. Commun Nonlinear Sci Numer Simul 2012;17(7):2943–59.
[4] Ye RS. A novel chaos-based image encryption scheme with an efficient permutation-diffusion mechanism. Opt Commun 2011;284(22):5290–8.
[5] Shen J, Jin X, Zhou C. A color image encryption algorithm based on magic cube transformation and modular arithmetic operation. Lect Notes Comput Sci 2005;3768:270–80.
[6] He X, Zhu Q, Gu P. A new chaos-based encryption method for color image. Lect Notes Artif Int 2006;4062:671–8.
[7] Wolfram S. Random sequence generation by cellular automata. Adv Appl Math 1986;7(2):123–69.
[8] Szaban, M, Seredynski, F, and Bouvry, P. Collective behavior of rules for cellular automatabased stream ciphers, evolutionary computation. In: Proceedings of the CEC. IEEE Congress: Jul 16–21, 2006. pp. 179–183.
[9] Chatzichristofis Savvas A, Mitzias Dimitris A, Sirakoulis Georgios Ch, Boutalis Yiannis S. Novel cellular automata based technique for visual multimedia content encryption. Opt Commun 2010;283(21):4250–60.

[10] Tomassini M, Sipper M, Perrenoud M. On the generation of high quality random numbers by two-dimensional cellular automata. IEEE Trans Comput 2000;49(10):1146–51.
[11] Seredynski M, Bouvary P. Block cipher based on reversible cellular automata. New Gener Comput 2005;23:245–58 (Ohmsha Ltd and Springer).
[12] Anghelescu Peter, Ionita Silviu, Safron Emil. Block encryption using hybrid additive cellular automata. In: Proceedings of the 7th International conference on Hybrid Intelligent Systems, IEEE- 2007.
[13] Ray A, Das D. Encryption algorithm for block ciphers based on programmable cellular automata. Inf Process Manag 2010:269–75.
[14] Chen RJ, Lai JL. Image security system using recursive cellular automata substitution. Pattern Recognit 2007;40(5):1621–31.
[15] Toffoli T T, Margolus N. Invertible cellular automata: a review. Physica D 2001;45:229–53.
[16] Wofram S. A new kind of science. Wolfram media: Champaign, United Kingdom; 2002. p. 437–40. ISBN 1-57955-008-8.
[17] BellareM, RogawayP. Chapter 3: Pseudorandom functions. Introduction to modern cryptography. Retrieved 2007.
[18] Luby M, Rackoff C. How to construct pseudorandom permutations from pseudorandom functions. SIAM J Comput 1988;17(2):373–86 (April).
[19] Liskov M, Rivest RL, Wagner D. Tweakable block ciphers. In advances in cryptology—CRYPTO 2002. Berlin Heidelberg: Springer; 2002; 31–46.
[20] Abdo AA, Lian SG, Ismail IA, Amin M, Diab H. A cryptosystem based on elementary cellular automata. Commun Nonlinear Sci Numer Simul 2013;18 (1):136–47.
[21] Kanso A, Ghebleh M. A novel image encryption algorithm based on a 3D chaotic map. Commun Nonlinear Sci Numer Simul 2012;17(7):2943–59.
[22] Soto J. Statistical testing of random number generators. In: Proceedings of the 22nd national information systems security conference. Crystal City (Virginia): October 1999.
[23] Chen G, Mao YB, Chui CK. A symmetric image encryption scheme based on 3D chaotic cat maps. Chaos Solitons Fractals 2004;12:749–61.