Received March 16, 2019, accepted April 7, 2019, date of publication April 16, 2019, date of current version June 4, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2911559

Fourth Order MCA and Chaos-Based Image Encryption Scheme

MD NAZISH ASLAM[1], AKRAM BELAZI[2], SOFIANE KHARBECH[3], (Member, IEEE), MUHAMMAD TALHA[4], (Member, IEEE), AND WEI XIANG[5,6], (Senior Member, IEEE)

[1]Piro Technologies Pvt. Ltd., New Delhi 110025, India
[2]RISC Laboratory, National Engineering School of Tunis, Tunis El Manar University, Tunis 1002, Tunisia
[3]Laboratory Sys'Com-ENIT (LR-99-ES21), Tunis El Manar University, Tunis 1002, Tunisia
[4]Deanship of Scientific Research, King Saud University, Riyadh 11543, Saudi Arabia
[5]College of Science and Engineering, James Cook University, Cairns, QLD 4878, Australia
[6]Peng Cheng Laboratory, Shenzhen 518000, China

Corresponding author: Wei Xiang (wei.xiang@jcu.edu.au)

This work was supported in part by the Stereoscopic Coverage Communication Network Verification Platform for China Sea Project under Grant PCL2018KP002, and in part by the Deanship of Scientific Research, King Saud University, Riyadh, Saudi Arabia.

ABSTRACT This paper presents a fast and efficient cryptosystem for enciphering digital images. It employs two of the most prominent dynamical systems-chaotic maps and cellular automata. The key streams in the proposed encryption scheme are derived from the SHA-256 hash function. Hash functions produce the digest of the input plaintext, known as a hash value, which can be considered as a unique signature of the input. This makes the keys more plaintext dependent, which is a desirable property of a robust cryptosystem. These key streams are used as the secret keys (i.e., initial conditions and control parameters) of an improved one-dimensional (1-D) chaotic map, i.e., the Logistic-Sine map. As far as we know, this paper is a first that combines the well-known diffusion-confusion architecture and the fourth order 1-D memory cellular automata (MCA) for image encryption. First, a pixel-wise XOR operation is applied to the original image, followed by a pixel-wise random permutation. The resulting image is decomposed into four blocks according to the quadtree decomposition strategy. Then, a fourth order reversible MCA is applied, the blocks obtained from the quadtree decomposition are considered as the initial MCA configurations, and the transition rules are determined using the chaotic map. The performance analyses show that the proposed encryption scheme presents a high immunity against all kind of attacks while maintaining a low complexity, which outcome a notably better performance/complexity trade-off compared to some recently proposed image schemes.

INDEX TERMS Logistic-Sine map, chaos, cellular automata, image encryption, quadtree decomposition, hash functions.

I. INTRODUCTION

In recent years, with the unprecedented advancement in technology and aggressive expansion in internet usage, the amount of data generated and its distribution over various networks has increased remarkably. As the number of people using the internet has passed four billion marks in 2018, it is obvious to observe a surge in both the generation and the distribution of multimedia data, especially digital images. However, the security, integrity, and authenticity of this huge amount of digital data have become a crucial challenge for both the users as well as the organizations dealing with it. In particular, sensitive information like confidential military and medical images are at high risk of being intercepted by intruders. To meet these security challenges, development of robust cryptographic schemes is quite essential. These schemes help secure communication of data between the sender and the authorized receivers having common security keys. Apart from the security aspects, high computational speed is also a desirable criterion for an efficient cryptosystem to be incorporated with a real-time communication.

Multimedia data like high-quality digital images are too bulky regarding data size, therefore, cryptographic schemes like DES, 3DES, and AES are unable to meet the requirements of high computational speed. Besides, image encryption schemes follow two main categories: modular arithmetic-based algorithms (like most of the algorithms which rely on the substitution-permutation-diffusion architecture [1]–[3]) and algorithms entirely based on an analytical mathematics approach (e.g., algebraic methods and

The associate editor coordinating the review of this manuscript and approving it for publication was Antonio J. Plaza.

methods derived from number theory). There is no doubt on the robustness of many of algorithms modeled wholly by mathematical approaches, but most of the recently developed ones fit the category of modular arithmetic due to its lower computational complexity [4]. Encryption schemes associated with modular arithmetic are introduced to offer alternatives which give a mathematical approaches-like performance while maintaining a decreased complexity. In this context, we propose a new image encryption scheme, based on modular arithmetic with one-time key generation [5], to boost the ratio performance/complexity.

A wide range of encryption algorithms that belong to the modular arithmetic class employs dynamical systems. Cryptosystems based on dynamical systems are gaining more attention as they surpass the issues of security, authenticity and computational speed.

Chaotic maps are one of the most promising dynamical systems to design cryptosystems because the discrete dynamic chaotic maps have an inherent property of hiding relationship between initial and final states [6]. This unpredictable nature of chaotic map components makes it more suitable for constructing image cryptosystems. Confusion and diffusion are two basic and important phases of a chaos-based system. Confusion withholds any correspondence between the key and the ciphertext, and it is usually achieved by permutation of pixels using chaotic maps without making any changes in the pixel values. Diffusion, on the other hand, ensures that changing any pixel in the original image leads to the change in multiple pixels of the cipher image. Owing to its significant relevance in the design of image cryptosystems, researchers have proposed many variants and improved algorithms to encrypt digital images using chaotic maps including [7]–[14]. In chaos-based image encryption, many confusion and diffusion techniques are reported in the literature such as DNA-based encryption [2], dynamic random growth technique [15], non-adjacent coupled map lattices system [16], etc. A new image encryption algorithm called the Tent-Logistic map-based Data Encryption Algorithm (TL-DEA) is reported in [14]. It adopts the substitution-permutation network that operates on the pixels, under two rounds. Firstly, the image is decomposed into fixed-length data blocks. Then, substitution and permutation processes are carried out on these blocks using the Tent-Logistic map.

Cellular automata (CA) are another essential dynamical system which has gained much attention from researchers in recent years. Efficient modeling of various physical, biological, and mathematical systems was performed using CA simulation. Stephen Wolfram, who for the first time, introduced the concept of cryptography using CA in 1985 [17]. Since then, many research work in this direction were presented including [18]–[21], which mainly aims at designing cryptosystems for encrypting digital images using CA. However, more robust and efficient cryptographic schemes were achieved using a combination of chaotic systems and CA [22]–[24]. [25] used the quadtree decomposition strategy for decomposing the image into several blocks at multiple levels and then reversible memory cellular automata (MCA) are used to induce confusion and diffusion in the original image. In [26], a mixture of chaos-based systems and CA is used for image encryption along with the quadtree decomposition strategy. Secret key streams are generated using LTS chaotic sequences, and an improved (one-dimensional) 1D chaotic system is used for bit level permutation in confusion phase. Then a combination of chaotic system and reversible MCA is used for inducing diffusion. Hash functions are an important cryptographic tool that computes a fixed length digest of plaintext, known as hash value [27]. This unique representation of the message can be considered as the fingerprint of the input plaintext. It is a one-way function in the sense that once the message is hashed it is impossible to recover the plaintext back from its hash value. Due to its crucial features, hash functions find application in many fields like message authentication codes and digital signatures. In cryptography, it is widely used for storing password hashes and key derivation.

This paper presents an efficient and robust image cryptosystem that combines chaos-based system and reversible MCA to encrypt/decrypt a digital image. Key streams are derived using a 1D chaotic map and an image dependent hash value generated by hash function SHA-256. The cryptographic scheme involves three stages of encryption. Firstly, to induce diffusion, image pixels are bitwise XORed with a chaotic sequence using Logistic-Sine (LS) map. The permutation of image pixels is then carried out using a pseudo-random number generator (PRNG) to induce confusion. Finally, the resulting image is split into four blocks of equal size using the quadtree decomposition strategy, and a fourth order reversible MCA is applied to them. These four blocks are considered as the initial configurations of the MCA, and the transition rules are determined using the chaotic map. This yields four output configurations constitute the final cipher image. Both confusion and diffusion are introduced in this phase. The performance analysis of the proposed encryption scheme and those of some well-known encryption schemes are provided in this paper. The given results demonstrate that the proposed scheme has a sound balance between the low complexity and the high level of security. It is worth noting that, trials to reduce the complexity of the proposed algorithm using a lower order MCA, give weaker performances.

The main contributions of this paper are the followings. First, to the best of our knowledge, no previous study has taken advantage of the fourth order 1D MCA to improve the efficiency of the classic diffusion-confusion architecture (i.e., the bit-wise XOR operation and the pixel-wise random permutation). Computer simulations show that the proposed combination CA–chaos, compared to a set of recently published works, gives competitive performances with a lower level of complexity. Second, the secret key (i.e., the initial condition and the control parameter) of the 1D chaotic map are generated using the 256-bit long hash value of the original image, this reliance boosted the sensitivity to minor changes in the original image or the initial keys, and then insured a high immunity to known/chosen plaintext attacks. Third, the time complexity of the proposed encryption scheme is significantly reduced as it employed a one-dimensional map instead of higher-dimensional ones and one iteration of the fourth order 1D MCA.

The rest of the paper is structured as follows. The preliminaries, including CA and chaotic system, are presented in Section II. Section III describes the proposed scheme, the analysis of which is provided in Section IV. Finally, concluding remarks are reached in Section V.

II. PRELIMINARIES

A. CELLULAR AUTOMATA

CA is a class of discrete dynamical system invented by John von Neumann [28] based on local rules. Using CA, emergence and complex behavior of a wide variety of natural systems can be demonstrated [29]. There are four elements of CA that define the overall structure and working of a cellular automaton:

1) A grid of identical objects called cells arranged in a circular register. Each of these cells assumes a finite number of states.
2) The cell states.
3) The neighborhood of a cell.
4) The transition rules that determine how a cell state changes over time.

The grid of cells, in general, can be a 1D or 2D array. The output obtained at discrete time intervals is another grid of the same size as input. These grids, usually known as CA configurations, form a pattern when visualized after serially stacking them for multiple iterations. The patterns can be categorized into four classes based on the behavior of subsequent configurations [30], [31]. Class 1 includes pattern with simple behavior in which a uniform final state is obtained irrespective of the initial configuration. In class 2, different final states are possible, but the pattern consists of simple structures that either remains same forever or repeats itself after few iterations. The behavior of class 3 is completely random in the sense that the location and frequency of structures formed in the pattern are quite unpredictable. And finally, the behavior of class 4 is a combination of order and randomness. Example of pattern from each of the above-mentioned classes for a 1D CA is shown in Fig. 1. In cryptography, we can exploit the property of randomness from class 3, for inducing confusion and diffusion in the ciphertext.

1) ELEMENTARY CA

An elementary CA is the simplest class of cellular automata with aforementioned characteristics. The grid of cells in the elementary CA is one dimensional or linear. Each cell can assume only two states, 0 or 1. The neighborhood is defined as the r left and right neighbors of the current cell including itself such that the total number of cells in the neighborhood is $2r + 1$. r is also called the radius. Generally, the transition rule is defined as follows

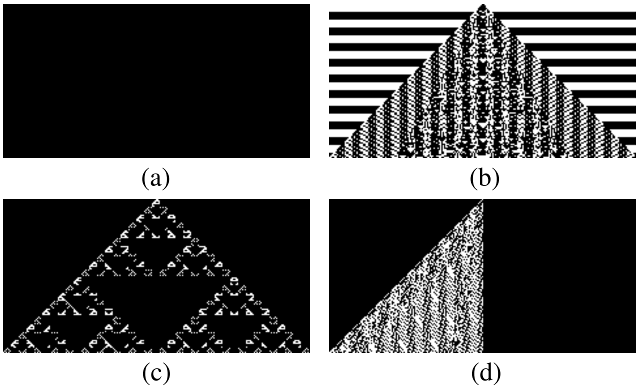


FIGURE 1. Example of patterns for the four classes of a 1D CA. (a) Class 1. (b) Class 2. (c) Class 3. (d) Class 4.

$$S_i^t = F\left\{S_{i-r}^{t-1}, S_{i-r+1}^{t-1}, \ldots, S_i^{t-1}, \ldots, S_{i+r-1}^{t-1}, S_{i+r}^{t-1}\right\}, \quad (1)$$

where S_i^t represents the state of i^{th} cell at discrete time t and F is a function of states of the neighborhood of cell i. A configuration C_i with cell size N is defined as the set of all the cell states at a given time t as follows

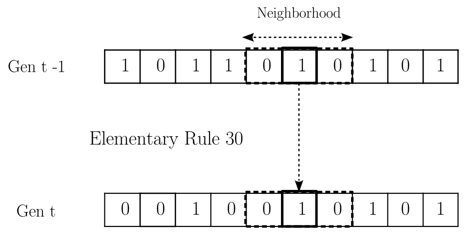
$$C_i = \left\{S_0^t, S_1^t, \ldots, S_{N-1}^t\right\}. \quad (2)$$

The discrete time at which a configuration is defined is known as a generation. Cell state at generation t is the function of states of neighborhood at generation $t - 1$.

Working of the simplest possible 1D CA with cell states 0 or 1 and radius $r = 1$ is depicted in Fig. 2. Since neighborhood size, in this case, is 3, there are $2^3 = 8$ possible ways in which it can be configured as shown in Fig. 2. For each of the neighborhood pattern in the current generation, a binary output is assigned to the cell in the next generation. So, in this case, a set of eight binary digits, known as ruleset, is needed for rule formation. Again, since the number of possible ways in which these rulesets can be configured is 28, there are 256 rules following which next configuration can be obtained. The ruleset shown in Fig. 2 is known as Rule-30 as the decimal equivalent of binary digits is 30.

2) MEMORY CELLULAR AUTOMATA

Standard CA discussed in the previous section are memoryless. That is, the states of the configuration at generation t depend only on the states of the configuration at generation $t - 1$. However, there is another class of CA in which the states of the configuration at generation t not only depend on the generation at $t - 1$ but also on the previous generations. Such class of CAs is known as memory cellular automata, MCA. MCA exhibit the property of reversibility which ensures the recovery of the original configuration using inverse MCA. A k^{th} order MCA implies that configuration at generation t, C_t depends on configurations of k previous generations $C_{t-1}, C_{t-2}, \ldots C_{t-k}$. Also, it is



Rule of CA definition:

r =1, rule $00011110_2 = 30_{10}$

Number	7	6	5	4	3	2	1	0
Neighborhood pattern	111	110	101	100	011	010	001	000
Transition Rule	0	0	0	1	1	1	1	0

FIGURE 2. CA evolution and ruleset.

evident that to ensure the reversibility of a k^{th} order MCA, $k-1$ transition rules are sufficient to retrieve the original configuration [25]. Therefore, an MCA of order k is defined as follows

$$C_t = F_1(C_{t-1}) \oplus F_2(C_{t-2}) \oplus \cdots \oplus F_{k-1}(C_{t-k+1}) \oplus (C_{t-k}), \tag{3}$$

where $F_1, F_2, \ldots, F_{k-1}$ are the transition rules and $\oplus$ denotes the bit-XOR operation. (3) is reversible for any set of transition rules, and the corresponding inverse MCA is defined by

$$P_t = F_{k-1}(P_{t-1}) \oplus F_{k-2}(P_{t-2}) \oplus \ldots \oplus F_1(P_{t-k+1}) \oplus (P_{t-k}), \tag{4}$$

where P_{t-i}, $i \in \{0, 1, \ldots, k\}$ are the possible configurations of the inverse MCA.

The reversibility of MCA can be proved as follows. Let the configuration C_t is obtained by applying k^{th} order MCA on k previous configurations $C_{t-1}, C_{t-2}, \ldots, C_{t-k}$ using (3). We prove that configurationC_{t-k} can be recovered from k latest configurations $C_t, C_{t-1}, \ldots, C_{t-k+1}$ using the inverse MCA defined by (4). It should be noted that configurations are placed in reverse order while applying the inverse MCA. That is,

$$P_{t-1} = C_{t-k+1}, P_{t-2} = C_{t-k+2}, \ldots, \\ P_{t-k+1} = C_{t-1}, P_t = C_{t-1}.$$

Therefore, (4) follows

$$\begin{aligned} P_t &\\ &= F_{k-1}(P_{t-1}) \oplus F_{k-2}(P_{t-2}) \oplus \cdots \oplus F_1(P_{t-k+1}) \oplus (P_{t-k}) \\ &= F_{k-1}(C_{t-k+1}) \oplus F_{k-2}(C_{t-k+2}) \oplus \cdots \oplus F_1(C_{t-1}) \oplus (C_t) \\ &= F_1(C_{t-1}) \oplus F_2(C_{t-2}) \oplus \cdots \oplus F_{k-1}(C_{t-k+1}) \oplus (C_t), \end{aligned} \tag{5}$$

substituting value of C_t in (5) from (3), we get

$$\begin{aligned} P_t &\\ &= F_1(C_{t-1}) \oplus F_2(C_{t-2}) \oplus \cdots \oplus F_{k-1}(C_{t-k+1}) \\ &\quad \oplus F_1(C_{t-1}) \oplus F_2(C_{t-2}) \oplus \cdots \oplus F_{k-1}(C_{t-k+1}) \oplus (C_{t-k}) \\ &= [F_1(C_{t-1}) \oplus F_1(C_{t-1})] \oplus [F_2(C_{t-2}) \oplus F_2(C_{t-2})] \ldots \\ &\quad \oplus [F_{k-1}(C_{t-k+1}) \oplus F_{k-1}(C_{t-k+1})] \oplus (C(t-k)) \\ &= 0 \oplus 0 \cdots \oplus 0 \oplus (C_{t-k}) \\ &= C_{t-k}. \end{aligned} \tag{6}$$

Hence, the configuration C_{t-k} can always be recovered by the MCA defined by (4) using $C_t, C_{t-1}, \ldots, C_{t-k+1}$ configurations. Therefore, (4) represents the inverse MCA defined by (3).

B. THE LOGISTIC-SINE MAP

The LS map is a 1D nonlinear combination of two existing seed maps, namely the Logistic map and the Sine map [32]. It is mathematically defined by

$$x_{n+1} = \left(\alpha x_n (1 - x_n) + (4 - \alpha) \frac{\sin(\pi x_n)}{4} \right) \bmod 1, \tag{7}$$

where $\alpha \in (0, 4]$ is the control parameter and $x_0 \in (0, 1)$ is the seed. The bifurcation diagram and the Lyapunov exponent of the LS map are illustrated in Figs. 3(a) and 3(b), respectively. An analysis reported in [32] shown the excellent chaoticness of (7). It was reached that the LS map outperforms its corresponding seed maps (i.e., the Logistic map and the Sine map). First, the output sequences of (7) spread out in the whole range of the seed $x_0 \in (0, 1)$. That is, the LS map has a uniform distribution within (0, 1) (cf. Fig. 3(a)). Second, the Lyapunov exponent of (7) is always greater than 0 in the entire range of the control parameter $\alpha \in (0, 4]$ (cf. Fig. 3(b)). However, the Lyapunov exponents of its corresponding seed maps are positive only in limited ranges. Finally, the Lyapunov exponent of the LS map is always larger than those of its seed maps. Considering all the above, the LS map outperforms its seed maps and is purely chaotic in the whole range of the secret key (i.e., control parameter and seed).

C. THE BACKGROUND OF COMBINING CA AND CHAOTIC MAP

Knowing that complexity does not imply performance, our paper aims at achieving better performance with the minimum of operations. This belongs to the background of the development of real-time applications and their implementation. With the purpose of improving complexity, the design of the proposed encryption scheme is based on the selection of efficient operations while avoiding rounds (for both key generation and encryption steps).

With the descriptions provided in subsection II.A, one can note that the evolution pattern of CAs matches well the principles of confusion and diffusion required by any encryption scheme obeying the well-known diffusion-confusion architecture. Using CA, confusion and diffusion are achieved

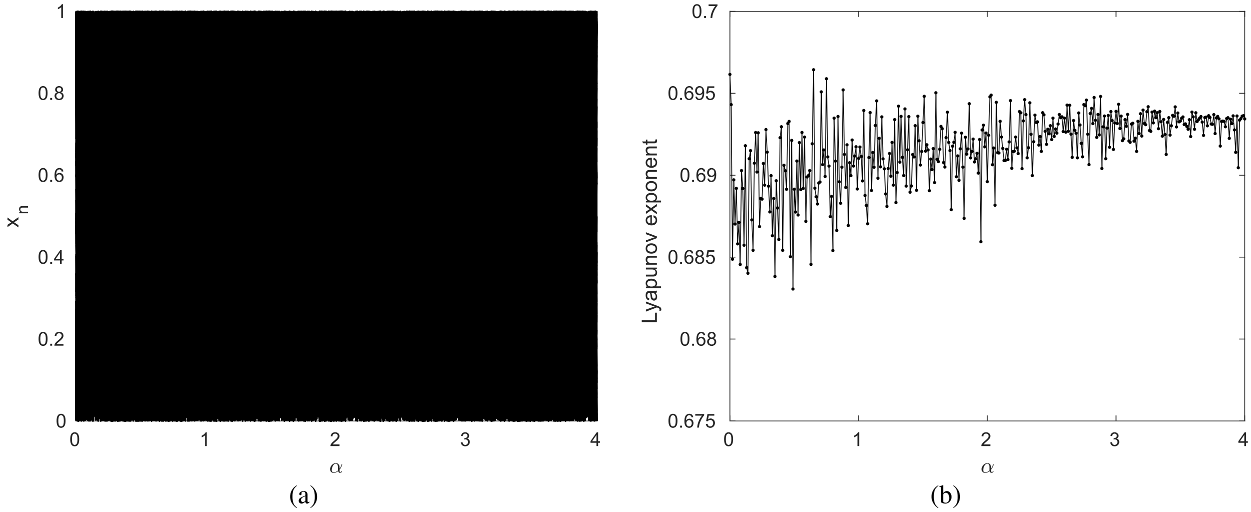


FIGURE 3. **Bifurcation diagram (a) and Lyapunov exponent (b) of the LS map.**

jointly and without loop neither on key generation nor on encryption steps. Through preliminary tests, we noted that (i) CA with a high number of iterations increases substantially its time complexity (ii) utilizing only CA with a lower number of iterations does not lead to competitive performance, especially for sensitivity tests. For this reason, we made use of a chaotic map to further improve the diffusion in the proposed algorithm. We remind that the main feature of chaotic maps is their high sensitivity to initial conditions, which makes them very useful in encryption schemes due to enhancing diffusion.

The paper presents a new architecture of image encryption scheme which takes benefit from both memory CA and chaos advantages, and the consideration of the complexity side in our proposal promotes its integration in real-time systems.

III. PROPOSED IMAGE ENCRYPTION ALGORITHM

In this section, we describe in detail the proposed scheme for both encryption and decryption. The encryption/decryption scheme involves a series of prominent cryptographic techniques including MCA, hash functions, chaotic systems, and random permutation. A hash function is used in the proposed scheme to get a unique signature of the original image. The hash value corresponding to the original image is used for derivation of various system parameters and keys required at different stages of encryption/decryption. Generation details of these keys are further discussed in this section. In our algorithm, the SHA-256 is employed, a custom designed cryptographic hash function that generates a 256-bit hash value.

At first, a string of hash value is generated, and its first 212 bits are split into four substrings of unequal lengths according to (8)-(11). These substrings are then converted to their corresponding decimal values, h_{1-52}, h_{53-106}, $h_{107-158}$, and $h_{159-212}$. System parameters and the seeds of (7) are obtained using these decimal values as follows

$$x'_0 = \frac{1}{2}(x_0 + h_{1-52}), \tag{8}$$

$$\alpha'_0 = \frac{1}{2}(\alpha_0 + h_{53-106}), \tag{9}$$

$$x'_1 = \frac{1}{2}(x_1 + h_{107-158}), \tag{10}$$

$$\alpha'_1 = \frac{1}{2}(\alpha_1 + h_{159-212}), \tag{11}$$

where x_0, α_0, x_1, and α_1 are the initial secret keys and h_{i-j} denotes the decimal conversion of the substring taking from the i^{th} bit to j^{th} bit of the hash value.

A. KEY GENERATION

A reversible MCA of k^{th} order requires $k-1$ transition rules and k previous configurations to obtain the next configuration. Since we are employing a fourth order MCA, three transition rules are needed for the MCA operation. These rules are defined using a set of sub-keys generated by (7) with system parameters evaluated through (8) and (9). The detailed procedure is described as follows.

1) Iterate (7) for l times ($l \geq 500$) to avoid the transient effect using the control parameter α'_0 and the seed x'_0. Continue iterating (7) 16×3 times to get the sequence $\mathbf{e}$.
2) Map $\mathbf{e}$ from [0, 1] to [0, 255] according to (12), which yields the sequence $\mathbf{q}$.

$$\mathbf{q} = \left(\mathbf{e} \times 10^{15}\right) \bmod 256 \tag{12}$$

3) Convert $\mathbf{q}$ into its binary form $\mathbf{k}$, a sequence of length $16 \times 3 \times 8 = 384$ bits. Then, divide $\mathbf{k}$ into 3 subsequences of 128 bits each to obtain the sub-keys set $\mathbf{k} = [\mathbf{k}_1\ \mathbf{k}_2\ \mathbf{k}_3]$.

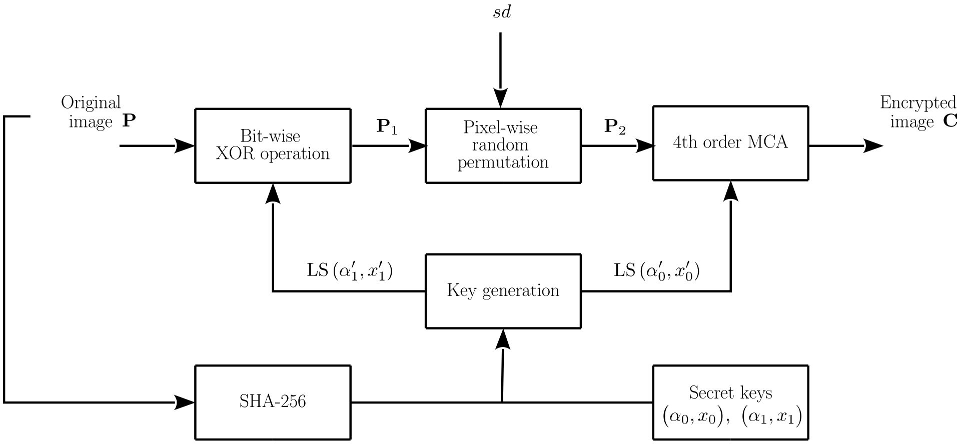


FIGURE 4. Flowchart of the proposed encryption scheme.

B. ENCRYPTION PROCESS

In this section stepwise procedure of the proposed encryption scheme, illustrated in Fig. 4, is described in detail. The overall process is composed of three phases under one round. First, a $M \times N$ chaotic matrix is generated using (7), M and N are the rows and columns of the original image. This matrix is then mapped into a suitable range and bit-wise XORed with the original image. The resulting image is then undergoing pixel-wise permutation. Finally, a fourth order reversible MCA is applied on the permuted image, which yields the encrypted image.

1) BIT-WISE CHAOTIC DIFFUSION

The original image is first bit-wise XORed with a chaotic matrix as follows.

- Iterate (7) $M \times N + 500$ times with the control parameter α'_1 and the seed x'_1. By ignoring the first 500 values and rearrange the rest, a $M \times N$ chaotic matrix $\mathbf{S}$ is obtained.
- Map $\mathbf{S}$ from $[0, 1]$ to $\{0, 1, 2, \ldots, 255\}$ according to the following

$$\mathbf{H} = \left(\mathbf{S} \times 10^{15}\right) \bmod 256. \tag{13}$$

- Perform bit-wise XOR operation between $\mathbf{H}$ and the original image $\mathbf{P}$ by

$$\mathbf{P}_1 = \mathbf{H} \oplus \mathbf{P}. \tag{14}$$

Simply, the bit-wise chaotic diffusion is referred as

$$\mathbf{P}_1 = \mathcal{F}_{\alpha'_1, x'_1}(\mathbf{P}). \tag{15}$$

2) PIXEL-WISE RANDOM PERMUTATION

In this step, $\mathbf{P}_1$ undergoes a pixel-wise random permutation in which its pixels are permuted using a PRNG. Positions of the pixels are updated according to a random sequence $\mathbf{w}$ obtained using Algorithm 1. This yields the permuted image $\mathbf{P}_2$.

Algorithm 1 Pixel-Wise Random Permutation

```
Input: Number of pixels $n = M \times N$ and seed of the PRNG $sd$.
Output: Permuted indices $\mathbf{w}$.
w = 1
prn = PRNG($n, sd$) {PRNG(.) can be any pseudo-random number generator}
for $i = 2$ to $n$ do
    $\mathbf{w} = [\mathbf{w}\ i]$
    $k = \lceil (i \times \mathbf{prn}(i-1)) \rceil$ {⌈.⌉ denotes the ceiling function}
    $\mathbf{w}([k\ i]) = \mathbf{w}([i\ k])$
end for
```

The operation of pixel-wise random permutation is referred as

$$\mathbf{P}_2 = \mathcal{G}_{sd}(\mathbf{P}_1). \tag{16}$$

3) FOURTH ORDER MCA

Image $\mathbf{P}_2$ is split into four $\frac{M}{2} \times \frac{N}{2}$ blocks, namely $\mathbf{B}_1, \mathbf{B}_2, \mathbf{B}_3$, and $\mathbf{B}_4$. Each of these blocks is first transformed into its binary form and then converted to a one-dimensional binary sequence of length $\frac{M}{2} \times \frac{N}{2} \times 8 = 2MN$. This yields the binary sequences $c_1, \mathbf{c}_2, \mathbf{c}_3$, and $\mathbf{c}_4$, which will be considered as the initial configurations of the fourth order MCA defined by (3). The transition rules of the MCA are derived using sub-keys $\mathbf{k}_1, \mathbf{k}_2$, and $\mathbf{k}_3$. The final configurations are obtained after applying the MCA evolution mechanism during r iterations. Therefore, these configurations are transformed into their decimal forms, rearranged into $\frac{M}{2} \times \frac{N}{2}$ matrices, and merged to obtain the encrypted image $\mathbf{C}$. The fourth Order MCA mechanism is referred as

$$\mathbf{C} = \mathcal{H}_{\alpha'_0, x'_0}(\mathbf{P}_2). \tag{17}$$

C. THE DECRYPTION PROCESS

To properly recover the original image, a total of six parameters must be transmitted to the decryption side.

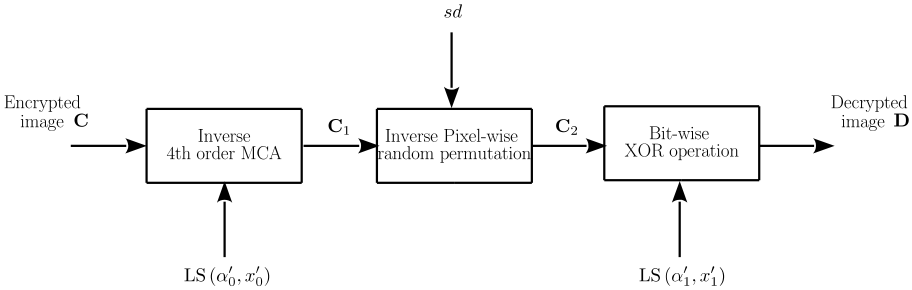


FIGURE 5. Flowchart of the proposed decryption scheme.

These parameters include the seeds and system parameters of the LS map (cf. x_0, α_0, x_1, and α_1), the seed of the PRNG (cf. sd), and the 256-bit hash value of the original image. The receiver can now generates the parameters x'_0, α'_0, x'_1, and α'_1 through (1)-(4). The decryption process, illustrated in Fig. 5, is performed in the reverse order of the encryption process. An encrypted image **C** can be recovered as

$$\mathbf{D} = \mathcal{F}^{-1}_{\alpha'_1,x'_1}\left(\overbrace{\mathcal{G}^{-1}_{sd}\left(\underbrace{\mathcal{H}^{-1}_{\alpha'_0,x'_0}(\mathbf{C})}_{\mathbf{C}_1}\right)}^{\mathbf{C}_2}\right), \tag{18}$$

where **D** is the decrypted image and $\mathcal{F}^{-1}_{\alpha'_1,x'_1}$, $\mathcal{G}^{-1}_{sd}$, $\mathcal{H}^{-1}_{\alpha'_0,x'_0}$ are the inverse functions of $\mathcal{F}_{\alpha'_1,x'_1}$, $\mathcal{G}_{sd}$, $\mathcal{H}_{\alpha'_0,x'_0}$, respectively.

IV. PERFORMANCE ANALYSIS

To show the effectiveness of the proposed encryption scheme, a set of tests have been performed. These tests include key space analysis, statistical analysis, sensitivity analysis, robustness analysis, and time complexity analysis. Note that the proposed scheme can be immediately applied to any image formats. One hundred of standard 8-bit grayscale images of size 512 × 512 are used as benchmarking [33]. This image database is thought to be the largest database ever to have used for evaluating an image cryptosystem. The seeds and the control parameters of (7), i.e., x_0, α_0, x_1, and α_1 are fixed at $0.2, 1.99, 0.7$, and 2.75, respectively. The seed of the PRNG is chosen to be 0.3. The number of iterations used for the MCA is $n = 1$. As they contain new concepts and present high performances surpassed most of the previous works, some state-of-the-art algorithms from [14], [25], [34] are selected for comparison with the proposed algorithm. It should be emphasized that the algorithm from [34] employs a new cryptographic technique, namely image filtering. Indeed, the images are encrypted using block-based scrambling and image filtering. The block-based scrambling consists of dividing the image into blocks and randomly shuffling the pixels of each block. In image filtering, random masks are employed to filter the shuffled image.

A. KEY SPACE ANALYSIS

An encryption scheme should possess a large key space to resist brute-force attacks. In the proposed encryption scheme, the key space consists of:

- The seeds x_0, x_1 and the control parameters α_0, α_1 (cf. (7)).
- The seed sd of the PRNG (cf. Algorithm 1).
- The 256-bit long hash value.

Assuming that the computational precision of the 64-bit double-precision numbers is 2^{-49}, the possible values of x_0 are more than 2^{49}, as are the values of x_1, α_0, α_1, and sd. The complexity of the currently best known practical collision attack on the SHA-256 hash function is 2^{128}. Therefore, the key space is larger than 373 bits, which rendered impossible any potential brute-force attacks.

B. STATISTICAL ANALYSIS

1) UNIFORMITY OF THE BIT DISTRIBUTION WITHIN EACH BIT-PLANE

A bit-plane of an image represents the set of bits corresponding to a bit position of the binary representations of the pixels. Indeed, the pixels of a grayscale image are usually encoded into 8 bits, and hence there are eight bit-planes corresponding to each grayscale image. It has been found that the distribution of bit-planes of a natural image regarding the percentage of 1's and 0's is usually nonuniform because of the shapes and texture present in the image. However, this distribution should be highly uniform (the expected value is 50%) in case of an encrypted image because of bit-level randomness introduced by diffusion and confusion in each bit-plane. Therefore a secure cryptosystem should introduce as much uniformity in the bit distribution as possible within each of its bit-plane. Table 1 shows the mean and variance of the percentage of 1's in different original images and their encrypted counterparts for different algorithms. It is observed that our scheme leads to the best mean percentage of 1's in five bit-planes. The studied algorithms exhibit identical variances, in which they are almost equal to zero with the exception of the algorithm from [25] where the variance is somewhat larger than zero.

2) CORRELATION OF ADJACENT PIXELS

In an original image, the correlation between neighboring pixels is always high as every pixel follows a certain pattern. However, this correlation should be sufficiently low in an

TABLE 1. Percentage of '1's (mean and variance) in different original images vs. their encrypted counterparts for different algorithms.

		8th bit	7th bit	6th bit	5th bit	4th bit	3rd bit	2nd bit	1st bit
Mean	**Original images**	61.5202	38.4798	50.7646	49.2354	49.4647	50.5353	47.8943	52.1057
	[14]	50.0015	49.9985	50.0138	49.9862	49.9924	50.0076	49.9952	50.0048
	[34]	49.9907	50.0093	49.9987	50.0013	49.9873	50.0127	50.0012	49.9988
	[25]	50.1850	49.8150	50.1478	49.8522	50.0497	49.9503	50.0199	49.9801
	Proposed	50.0095	49.9905	49.9997	50.0003	49.9963	50.0037	50.0018	49.9982
Variance	**Original images**	630.4133	630.4133	283.8726	283.8726	153.4690	153.4690	85.2285	85.2285
	[14]	0.0109	0.0109	0.0096	0.0096	0.0087	0.0087	0.0089	0.0089
	[34]	0.0084	0.0084	0.0101	0.0101	0.0105	0.0105	0.0068	0.0068
	[25]	1.2344	1.2344	0.4495	0.4495	0.0411	0.0411	0.0704	0.0704
	Proposed	0.0083	0.0083	0.0094	0.0094	0.0092	0.0092	0.0099	0.0099

encrypted image to ensure randomness and unpredictability among the cipher image pixels. The correlation among pixels can be exploited by a cryptanalyst for statistical attacks to recover the secret key and subsequently the original image. Therefore to make the encryption scheme secure, adjacent pixels should be uncorrelated.

Correlation analysis of the original and encrypted images is performed by computing and comparing 3000 pairs of two adjacent pixels in horizontal, vertical and diagonal directions. These pixels are randomly selected from the original and cipher images. The correlation coefficient between two neighboring pixels can be calculated using the following formula:

$$C_{ab} = \frac{\sum_{i=1}^{K} (a_i - \mathbb{E}\{a\})(b_i - \mathbb{E}\{b\})}{\sqrt{\sum_{i=1}^{K} (a_i - \mathbb{E}\{a\})^2}\sqrt{\sum_{i=1}^{K} (b_i - \mathbb{E}\{b\})^2}}, \quad (19)$$

where a_i and b_i are the grayscale values of two randomly neighboring pixels, K is the total number of (a_i, b_i) combinations obtained from the image, and $\mathbb{E}\{.\}$ denotes the expected value of a random variable.

Table 2 reports the mean of absolute values of the correlation coefficients in the horizontal, vertical and diagonal directions from the original and encrypted images. It is observed that, in all directions, the obtained correlation coefficients are quite close to one for the original images, and are quite close to zero for their encrypted ones. Therefore, the neighboring pixels of the original images are uncorrelated, and this for all directions. Similar results are shown graphically in Figs. 6(a)-(c) and Figs. 6(d)-(f) which plot the correlation coefficients between pixel pairs in the horizontal, vertical, and diagonal directions of the original and encrypted images of Cameraman, respectively. As a result, the proposed algorithm can successfully cancel the correlation between adjacent pixels, which makes correlation-based attacks impossible.

TABLE 2. Mean of absolute values of the correlation between pairs of the original and encrypted images.

	Scan direction	Original images	Encrypted images
Mean	Horizontal	0.9159	0.0022
	Vertical	0.9155	0.0022
	Diagonal	0.8704	0.0020

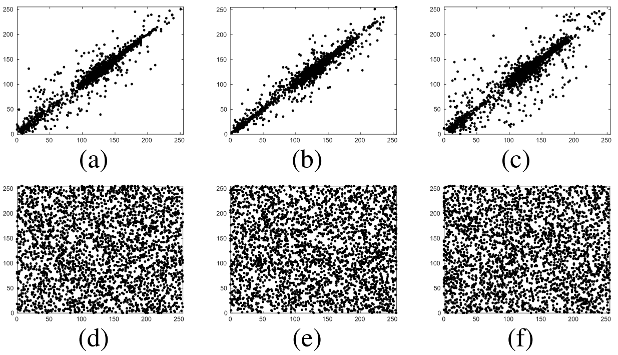


FIGURE 6. Distribution of pairs of adjacent pixels in the original and encrypted images of Cameraman. Frames (a) and (d): Distributions of two horizontally adjacent pixels in the original and encrypted images, respectively. Frames (b) and (e): Distributions of two vertically adjacent pixels in the original and encrypted images, respectively. Frames (c) and (f): Distributions of two diagonally adjacent pixels in the original and encrypted images, respectively.

3) HISTOGRAM AND CHI-SQUARE TEST

An image histogram is a graphical representation of the distribution of pixel intensities in the image. It gives an idea about the image content concerning the intensity range of the image pixels. Most of the original images have a nonuniform histogram, however, a well-encrypted image reflects a uniform histogram to conceal the intensity information. Fig. 7 shows the original images of Cameraman and Airplane along with their histograms, while the corresponding cipher images and their histograms are shown in Fig. 8. It can be observed from the figures that the encrypted image histograms are fairly uniform and completely different from that of the corresponding original images.

Chi-square test [35], [36] is performed to check the extent of equiprobable frequency counts (H_0 - null hypothesis), that is, how much the distribution of encrypted image's histogram matches with the uniform distribution. It can be calculated as follows

$$\chi^2 = \sum_{i=0}^{L_p-1} \frac{(o_i - e_i)^2}{e_i}, \quad (20)$$

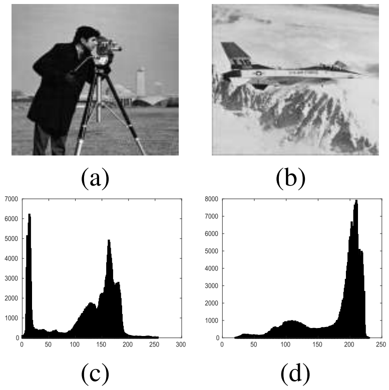


FIGURE 7. Original images and their histograms. Frames (a) and (b) are the original images of Cameraman and Airplane, respectively. Frames (c) and (d) are the histograms of the original images of Cameraman and Airplane, respectively.

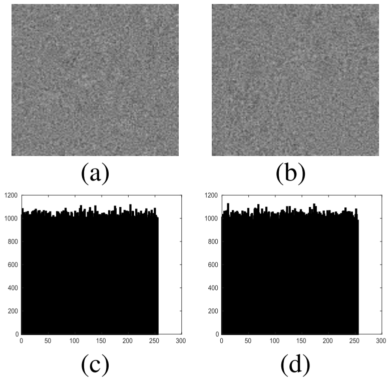


FIGURE 8. Encrypted images and their histograms. Frames (a) and (b) are the encrypted images of Cameraman and Airplane, respectively. Frames (c) and (d) are the histograms of the encrypted images of Cameraman and Airplane, respectively.

where L_p denotes the number of pixel levels (for a grayscale image, $L_p = 256$), o_i represents the observed occurrence frequency of each pixel value $(0-255)$ in the histogram of the encrypted image, and e_i is the expected occurrence frequency of the uniform distribution, i.e., $e_i = (M \times N)/256$. If the p-value exceeds the significance level l_s $(l_s \in [0, 1])$, the null hypothesis is accepted and the histogram follows the uniform distribution. Table 3 shows the Chi-square test of histograms (mean, variance and success rate), with a significance level of 0.05, for different algorithms. It is obvious from the table that the proposed algorithm outperforms, with a success rate of 97%, the comparative algorithms. Therefore, the proposed algorithm can remove spatial redundancies in the original image, and then, no useful information can be found to apply histogram-based attacks. It is worth noting that, in literature, histogram analysis is often performed as a hypothesis test. However, some works present the analysis using variance calculation [37].

TABLE 3. Chi-square test of histograms (mean, variance, and success rate) for different algorithms.

	χ^2 test (p-value)			
	[14]	**[34]**	**[25]**	**Proposed**
Mean	0.5115	0.5113	0.3661	0.4909
Variance	0.0837	0.0781	0.0958	0.0721
Success rate (%)	95	96	77	**97**

4) GLOBAL ENTROPY

Global entropy is evaluated to perform statistical measurement of randomness in the image usually to characterize the texture of the image. The global entropy of a source X is given by

$$H(X) = -\sum_{i=1}^{K} p(x_i) \log_2 (p(x_i)) \quad \text{[bits]}, \tag{21}$$

where K is the number of symbols in X and $p(x_i)$ is the probability of symbol x_i. A high value of entropy correspond to high randomness in the image while the predictability of the information rises as the entropy of image gets lower. The optimum entropy (i.e., 8 bits) is obtained when the pixels are equiprobable, which means a uniform distribution of the pixels. The global entropies (mean and variance) of the studied algorithms are reported in Table 4. The proposed algorithm and the ones from [14], [34] present similar global entropy values that approach to the optimum value, while the obtained entropy for the algorithm in [25] is a bit far from the optimum value. Consequently, the information loss from the proposed algorithm is negligible.

5) LOCAL ENTROPY

Local Shannon entropy is evaluated to measure the image's randomness qualitatively rather than quantitative measure (as for the global entropy) that cannot necessarily guarantee the randomness of the image [38]. The calculation of the local entropy is detailed in [38]. Comparison of local entropies is reported in Table 5, where the proposed algorithm and those from [14], [34] have average local entropies approximately equal to the optimum value, 7.9024693 [38].

C. SENSITIVITY ANALYSIS

1) KEY SENSITIVITY

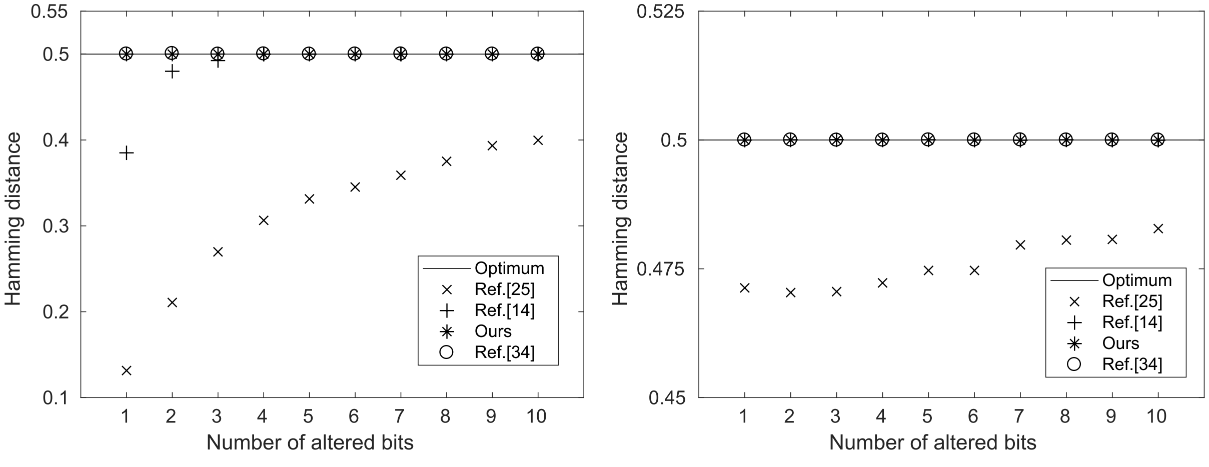
A secure encryption scheme should be sensitive to the secret keys such that even a tiny change in the keys should completely change the encrypted image as well as the decrypted one. Fig. 9(a) shows a comparison of the mean Hamming distances as a function of the number of altered bits in the secret keys $\big(\text{cf. } (7)\big)$. During this test, the indices of the n $(n = 1, \ldots, 10)$ altered bits as well as the 512×512 images are randomly generated under 200 iterations. It is seen that even a single altered bit, the proposed algorithm and the one from [34] are quite close to the optimal value of 0.5. Therefore, the proposed algorithm is extremely sensitive to minor change within its secret keys, and the avalanche effect is reached. Moreover, it is noted that the algorithm reported in [25] fails in this test and presents critical security issues.

TABLE 4. Global entropy (mean and variance) analysis.

	Global entropy			
	[14]	**[34]**	**[25]**	**Proposed**
Mean	7.999 300	7.999 300	7.986 154	7.999 298
Variance	$3.814\,091 \times 10^{-9}$	$3.197\,235 \times 10^{-9}$	$9.646\,547 \times 10^{-3}$	$3.078\,801 \times 10^{-9}$

TABLE 5. Local entropy (mean and variance) analysis.

	Local entropy			
	[14]	**[34]**	**[25]**	**Proposed**
Mean	7.902 454	7.902 396	7.832 426	7.902 426
Variance	$3.048\,803 \times 10^{-7}$	$3.820\,000 \times 10^{-7}$	$2.080\,030 \times 10^{-1}$	$3.610\,543 \times 10^{-7}$



(a) (b)

FIGURE 9. Sensitivity analysis: (a) key sensitivity and (b) plaintext sensitivity for the proposed algorithm (where $\alpha_0 = 2.6484375$, $x_0 = 0.3515625$, $\alpha_1 = 1.078125$, and $x_1 = 0.712890625$) and the comparative ones.

2) PLAINTEXT SENSITIVITY

An encryption scheme should be sensitive to plaintext such that even one-bit change in the original image yields a completely different cipher image. Cryptosystems with better plaintext sensitivity are less prone to chosen plaintext attacks where more than one original images are encrypted to analyze the change in their corresponding cipher images. Fig. 9(b) shows a comparison of the mean Hamming distances as a function of the number of altered bits in the original image. During this test, the indices of the n ($n = 1, \ldots, 10$) altered bits as well as the 512 × 512 images are randomly generated under 200 iterations. It is observed that even one altered bit, the proposed algorithm and the ones from [14], [34] are quite close to the optimal value of 0.5. As a result, the proposed algorithm is quite sensitive to small change within the original image, and no useful information can be obtained from the algorithm. As for the algorithm in [25] there is a harmony between the obtained Hamming distances for plaintext and key sensitivity which supports the hypothesis of a potential vulnerability in the algorithm.

3) UACI AND NPCR

One of the well-known chosen-plaintext attacks is the differential attack. By analyzing how small changes in the original images can significantly affect the encrypted images, the differential attack reaches to find correlations between the original and encrypted images, which leads to deciphering the encrypted images without using the secret keys. The ability of an image cryptosystem to resist differential attacks can be quantitatively examined using the number of changing pixel rate (NPCR), and the unified averaged changed intensity (UACI). A high UACI/NPCR value corresponds to high resistance against various differential attacks [39]. For two encrypted images $\mathbf{C}_1$ and $\mathbf{C}_2$ generated from two original images with a one-pixel difference, the NPCR and UACI are defined as follows

$$\mathtt{NPCR} = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\mathcal{D}(i,j), \tag{22}$$

$$\mathtt{UACI} = \frac{1}{255 \times MN}\sum_{i=1}^{M}\sum_{j=1}^{N}|\mathbf{C}_1(i,j) - \mathbf{C}_2(i,j)|, \tag{23}$$

where M and N denotes the image's width and height, respectively, and $\mathcal{D}(i,j)$ represents the difference between $\mathbf{C}_1$ and $\mathbf{C}_2$ which is given by

$$\mathcal{D}(i,j) = \begin{cases} 0, & \text{if } \mathbf{C}_1(i,j) = \mathbf{C}_2(i,j), \\ 1, & \text{if } \mathbf{C}_1(i,j) \neq \mathbf{C}_2(i,j). \end{cases}$$

The NPCR and UACI (mean and variance) values are compared in Table 6. As for the proposed algorithm, the mean

TABLE 6. NPCR and UACI (mean and variance) of the encrypted images due to one-bit change in the original image.

	NPCR (%)				UACI (%)			
	[14]	[34]	[25]	Proposed	[14]	[34]	[25]	Proposed
Mean	99.6103	99.6100	99.2000	99.6103	33.4674	33.4526	31.9249	33.4615
Variance	0.0002	0.0002	0.0926	0.0002	0.0018	0.0020	0.4232	0.0022

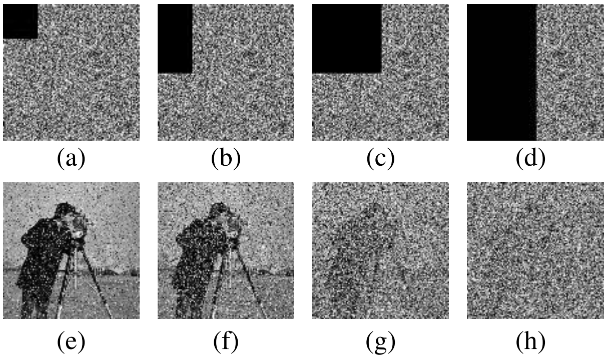


FIGURE 10. Test of image occlusion attacks: encrypted images with (a) 1/16, (b) 1/8, (c) 1/4, and (d) 1/2 data loss; corresponding decrypted images (e)-(h) in accordance with (a)-(d).

values of NPCR and UACI are 99.6103% and 33.4615%, respectively. They are approximately equal to the theoretically ideal values of 99.6% and 33.4%, respectively. That is, the proposed algorithm possesses high immunity against differential attacks. For this test, the algorithm reported in [25] has the lowest perfromance, while the proposed algorithm and the tested ones from [14], [34] have competitive performances.

D. ROBUSTNESS ANALYSIS

1) OCCLUSION ATTACK

Ciphered images are inevitably affected by occlusions during transmission. Quality of the decrypted image sharply deteriorates as the occlusion size increases. Therefore an encryption scheme should be robust enough to resist such occlusion attacks. Figs. 10 (a)-(d) depict the encrypted images with a data loss of 1/16, 1/8, 1/4, and 1/2, respectively. The corresponding decrypted images from Figs. 10 (a)-(d) are shown in Figs. 10 (e)-(h), respectively. It is seen that even a data loss of 50% from the encrypted image its corresponding decrypted image maintains the most visual information of the original image. Table 7 quantitatively compares the effect of these occlusion attacks on the decrypted images by measuring the PSNR (mean) between the original and the decrypted images. It is evident that the proposed algorithm performs better against the occlusion attacks.

2) NOISE ATTACK

An encrypted image gets affected by several noises during transmission through different channels. The robustness of an encryption scheme against these noises should be analyzed. Fig. 11 shows the effect of salt & pepper noise with various densities on the decrypted images of Cameraman. As can see, the decrypted images are noisy but cognizable. Table 8 quantitatively compares the ability to defend noise attacks by evaluating the PSNR (mean) between original and decrypted images subjected to salt & pepper noises. It is found that the proposed algorithm outperforms algorithms from [14], [25], [34] in withstanding noise attacks.

TABLE 7. PSNR (mean) between original and decrypted images subject to occlusion attacks.

		Occlusion			
	Algorithm	1/16	1/8	1/4	1/2
PSNR (dB)	[14]	12.1619	10.0859	8.7538	8.2839
	[34]	8.2354	8.2305	8.2316	8.2297
	[25]	8.5661	8.4755	8.7571	8.4072
	Proposed	15.6097	12.6328	9.6207	8.7128

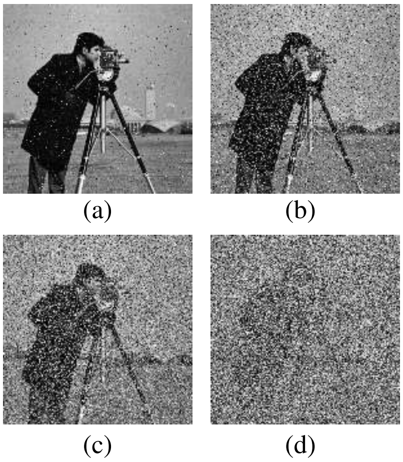


FIGURE 11. Decrypted images subject to the salt & pepper noise with a noise density of (a) 0.005; (b) 0.05; (c) 0.100; and (d) 0.300.

TABLE 8. PSNR (mean) between the original and decrypted images under salt & pepper noise.

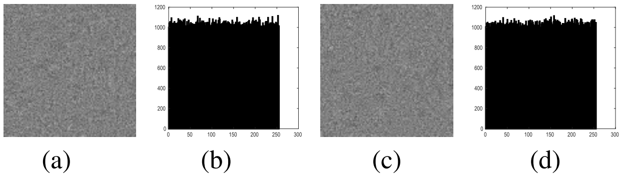
		Density of salt & pepper noise			
	Algorithm	0.005	0.050	0.100	0.300
PSNR (dB)	[14]	21.8814	12.6316	10.4252	8.4225
	[34]	8.2274	8.2192	8.2210	8.2191
	[25]	12.2487	8.3992	8.3218	8.2416
	Proposed	23.3479	14.0162	11.6727	9.0545

E. KNOWN-PLAINTEXT AND CHOSEN-PLAINTEXT ATTACKS

In known-plaintext attack (KPA), the cryptanalyst has access to both cipher image and its corresponding original image,

TABLE 9. Statistical and differential analyses of the encrypted ones of the all white and all black images.

Image	Algorithm	χ^2 test of histogram	Correlation			Global entropy	Local entropy	NPCR (%)	UACI (%)
		p-value	Horizontal	Vertical	Diagonal				
All white	**[14]**	0.7837	0.0043	–0.0007	0.0003	7.9993	7.9022	99.5861	33.4615
	[34]	0.2299	0.0043	0.0028	0.0003	7.9993	7.9030	99.5983	33.4682
	[25]	0	0.9991	0.0381	0.0366	3.1642	1.9065	62.5015	10.9806
	Proposed	0.8677	–0.0033	0.0005	–0.0007	7.9994	7.9022	99.6143	33.5214
All black	**[14]**	0	0.0002	–0.0009	0.0046	2.0000	1.9989	75.0347	00.4900
	[34]	0	–0.0015	0.0004	–0.0004	0.0384	0.0381	0.7969	0.0031
	[25]	0	NaN	NaN	NaN	0.0012	0.0002	12.5031	00.0505
	Proposed	0.7989	–0.0027	0.0055	–0.0016	7.9994	7.9022	99.6044	33.4588



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

FIGURE 12. Encrypted images of the all white (a) and all black (c); their histograms (b) and (d), respectively.

TABLE 10. Time complexity orders and their orders of magnitude for a grayscale image of size $M \times N = 512 \times 512$. For the given simulation results in [14] and [25], $L = MN$ and $n = 1$, respectively.

	Complexity order	Order of magnitude $(\times 10^6)$
[14]	$O\left(MN\left(34+\log(L)+2\sqrt{L}\right)\right)$	282
[34]	$O\left(108MN+72L^4\right)$	45
[25]	$O\left((36n+2)MN\right)$	10
Proposed	$O\left((27n+20)MN\right)$	12

which allows him to recover secret information such as secret keys, and thus break the cipher [40]. As for the chosen-plaintext attack (CPA), the attacker selects an arbitrary original image, and then he got its cipher version [40]. Doing so, the attacker tries to obtain the secret keys or to develop an algorithm that allows him to decipher an encrypted image that was encrypted using these keys but without actually accessing the keys. In the proposed algorithm, the SHA-256 hash value of the input image is employed to generate the secret keys of (7). So, when the input image changes the secret keys change too as is the output image. That is, an attacker cannot get sensitive information about the encryption by ciphering some special images. Therefore, the proposed algorithm possess high ability in resisting chosen-plaintext and known-plaintext attacks. During a hacking attempt, the attacker profit from encrypting all white and all black images since they completely disable the permutation/substitution process. Various statistical and differential analysis of the proposed algorithm for all white and all black images are showcased in Fig. 12 and Table 9, which contains a comparison with other tested algorithms. It is clear that the proposed algorithm produces unrecognizable images with uniform histograms, zero correlations, global and local entropies quite close to the optimal values of 8 and 7.9024693, respectively, and NPCR and UACI values approximate to the expected ones of 99.6% and 33.4%, respectively. The comparative algorithms fail in ciphering all black image. Considering all the above, the proposed scheme performs better against chosen-plaintext and known-plaintext attacks.

F. TIME COMPLEXITY ANALYSIS

The time complexity orders and their orders of magnitude for a 512 × 512 grayscale image are compared in Table 10. The details of complexity computation are available in Appendix V. It is shown from Table 10 that the proposed algorithm and the one from [25] have the same order of magnitude of computational complexity. Moreover, the algorithms from [14], [34] are the slowest ones.

V. CONCLUSION

In this paper we proposed a fast and efficient image encryption scheme that combines two familiar dynamical systems: chaotic map and cellular automata. The process is composed of a key generation layer followed by a diffusion-confusion layer and a fourth order 1D MCA, under one round. The secret keys of the LS map are derived using the SHA-256 hash value of the original image, leading to produce one-time keystreams. This fostered a high sensitivity to small alterations in the original image, and hence guaranteed an excellent ability in defending chosen-plaintext and known-plaintext attacks. The keystream of the diffusion process and the transition rules of the fourth order 1D MCA are both derived from the LS map. The diffusion-confusion layer consists of a bit-wise XOR operation followed by a pixel-wise random permutation. The resulting image is decomposed into four blocks according to the quadtree decomposition strategy. Then, a fourth order MCA is applied, the blocks obtained from the quadtree decomposition are considered as the initial MCA configurations. Performance of the proposed algorithm is evaluated using several analyses such as key space analysis, statistical analysis, sensitivity analysis, robustness analysis, and complexity analysis. We found that the proposed algorithm is fast running and can efficiently resisting all kind of attacks, which gives a better performance/complexity trade-off, in particular with regard to the selected state-of-the-art algorithms. Furthermore, the lower computational complexity makes our encryption scheme prominent for real-time applications and hardware implementation.

TABLE 11. Time complexity analysis for the evaluated image encryption algorithms.

	[14]$^{(1)}$	[34]$^{(4)}$	[25]$^{(6)}$	Proposed
Addition	$18MN + 260$	$44MN+36L^4+8L^2+32$	–	$4MN + 148$
Multiplication	$10MN + 266$	$40MN+16L^4+8L^2+36$	–	$8MN + 291$
Exponentiation	259	128	–	–
Trigonometric functions	–	–	–	$MN + 48$
Mod	$4MN + 6$	$16MN+12L^4+8L^2+32$	–	$3MN + 96$
Rounding functions	$2MN$	$8MN + 8L^4 + 8L^2 + 34$	–	$MN - 1$
XOR	–	16	$\left(2 + 12n \sum_{i=1}^{m-1} 4^{-i}\right) MN$ $+62$	$(3n + 1)MN$
Substitutions	–	–	$96MNn \sum_{i=1}^{m-1} 4^{-i}$ $+768m$	$(24n + 2)MN - 1$
Sorting vector of length	$\sqrt{L}^{(2)}$	$L^{2(5)}$; $L^{2(5)}$	–	–
Searching in vector of length	$\sqrt{L}^{(3)}$	–	–	–
SHA-256 operations	–	–	–	$\frac{8}{512}MN$

(1) The data block is of length L. Then, each block is assumed to be of size $\sqrt{L} \times \sqrt{L}$;
(2) One sorting operation for each round, data block, and block row. The sorting-time complexity should be multiplied by $2 \times \frac{MN}{L} \times \sqrt{L} = \frac{2MN}{\sqrt{L}}$;
(3) One searching operation for each round, data block, block row, and block column. The searching-time complexity should be multiplied by $2 \times \frac{MN}{L} \times \sqrt{L} \times \sqrt{L} = 2MN$;
(4) The data block is of size $L \times L$. In [34], $L = \min\{\lfloor\sqrt{M}\rfloor, \lfloor\sqrt{N}\rfloor\}$, where $\lfloor . \rfloor$ denotes the floor function;
(5) The sorting-time complexity should be multiplied by 4, i.e., the number of rounds;
(6) The parameter m refers to the quadtree decomposition depth. Assuming a square image of size $M \times M$ pixels, where M is equal to two to the nth power (not to be confused with the parameter n), according to [25], $m = \log_2(M) - 2$.

APPENDIX

TIME COMPLEXITY ANALYSIS FOR THE PROPOSED IMAGE ENCRYPTION ALGORITHM AND THE ONES IN [14], [34], AND [25]

Table 11 reports a count of occurrences of each operation according to MN, the number of pixels. [14]'s encryption algorithm calls searching and sorting operations. Let a vector of k elements, searching is of time complexity $O(k)$, and sorting is of average time complexity $k \log(k)$ if one considers the *quicksort* algorithm [41]. The proposed algorithm and the one in [34] use randomly generated numbers. To derive their time complexities, we assume a linear congruential generator since it is widely used as a uniform random number generator [42]. About SHA-256 hash algorithm, note that the algorithm has an entry size of 512 bits and a constant number of operations.

After gathering the operations given by Table 11, then ignoring lower-orders terms, complexity orders are expressed as

$$\begin{cases} O\left(MN\left(34 + \log(L) + 2\sqrt{L}\right)\right) & \text{for [14]} \\ O\left(108MN + 72L^4\right) & \text{for [34]} \\ O\left(\left(2 + 108n\sum_{i=1}^{m-1} 4^{-i}\right)MN\right) & \text{for [25]} \\ O\left((27n + 20)MN\right) & \text{for our proposed algorithm.} \end{cases} \tag{24}$$

To make the comparison easier, let simplify the time complexity of [25]. The term

$$\sum_{i=1}^{m-1} 4^{-i} = \sum_{i=1}^{\log_2(M)-3} \left(\frac{1}{4}\right)^i \tag{25}$$

is, in fact, a geometric series with common ratio $\frac{1}{4}$, the sum then becomes

$$\begin{aligned} \sum_{i=1}^{\log_2(M)-3} \left(\frac{1}{4}\right)^i &= \frac{1}{3} - \frac{1}{3}4^{3-\log_2(M)} \\ &= \frac{1}{3} - \frac{64}{3M^2}. \end{aligned} \tag{26}$$

Disregarding the lower-order term, the derived complexity order of [25] is simplified as $O\left((36n + 2)MN\right)$. It is worth noting that, since $36n + 2 \geq 27n + 20\ \forall n \geq 2$, complexity order of our proposed encryption algorithm is lower than that of [25] for n greater than 2.

REFERENCES

[1] H. Liu and X. Wang, "Color image encryption using spatial bit-level permutation and high-dimension chaotic system," *Opt. Commun.*, vol. 284, nos. 16–17, pp. 3895–3903, 2011.

[2] H. Liu, X. Wang, and A. Kadir, "Image encryption using DNA complementary rule and chaotic maps," *Appl. Soft. Comput.*, vol. 12, no. 5, pp. 1457–1466, 2012.

[3] X.-Y. Wang, L. Yang, R. Liu, and A. Kadir, "A chaotic image encryption algorithm based on perceptron model," *Nonlinear Dyn.*, vol. 62, no. 3, pp. 615–621, 2010.

[4] S. E. Assad and M. Farajallah, "A new chaos-based image encryption system," *Signal Process., Image Commun.*, vol. 41, pp. 144–157, Feb. 2016.

[5] H. Liu and X. Wang, "Color image encryption based on one-time keys and robust chaotic maps," *Comput. Math. Appl.*, vol. 59, no. 10, pp. 3320–3327, 2010.

[6] P. Stavroulakis and M. Stamp, *Handbook of Information and Communication Security*, 1st ed. Berlin, Germany: Springer, 2014.

[7] A. Belazi, R. Rhouma, and S. Belghith, "A novel approach to construct S-box based on Rossler system," in *Proc. Int. Wireless Commun. Mobile Comput. Conf. (IWCMC)*, Aug. 2015, pp. 611–615.

[8] A. Belazi, A. A. A. El-Latif, R. Rhouma, and S. Belghith, "Selective image encryption scheme based on DWT, AES S-box and chaotic permutation," in *Proc. Int. Wireless Commun. Mobile Comput. Conf. (IWCMC)*, Aug. 2015, pp. 606–610.

[9] A. A. Alzaidi, M. Ahmad, M. Doja, E. Al Solami, and M. S. Beg, "A new 1D chaotic map and β-hill climbing for generating substitution-boxes," *IEEE Access*, vol. 6, pp. 55405–55418, Oct. 2018.

[10] C. Zhu and K. Sun, "Cryptanalyzing and improving a novel color image encryption algorithm using RT-enhanced chaotic tent maps," *IEEE Access*, vol. 6, pp. 18759–18770, Mar. 2018.

[11] J.-X. Chen, Z.-L. Zhu, and H. Yu, "A fast chaos-based symmetric image cryptosystem with an improved diffusion scheme," *Optik-Int. J. Light Electron Opt.*, vol. 125, no. 11, pp. 2472–2478, Jun. 2014.

[12] X. Wu, B. Zhu, Y. Hu, and Y. Ran, "A novel color image encryption scheme using rectangular transform-enhanced chaotic tent maps," *IEEE Access*, vol. 5, pp. 6429–6436, 2017.

[13] Y. Zhang and D. Xiao, "An image encryption scheme based on rotation matrix bit-level permutation and block diffusion," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 19, no. 1, pp. 74–82, 2014.

[14] Y. Zhou, Z. Hua, C.-M. Pun, and C. L. P. Chen, "Cascade chaotic system with applications," *IEEE Trans. Cybern.*, vol. 45, no. 9, pp. 2001–2012, Sep. 2015.

[15] X. Wang, L. Liu, and Y. Zhang, "A novel chaotic block image encryption algorithm based on dynamic random growth technique," *Opt. Lasers Eng.*, vol. 66, pp. 10–18, Mar. 2015.

[16] Y.-Q. Zhang and X.-Y. Wang, "A new image encryption algorithm based on non-adjacent coupled map lattices," *Appl. Soft. Comput.*, vol. 26, pp. 10–20, Jan. 2015.

[17] S. Wolfram, "Cryptography with cellular automata," in *Proc. Conf. Theory Appl. Cryptograph. Techn.* Berlin, Germany: Springer, Nov. 1985, pp. 429–432.

[18] R.-J. Chen and J.-L. Lai, "Image security system using recursive cellular automata substitution," *Pattern Recognit.*, vol. 40, no. 5, pp. 1621–1631, May 2007.

[19] R.-J. Chen and S.-J. Horng, "Novel SCAN-CA-based image security system using SCAN and 2-D von Neumann cellular automata," *Signal Process., Image Commun.*, vol. 25, no. 6, pp. 413–426, Jul. 2010.

[20] A. A. Abdo, S. Lian, I. A. Ismail, M. Amin, and H. Diab, "A cryptosystem based on elementary cellular automata," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 18, no. 1, pp. 136–147, 2013.

[21] J. Jin, "An image encryption based on elementary cellular automata," *Opt. Lasers Eng.*, vol. 50, no. 12, pp. 1836–1843, 2012.

[22] A. M. del Rey, G. R. Sánchez, and A. De La Villa Cuenca, "Encrypting digital images using cellular automata," in *Proc. Int. Conf. Hybrid Artif. Intell. Syst.* Berlin, Germany: Springer, 2012, pp. 78–88.

[23] X. Wang and D. Luan, "A novel image encryption algorithm using chaos and reversible cellular automata," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 18, no. 11, pp. 3075–3085, 2013.

[24] A. Bakhshandeh and Z. Eslami, "An authenticated image encryption scheme based on chaotic maps and memory cellular automata," *Opt. Lasers Eng.*, vol. 51, no. 6, pp. 665–673, 2013.

[25] A. Souyah and K. M. Faraoun, "Fast and efficient randomized encryption scheme for digital images based on quadtree decomposition and reversible memory cellular automata," *Nonlinear Dyn.*, vol. 84, no. 2, pp. 715–732, Apr. 2016.

[26] A. Souyah and K. M. Faraoun, "An image encryption scheme combining chaos-memory cellular automata and weighted histogram," *Nonlinear Dyn.*, vol. 86, no. 1, pp. 639–653, Oct. 2016.

[27] C. Paar and J. Pelzl, *Understanding Cryptography: A Textbook for Students and Practitioners*, 1st ed. Berlin, Germany: Springer, 2009.

[28] J. V. Neumann, *Theory of Self-Reproducing Automata*, A. W. Burks, Ed. Champaign, IL, USA: Univ. of Illinois Press, 1966.

[29] K. C. Clarke, "Cellular automata and agent-based models," in *Handbook of Regional Science*. Berlin, Germany: Springer, 2014, pp. 1217–1233.

[30] S. Wolfram, "Random sequence generation by cellular automata," *Adv. Appl. Math.*, vol. 7, pp. 123–169, 1986.

[31] S. Wolfram, *A New Kind of Science*, vol. 5. Champaign, IL, USA: Wolfram Media, 2002.

[32] Y. Zhou, L. Bao, and C. P. Chen, "A new 1D chaotic system for image encryption," *Signal Process.*, vol. 97, no. 11, pp. 172–182, 2014.

[33] Computer Vision Group, University of Granada (CVG-UGR). *Image Database*. Accessed: Jan. 1, 2019. [Online]. Available: http://decsai.ugr.es/cvg/dbimagenes/

[34] Z. Hua and Y. Zhou, "Design of image cipher using block-based scrambling and image filtering," *Inf. Sci.*, vol. 396, pp. 97–113, Aug. 2017.

[35] N. D. Gagunashvili, "CHICOM: A code of tests for comparing unweighted and weighted histograms and two weighted histograms," *Comput. Phys. Commun.*, vol. 183, no. 1, pp. 193–196, Jan. 2012.

[36] N. D. Gagunashvili, "CHIWEI: A code of goodness of fit tests for weighted and unweighted histograms," *Comput. Phys. Commun.*, vol. 183, no. 2, pp. 418–421, Oct. 2012.

[37] Y.-Q. Zhang and X.-Y. Wang, "A symmetric image encryption algorithm based on mixed linear–nonlinear coupled map lattice," *Inf. Sci.*, vol. 273, pp. 329–351, Jul. 2014.

[38] Y. Wu, Y. Zhou, G. Saveriades, S. Agaian, J. P. Noonan, and P. Natarajan, "Local Shannon entropy measure with statistical tests for image randomness," *Inf. Sci.*, vol. 222, pp. 323–342, Feb. 2013.

[39] Y. Wu, J. P. Noonan, and S. Agaian, "NPCR and UACI randomness tests for image encryption," *Cyber J., Multidiscipl. J. Sci. Technol., J. Sel. Areas Telecommu.*, vol. 1, pp. 31–38, Apr. 2011.

[40] X. Wang, L. Teng, and X. Qin, "A novel colour image encryption algorithm based on chaos," *Signal Process.*, vol. 92, no. 4, pp. 1101–1108, Apr. 2012.

[41] C. A. R. Hoare, "Algorithm 64: Quicksort," *Commun. ACM*, vol. 4, no. 7, p. 321, Jul. 1961.

[42] P. L'Ecuyer, "Uniform random number generation," *Ann. Oper. Res.*, vol. 53, no. 1, pp. 77–120, Dec. 1994.

MD NAZISH ASLAM received the M.Tech. degree in control and instrumentation system from the Department of Electrical Engineering, Jamia Millia Islamia University, New Delhi, India, in 2016.

From 2016 to 2017, he was with PIRO Technologies Pvt. Ltd., New Delhi, as a Research Associate and has been a Senior Research Associate, since 2017, where he is currently involving in image processing, wireless networks, image cryptography, and machine learning. He has also delivered various research oriented training program to the research scholars and faculty members in the above-mentioned fields. His research interests include cryptography, machine learning, and deep learning.

AKRAM BELAZI received the B.Eng. degree in telecommunications and networks from the National Engineering School of Gabes, ENIG Tunisia, in 2011, the M.Sc. degree in electronic systems and networks communications from the Tunisia Polytechnic School, EPT, in 2013, and the Ph.D. degree in telecommunications from the National Engineering School of Tunis, ENIT Tunisia, in 2017. His main research interests include multimedia cryptography, machine learning, deep learning, and optimization techniques.

SOFIANE KHARBECH (M'17) received the Engineering degree in networking and telecommunications from the National Institute of Applied Science and Technology, Tunis, Tunisia, in 2009, the M.Sc. degree in electronic systems and communication networks from the Tunisia Polytechnic School, Carthage, Tunisia, in 2012, and the Ph.D. degree in electrical engineering from the University of Valenciennes and Hainaut-Cambresis, Valenciennes, France, in 2015.

From 2012 to 2015, he was a Research Engineer with the Laboratory IEMN/DOAE (CNRS UMR 8520, France). Since 2015, he has been an Assistant Professor with the Higher Institute for Technological Studies of Gabes, Gabes, Tunisia, and a Senior Researcher with the Laboratory Sys'Com (ENIT, Tunisia). His main research interests include cognitive radio, wireless communications, and cryptography.

MUHAMMAD TALHA received the Ph.D. degree in computer science from the Faculty of Computing, Universiti Teknologi Malaysia. He is currently a Researcher with the Deanship of Scientific Research, King Saud University, Riyadh, Saudi Arabia. He has authored more than 25 international journals and conferences. His research interests include image processing, medical imaging, features extraction, classification and machine learning techniques.

WEI XIANG (S'00–M'04–SM'10) received the B.Eng. and M.Eng. degrees in electronic engineering from the University of Electronic Science and Technology of China, Chengdu, China, in 1997 and 2000, respectively, and the Ph.D. degree in telecommunications engineering from the University of South Australia, Adelaide, Australia, in 2004. From 2004 to 2015, he was with the School of Mechanical and Electrical Engineering, University of Southern Queensland, Toowoomba, QLD, Australia. He is currently the Founding Professor and the Head of the Discipline of Internet of Things Engineering, James Cook University, Cairns, QLD, Australia. Due to his instrumental leadership in establishing Australia's first accredited Internet of Things Engineering degree program, he was selected into Pearcy Foundation's Hall of Fame, in 2018. He has published over 250 peer-reviewed papers with over 130 journal articles. He has severed in a large number of international conferences in the capacity of General Co-Chair, TPC Co-Chair, and Symposium Chair. His research interests include communications and information theory, particularly the Internet of Things, and coding and signal processing for multimedia communications systems. He is an elected fellow of the IET in U.K. and Engineers Australia. He received the TNQ Innovation Award, in 2016, and the Pearcey Entrepreneurship Award, in 2017, and Engineers Australia Cairns Engineer of the Year, in 2017. He was a co-recipient of four Best Paper Awards system 2019 WiSATS, 2015 WCSP, 2011 IEEE WCNC, and 2009 ICWMC. He has been awarded several prestigious fellowship titles. He was named a Queensland International Fellow, from 2010 to 2011, by the Queensland Government of Australia, an Endeavour Research Fellow, from 2012 to 2013, by the Commonwealth Government of Australia, a Smart Futures Fellow, from 2012 to 2015, by the Queensland Government of Australia, and a JSPS Invitational Fellow jointly by the Australian Academy of Science and Japanese Society for Promotion of Science, from 2014 to 2015. He is the Vice Chair of the IEEE Northern Australia Section. He was an Editor for IEEE COMMUNICATIONS LETTERS, from 2015 to 2017, and is an Associate Editor for *Telecommunications Systems* (Springer).