IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received April 24, 2019, accepted May 8, 2019, date of publication May 14, 2019, date of current version June 19, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2916600

Cryptanalysis and Enhancement of an Image Encryption Scheme Based on Bit-Plane Extraction and Multiple Chaotic Maps

YU LIU[1], ZHENG QIN[1], AND JIAHUI WU[2]
[1]College of Computer Science and Electronic Engineering, Hunan University, Changsha 410082, China
[2]College of Electronic and Information Engineering, Southwest University, Chongqing 400715, China

Corresponding author: Zheng Qin (zqin@hnu.edu.cn)

This work was supported in part by the National Science Foundation of China under Grant 61472131, Grant 61772191, Grant 61872130, and in part by the Science and Technology Key Projects of Hunan Province under Grant 2015TP1004 and Grant 2016JC2012, and in part by the Science and Technology Key Projects of Changsha under Grant kq1801008.

ABSTRACT Recently, an image encryption scheme combining bit-plane extraction with multiple chaotic maps (IESBC) was proposed. The scheme extracts binary bit planes from the plain-image and performs bit-level permutation and confusion, which are controlled by a pseudo-random sequence and a random image generated by the Logistic map, respectively. As the rows and columns of the four MSBPs are permuted with the same pseudo-random sequence and the encryption process does not involve the statistical characteristics of the plain-image, the equivalent secret key of IESBC can be disclosed in the scenario of known/chosen-plaintext attacks. This paper analyzes the weak points of IESBC and proposes a known-plaintext attack and a chosen-plaintext attack on it. Furthermore, we proposed an enhanced scheme to fix the shortcomings and resist the proposed plaintext attacks. The experimental simulation results demonstrated that the enhanced scheme is excellent in terms of various cryptographic metrics.

INDEX TERMS Image encryption, plaintext attacks, security analysis, dynamic diffusion, chaotic cryptanalysis.

I. INTRODUCTION

With the rapid development of network communication, information security technology has entered a comprehensive digital era [1]. Digital image is one of the most popular information media among multiple multimedia representation forms because of its impressive advantages, such as intuitive nature, rich content, low cost and rapid transmission. However, there exist many security risks in the network transmission of digital images, such as interception, malicious falsification and illegal destruction by adversaries, which may lead to immeasurable but unperceived losses [2]–[4]. Therefore, ensuring secure transmission of digital images has become an important concern of every person living with cyberspace.

Compared with textual data, image data has special characteristics, such as strong redundancy and high correlation among adjacent pixels. Therefore, the traditional encryption algorithms, such as DES, AES, and IDEA, cannot protect image data efficiently [5], [6]. To address this challenge, many digital image encryption algorithms were proposed in the past three decades [7]. According to their essential structures, the algorithms can be classified into three classes:

- Image encryption algorithms based on pixel permutation [8]–[10], which changes the pixel positions via a finite replacement operation on the pixel matrix, destroys the correlation of adjacent pixels, and generates unrecognized cipher images [11]. Typical permutation encryption methods include image encryption algorithms based on various affine transformations: Arnold transform, and cube transformation [7]. The pixel replacement method is simple and easy to calculate. Since the statistical characteristics of the original image are not destroyed, the resistance against statistical attacks is not effective enough.
- Image encryption algorithms based on chaotic systems [12]–[15]. Such kind of schemes utilize the ergodicity, pseudo-randomness and initial condition sensitivity of chaotic sequences to realize the diffusion and

The associate editor coordinating the review of this manuscript and approving it for publication was Resul Das.

confusion requirements of a secure image encryption scheme [16]. Typical chaotic image encryption algorithms are built on all kinds of chaotic systems: Logistic map [17]–[19], PWLCM [5], Tent map [20], and Henon map [21]. These methods have a large secret-key space and high security. Therefore, image encryption methods based on chaos theory received attentions of many researchers [22], [23].

- Bit-plane-based image encryption algorithms [24], which decompose the pixels to the bit level and perform scrambling and spreading operations on the bit plane [25]. Representative methods include a bit-plane encryption method based on a matrix transformation [26], a selective encryption method based on the bit-plane information distribution, and a combination of bit-level and pixel-level encryption approaches [27]. The bit-plane cryptographical operation enhances the sensitivity of the algorithm to tiny changes in the plaintext, and increases its resistance to differential attacks [28], [29]. In addition, the permutation operation on the bit plane can change the pixel position and value at the same time, reducing the number of calculations performed by the encryption algorithm. Some bit-plane encryption algorithms can realize the ideal encryption effect via only one round of encryption.

This paper focuses on security analysis and improvement of an image encryption scheme based on the extraction of binary bit planes and multiple chaotic maps (IESBC) proposed in [30]. In IESBC, the authors used multiple chaotic maps to design a chaos-based image encryption scheme at the bit level. In the permutation phase, the algorithm selected the four most significant bit-planes (MSBPs), which contains the critical information of an natural image, to perform row-column permutations. Instead, the four LSBPs containing some edge information of the image are kept unchanged. In the confusion phase, the scheme used chaotic maps to generate a pseudo-random image, extracted the bit planes of the random image, and performed the XOR operation with the corresponding bit planes of the plain-image. The most important advantage of this encryption process is its high computational efficiency. However, there are security issues with IESBC: the encryption algorithm is fixed, namely the ciphertexts of the same plaintexts that are encrypted by the algorithm with the identical secret keys are exactly the same. An attacker can derive an equivalent secret key using multiple pairs of the plaintext and the corresponding ciphertext. Moreover, the rows and columns of the four MSBPs of the plain-image are permuted with the same random sequence. Therefore, an adversary can identify the mathematical law of the permutation mechanism and use it to obtain the secret key. From this point of view, IESBC is not secure. We performed a known-plaintext attack and a chosen-plaintext attack on it and recover the plain-image successfully. In addition, we enhance the scheme by using a statistical value of the plain-image and a random number K, generated by a pseudo-random number generator in the diffusion phase. The number K is not transmitted as a secret key as it is not needed in the decryption process. The same plaintext is encrypted twice with the same secret key and the resulting ciphertexts are different because K is random. Therefore, our enhanced scheme is a probabilistic algorithm and can resist the proposed plaintext attacks efficiently. Experimental results and cryptographic analysis both demonstrated that the enhanced encryption scheme shows excellent performance in terms of the key space, key sensitivity, correlation analysis, information entropy, local Shannon entropy, cropping attack analysis, noise attack analysis, differential attack analysis, computational complexity analysis and speed analysis.

The rest of this paper is organized as follows. Section II concisely reviews the encryption process of IESBC. Section III presents the cryptanalysis of IESBC against the known-plaintext and chosen-plaintext attacks. Section IV proposes an enhanced version of IESBC. Section V presents a simulation experiment and a performance analysis of the enhanced scheme. Some conclusions are presented in the last section.

II. OVERVIEW OF THE IESBC ENCRYPTION ALGORITHM

A. PRELIMINARY WORK

1) BINARY BIT-PLANE EXTRACTION

An 8-bit plain-image, which is denoted by I, can be extracted as eight binary bit-planes $I_1, I_2, \cdots, I_8$, and each binary bit plane contains one bit information of the image. Binary matrixes I_1, I_2, I_3, and I_4 represent the least significant bits planes (LSBPs). In contrast, matrixes I_5, I_6, I_7, and I_8 represent the MSBPs [30]. Every bit plane is different in terms of the amount of contained visual information. The quantity of visual information contained in the binary bit planes is incremented from I_1 to I_8 in order. The smallest amount of information of the plain-image is presented in the bit plane LSB_1, while the largest amount of visual information is contained in MSB_8. The bit planes are extracted by using

$$I_n = I \bmod 2^n, \tag{1}$$

where $n = 0, 1, 2, \cdots, 7$. The pixel values of the plain-image I range in [0, 255], and those of the bit planes range in [0, 1].

2) LOGISTIC MAP

The Logistic map is simple dynamical equation exhibiting complex nonlinear behavior. Its output is highly sensitive to the initial condition and system parameters. Thus, it is often used to generate pseudo-random number sequences, especially in the field of image security. The Logistic map is expressed as

$$\Omega(n+1) = \mu * \Omega(n)(1 - \Omega(n)), \tag{2}$$

where the initial parameters are Ω_0 and μ_0. When μ falls in the range of (3.56995, 4), a chaotic sequence, denoted by Ω, is obtained in the range (0, 1). Detailed dynamical analysis of the Logistic map implemented in a digital computer can be found in [20].

The Logistic map is highly sensitive to the value of control parameter μ. The behavior of the map changes with μ. When the value of μ is small, the mapping gradually stabilizes after a certain number of iterations. With increase of μ, the stable state is decomposed into bifurcations and becomes a two-state periodic form. Then, these states are further divided into four-state periodic forms and subsequently into eight-states [30]. Eventually, the system enters a chaotic state. The chaotic cubic-logistic map, a combination of the Logistic map and the cubic map, is defined as

$$\Omega_{(n+1)} = \mu * \Omega_n(1 - \Omega_n)(2 + \Omega_n), \tag{3}$$

where $\mu \in (1.41, 1.59)$ and $\Omega \in (0, 1)$.

3) THE RANDOM IMAGE GENERATION

The specific steps for generating a random image in IESBC are described as follows:

- Set the initial condition and the control parameter of the chaotic Logistic map;
- Generate a random sequence R' of length mn by iterating the Logistic map, where mn is the size of the plain-image I_{mn}, and every element of R' belongs to the range [0, 1];
- Transform the decimals in R' into positive integers by multiplying them with a large integer θ and remove their decimal parts;
- Adopt a modulo operation $P(i) = R'(i) \bmod 256$ to ensure that each value of R' belongs to [0, 255], where $i \in [1, mn]$;
- Reshape the sequence R' into a two-dimensional matrix, denoted as $R_{m\times n}$, as the required random image.

B. ENCRYPTION ALGORITHM

This section describes the encryption algorithm in detail. The key steps of IESBC are described as follows:

- Extract the eight binary bit planes of the plain-image I: $I_1, I_2, \cdots, I_8$;
- Generate a random sequence K' using the chaotic cubic-Logistic map, where $K' = [K_1, K_2, K_3, \cdots, K_{256}]$ and no value of K' is repeated. If there is a duplicate value in the generated sequence, only the first instance is selected for K'. Perform the modulo operation to scale the values of K' to the range [0, 255], namely, $K = K' \bmod 256$;
- Perform the row-column permutation operation on the four MSBPs (I_5, I_6, I_7, I_8) according to the random sequence K. The permuted MSBPs ($I_{5p}, I_{6p}, I_{7p}, I_{8p}$) can be obtained. Here, only four MSBPs are permuted since they contain the main information of the original plain-image, whereas the four LSBPs contain only a small amount of information;
- Generate a random image R using the chaotic Logistic map. Then extract eight binary bit planes of the random image R: $R \rightarrow R_1, R_2, \cdots, R_8$. So far, there are a total of sixteen different binary bit planes: $I_1, I_2, I_3, I_4, I_{5p}, I_{6p}, I_{7p}, I_{8p}$ and $R_1, R_2, \cdots, R_8$;
- Perform the XOR operation on the corresponding binary bit planes between the plaintext and the random image: $C_i = I_i \oplus R_i$ for $i = 1, \cdots, 4$; $C_i = I_{ip} \oplus R_i$ for $i = 5, \cdots, 8$;
- Compose the eight manipulated planes to obtain the cipher-image $C = \sum_{i=1}^{7} C_i \cdot 2^{8-i}$.

III. CRYPTANALYSIS

A. PRELIMINARIES

A secure encryption scheme should be able to withstand all types of attacks; otherwise the encryption scheme cannot be used in secure communications [31]. Cryptographic analysis focuses on recovering the plaintext without knowing the secret key. Four common types of attacks for analyzing cryptosystem security are described as follows:

- Ciphertext-only attack (COA): The attackers do not know the cryptographic algorithm and can only analyze it based on a series of intercepted ciphertext to obtain some information of the plaintext or the secret keys.
- Known-plaintext attack (KPA): In addition to the intercepted ciphertext, the attackers have some known plaintext-ciphertext pairs for deciphering the cryptosystem.
- Chosen-plaintext attack (CPA): The attackers not only have some plaintext-ciphertext pairs but can also choose or create some specific plaintexts and obtain their corresponding ciphertexts to enhance the attacking performance.
- Chosen-ciphertext attack (CCA): The attackers can choose or create some specific ciphertexts and obtain their corresponding plaintexts because they can access the decryption mechanism.

These attacks demonstrate that the goal of the attackers is to recover information regarding the plaintext and secret keys. In IESBC, the four MSBPs (I_5, I_6, I_7 and I_8) are permuted using the random sequence K successively. In contrast to another approach, in which the four planes are composed together and subsequently permuted. Those two approaches have the same effect; however, the former takes longer. IESBC can be analogized to the following process, with the key difference being that each step operates at the bit plane level rather than the pixel level: plain-image I is permuted by the random sequence K and the XOR operation is performed on the permuted image and the random image R to obtain cipher image C. Based on this analogous algorithm, the security of the encryption algorithm only depends on the random sequence K and the random image R. This encryption algorithm can be deciphered by an adversary owning K and R. In IESBC, K and R are only generated by the Logistic map and are not related to the plain-image. In addition, the LSBPs of the plain-image are not related to the MSBPs in the encryption process. Moreover, the rows and columns of the plain-image's four MSBPs (I_5, I_6, I_7 and I_8) are permuted with the same random sequence K; therefore, the adversary can identify the mathematical law of the permutation mechanism. Besides, compared with point permutation, the row-column permutation is more likely to contain relevant information

of the plain-image's neighboring pixels. These shortcomings undermine the security of IESBC and render it vulnerable to plaintext attacks.

B. KNOWN-PLAINTEXT ATTACK

In this section, we use some known plain-images $I_{n\times n}$ and the corresponding cipher-images $C_{n\times n}$ to break IESBC, where the random image is denoted as $R_{n\times n}$. As for the four LSBPs, there is only XOR operation, so $R(i) = I(i) \oplus C(i)$ can be obtained for each $i \in \{1, 2, 3, 4\}$. When the attacker get another cipher image C', he can obtain the first four binary bit planes of plain-image I' using $I'(i) = R(i) \oplus C'(i)$ for each $i \in \{1, 2, 3, 4\}$.

C. SIMULATION RESULTS OF KPA

In this section, we use a pair of plain-image and its corresponding cipher-image to obtain the four LSBPs of the random image in the simulation experiment. Then, we decipher two encrypted images shown in Fig. 1(b) and Fig. 1(e). The results of the cracking images are shown in Fig. 1(c) and Fig. 1(f). Compared with the original plain-images, one can see that although only the four LSBPs are recovered, they still contain a lot of visual information and the attacker can see the skeleton of plain-image clearly. More importantly, the computational load on the proposed attack is very low.

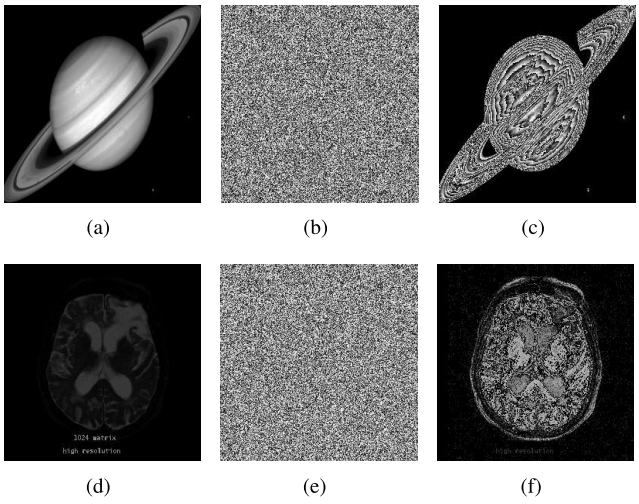


FIGURE 1. Simulation results of the known-plaintext attack. (a) Plain-image A. (b) Encrypted image of (a). (c) Recovered the four LSBPs image of (b). (d) Plain-image B. (e) Encrypted image of (d). (f) Recovered the four LSBPs image of (e).

D. CHOSEN-PLAINTEXT ATTACK

In this section, we use some chosen plain-images and the corresponding cipher-plaintexts to decipher IESBC. First, the attacker chooses the plain-image $I_0 = 0$ and obtains its cipher image C since he can access the encryption mechanism. Because I_0 is a matrix of all zeros, it is unchanged by permutations. It follows that $I_0 \oplus R = C$. Thus, $R = C$ can be obtained. Hence, the binary bit planes of the random image can be extracted: $R \to R_1, R_2, \cdots, R_8$. Next, the attacker chooses another plain-image $I_{n\times n}$ and its cipher image $C_{n\times n}$ whose binary bit planes are $\{I_i = 0\}_{i=1}^{7}$ and

$$I_8 = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \end{bmatrix}_{n\times n},$$

namely $I_8(1, 1) = 1, I_8(1, 2) = 1, I_8(3, 2) = 1, I_8(3, 4) = 1, I_8(5, 4) = 1, \ldots, I_8(n-3, n-2) = 1, I_8(n-1, n-2) = 1$, and $I_8(n-1, n) = 1$. The plain-image is encrypted into the cipher image $C_{n\times n}$ whose eight binary bit planes are denoted as $C_1, C_2, \cdots$, and C_8. The XOR operations are performed on the corresponding binary bit planes between the cipher image C and the random image R: $D_i = C_i \oplus R_i, i = 1, \cdots, 8$. D_5, D_6, D_7, and D_8 are permuted from the four MSBPs of the plain-image (I_5, I_6, I_7, I_8, respectively) using the random sequence K generated by the chaotic cubic-Logistic map.

The values of the random sequence K is calculated by Algorithm 1. Since the rows and columns are permuted with the same random sequence K, only one position's value equal to 1 on the diagonal line of matrix D_8, which is permuted by I_8 (1, 1). Suppose that the location of the diagonal entry of D_8 that equals 1 is $D_8(x, x) = 1$. Then, the first row of I_8 is permuted to the x-th row of D_8 and the first column of I_8 is permuted to the x-th column of D_8. Thus, the x-th value of the random sequence K is 1, namely $K(x) = 1$. Next, it is easy to show that $I_8(1, 2)$ is permuted to another position whose value equals 1 in the x-th row of D_8. Suppose another location in the xth row of I_{8p} that equals 1 is $D_8(x, y) = 1$. The yth element of the random sequence K can be obtained, namely, $K(y) = 2$. Similarly, all the values of the random sequence K can be obtained.

With the random sequence K and the random image R, the original plain-image can be recovered from the corresponding cipher-image. Extract the binary bit planes of the cipher image. Then, the XOR operation is performed on the corresponding binary bit planes between cipher image C and random image R: $C_i \oplus R_i = D_i, i = 1, 2, \cdots, 8$. Inversely permute the four MSBPs of D (D_5, D_6, D_7, and D_8) according to the random sequence K and save the resulting planes in D_{5p}, D_{6p}, D_{7p}, and D_{8p}. Last, combine D_1, D_2, D_3, D_4, D_{5p}, D_{6p}, D_{7p}, and D_{8p} to obtain the plain-image I. Therefore, IESBC is not secure and can be deciphered with two chosen plain-images.

E. SIMULATION RESULTS OF CPA

In this section, a simulation experiment is conducted to demonstrate performance of the proposed attacks. The sim-

ulation is run on a desktop PC with Intel(R) Core(TM) i7-6500 CPU 2.50GHz, 8GB RAM and a 500GB hard drive. The computational platform is MATLAB 8.3.0.532 (R2017a) and the operating system is Microsoft Windows 10. The concrete steps are presented as follows:

- Choose the plain-image $I_{0(256\times256)} = 0$ and obtain its cipher image $C_{0(256\times256)}$ from the encryption mechanism. Thus, the random image R can be obtained: $R_{256\times256} = C_{0(256\times256)}$;
- Extract the binary bit planes of the random image $R \rightarrow R_1, R_2, R_3, R_4, R_5, R_6, R_7$, and R_8;
- Choose another plain-image $I_{256\times256}$ and obtain its cipher image $C_{256\times256}$. Eight binary bit planes of the chosen plain-image satisfy: $I_1 = 0, I_2 = 0, I_3 = 0, I_4 = 0, I_5 = 0, I_6 = 0, I_7 = 0$ and

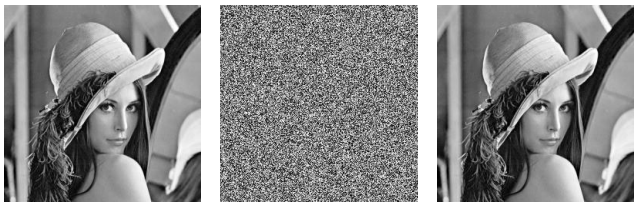
$$I_8 = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \end{bmatrix}_{256\times256}.$$

  Precisely, $I_8(1,1) = 1, I_8(1,2) = 1, I_8(3,2) = 1, I_8(3,4) = 1, I_8(5,4) = 1, \ldots, I_8(253,254) = 1, I_8(255,254) = 1, I_8(255,256) = 1$;
- Extract the binary bit planes of the cipher image $C \rightarrow C_1, C_2, \cdots, C_8$;

Algorithm 1 Obtain the Random Sequence K

Input: Two bit planes $I_{8n\times n}$ and $D_{8n\times n}$.
Output: The random sequence K_n.

```
Find the position (x, x) in D_8 where the diagonal value equals 1;
K(x) = 1 ;
for i = 1 : n − 1, i = i + 2 do
    Find another position (x, y) of the xth row in D_8 which equals to 1 ;
    K(y) = i + 1;
    Find another position (p, y) of the yth column in D_8 which equals to 1 ;
    K(p) = i + 2;
    x = p ;
end
```



(a) (b) (c)

FIGURE 2. Simulation results of the chosen-plaintext attack: (a) Plain-image; (b) Encrypted image; (c) Recovered image.

- Performe the XOR operation on the corresponding binary bit planes between the cipher image C and the random image R: $D_i = C_i \oplus R_i$ for any $i \in \{1, 2, \cdots, 8\}$;
- Extract the binary bit planes of the plain-image $I \rightarrow I_1, \cdots, I_8$;
- Input I_8 and D_8, and the random sequence K can be calculated via Algorithm 1.

Figure 2 depicts the results of the simulation experiment. (a) is the chosen plain-image of size 256×256, (b) is the cipher-image of (a), and (c) is the image that is recovered using the chosen-plaintext attack.

Table 1 presents the performance analysis results for extracting two equivalent secret keys: R and K. We only use two special plain-images and a little time to obtain the equivalent secret keys. The computational complexity of the proposed chosen-plaintext attack is $O(n)$ where n is the larger of the height and width of the known plain-image.

IV. ENHANCED ENCRYPTION ALGORITHM

A. SHORTCOMINGS OF IESBC

The key shortcomings of IESBC are summarized as follows:

- Sequences K and R are only generated by the Logistic map and the encryption process does not involve the statistical characteristics of the plain-image.
- The rows and columns of the plain-image's four MSBPs (I_5, I_6, I_7 and I_8) are permuted with the same random sequence K; therefore, the adversaries can identify the mathematical law of the permutation mechanism.
- This algorithm can only be applied to square plain-images due to the use of the same random sequence K, which is not applicable to non-square images.
- Compared with a point permutation, a row-column permutation is more likely to preserve relevant information of the plain-image's neighboring pixels.
- When performing the XOR operation, this algorithm simply operates on the corresponding bit plane. The

TABLE 1. Performance analysis of equivalent secret key extraction.

Operation	Extract random image R	Extract random sequence K	Total
Time (seconds)	0.1070	0.0435	0.1505
Number of the chosen plain-image	1	1	2

LSBPs of the plain-image are not related to the MSBPs in the encryption process.

Our enhanced encryption scheme is able to encrypt rectangular plain-images. It performs point permutations instead of row-column permutations in the permutation phase to reduce the correlation of the plain-image's neighboring pixels. In the confusion phase, each position of the LSBs plane is closely related to the same position on the MSBs plane and possesses the statistical characteristics of the plain-image. Moreover, we introduce a random number K, which is generated by a pseudo-random number generator. The same plaintext is encrypted twice with the same secret key and the resulting cipher-texts are different because K is random. Thus, it is sufficiently secure against all types of attacks. Our enhanced encryption and decryption algorithms are described as follows.

B. ENCRYPTION

Assume that the original plain-image is $I_{m\times n}$. The encryption steps are presented as follows:

- Extract the low four bits of the plain-image as LSBs plane I_L and the high four bits of the plain image as MSBs plane I_M: $I \rightarrow I_L, I_M$. The pixel values of the two image planes fall in interval $[0, 2^4 - 1]$;
- Choose the control parameters μ, initial values x_0, and lengths of the sequences l: $\{\mu_i;\ x_{i_0};\ l_i \mid i = 1,\ 2,\ 3\}$ and generate three random sequences S', R'_L, R'_M by using the chaotic cubic-Logistic map. Sequences R'_L, R'_M are both of length mn. No value of S is repeated. If there is a duplicate value in the generated sequence S', only the first is retained. Therefore, we should generate more than mn values in S' to ensure that the length of S is mn after the removal of the duplicate values. Then, we apply the modulo operation to scale the values of the random sequence as follows: $S = S' \bmod mn$, $R_L = R'_L \bmod 2^4$, and $R_M = R'_M \bmod 2^4$;
- Transform $I_{L_{m\times n}}$ and $I_{M_{m\times n}}$ into one-dimensional vectors $V_{L_{1\times mn}}$ and $V_{M_{1\times mn}}$, respectively, and permute $V_{M_{1\times mn}}$ to $V_{MP_{1\times mn}}$ with the random sequences S;
- Encrypt $V_{L_{1\times mn}}$ and $V_{MP_{1\times mn}}$ to make them interrelated and obtain corresponding cipher-texts $V'_{L_{1\times mn}}$ and $V'_{MP_{1\times mn}}$ by setting

$$\begin{cases} V'_L(i+1) = ((V_L(i+1) + V'_L(i) + V'_{MP}(i)) \\ \qquad\qquad \bmod 2^4) \oplus R_L(i+1), \\ V'_{MP}(i+1) = ((V_{MP}(i+1) + V'_{MP}(i) + V'_L(i+1)) \\ \qquad\qquad \bmod 2^4) \oplus R_M(i+1), \end{cases} \tag{4}$$

for any $i \in \{0, 1, 2, \cdots, mn-1\}$, where

$$V'_L(0) = (\bar{I_L} + \theta) \bmod 2^4, \tag{5}$$

$$V'_{MP}(0) = (\bar{I_M} + \theta) \bmod 2^4, \tag{6}$$

$$\bar{I_L} = \frac{\sum_{i=1}^{i=m}\sum_{j=1}^{j=n} I_L(i,j)}{m \times n}, \tag{7}$$

$$\bar{I_M} = \frac{\sum_{i=1}^{i=m}\sum_{j=1}^{j=n} I_M(i,j)}{m \times n}, \tag{8}$$

$$\bar{I} = \frac{\sum_{i=1}^{i=m}\sum_{j=1}^{j=n} I(i,j)}{m \times n}, \tag{9}$$

and

$$\theta = \left\lfloor \frac{\sum (I - \bar{I} + K)^2}{m \times n} \times 10^{10} \right\rfloor \bmod 2^4, \tag{10}$$

K is a random number generated by a pseudo-random number generator such that $K \in (-\infty, +\infty)$;

- Reshape 1D vectors $V'_{L_{1\times mn}}$, $V'_{MP_{1\times mn}}$ into matrices $C_{L_{m\times n}}$, $C_{MP_{m\times n}}$, respectively;
- Use the two corresponding planes $C_{L_{m\times n}}$ and $C_{MP_{m\times n}}$ to compose the final cipher image $C_{m\times n}$.

This encryption scheme is mainly used for gray-scale images, and it can also encrypt color images. For an RGB color image, the scheme decomposes the plain-image into R, G and B components, encrypts each channel separately and then combine them together to form the final cipher image.

C. DECRYPTION

Assume that the ciphertext image is $C_{m\times n}$. The decryption steps are described as follows:

- Extract the low four bits of the ciphertext image as LSBs plane C_L and the high four bits of the ciphertext image as MSBs plane C_M: $C \rightarrow C_L, C_M$. The pixel values of the two image planes fall in interval $[0, 2^4 - 1]$;
- Receive the parameter set $\{\mu_i;\ x_{i_0};\ l_i \mid i = 1,\ 2,\ 3\}$ and obtain three random sequences S', R'_L, R'_M via the chaotic cubic-Logistic map. Then, we apply the modulo operation to scale the values of the random sequences as follows: $S = S' \bmod mn$, $R_L = R'_L \bmod 2^4$, and $R_M = R'_M \bmod 2^4$. Sequences S, R_L, R_M are all of length mn. No value of S is repeated;
- Transform $C_{L_{m\times n}}$ and $C_{M_{m\times n}}$ into one-dimensional vectors $V_{L_{1\times mn}}$ and $V_{M_{1\times mn}}$, respectively;
- Inversely diffuse $V'_{L_{1\times mn}}$ and $V'_{MP_{1\times mn}}$ to obtain corresponding vectors $V_{L_{1\times mn}}$ and $V_{MP_{1\times mn}}$ via

$$\begin{cases} V_L(i+1) = (V'_L(i+1) \oplus R_L(i+1) - V'_L(i) \\ \qquad\qquad -V'_{MP}(i)) \bmod 2^4, \\ V_{MP}(i+1) = (V'_{MP}(i+1) \oplus R_M(i+1) - V'_{MP}(i) \\ \qquad\qquad -V'_L(i+1)) \bmod 2^4, \end{cases} \tag{11}$$

where $i = 1, 2, \cdots, mn-1$;

- Permute $V'_{MP_{1\times mn}}$ to $V'_{M_{1\times mn}}$ with the random sequences S;
- Reshape 1D vectors $V'_{L_{1\times mn}}$ and $V'_{M_{1\times mn}}$ to matrices $I_{L_{m\times n}}$ and $I_{M_{m\times n}}$, which have m rows and n columns;
- Use the two corresponding binary bit planes $I_{L_{m\times n}}$ and $I_{M_{m\times n}}$ to compose the final plain-image $I_{m\times n}$.

V. PERFORMANCE ANALYSIS AND COMPARISON

A. RESISTANCE TO THE KNOWN-PLAINTEXT AND CHOSEN-PLAINTEXT ATTACKS

In IESBC, ciphertexts of the same plaintexts encrypted via the algorithm with identical secret keys are exactly the same; hence, it is a fixed algorithm. An attacker can derive an equivalent secret key from multiple plaintext-ciphertext pairs. In our enhanced scheme, a random number K generated by a pseudo-random number generator in the encryption process. Since K is not needed in decryption, K is not transmitted as a secret key. Therefore, our enhanced scheme is a probabilistic algorithm. If the same plaintext is encrypted twice with the same secret keys, the resulting ciphertexts will be different because K is random. Besides, the enhanced scheme extract an intrinsic feature of the plain-image in diffusion process. Due to the relation mechanism between the LSBs plane and the MSBs plane, each position of those planes can be affected by the statistical characteristics of the plain-image. Therefore, our enhanced scheme can resist the known-plaintext attack and the chosen-plaintext attack effectively.

B. EXPERIMENTS OF ENCRYPTION AND DECRYPTION

This section presents experimental results for the enhanced algorithm. We set the original parameters and initial conditions as $\mu_1 = 1.45, x_{1_0} = 0.3$, and $l_1 = 10mn + 1000$ to obtain S; set $\mu_2 = 1.49, x_{2_0} = 0.1$, and $l_2 = mn + 1000$ to obtain R_L; and set $\mu_3 = 1.52, x_{3_0} = 0.2$, and $l_3 = mn + 1000$ to obtain R_M. Fig. 3(a) depicts the plain-image "Lena" of size 256×256, and its histogram is shown in (d). Fig. 3(b) shows the encrypted image obtained via the enhanced algorithm and its histogram is demonstrated in Fig. 3(e). According to the histogram of the cipher image, the pixel distribution is highly uniform and has satisfactory statistical properties. Fig. 3(c) and (f) presents the decrypted image and its histogram, respectively.

C. KEY SPACE

A secure encryption algorithm should have a sufficiently large key space to resist the brute-force attack. When we evaluate a secure encryption algorithm, the key space is typically larger than 2^{100} [32], [33]. In the enhanced scheme, the key space can be calculated from the parameters μ_1, μ_2, and μ_3 and the initial values x_{1_0}, x_{2_0}, and x_{3_0} of the chaotic cubic-Logistic map. Assuming the accuracy range of a floating-point number is 10^{-14}, the key space is of size $(10^{14})^6 = 10^{84}$, which well exceeds 2^{100}. Therefore, the enhanced encryption algorithm is sufficiently secure for resisting the brute-force attack. Table 2 shows the key space comparisons of different encryption schemes. As can be seen, the key space of our enhanced scheme is larger than that of some existing scheme, such as Ref. [30], [34]–[37].

D. KEY SENSITIVITY ANALYSIS

A good encryption scheme should be highly sensitive to the secret keys in both the encryption process and decryption process. To test the key sensitivity in encryption part, the original keys $K_0 = (\mu_1 = 1.45, x_{1_0} = 0.3, \mu_2 = 1.49, x_{2_0} = 0.1, \mu_3 = 1.52, x_{3_0} = 0.2)$ are added a slight disturbance 10^{-14} to construct two slightly different keys $K_1 = (\mu_1 = 1.45 + 10^{-14}, x_{1_0} = 0.3, \mu_2 = 1.49, x_{2_0} = 0.1, \mu_3 = 1.52, x_{3_0} = 0.2)$ and $K_2 = (\mu_1 = 1.45, x_{1_0} = 0.3 + 10^{-14}, \mu_2 = 1.49, x_{2_0} = 0.1, \mu_3 = 1.52, x_{3_0} = 0.2)$. Then the same plain-image "Lena" is encrypted independently using K_0, K_1, K_2. The encrypted images are shown in Fig. 4(a)-(c), respectively. The different pixels between the original cipher image encrypted by K_0 and two changed cipher images encrypted independently by K_1, K_2 account for 99.5758% and 99.5941% of the total ones, respectively. This means that the same plain-image can generate totally different cipher images using slight different keys. In decryption part,

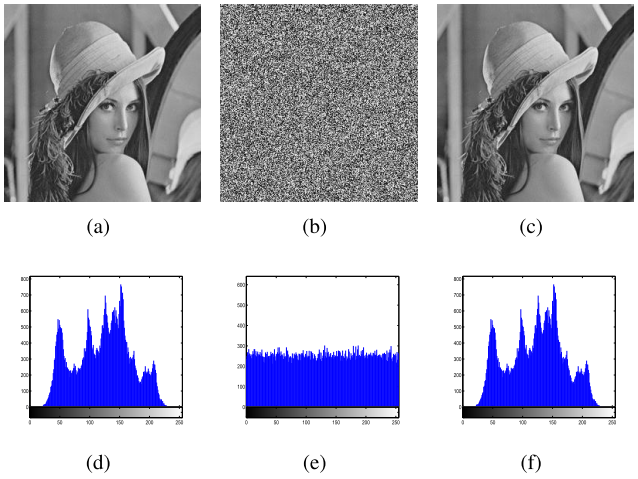


FIGURE 3. The plain, encrypted and decrypted images and their histograms. (a) Plain-image. (b) Encrypted image. (c) Decrypted image. (d) Histogram of plain-image. (e) Histogram of encrypted image. (f) Histogram of decrypted image.

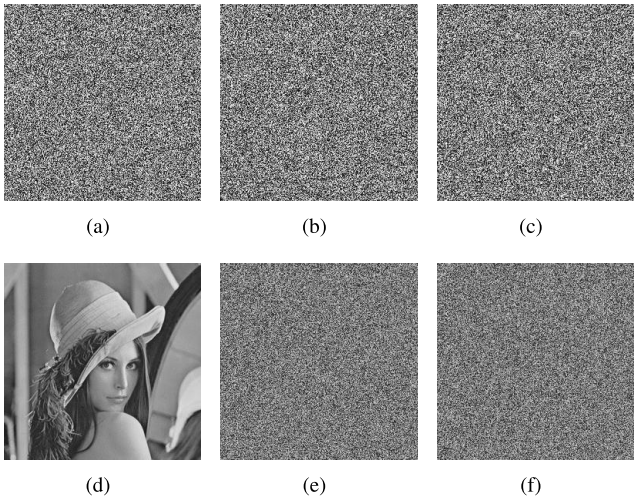


FIGURE 4. Key sensitivity analysis. (a) "Lena" with original keys K_0. (b) "Lena" with changed keys K_1. (c) "Lena" with changed keys K_2. (d) The decrypted image with correct keys K_0. (e) The decrypted image with changed keys K_1. (f) The decrypted image with changed keys K_2.

TABLE 2. Key space comparisons of different encryption schemes.

Schemes	Ref. [34]	Ref. [36]	Ref. [37]	Ref. [38]	Ref. [35]	Ref. [30]	Enhanced
Key space	2^{240}	10^{70}	10^{48}	10^{112}	10^{32}	10^{56}	10^{84}

the original cipher-image (Fig. 4(a)) is decrypted independently using K_0, K_1, K_2. The decipher images are shown in Fig. 4(d)-(f), respectively. As can be seen, plain-image can only be recovered with the correct key K_0; otherwise, the process will completely fail although decrypting with a tiny disturbed key. Conclusively, the above two experiments demonstrate that our enhanced scheme is extremely sensitive to the secret keys in encryption and decryption processes.

E. CORRELATION ANALYSIS

Two neighboring pixel pairs of a natural image always have a high correlation degree. An effective encryption scheme is expected to substantially reduce the correlation information of the cipher image to resist the statistical analysis attacks.

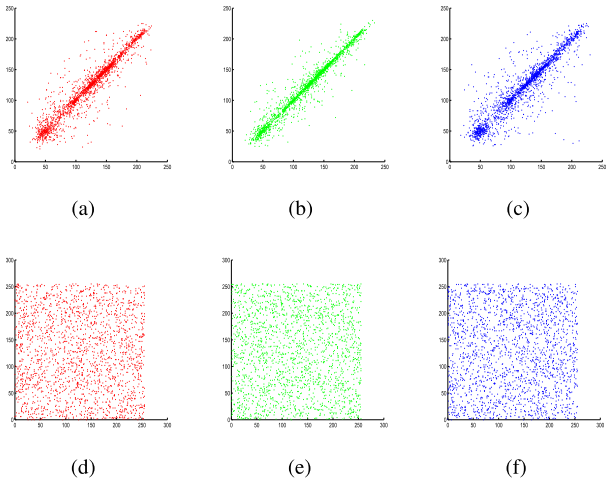


FIGURE 5. Correlation of plain "Lena" image and encrypted "Lena" image. (a) Horizontal correlation of plain-image. (b) Vertical correlation of plain-image. (c) Diagonal correlation of plain-image. (d) Horizontal correlation of encrypted image. (e) Vertical correlation of encrypted image. (f) Diagonal correlation of encrypted image.

We randomly select 2000 neighboring pixels in the horizontal, vertical and diagonal directions in the plain-image and the encrypted image and calculate their correlation degree by

$$r_{xy} = \frac{E((x - E(x))(y - E(y)))}{\sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))^2}\sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - E(y))^2}},$$

where x, y are two neighboring pixels in different directions, and $E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i$.

Figure 5 shows the correlation of the plain-image "Lena" and its cipher-image. The correlations of the encrypted image are random-like. Table 3 compares the correlations of different encryption schemes. The results demonstrate that the enhanced algorithm yields a lower correlation coefficient after encryption.

F. DIFFERENTIAL ATTACK ANALYSIS

The basic strategy of differential cryptanalysis is to obtain the largest possible keys by analyzing the impact of the specific plaintext difference on the corresponding ciphertext difference. Typically, the attackers make a tiny change in the plaintext and compare the corresponding ciphertext pairs with the difference in the plaintext. The number of pixels change rate

$$NPCR = \frac{\sum_{i,j} D(i,j)}{m \times n} \times 100\% \tag{12}$$

TABLE 4. NPCR and UACI of the encrypted Lena image.

Position	Original pixel	Changed pixel	NPCR	UACI
(100,1)	99	100	1	0.3343
(100,100)	93	92	1	0.3347
(233,233)	72	71	1	0.3347
(256,256)	98	97	1	0.3348

TABLE 3. Correlation comparison of different encryption schemes.

Test image	Direction	Plain-image	Cipher image				
			Ref. [34]	Ref. [38]	Ref. [35]	Ref. [30]	Enhanced
Lena(256×256)	Horizontal	0.9238	0.0023	0.0056	0.0041	0.0259	0.0023
	Vertical	0.9701	0.0019	0.0037	0.0021	0.0690	0.0031
	Diagonal	0.9082	0.0011	0.0032	0.0009	0.0617	0.0039
Baboon(256×256)	Horizontal	0.8658	0.0059	0.0026	0.0055	0.0100	0.0019
	Vertical	0.8121	0.0041	0.0009	0.0015	0.0082	0.0013
	Diagonal	0.7604	0.0028	0.0052	0.0041	0.0341	0.0039
Peppers(256×256)	Horizontal	0.9425	0.0037	0.0016	0.0021	0.0107	0.0018
	Vertical	0.9466	0.0258	0.0059	0.0218	0.0462	0.0030
	Diagonal	0.9143	0.0079	0.0034	0.0096	0.0345	0.0056
Boat(256×256)	Horizontal	0.9253	0.0073	0.0001	0.0073	0.0381	0.0050
	Vertical	0.9630	0.0109	0.0031	0.0216	0.0105	0.0035
	Diagonal	0.9013	0.0016	0.0015	0.0035	0.0302	0.0016

TABLE 5. Comparison for different encryption schemes in terms of NPCR and UACI scores.

Test image		Ref. [34]	Ref. [38]	Ref. [35]	Ref. [30]	Enhanced
Lena(256×256)	NPCR	0.9951	0.9962	1.5258×10^{-5}	1.5258×10^{-5}	1
	UACI	0.3358	0.3341	5.9838×10^{-8}	5.9838×10^{-8}	0.3337
Baboon(256×256)	NPCR	0.9910	0.9959	1.5258×10^{-5}	1.5258×10^{-5}	1
	UACI	0.3325	0.3338	5.9838×10^{-8}	5.9838×10^{-8}	0.3346
Peppers(256×256)	NPCR	0.9849	0.9960	1.5258×10^{-5}	1.5258×10^{-5}	1
	UACI	0.3294	0.3349	5.9838×10^{-8}	5.9838×10^{-8}	0.3348
Boat(256×256)	NPCR	0.9925	0.9961	1.5258×10^{-5}	1.5258×10^{-5}	1
	UACI	0.3339	0.3366	5.9838×10^{-8}	5.9838×10^{-8}	0.3345

and unified average changing intensity

$$UACI = \frac{1}{m \times n}\left[\sum_{i,j}\frac{|C(i,j) - C'(i,j)|}{255}\right] \times 100\% \quad (13)$$

are used to evaluate the strength of the encryption scheme against the differential attacks, where

$$D_{i,j} = \begin{cases} 1, & \text{if } C(i,j) \neq C'(i,j), \\ 0, & \text{if } C(i,j) = C'(i,j). \end{cases}$$

m and n are the height and width of the plain-image and C, C' are two cipher-images [39]. For resisting a differential attack, the best theoretical score of UACI is 0.33 and that of NPCR is 1. We select plain-image "Lena" and then change one bit of the pixel value in different positions. Then, we encrypt these plain-images with the same keys and calculate the NPCR and UACI scores of the encrypted image via Eq. (12). Table 4 lists the NPCR and UACI scores when a pixel value is changed. The NPCR values are always equal to the ideal value of 1, and the UACI values are close to the ideal value of 0.33. Hence, the enhanced scheme is highly sensitive to tiny changes in the plain-image: even if the two encrypted plain-images differ in only one bit, the two decrypted images will be entirely different. Table 5 compares the NPCR and UACI scores for different encryption schemes on some typical images. It is clear that the enhanced scheme obtains better scores. Thus, the proposed encryption scheme performs well in terms of robustness against the differential attacks.

G. INFORMATION ENTROPY ANALYSIS

For a cryptosystem, the information entropy is used to assess the randomness of the encrypted image. The information entropy of an image I is defined as

$$H(I) = \sum_{i=0}^{255} p(I_i) \log \frac{1}{p(I_i)}, \quad (14)$$

where I_i is the ith gray value of I and $P(I_i)$ is the probability of I_i [40]. For a truly random 8-bit image, the ideal information entropy value is 8. If a well-performing encryption algorithm is utilized, the cipher image is typically as random as possible; hence, the cipher image does not reveal any useful information to attackers.

Table 6 lists the entropy values of different encryption schemes for several typical images. According to Table 6, the information entropy values of the tested images encrypted via the enhanced scheme exceed those encrypted via other schemes, which indicates the enhanced scheme can protect information better.

TABLE 6. Information entropy comparison for different images encrypted by different schemes.

Test image	Ref. [34]	Ref. [38]	Ref. [35]	Ref. [30]	Enhanced
Lena	7.9994	7.9994	7.9924	7.9993	7.9993
Baboon	7.9981	7.9992	7.9922	7.9992	7.9994
Peppers	7.9983	7.9993	7.9921	7.9992	7.9993
Airplane	7.9991	7.9992	7.9925	7.9992	7.9993
Boat	7.9988	7.9994	7.9924	7.9993	7.9993

TABLE 7. Local Shannon entropy of different images encrypted by the enhanced scheme.

Test image	Local Shannon entropy	Result
Lena(512 × 512)	7.902976	Success
Baboon(512×512)	7.902303	Success
Peppers(512×512)	7.902521	Success
Couple(512×512)	7.902276	Success
Boat(512×512)	7.902876	Success

H. LOCAL SHANNON ENTROPY ANALYSIS

Local Shannon entropy is usually used to measure the randomness of encrypted images. In [41], Local Shannon entropy is defined as

$$\overline{H_{k,T_B}}(S) = \sum_{i=1}^{k} \frac{H(S_i)}{k}, \quad (15)$$

where $S_1, S_2, \cdots, S_k$ are non-overlapping image blocks selected randomly from encrypted images, and each block has T_B pixels. For $i = 1 \sim k$, $H(S_i)$ denotes the Shannon entropy for the image block S_i, which are computed by Eq. (14). We encrypted the test images using our enhanced scheme and then calculate the local Shonnon entropy of the encrypted images. Table 7 presents the local entropy values where $K = 30$ and $T_B = 1936$. According to [41], [42], the interval of (30, 1936)-local Shannon entropy of a random image should be between [7.901901305, 7.903037329] with confidence $\alpha = 0.05$. From Table 7, it is obvious that all the cipher images have passed local Shannon entropy test, which means the encrypted images generated by our enhanced scheme have good randomness.

I. ROBUSTNESS AGAINST THE CROPPING ATTACKS AND NOISE ATTACKS

The goal of cropping attack analysis is to determine whether the encrypted images can be deciphered when it is incom-

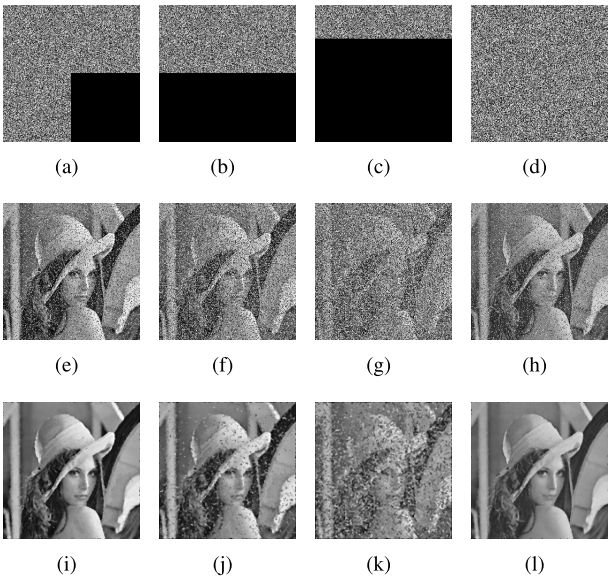


FIGURE 6. Robustness against the cropping attacks and noise attacks. (a) Encrypted image cropped by 25%. (b) Encrypted image cropped by 50%. (c) Encrypted image cropped by 75%. (d) Encrypted image with salt-and-pepper noise. (e) Decrypted image of (a). (f) Decrypted image of (b). (g) Decrypted image of (c). (h) Decrypted image of (d). (i) Filtered image of (e). (j) Filtered image of (f). (k) Filtered image of (g). (l) Filtered image of (h).

plete. And the noise attack analysis can assess the robustness of the encrypted image affected by noise in real-time transmission. We selected three encrypted images that have been cropped by 25% (Fig. 6(a)), 50% (Fig. 6(b)), 75% (Fig. 6(c)) and one encrypted images added a salt-and-pepper noise with signal noise rate 0.1 (Fig. 6(d)) and then decrypt them via the enhanced scheme using the correct secret keys respectively. Fig. 6 (e–h) show the corresponding decrypted images. We perform a simple median filtering process on the decrypted image respectively. Fig. 6 (i–l) show the corresponding filtered images. The results demonstrate that the decrypted images can be recognized when the encrypted images are subjected to the cropping attacks and noise attacks. Even if the encrypted images have been cropped by 75%, the outline of the decrypted image can still be recognized visually. Thus, the enhanced encryption scheme has good robustness against the cropping attacks and noise attacks.

J. COMPUTATIONAL COMPLEXITY ANALYSIS

For an encryption algorithm, the computational complexity is another important evaluation index. In this section, we compare the computational complexity of our enhanced scheme with that of other encryption algorithms. The size of plain-image is denoted as $m \times n$.

In [34], there are three steps in the encryption process, that is permutation, diffusion and permutation, and the number of operations are mn, $9mn$, and mn, respectively. Thus, its total computational complexity is $11mn$. In [38], the diffusion part includes DNA encoding, DNA decoding and two XOR operations. Besides, there is also a permutation after diffusion. Thus, the total number of operations are $5mn$. In [35], the encryption process has only one simple XOR operation, so its complexity is mn. However, it is highly insecure. The encryption process of [30] is performed at bit level, and the number of operations in permutation and diffusion are $4m + 4n$ and $8mn$, respectively. In our enhanced scheme, the encryption process is divided into two parts. The permutation of MSBs plane has the computation complexity mn. Besides, The confusion part includes a relation mechanism and XOR operations, whose computational complexity is $2mn$. In general, the number of operations of the entire enhanced algorithm is $3mn$, which depends on the size of the plain-image.

TABLE 8. Encryption speed comparisons of different encryption schemes (seconds).

Schemes	Ref. [34]	Ref. [38]	Ref. [35]	Ref. [30]	Ref. [9]	Enhanced
Time	5.687	6.610	0.113	0.278	2.055	0.760

K. SPEED ANALYSIS

In this section, we measure the encryption speed and decryption speed of our enhanced algorithm on image "Lena" of size 256×256. Our experiments are run on the compiler platform of MATLAB 8.3.0.532 (R2017a) with 8GB RAM and a 2.50GHz preprocessor. Table 8 lists the execution times of different encryption schemes. As can be seen, the encryption speed of our enhanced scheme is sufficiently fast to meet real-time performance requirements.

VI. CONCLUSION

This paper analyzed the security performance of an image encryption scheme based on bit-plane extraction and multiple chaotic maps. Based on the identified security defects, we proposed efficient know-plaintext and chosen-plaintext attacks for recovering some information of the original plain-image. Furthermore, we revised the defects of IESBC via multiple modifications: adopting a statistical value of the plain-image in the diffusion phase; building a relation mechanism between each position of the LSBs plane with the corresponding position in the MSBs plane to reduce the correlation among neighboring pixels of the plain-image; and utilizing a random number source making IESBC compose a probabilistic algorithm and can resist the plaintext attacks efficiently. In all, this paper provides a new way to enhance an insecure image encryption scheme.

REFERENCES

[1] H. Yin, Z. Qin, J. Zhang, L. Ou, and K. Li, "Achieving secure, universal, and fine-grained query results verification for secure search scheme over encrypted cloud data," *IEEE Trans. Cloud Computing*. to be published.
[2] X. Li, Y. Wang, and Q.-H. Wang, "Modified integral imaging reconstruction and encryption using an improved SR reconstruction algorithm," *Opt. Lasers Eng.*, vol. 112, pp. 162–169, Jan. 2019.
[3] X. Liao, J. Yin, S. Guo, X. Li, and A. K. Sangaiah, "Medical JPEG image steganography based on preserving inter-block dependencies," *Comput. Electr. Eng.*, vol. 67, pp. 320–329, Apr. 2018.

[4] H. Yin, Z. Qin, L. Ou, and K. Li, "A query privacy-enhanced and secure search scheme over encrypted data in cloud computing," *J. Comput. Syst. Sci.*, vol. 90, pp. 14–27, Dec. 2017.
[5] S. Li, G. Chen, and X. Mou, "On the dynamical degradation of digital piecewise linear chaotic maps," *Int. J. Bifurcation Chaos*, vol. 15, no. 10, pp. 3119–3151, Oct. 2005.
[6] F. Özkaynak, "Brief review on application of nonlinear dynamics in image encryption," *Nonlinear Dyn*, vol. 92, no. 2, pp. 305–313, Apr. 2018.
[7] E. Y. Xie, C. Li, S. Yu, and J. Lü, "On the cryptanalysis of Fridrich's chaotic image encryption scheme," *Signal Process.*, vol. 132, pp. 150–154, Mar. 2017.
[8] C. Li, Y. Liu, L. Y. Zhang, and K.-W. Wong, "Cryptanalyzing a class of image encryption schemes based on Chinese Remainder Theorem," *Signal Process.: Image Commun.*, vol. 29, no. 8, pp. 914–920, Sep. 2014.
[9] W. Zhang, H. Yu, Y.-L. Zhao, and Z.-L. Zhu, "Image encryption based on three-dimensional bit matrix permutation," *Signal Process.*, vol. 118, pp. 36–50, Jan. 2016.
[10] Z. Hua, S. Yi, and Y. Zhou, "Medical image encryption using high-speed scrambling and pixel adaptive diffusion," *Signal Process.*, vol. 144, pp. 134–144, Mar. 2018.
[11] Q. Yin and C. Wang, "A new chaotic image encryption scheme using breadth-first search and dynamic diffusion," *Int. J. Bifurcation Chaos*, vol. 28, no. 04, Apr. 2018, Art. no. 1850047.
[12] L. Zhang, X. Liao, and X. Wang, "An image encryption approach based on chaotic maps," *Chaos Solitons Fractals*, vol. 24, no. 3, pp. 759–765, May 2005.
[13] S. E. Assad and M. Farajallah, "A new chaos-based image encryption system," *Signal Process. Image Communication*, vol. 41, pp. 144–157, 2016.
[14] C. Li, D. Lin, B. Feng, and J. Lü, and F. Hao, "Cryptanalysis of a chaotic image encryption algorithm based on information entropy," *IEEE Access*, vol. 6, pp. 75834–75842, 2018.
[15] X. Chai, Z. Gan, Y. Ke, Y. Chen, and X. Liu, "A novel image encryption scheme based on DNA sequence operations and chaotic systems," *Neural Comput. Appl.*, vol. 31, no. 1, pp. 219–237, Jan. 2019.
[16] Z. Hua, F. Jin, B. Xu, and H. Huang, "2D Logistic-Sine-coupling map for image encryption," *Signal Process.*, vol. 149, pp. 148–161, Aug. 2018.
[17] L. Y. Zhang, C. Li, K.-W. Wong, S. Shu, and G. Chen, "Cryptanalyzing a chaos-based image encryption algorithm using alternate structure," *J. Syst. Softw.*, vol. 85, no. 9, pp. 2077–2085, Sep. 2012.
[18] C. Li, D. Lin, and J. Lü, and F. Hao, "Cryptanalyzing an image encryption algorithm based on autoblocking and electrocardiography," *IEEE Multimedia*, vol. 25, no. 4, pp. 46–56, Dec. 2018.
[19] M. Garcia-Bosque, A. Pérez-Resa, and C. Sánchez-Azqueta, C. Aldea, and S. Celma, "Chaos-based bitwise dynamical pseudorandom number generator On FPGA," *IEEE Trans. Instrum. Meas.*, vol. 68, no. 1, pp. 291–293, Jan. 2019.
[20] C. Li, B. Feng, S. Li, J. Kurths, and G. Chen, "Dynamic analysis of digital chaotic maps via state-mapping networks," *IEEE Trans. Circuits Syst. I, Reg. Papers*, to be published.
[21] S. J. Sheela, K. V. Suresh, and D. Tandur, "Image encryption based on modified Henon map using hybrid chaotic shift transform," *Multimedia Tools Appl.*, vol. 77, no. 19, pp. 25223–25251, Oct. 2018.
[22] F. Peng, X.-W. Zhu, and M. Long, "An ROI privacy protection scheme for H.264 video based on FMO and chaos," *IEEE Trans. Inf. Forensics Security*, vol. 8, no. 10, pp. 1688–1699, Oct. 2013.
[23] H. Yang, X. Liao, K.-W. Wong, W. Zhang, and P. Wei, "A new cryptosystem based on chaotic map and operations algebraic," *Chaos Solitons Fractals*, vol. 40, no. 5, pp. 2520–2531, Jun. 2009.
[24] R. Lukac and K. N. Plataniotis, "Bit-level based secret sharing for image encryption," *Pattern Recognit.*, vol. 38, no. 5, pp. 767–772, May 2005.
[25] Z. Tang, J. Song, X. Zhang, and R. Sun, "Multiple-image encryption with bit-plane decomposition and chaotic maps," *Opt. Lasers Eng.*, vol. 80, pp. 1–11, May 2016.
[26] L. Xu, Z. Li, J. Li, and W. Hua, "A novel bit-level image encryption algorithm based on chaotic maps," *Opt. Lasers Eng.*, vol. 78, no. 21, pp. 17–25, Mar. 2012.
[27] Y. Li, C. Wang, and H. Chen, "A hyper-chaos-based image encryption algorithm using pixel-level permutation and bit-level permutation," *Opt. Lasers Eng.*, vol. 90, pp. 238–246, Mar. 2017.
[28] M. Xu and Z. Tian, "A novel image cipher based on 3D bit matrix and latin cubes," *Inf. Sci.*, vol. 478, pp. 1–14, Apr. 2018.
[29] C. Li, D. Lin, and J. Lü, "Cryptanalyzing an image-scrambling encryption algorithm of pixel bits," *IEEE Multimedia*, vol. 24, no. 3, pp. 64–71, Aug. 2017.
[30] A. Shafique and J. Shahid, "Novel image encryption cryptosystem based on binary bit planes extraction and multiple chaotic maps," *Eur. Phys. J. Plus*, vol. 133, no. 8, p. 331, Aug. 2018.
[31] J. Wu, X. Liao, and B. Yang, "Cryptanalysis and enhancements of image encryption based on three-dimensional bit matrix permutation," *Signal Process.*, vol. 142, pp. 292–300, Jan. 2017.
[32] X. Chai, X. Fu, Z. Gan, Y. Lu, and Y. Chen, "A color image cryptosystem based on dynamic DNA encryption and chaos," *Signal Process.*, vol. 155, pp. 44–62, Feb. 2019.
[33] G. Alvarez and L. I. Shujun, "Some basic cryptographic requirements for chaos-based cryptosystems," *Int. J. Bifurcation Chaos*, vol. 16, no. 08, pp. 2129–2151, Aug. 2006.
[34] R. Enayatifar, A. H. Abdullah, I. F. Isnin, A. Altameem, and M. Lee, "Image encryption using a synchronous permutation-diffusion technique," *Opt. Lasers Eng.*, vol. 90, pp. 146–154, Mar. 2017.
[35] C. Li, G. Luo, Q. Ke, and C. Li, "An image encryption scheme based on chaotic tent map," *Nonlinear Dyn*, vol. 87, no. 1, pp. 127–133, Jan. 2017.
[36] X. Chai, X. Zheng, Z. Gan, D. Han, and Y. Chen, "An image encryption algorithm based on chaotic system and compressive sensing," *Signal Process.*, vol. 148, pp. 124–144, Jul. 2018.
[37] X. Wang and K. Guo, "A new image alternate encryption algorithm based on chaotic map," *Nonlinear Dyn*, vol. 76, no. 4, pp. 1943–1950, Jun. 2014.
[38] J. Wu, X. Liao, and Y. Bo, "Image encryption using 2D Hénon-Sine map and DNA approach," *Signal Process.*, vol. 153, pp. 11–23, Dec. 2018.
[39] Y. Wu, J. P. Noonan, and S. Agaian, "NPCR and UACI randomness tests for image encryption," *Cyber J.: Multidisciplinary J. Sci. Technol., J. Sel.Areas Telecommun.*, vol. 1, no. 2, pp. 31–38, Apr. 2011.
[40] C. Pak and L. Huang, "A new color image encryption using combination of the 1D chaotic map," *Signal Process.*, vol. 138, pp. 129–137, Sep. 2017.
[41] Y. Wu *et al.*, "Local shannon entropy measure with statistical tests for image randomness," *Inf. Sci.*, vol. 222, pp. 323–342, Feb. 2013.
[42] X. Wang and C. Liu, "A novel and effective image encryption algorithm based on chaos and DNA encoding," *Multimedia Tools Appl.*, vol. 76, no. 5, pp. 6229–6245, Mar. 2017.

YU LIU received the B.S. degree in information security from Yunnan University, Yunnan, China, in 2016. She is currently pursuing the Ph.D. degree in computer science and technology with Hunan University, China. Her research interests include information security and social network analysis.

ZHENG QIN received the Ph.D. degree in computer software and theory from Chongqing University, China, in 2001. He is currently a Full Professor with the College of Computer Science and Electronic Engineering, Hunan University, China. His main interests include information security, cloud computing, big data processing, and software engineering. He has accumulated rich experiences in product development and application services, especially in the area of financial, medical and education sectors. He is a member of the China Computer Federation (CCF) and the ACM.

JIAHUI WU received the M.S. degree in signal and information processing from Southwest University, Chongqing, China, in 2018, where she is currently pursuing the Ph.D. degree in intelligent computing and information processing. Her primary research interests include information security, cloud computing security, and data mining.