Multimed Tools Appl (2016) 75:14685–14706
DOI 10.1007/s11042-016-3274-9

A fast ARX model-based image encryption scheme

Jongseok Choi[1] · Seonhee Seok[1] · Hwajeong Seo[1] · Howon Kim[1]

Received: 22 February 2015 / Revised: 5 November 2015 / Accepted: 11 January 2016 /
Published online: 2 February 2016

Abstract This paper proposes a novel ARX model-based image encryption scheme that uses addition, rotation, and XOR as its confusion and diffusion mechanism instead of S-Box and permutation as in SP networks. The confusion property of the proposed scheme is satisfied by rotation and XOR with chaotic sequences generated from two logistic maps. Unlike classical image encryption schemes that adopt S-Box or permutation of the entire plain image, the diffusion property is satisfied using addition operations. The proposed scheme exhibits good performance on correlation coefficients (horizontal, vertical and diagonal), Shannon's entropy and NPCR (Number of Pixels Change Rate). Furthermore, simulation results indicate that its time complexity is 9.2 times more efficient than the fastest algorithm(Yang's algorithm).

Keywords Image encryption · Chaotic encryption · ARX

✉ Howon Kim
howonkim@gmail.com

Jongseok Choi
js.choi.85@gmail.com

Seonhee Seok
seokseonhee@gmail.com

Hwajeong Seo
hwajeong84@gmail.com

[1] Pusan National University, Room 6512, 6-Eng. Bldg., Jangjeon 2(i)-dong, Geumjeong-gu, Busan, 609-735 Republic of Korea

1 Introduction

Technological developments and the increasing prevalence of internet and mobile devices have resulted in SNSs (Social Network Services) being used all over the world and producing significant amounts of digital content. Images and videos are the content types that are the most familiar to users, and they can be used to deliver information effectively. However, such contents can infringe on the privacy of individuals because they may contain private information that can be used to identify individuals, from the contents themselves or via advanced image processing technologies such as face recognition and machine learning. From a privacy viewpoint, the dangers inherent in the characteristics of digital contents, such as ease of modification, fabrication, and distribution, can result in the privacy infringement situation being exacerbated. Consequently, studies are actively investigating methods for preventing forgeries and protecting privacy and copyrights through proper cryptographic approaches such as encryption and DRM (Digital Rights Management).

Naturally, the simplest image encryption method [6, 7, 24, 28] is to encrypt images by only conducting XOR with a plain image and secret key sequences. Unfortunately, these algorithms are vulnerable to chosen-plaintext attacks [1, 23]. Consequently, many other encryption schemes that are not susceptible to this type of attack have been proposed [3, 5, 12, 13, 21].

Recently, double image encryption schemes [14, 15, 17–19, 30] have attracted research attention. These schemes can improve robustness because they perform encryption using two original images. However, simply evaluating the randomness of a cipher image is not sufficient to guarantee a scheme's security.

Many image encryption schemes using traditional symmetric cryptography, such as AES (Advanced Encryption Standard) and DES (Data Encryption Standard) [2, 29], as well as asymmetric cryptography, such as RSA and ECC [10, 32], have been proposed to prevent forgery and safeguard privacy. However, the direct use of conventional cryptographic algorithms is fraught with difficulties because they were designed to encrypt text data transmission. In contrast to text data transmission, in the image encryption field, however, fast and real-time interaction between the camera and the storage is required. Consequently, many studies have proposed specialized image encryption techniques [4, 8, 22].

Chaotic map-based image encryption methods are widely used as a mathematical tool to provide secret key sequences from simple equations. A chaotic map such as the Arnold cat map and generalized cat map are usually used to realize the diffusion property [4, 9]. Furthermore, as most image encryption schemes aim to increase randomness and reduce time complexity, chaotic map-based image encryption schemes generate more randomness and lower time complexity than conventional encryption schemes.

The performance of these schemes can be evaluated via statistical analysis and time complexity. Statistical analysis, which is generally used to evaluate randomness, can be performed on the histogram and entropy of encrypted images and the correlation coefficients between a cipher image and its original image. Further, NPCR (Number of Pixels Change Rate) and UACI (Unified Average Changing Intensity), proposed by Wu et al. [27], can be used to evaluate the randomness of image encryption. Most of the schemes previously proposed are unable to satisfy either NPCR or UACI. Some schemes, such as [16, 26, 31, 33], are able to satisfy either or both of the tests. Even if most of the schemes exhibit high performance on NPCR and UACI, they also exhibit high time complexity.

In this paper, we propose a new ARX (Addition, Rotation, and XOR) model-based image encryption scheme. Our proposed scheme performs encryption and decryption using only addition, rotation, and XOR. To satisfy Shannon's properties, it performs confusion and

diffusion processes generated via addition operations, in contrast to earlier schemes [4, 9] that use cat maps for permutation processes. The results of evaluations conducted of the proposed scheme on key space analysis, statistical analysis, sensitivity, and time complexity show that it has outstanding performance for statistical analysis and time complexity. In addition, the results of comparisons with Wang's algorithm and Yang's algorithm indicate that in terms of time complexity, it is about ten times more efficient than Yang's algorithm, which is the fastest algorithm to date.

The remainder of this paper is organized as follows:

We discuss logistic maps and LEA encryption algorithms that utilize ARX in Section 2. Section 3 presents our proposed ARX model-based image encryption scheme. In Section 4 we evaluate and compare the proposed scheme to other schemes. Finally, we present our conclusions in Section 5.

2 Preliminaries

2.1 Logistic map

The logistic map proposed by May [20] in 1976 is a chaotic map with a chaos feature. The chaos feature means that a small difference in initial conditions has a significant effect on the result. The proposed map is defined by (1).

$$X_{t+1} = \mu X_t (1 - X_t) \tag{1}$$

The logistic map is defined by (1) for $0 < X_0 < 1$ and $0 < \mu \leq 4$. However, the chaotic feature can be observed when $3.56 < \mu \leq 4$ because a logistic map has a fractal structure.

Figure 1b shows the structure of the map for $\mu = 4$. Following the proposal of the first cipher using a chaos map in the 1990s, they have since been actively studied. This has resulted in logistic maps, which are a type of chaos map, being used extensively in the image encryption field, owing to its simple equation.

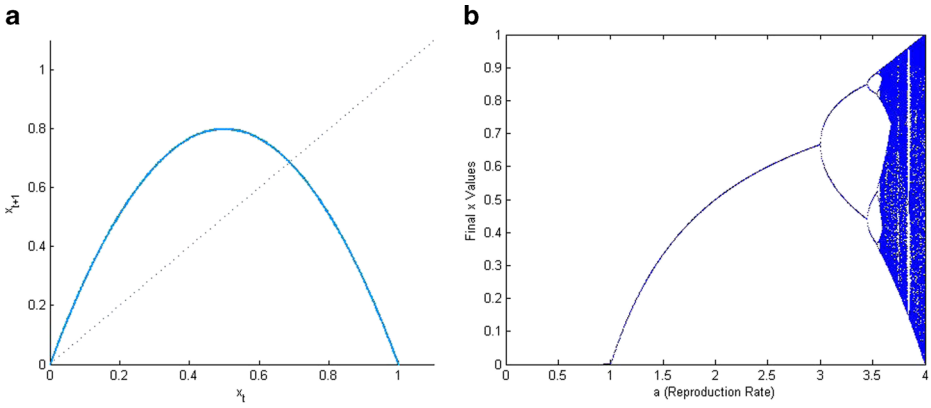

Fig. 1 The logistic function: **a** function graph, **b** bifurcation diagram

2.2 LEA: a lightweight 128-bit block cipher

LEA, proposed by Hong et al. [11], is a lightweight block cipher that can utilize a 128-bit, 192-bit, or 256-bit key. LEA carries out encryption and decryption at a higher speed than traditional block ciphers because it uses simpler operations. It employs only three types of operations: modular Addition, bitwise Rotation, and bitwise XOR (ARX). LEA's en/decryption speed is faster than that of AES in both software and hardware environments. Furthermore, its performance is optimized on 32-bit and 64-bit processors. Table 1 shows the LEA encryption algorithm.

To encrypt plain data, LEA starts key expansion from the $keySchedule$ function and iteration of the $Round$ function. The $keySchedule$ function generates 192-bit round keys from secret key K. Then, it encrypts plaintext X_i to ciphertext X_{i+1} using round keys RK_i and round function $Round(X_i, RK_i)$ which consists of ARX operations. Fast en/decryption is enabled by the $Round$ function, which comprises a simple combination of ARX operations. Table 2 outlines the procedure used by the LEA $Round$ function.

Notation 1 *Addition ($x \boxplus y$) : An operation defined as $IntToBit\ (\ (BitToInt(x) + BitToInt(y)\ )\ mod\ 2^{32})$ for 32-bit array x and y.*

- ⁎ **1** *BitToInt (x) : A function that returns an integer $n = x_0 \cdot 2^0 + x_1 \cdot 2^1 + \cdots + x_{31} \cdot 2^{31}$ for 32-bit binary stream $x = x_{31}||x_{30}||\cdots||x_0$.*
- ⁎ **2** *IntToBit (n) : A function that returns a 32-bit binary stream $x = x_{31}||x_{30}||\cdots||x_0$ for an integer $n = x_0 \cdot 2^0 + x_1 \cdot 2^1 + \cdots + x_{31} \cdot 2^{31}$.*

Notation 2 *Rotation ($ROR_n(x)$ or $ROL_n(x)$) : Used for 32-bit bitstream x, n-bit right or left rotation.*

Notation 3 *XOR ($x \oplus y$) : An exclusive OR for two bitstreams x and y that have the same length.*

3 Proposed encryption

In this section, we propose an ARX (modular Addition, bitwise Rotation, and eXclusive OR)-based image encryption scheme. A combination of S-Box and chaotic map is commonly used to satisfy Shannon's confusion and diffusion properties in the image encryption field. AES and DES, which are used to be the most common techniques, have proved the

Table 1 LEA encryption algorithm

Simple code for an encryption function : $C \leftarrow Encrypt(P, RKset)$
Input : P is a 128-bit plaintext, RK is a round key set.
Output : C is a 128-bit ciphertext.
1 : $X_0 \leftarrow P$
2 : $for\ i = 0\ to\ numberOfRound$
3 : $\quad X_{i+1} \leftarrow Round(X_i, RK_i)$
4 : $end\ for$
5 : $C \leftarrow X_{numberOfRound}$

Table 2 A round function of LEA

i-th round function : $X_{i+1} \leftarrow Round(X_i, RK_i)$
Input : X_i is a 128-bit value of previous round function, RK_i is a 192-bit round key.
Output : X_{i+1} is a 128-bit value.
$1 : X_{i+1}[0] \leftarrow ROL_9((X_i[0] \oplus RK_i[0]) \boxplus (X_i[1] \oplus RX_i[1]))$
$2 : X_{i+1}[0] \leftarrow ROR_5((X_i[1] \oplus RK_i[2]) \boxplus (X_i[2] \oplus RX_i[3]))$
$3 : X_{i+1}[0] \leftarrow ROR_3((X_i[2] \oplus RK_i[4]) \boxplus (X_i[3] \oplus RX_i[5]))$
$4 : X_{i+1}[0] \leftarrow X_i[0]$

security of S-Box in previous years. Further, these two algorithms are designed to perform on lightweight devices. Consequently, the S-Box of these algorithms can be designed with six bits or eight bits. The S-Box of AES is different from that of DES, which has not been proved mathematically, and can be used in the image encryption and various other fields because it has been mathematically proved to be nonlinear. However, most applications implement S-Box in the form of a lookup table, in which case a large memory space is required for storing its values. The amount of memory required can be calculated using (2).

$$row \times column \times bits \times sboxes \tag{2}$$

Eq. (2) gives the amount of memory needed to store the respective S-Box of AES and DES as 512 bytes and 256 bytes, respectively. The S-Box of AES can be implemented without a lookup table, but such a scenario results in high computational complexity because the Galois field has to be calculated. We propose an ARX model-based image encryption algorithm that is both appropriate for lightweight devices and overcomes these disadvantages. The proposed algorithm also consists of confusion and diffusion processes. The main property of the proposed algorithm is that, unlike existing S-Box implementations, no substitution process is used. Instead of substitution, the proposed algorithm employs Addition, Rotation, and XOR for its encryption logic (Fig. 2).

The encryption process in the proposed scheme comprises three phases: XOR phase, Round phase, and Rotation phase. The XOR and Rotation phases are used to satisfy the confusion property, while the Round phase is used to satisfy the diffusion property. In the confusion process, two key sequences are calculated via two logistic maps using two key pairs. The decryption process is the inverse of the encryption process.

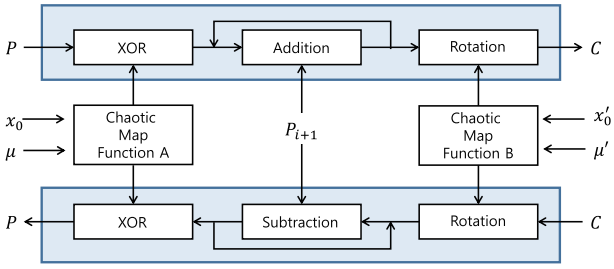

Fig. 2 The entire concept of ARX based image encryption

3.1 Confusion process

In the confusion process, several bits of ciphertext are changed whenever a bit of a key is changed in order for the process to satisfy Shannon's confusion property. In the proposed algorithm's confusion process, the parameters of the logistic maps are the secret keys(μ, x_0, μ' *and* x_0') and they confuse the entire cipherimage via the XOR and Rotation phases. To encrypt the image pixels as a unit, two random sequences generated by the logistic maps are divided by eight bits. One of the sequences is the addition operation parameter, and the other uses only three bits for the rotation operation parameter. Two logistic maps are calculated independently in the XOR and Rotation phases of our scheme. They help to reduce the memory space required to save the secret information, thereby generating a sequence using secret keys. The logistic map can be represented by (3):

$$x_{n+1} = \mu x_n(1 - x_n) \tag{3}$$

where μ is a control parameter of the logistic map and the initial value x_0 of the random variable is in the range $(0 < x_0 < 1)$. x_1 to x_n is a chaotic sequence with a pattern determined by μ and x_0.

3.2 Diffusion process

Shannon's diffusion property specifies that the corresponding cipher text should have a completely different value each time the plaintext is changed by one bit. The proposed algorithm performs the diffusion process as an accepting Addition calculation. This addition operation is calculated using the previous plaintext value and the IV (Initial Vector). The addition process by which IV is updated is defined by (4):

$$IV_{n+1} = IV_n \oplus P_n \tag{4}$$

Each bit of the original image has an effect on the next cipher image in that the default value is configured to IV in the first round of the proposed encryption technique. From the second round, one bit of the original image has an effect on the overall cipher image because IV accumulates all the pixel values of original image at the first round.

3.3 Encryption

The overall encryption process combines confusion logic and diffusion logic in pixel units using two secret key pairs(μ, x_0, μ', x_0'). In the initialization process, the plain image and chaotic value are calculated using secret key μ, x_0 is calculated via the first logistic map, such as (5), and eight bits are cut from the generated chaotic sequence and used as the chaotic value.

$$\begin{aligned} x_{n+1} &= \mu x_n(1 - x_n) \\ C_m &= P_m \oplus truncate_8(x_{n+1}) \end{aligned} \tag{5}$$

In (5), n is the iterator of the chaotic map and is in the range $0 \leq n < T$, where T is the number of pixels in the original image. In the next process, the diffusion property is satisfied by performing calculations on the surrounding pixels and addition of the plaintext. The diffusion rate is up to the number of encryption rounds in the *Round* function defined below (Table 3).

The *Round* function consists of addition and the XOR operations for the diffusion property. Diffusion through addition is performed using (6) in the *Round* function. As given in

Table 3 Round process of proposed scheme

r-th round function : $C^{r+1} \leftarrow Round(C^r, IV_N^r)$
Input : C^r is a result image of previous *Round* function,
IV_N^r is a 8-bit last value of r-th *Round* function.
Output : C^{r+1} is a result image with N Pixels.
1 : $for\ i = 1\ to\ N$(N is the number of image pixel)
2 : $IV_{i+1}^r \leftarrow IV_i^r \oplus C_i^r$
3 : $C_{i+1}^r \leftarrow IV_i^r \boxplus C_i^r$
4 : $end\ for$

(6), IV is influenced by the chaotic value and pixel of the previous plain image. In (6), $\boxplus$ represents modular addition. In the proposed scheme, the modular addition is calculated on the modulus 2^8 because the grayscale pixel is expressed as eight bits.

$$\begin{aligned} IV_{n+1} &= IV_n \oplus C_{m+1} \\ C_m &= C_m \boxplus IV_n \end{aligned} \tag{6}$$

Another property of our scheme is that the *Round* function iterates only the process of (6). As a result of these properties, the proposed algorithm can be implemented in three types of processes. Further, it is possible to construct a pipeline when it is implemented in hardware.

$$\begin{aligned} x_{n+1} &= x_n' r'(1 - x_n') \\ k &= truncate_3(x_{n+1}) \\ C_m &= Rotate_k(C_m) \end{aligned} \tag{7}$$

Table 4 Encryption algorithm of proposed scheme

Simple code for encryption process : $C \leftarrow Encrypt\ (P, keys)$
Input : P is a plain image,
$keys$ are parameters(x_0, μ, x_0', μ') for logistic maps.
Output : C is a cipher image.
Generate chaotic sequences, IV
1 : $x^a \leftarrow GenerateChaoticSeq(x_0, \mu)\ for\ XOR\ phase$
2 : $x^b \leftarrow GenerateChaoticSeq(x_0, \mu)\ for\ Rotation\ phase$
3 : $IV^0 \leftarrow SelectRandomValue()$

Phase 1. XOR with chaotic sequence x^a
4 : $C^0 \leftarrow P \oplus x^a$

Phase 2. Round
5 : $for\ i = 1\ to\ 7$
6 : $C^i \leftarrow Round(C^{i-1}, IV^{i-1})$
4 : $end\ for$

Phase 3. Rotation with chaotic sequence x^b
7 : $C \leftarrow C^7 \boxplus x^b$

Rotation of C_i, which is performed after the last execution of the *Round* function, is calculated using the chaotic sequence made by the second key pair μ', x_0' and the logistic map. Rotation calculation on a pixel is meaningless when performed only once because the bit order is not mixed or changed. The truncate function generates a rotation key by cutting three bits from the chaotic sequence, generating eight bits each time. In other words, the result of the Rotation calculation has virtually no effect on randomness in terms of histogram analysis because the Rotation calculation can have only one result among the resulting values of the eight units made by moving the eight unit bits (Table 4).

The encryption process comprises the following steps:

Step 1. Initialize the parameters (μ, x_0, μ', x_0') for two logistic maps and select the IV randomly.
Step 2. Generate the first chaotic sequence(x^a) using (3) with x_0 and μ;.
Step 3. Calculate the XOR of the chaotic sequences and pixels, $C_m = P_m \oplus x_m$.
Step 4. Update IV by calculating the XOR of IV and C_m. $IV_{n+1} = IV_n \oplus C_m$.
Step 5. Calculate the addition value of (3) and (4), $C_m = IV_{n \bmod m}^{floor(\frac{n}{m})} \boxplus C_m$
Step 6. Generate the second chaotic sequence(x^b) by using x_0', μ' in (3)
Step 7. Perform rotation using the second chaotic sequence, $C_m = Rotate_{x_m^b}(C_m)$

In the steps above, m signifies the pixel position in each round and IV is not dependent on the round, so it has the range $0 < n < T \times rounds$ where T is the total number of original image pixels. Figure 3 shows the encryption process logic in detail. The keys are obtained from the logistics maps, which are two chaotic maps. For addition calculation, we use the chain of IV accumulating pixels in the original image. As depicted in Fig. 3, the

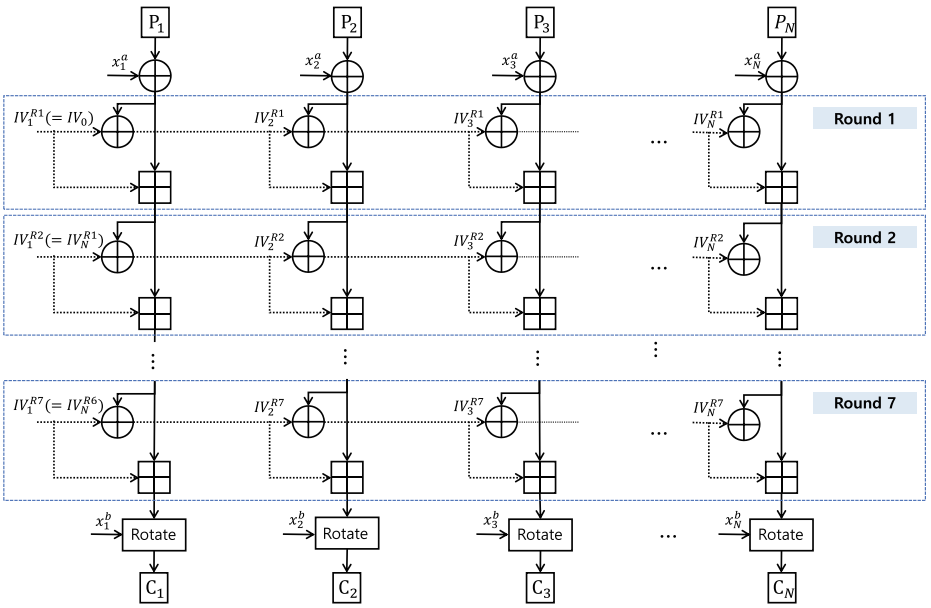

Fig. 3 Encryption process of the proposed scheme

rounds of chaining blocks are iterated through addition calculation to increase the diffusion effect of the proposed technique.

3.4 Simulated result

The proposed encryption/decryption algorithm was implemented using Xcode on a computer with the following system configuration: 2.4 GHz CPU, 4 GB RAM, 500 GB HDD, Mac OS X Yosemite. We used the 512 × 512 Lena grayscale image and $x_0 = 0.02$, $\mu = 3.923$, $x_0' = 0.005$ and $\mu' = 3.955$ as the secret key in the encryption/decryption performance evaluation.

4 Security analysis

In this section, we discuss the evaluation conducted of our scheme in terms of key space, statistical analysis, sensitivity, and time complexity. For statistical analysis, we evaluated histogram, correlation, and entropy. To analyze sensitivity, the performances of NPCR and UACI were estimated using (11) and (13). Subsequently, we compared the results with those of Wang's algorithm and Yang's algorithm.

4.1 Key space analysis

The key space needs to be sufficiently large to overcome brute force attack. In cryptography, a computational complexity above 128 bits is considered sufficiently secure. In this section, we analyze whether the key space of the proposed scheme is above 128 bits. The keys in our proposed encryption algorithm consist of two pairs of x_0 and r from the logistic map. More specifically, x_0, x_0', μ and μ' are used for our proposed encryption process. Each parameter can be 10-bits, 23-bits, 52-bits, or 112-bits according to the floating-point precision standards of [25]. Therefore, our proposed encryption satisfies the security criteria from a cryptographic perspective because the key length of our scheme can support up to 448-bits.

4.2 Statistical analysis

We performed statistical security analysis of the proposed encryption with three analyses: histogram, correlation, entropy. In the histogram analysis, we evaluated the fairness of color distribution. To estimate the relationship between the original image and the encrypted image, we calculated their correlation coefficient for three directional features. In the final statistical analysis, we proved the randomness of the proposed encryption through Shannon's entropy.

4.2.1 Histogram analysis

We used the Lena 512 × 512 grayscale image as a sample image in our histogram analysis. First, we evaluated the color distribution of the diffusion process. Figure 4 depicts the addition feature, which is a key operation of the diffusion process. As can be seen, the addition gradually hides the original image. In other words, the addition has no periodic feature, unlike the Arnold cat map. Figure 5 shows the variances in the addition histogram. In the

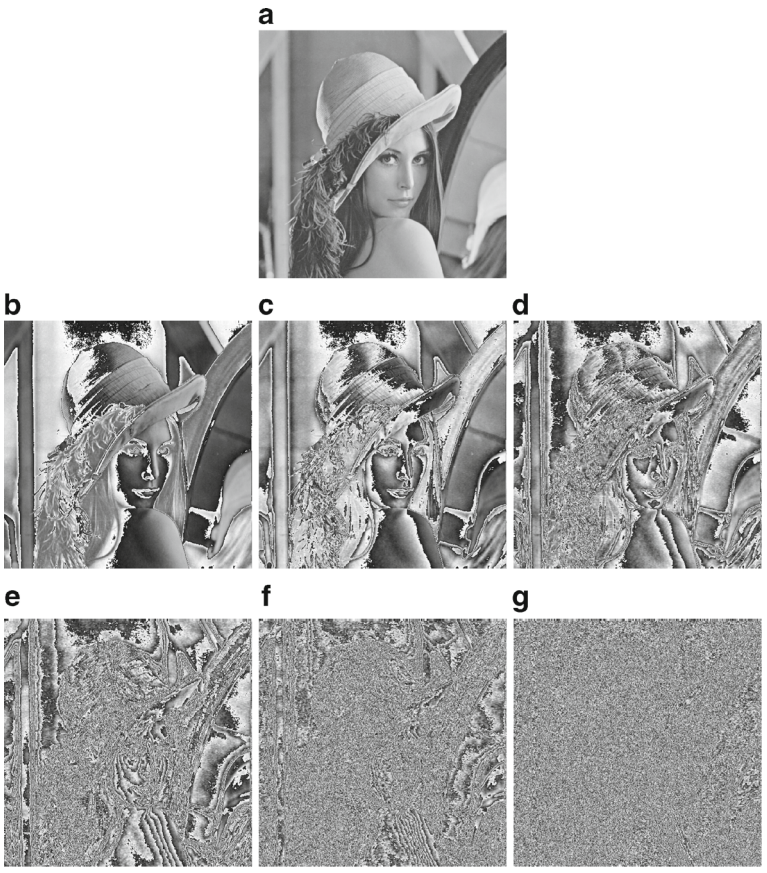

Fig. 4 The feature of addition operation: **a** Original image, **b** 1-round addition, **c** 2-rounds addition, **d** 3-rounds addition, **e** 4-rounds addition, **f** 5-rounds addition, **g** 6-rounds addition

case of addition at the first round, the distribution is similar to that of the original image. However, as the rounds progress, the distribution becomes straighter. As shown in Fig. 5, the most uniform distribution occurs after five rounds.

The histogram in Fig. 6 shows the results of encryption and decryption. The original and decrypted images are shown as fractal graphs, which provide substantial information about the images. In contrast, the histogram of the encrypted image is uniformly distributed. Thus, our proposed encryption is sufficiently secure in terms of histogram analysis.

4.2.2 Correlation analysis

In order to evaluate the correlation, we calculated the correlation coefficients of the original image and the cipher image for the horizontal, vertical, and diagonal directions. For the correlation coefficients, we randomly chose a thousand pixels and then computed the correlation coefficients between a chosen pixel and its adjacent pixel from the original image and the encrypted image for the horizontal, vertical, and diagonal directions, respectively. Figure 7 presents the correlations between the original image and cipher image for the

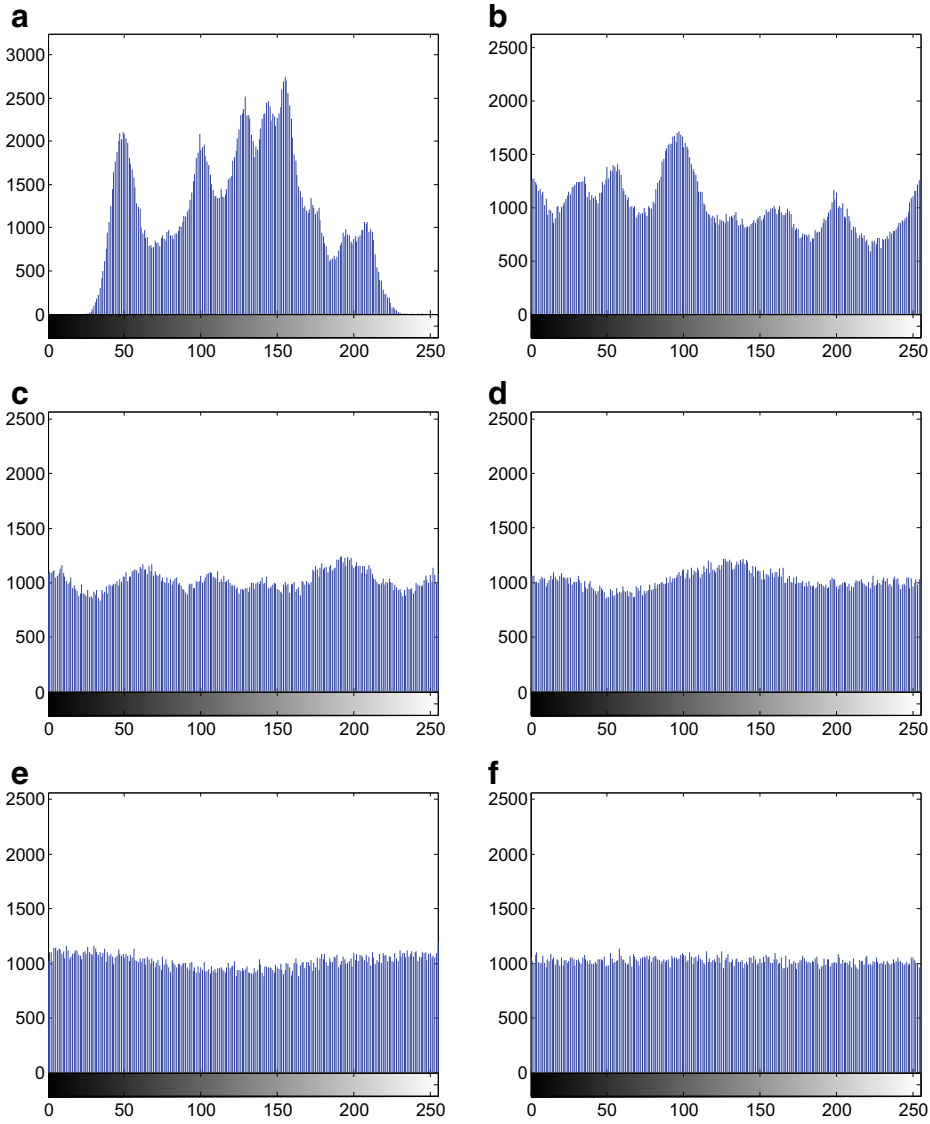

Fig. 5 Histograms of original and addition-operated images: **a** Original image, **b** 1-round addition, **c** 2-rounds addition, **d** 3-rounds addition, **e** 4-rounds addition, **f** 5-rounds addition

horizontal, vertical, and diagonal directions. In Fig. 7, it is clear that the pixels of the original image have a high dependency in all directions. Conversely, the cipher image shows negligible correlation features in all directions.

For numerical analysis of correlation, we calculated the correlation coefficients for three directions of the original and encrypted images using (8).

$$r_{xy} = \frac{cov(x, y)}{\sqrt{D(x)}\sqrt{D(y)}} \tag{8}$$

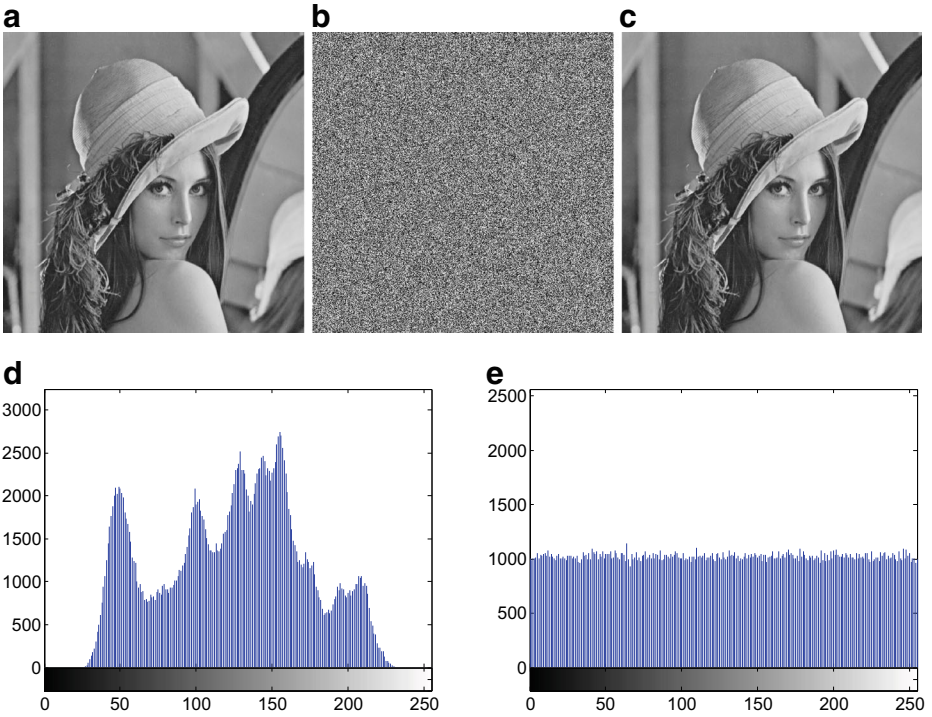

Fig. 6 Encryption and decryption results: **a** original image, **b** cipher image, **c** decrypted image, **d** histogram of original image, **e** histogram of histogram of cipher image

where $cov(x, y)$ and $D(x)$ can be calculated using (9)

$$\begin{aligned} cov(x, y) &= \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))(y_i - E(y)), \\ D(x) &= \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))^2, \\ E(x) &= \frac{1}{N}\sum_{i=1}^{N}x_i \end{aligned} \tag{9}$$

In (8) and (9), x is a set of selected pixels and y is a set of adjacent pixels to x for the three directions. Assume $x = p(i, j)$, then y becomes $p(i, j+1)$, $p(i+1, j)$, or $p(i+1, j+1)$ for the horizontal, vertical, or diagonal directions, respectively. The correlation coefficients are presented in Table 5. There is high correlation in the original image, but the cipher image shows negligible correlation coefficients.

4.2.3 Information entropy

Information entropy is one of the key methods used to measure randomness. We calculated the entropy of the images using (10). Let $p(x)$ be the probability of occurrence of x and N be $2^b - 1$ where b is the bit length per pixel. A grayscale image can be 8-bit or 16-bit. In

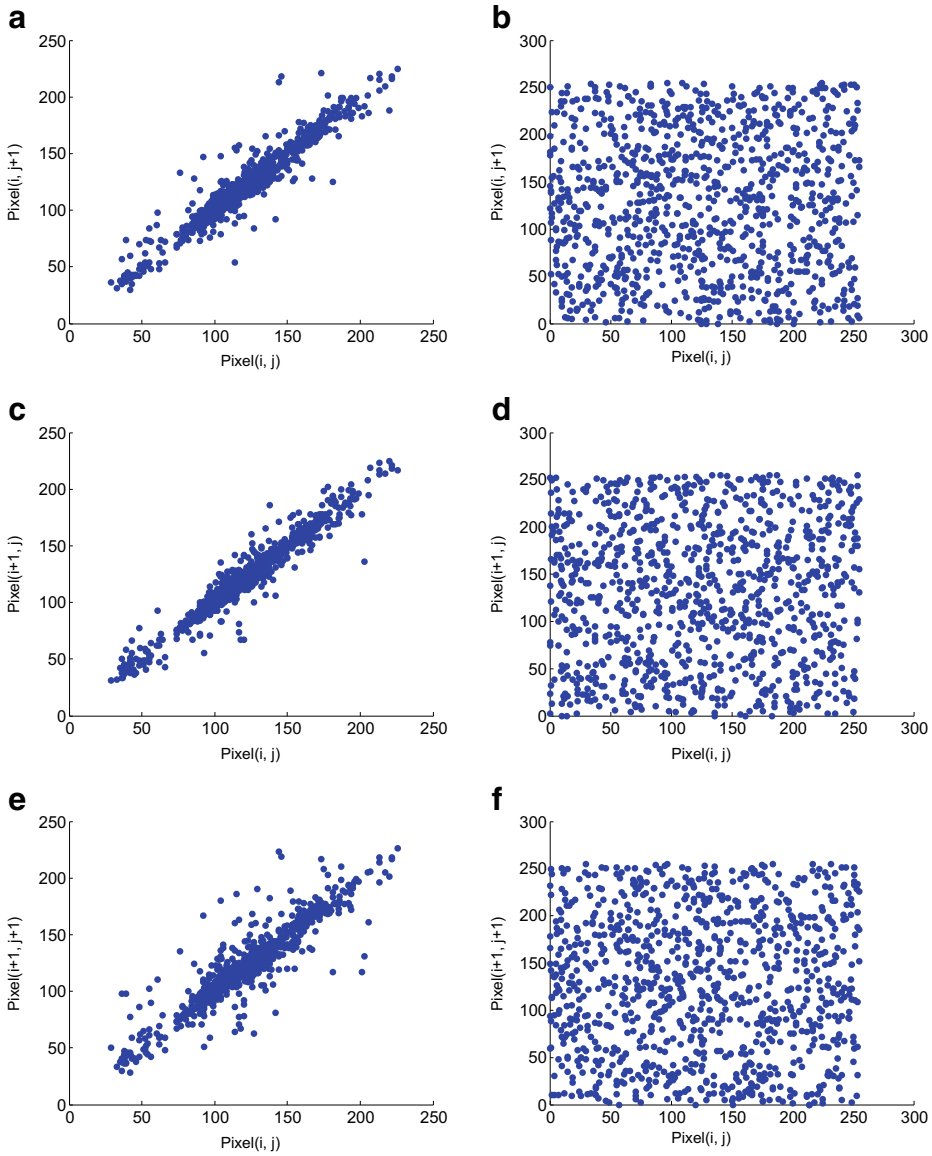

Fig. 7 Correlations: **a** horizontal correlation of original image, **b** horizontal correlation of cipher image, **c** vertical correlation of original image, **d** vertical correlation of cipher image, **e** diagonal correlation of original image, **f** diagonal correlation of cipher image

this study, all of the grayscale images used were based on 8-bit grayscale; thus, N was 255. Because the bit length per pixel was 8-bit, it is called "true random" when the entropy is eight. In other words, the algorithm grants higher randomness closer to eight.

$$H(s) = \sum_{i=1}^{N} p(s_i) log_2 \frac{1}{p(s_i)} \tag{10}$$

Table 5 Correlation coefficients with one round cipher image

	Original image	Cipher image
Horizontal	0.9706754758	−0.0530156415
Vertical	0.9838773042	−0.0246287836
Diagonal	0.9584541924	−0.0098274783

Table 6 shows the entropies of four images. The best entropy reaches up to 7.99. In addition, the entropy of the cipher image marginally depends on that of the original image. Therefore, it is similar to the ranks of the entropy between original images and cipher images. For example, the two highest entropy values both appear in the pirate image. Conversely, the two lowest entropy values are both appear in the cameraman image. Consequently, the original images with entropy values more than 7.2 are expected to have entropy values close to true randomness.

4.3 Sensitivity

Claude Shannon identified two properties of operation of a secure cipher: confusion and diffusion. Confusion is the relationship between the ciphertext and the key whereas diffusion is the relationship between the ciphertext and the plaintext. In other words, if a character of the key or the plaintext varies, several characters of the ciphertext will be changed. Modern cryptographic algorithms use a SP(substitution-permutation) network as the simplest approach to achieve both properties. For the typical case, the S-Box of AES is one of the popular methods. However this kind of substitution requires a large memory or high computational complexity.

To overcome this drawback, we adopted the ARX model to achieve both confusion and diffusion. The ARX model consists of addition, rotation, and XOR, which are very efficient in both software and hardware implementation. In order to show that our scheme satisfies Shannon's properties, we evaluated NPCR and UACI using (11) and (13).

NPCR signifies the number of pixels from the original image that have changed in the cipher image. In other words, if we assume that a plain image is 512×512 pixels, then the total number of pixels will be 2^{18}. If ten percent of the pixels are changed then NPCR would be 0.1. For confusion and diffusion tests, we compared NPCR and UACI with cipher images encrypted with 1-bit difference keys and 1-bit difference plain images.

$$NPCR = \frac{\sum_{j=1}^{H}\sum_{i=1}^{W} D(i,j)}{WH} \times 100\ \% \tag{11}$$

Table 6 Entropy with one round cipher image

	Original image	Cipher image
Lena.tif(512×512)	6.4383454397	7.8198246494
Mandril.tif(512×512)	6.9787580094	7.7923844695
Cameraman.tif(512×512)	5.6124126995	7.7771036488
Pirate.tif(512×512)	7.2367078128	7.9992695463

Table 7 NPCR performance of the four images

Round	Lena	Mandril	Cameraman	Unit : percentage(%) Pirate
1	99.6025085449	99.6112823486	99.6017456054	99.6154785156
2	99.6196746826	99.6002197265	99.6158599853	99.6345520019
3	99.6131896972	99.6105194091	99.6208190917	99.6170043945
4	99.6028900146	99.5845794677	99.6212005615	99.6078491210
5	99.6170043945	99.6025085449	99.6284484863	99.5960235595
6	99.6364593505	99.6166229248	99.6101379394	99.6063232421
7	99.5899200439	99.6089935302	99.6070861816	99.6109008789

Let W be width and H be height and D be a function. The function D returns one when $c_1(i, j)$ and $c_2(i, j)$ are the same values, otherwise it returns zero.

$$D(i, j) = \begin{cases} 1 \text{ if } c_1(i, j) \neq c_2(i, j) \\ 0 \text{ if } c_1(i, j) \equiv c_2(i, j) \end{cases} \tag{12}$$

UACI signifies the number of pixels that vary from the original image. For instance, if we assume that the total average number of pixels in the original image is 100. After encryption, if the total average number of pixels of the cipher image is 130, then the UACI test gives a value of 30. P in (13) denotes the value for the number of pixels. For example, there are 255 possible values in an 8-bit grayscale image. Therefore, P becomes 255 in 8-bit grayscale images.

$$UACI = \sum_{j=1}^{H} \sum_{i=1}^{W} \frac{|c_1(i, j) - c_2(i, j)|}{P \times WH} \times 100\ \% \tag{13}$$

Increasing the values of NPCR and UACI practically means that the algorithm has more secure features against differential attack. Tables 7 and 8 show the results of NPCR and UACI, respectively, for the intermediate image in each round from the first round to the seventh round. As Tables 7 and 8 show, our proposed scheme provides reasonable values for NPCR and UACI from the first round and the values remain unaffected as the number of rounds increases.

Table 8 UACI performance of the four images

Round	Lena	Mandril	Cameraman	Unit : percentage(%) Pirate
1	28.6225621840	27.5959912468	31.0880324419	28.9336933809
2	28.6237125770	27.5252368403	31.1555720310	29.0161835913
3	28.6558009128	27.6073156618	31.1159769694	29.0058869006
4	28.5922345928	27.5764869241	31.0359580844	28.9165018119
5	28.5897079168	27.4927251479	31.0862761852	28.9546801997
6	28.6173457725	27.5925864425	31.0990082983	29.0111541748
7	28.6412601844	27.5508731019	31.1118271771	29.0000825769

4.3.1 Key sensitivity

To evaluate the key sensitivity, we encrypted the same image with two pairs of keys. The key pair consisted of x, x', μ and μ'. We encrypted the grayscale Lena image with (3.923,3.955,4.94e–324,0.005) and (3.923,3.955,9.88e–324,0.005). The values 4.94e–324 and 9.88e–324 are different by only one bit in double precision; they are 0×1 and 0×2 in hexadecimal, respectively. Figure 8 shows the variance of the cipher image from a few different key pairs. In addition, it can be seen that the encrypted image also has a virtually uniform distribution. Therefore, our proposed encryption satisfies the confusion property.

4.3.2 Plain image sensitivity

To evaluate the influence of the plain image, we encrypted four images of size 512×512 pixels with an arbitrary pixel added. The modified images gave similar NPCR and UACI performances to those of original images, as shown in Tables 7 and 8, the NPCR and UACI performance of original images. The results indicate that our proposed encryption is sensitive to the plain image. This feature is important for defense against differential attacks.

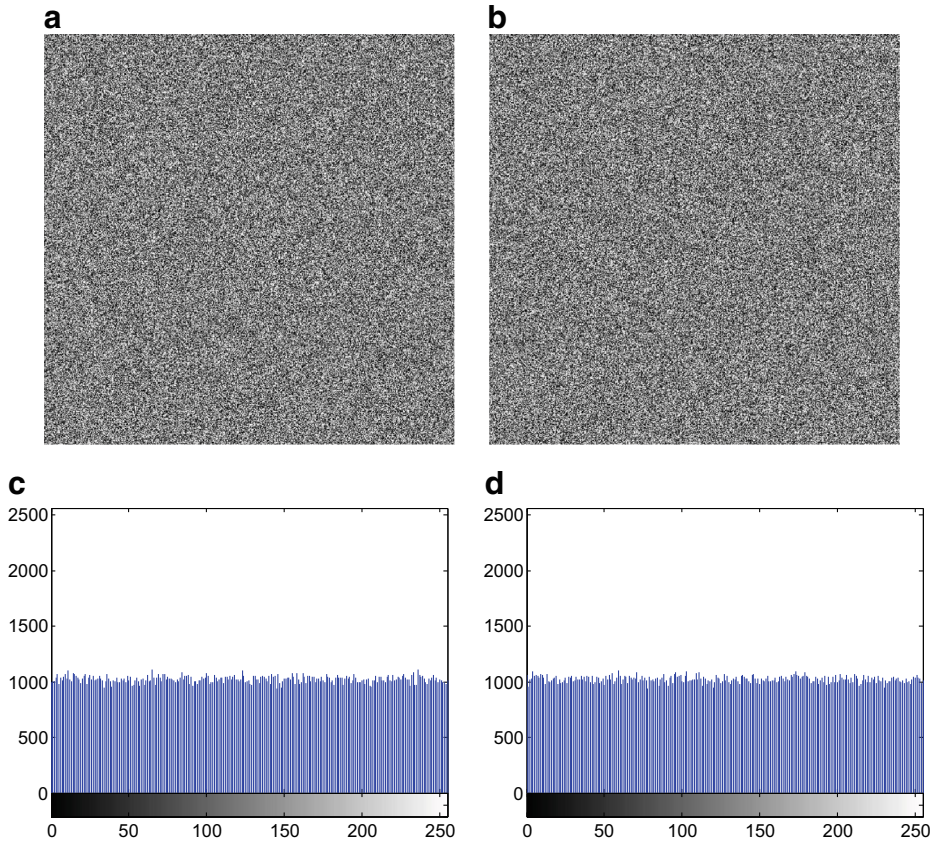

Fig. 8 Key sensitivity from 1-bit different key: **a** encrypted image with 4.94e–324, **b** encrypted image with 9.88e–324, **c** histogram of (a), **d** histogram of (b)

Table 9 Encryption and decryption speed with increased number of rounds

	1-Round	2-Round	3-Round	4-Round	5-Round	Unit : *ms* 6-Round
Encryption	3.407	4.062	4.776	5.392	6.059	6.739
Decryption	3.402	4.075	4.786	5.396	6.099	6.701

4.4 Time complexity

Encryption speed is a very important feature of image encryption. In particular, in order to satisfy the requirement of the real-time feature in the area of image encryption, lightweight computation is needed because image data requires more computation then text data. Therefore, we measured the encryption and decryption time complexity of the proposed encryption. Table 9 shows the time complexity from round one to round six. As the number of rounds increased, the time complexity linearly increased by approximately 0.7 ms.

4.5 Linear and differential cryptanalysis

Linear cryptanalysis is used to obtain information about a key by finding the XOR of two or more plaintexts and ciphertexts. In the proposed encryption, a pixel of a cipher image can be denoted as a chaining of a plain image because IV accumulates the pixels of the original image. Moreover, although two or more pairs of plain images and cipher images are used to acquire key information, it is very difficult because all pixels in a cipher image are calculated by addition of accumulative IV. Even if a combination of two or more keys could be revealed, it is close to impossible because a logistics map is a forward problem algorithm. In other words, calculating a key from a combination of keys is based on the difficulty of the inverse of the logistic map.

Differential cryptanalysis is an analysis scheme that uses the distance between two ciphertexts. For this cryptanalysis, let a white image and a black image be denoted as X and X', respectively, and let the differential value of the white image and black image be $\Delta X = X \oplus X'$, where ΔY is the differential value of these cipher images. Assume that in the first round of our proposed scheme, each pixel can be XORed twice, one addition and one rotation. The first XOR operation is performed with the original image and a chaos sequence from the logistics map, for the i-th pixel probability to obtain ΔY_i from ΔX_i is approximately $2^{8-Pr[CS_n|CS_n=CS_{n+1}]}$, where CS denotes the chaotic sequence. The second XOR operation updates IV with the current pixel, and then an addition operation is performed with the current pixel and the updated IV, where $Pr[\Delta X = \Delta Y]$ is 2^{-8} and the probability of obtaining ΔY_i from ΔX_i becomes $\frac{1}{8}$ because the final rotation also has an n-bit probability. Consequently, the proposed encryption has the same differential safety as the logistic map.

4.6 Comparison

In this section, we compare our proposed scheme with Wang's algorithm and Yang's algorithm in terms of correlation, NPCR, UACI, and time complexity. Table 10 shows the results obtained. As can be seen, the proposed algorithm has good values for correlation coefficients, Shannon's entropy, and time complexity. In general, correlation coefficients in the

Table 10 Comparisons in terms of entropy, NPCR, UACI and time complexity

	Correlation			NPCR	UACI	Speed
	Horizontal	Vertical	Diagonal	(%)	(%)	(ms)
Proposed algorithm	−0.003	−0.004	−0.009	99.636	31.155	3.407
Wang's algorithm	0.001	0.003	−0.001	99.737	37.572	174.99
Yang's algorithm	−0.002	−0.016	0.178	99.618	33.479	31.27

range above 0.1 and less than −0.1 signify that two images have a positive or negative relationship. A correlation coefficient in the range $-0.1 < r < 0.1$ signifies that no relationship exists between two images. As can be seen in Table 10, the proposed scheme has no relationship in three directions. In terms of time complexity, our scheme is ten times better than that of Yang's algorithm.

5 Conclusion and future works

In this paper, we proposed a new chaotic image encryption scheme based on the ARX model. The proposed scheme consists of the confusion and diffusion processes typical of chaotic image encryption schemes. In the proposed scheme, several sub-keys are derived from logistic maps with two given key pairs, which results in a bit of the given keys influencing all the bits of the cipher image. Consequently, our proposed scheme satisfies Shannon's confusion property. Further, to satisfy the diffusion property, all pixels in the cipher image are calculated by addition with accumulative IV. In the first round, a bit of the plain image influences the position of the next bits. After the second round, the bit is extended to the entire cipher image as a result of the accumulative feature of IV. The main feature of the proposed scheme is use of addition instead of cat map or S-Box for the diffusion process. Consequently, the encryption and decryption speeds are approximately ten times better than those of Yang's algorithms, the fastest known algorithm. In addition, the combination of XOR, addition, and rotation provides very good entropy values in the three directions. Even though the NPCR and UACI performances are poorer than those of Wang's algorithm, our proposed scheme is secure against linear and differential cryptanalysis. Through various analyses, we showed that the proposed ARX-based chaotic image encryption scheme has secure and good features via various analyses. In particular, our proposed scheme has a time complexity that is around ten times better than that of other schemes. Therefore, we expect that our scheme can be very useful for real-time applications.

Acknowledgments This work was supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIP) (No.10043907, Development of high performance IoT device and Open Platform with Intelligent Software)

References

1. Arroyo D, Li C, Li S, Alvarez G, Halang WA (2009) Cryptanalysis of an image encryption scheme based on a new total shuffling algorithm. Chaos, Solitons Fractals 41(5):2613–2616. doi:10.1016/j.chaos.2008.09.051. http://www.sciencedirect.com/science/article/pii/S0960077908004645
2. Ashtiyani M, Birgani PM, Hosseini HM (2008) Chaos-based medical image encryption using symmetric cryptography. In: 3rd international conference on Information and communication technologies: from theory to applications, 2008. ICTTA 2008. IEEE, pp 1–5
3. Bakhshandeh A, Eslami Z (2013) An authenticated image encryption scheme based on chaotic maps and memory cellular automata. Opt Lasers Eng 51(6):665–673. doi:10.1016/j.optlaseng.2013.01.001. http://www.sciencedirect.com/science/article/pii/S0143816613000079
4. Chen G, Mao Y, Chui CK (2004) A symmetric image encryption scheme based on 3d chaotic cat maps. Chaos, Solitons Fractals 21(3):749–761. doi:10.1016/j.chaos.2003.12.022. http://www.sciencedirect.com/science/article/pii/S0960077903006672
5. Enayatifar R, Abdullah AH, Lee M (2013) A weighted discrete imperialist competitive algorithm (wdica) combined with chaotic map for image encryption. Opt Lasers Eng 51(9):1066–1077. doi:10.1016/j.optlaseng.2013.03.010. http://www.sciencedirect.com/science/article/pii/S0143816613001048
6. Gao T, Chen Z (2008) Image encryption based on a new total shuffling algorithm. Chaos, Solitons Fractals 38(1):213–220. doi:10.1016/j.chaos.2006.11.009. http://www.sciencedirect.com/science/article/pii/S0960077906010447
7. Gao T, Chen Z (2008) A new image encryption algorithm based on hyper-chaos. Phys Lett A 372(4):394–400. doi:10.1016/j.physleta.2007.07.040. http://www.sciencedirect.com/science/article/pii/S0375960107010596
8. Guan ZH, Huang F, Guan W (2005) Chaos-based image encryption algorithm. Phys Lett A 346(1):153–157
9. Guan ZH, Huang F, Guan W (2005) Chaos-based image encryption algorithm. Phys Lett A 346(1–3):153–157. doi:10.1016/j.physleta.2005.08.006. http://www.sciencedirect.com/science/article/pii/S0375960105011904
10. Gupta K, Silakari S, Gupta R, Khan SA (2009) An ethical way of image encryption using ecc. In: 1st international conference on computational intelligence, communication systems and networks, 2009. CICSYN'09. IEEE, pp 342–345
11. Hong D, Lee JK, Kim DC, Kwon D, Ryu KH, Lee DG (2014) Lea: a 128-bit block cipher for fast encryption on common processors. In: Information security applications. Springer, Berlin Heidelberg New York, pp 3–27
12. Jin J (2012) An image encryption based on elementary cellular automata. Opt Lasers Eng 50(12):1836–1843. doi:10.1016/j.optlaseng.2012.06.002. http://www.sciencedirect.com/science/article/pii/S0143816612001674
13. Kwok H, Tang WK (2007) A fast image encryption system based on chaotic maps with finite precision representation. Chaos, Solitons Fractals 32(4):1518–1529. doi:10.1016/j.chaos.2005.11.090. http://www.sciencedirect.com/science/article/pii/S0960077905011999
14. Li H, Wang Y (2011) Double-image encryption based on discrete fractional random transform and chaotic maps. Opt Lasers Eng 49(7):753–757. doi:10.1016/j.optlaseng.2011.03.017. http://www.sciencedirect.com/science/article/pii/S0143816611000984
15. Li H, Wang Y, Yan H, Li L, Li Q, Zhao X (2013) Double-image encryption by using chaos-based local pixel scrambling technique and gyrator transform. Opt Lasers Eng 51(12):1327–1331. doi:10.1016/j.optlaseng.2013.05.011. http://www.sciencedirect.com/science/article/pii/S0143816613001620
16. Lian S, Sun J, Wang Z (2005) A block cipher based on a suitable use of the chaotic standard map. Chaos, Solitons Fractals 26(1):117–129. doi:10.1016/j.chaos.2004.11.096. http://www.sciencedirect.com/science/article/pii/S0960077905000378
17. Liu Z, Guo Q, Xu L, Ahmad MA, Liu S (2010) Double image encryption by using iterative random binary encoding in gyrator domains. Opt Express 18(11):12,033–12,043. doi:10.1364/OE.18.012033. http://www.opticsexpress.org/abstract.cfm?URI=oe-18-11-12033
18. Liu Z, Liu S (2007) Double image encryption based on iterative fractional fourier transform. Opt Commun 275(2):324–329. doi:10.1016/j.optcom.2007.03.039. http://www.sciencedirect.com/science/article/pii/S0030401807003240

19. Liu Z, Zhang Y, Li S, Liu W, Liu W, Wang Y, Liu S (2013) Double image encryption scheme by using random phase encoding and pixel exchanging in the gyrator transform domains. Opt Laser Technol 47:152–158
20. May RM et al. (1976) Simple mathematical models with very complicated dynamics. Nature 261(5560):459–467
21. Pareek N, Patidar V, Sud K (2006) Image encryption using chaotic logistic map. Image Vis Comput 24(9):926–934. doi:10.1016/j.imavis.2006.02.021. http://www.sciencedirect.com/science/article/pii/S026288560600103X
22. Pareek NK, Patidar V, Sud KK (2006) Image encryption using chaotic logistic map. Image Vis Comput 24(9):926–934
23. Rhouma R, Belghith S (2008) Cryptanalysis of a new image encryption algorithm based on hyper-chaos. Phys Lett A 372(38):5973–5978. doi:10.1016/j.physleta.2008.07.057. http://www.sciencedirect.com/science/article/pii/S0375960108011353
24. Srividya G, Nandakumar P (2011) A triple-key chaotic image encryption method. In: 2011 international conference on communications and signal processing (ICCSP), pp 266–270. doi:10.1109/ICCSP.2011.5739316
25. Stevenson D et al. (1987) An american national standard: Ieee standard for binary floating point arithmetic. ACM SIGPLAN Not 22(2):9–25
26. Wang X, Liu L, Zhang Y (2015) A novel chaotic block image encryption algorithm based on dynamic random growth technique. Opt Lasers Eng 66:10–18. doi:10.1016/j.optlaseng.2014.08.005. http://www.sciencedirect.com/science/article/pii/S0143816614001973
27. Wu Y, Noonan JP, Agaian S (2011) Npcr and uaci randomness tests for image encryption. Cyber journals: multidisciplinary journals in science and technology, Journal of Selected Areas in Telecommunications (JSAT), pp 31–38
28. Yen J-C, Guo JI (2000) A new chaotic key-based design for image encryption and decryption. In: Proceedings of the 2000 IEEE international symposium on circuits and systems, 2000. ISCAS 2000 Geneva, vol 4, pp 49–52. doi:10.1109/ISCAS.2000.858685
29. Zeghid M, Machhout M, Khriji L, Baganne A, Tourki R (2007) A modified aes based algorithm for image encryption. Int J Comput Sci Eng 1(1):70–75
30. Zhang Y, Xiao D (2013) Double optical image encryption using discrete chirikov standard map and chaos-based fractional random transform. Opt Lasers Eng 51(4):472–480. doi:10.1016/j.optlaseng.2012.11.001. http://www.sciencedirect.com/science/article/pii/S0143816612003077
31. Zhang YQ, Wang XY (2014) A symmetric image encryption algorithm based on mixed linear nonlinear coupled map lattice. Inf Sci 273:329–351. doi:10.1016/j.ins.2014.02.156. http://www.sciencedirect.com/science/article/pii/S0020025514002783
32. Zhao G, Yang X, Zhou B, Wei W (2010) Rsa-based digital image encryption algorithm in wireless sensor networks. In: 2010 2nd international conference on signal processing systems (ICSPS), vol 2. IEEE, pp v2–640
33. Zhu Z-L, Zhang W, Wong K-W, Yu H (2011) A chaos-based symmetric image encryption scheme using a bit-level permutation. Inf Sci 181(6):1171–1186. doi:10.1016/j.ins.2010.11.009. http://www.sciencedirect.com/science/article/pii/S0020025510005542

Jongseok Choi received the BSEE degree from TONGMYONG University, Pusan, and Republic of Korea in 2011, and he received the MS degree in Computer Engineering at Pusan National University. He is in PhD degree in computer engineering from Pusan National University. His research interests include IoT security, Elliptic Curve Cryptography, and cryptographic algorithms.

Seonhee Seok received the BSEE degree from Pusan National University, Pusan, and Republic of Korea in 2011. She is in MS degree in computer engineering from Pusan National University. Her research interests include IoT, Data mining and cryptographic algorithms.

Hwajeong Seo received the BSEE degree from Pusan National University, Pusan, and Republic of Korea in 2010, and he received the MS degree in Computer Engineering at Pusan National University. He is in PhD degree in computer engineering from Pusan National University. His research interests include sensor networks, information security, Elliptic Curve Cryptography, and RFID security.

Howon Kim received the B.Eng. degree from Kyungpook National University, Daegu, Republic of Korea, in 1993 and the MS and PhD degrees in electronic and electrical engineering from the Pohang University of Science and Technology (POSTECH), Pohang, Republic of Korea, in 1995 and 1999, respectively. From July 2003 to June 2004, he worked with the COSY group at the Ruhr-University of Bochum, Germany. He was a senior member of the technical staff at the Electronics and Telecommunications Research Institute (ETRI), Daejeon, Republic of Korea. He is currently working as an associate professor with the Department of Computer Engineering, School of Computer Science and engineering, Pusan National University, Pusan, Republic of Korea. His research interests include RFID technology, sensor networks, information security, and computer architecture. Currently, his main research focus is on mobile RFID technology and sensor networks, public key cryptosystems, and their security issues. Dr. Kim is a member of the IEEE, and the International Association for Cryptologic Research (IACR).