IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 21, 2019, accepted June 3, 2019, date of publication June 14, 2019, date of current version July 12, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2923018

Efficient and Secure Image Encryption Algorithm Using a Novel Key-Substitution Architecture

YANJIE SONG, ZHILIANG ZHU, (Member, IEEE), WEI ZHANG, HAI YU, AND YULI ZHAO
Software College, Northeastern University, Shenyang 110819, China

Corresponding author: Zhiliang Zhu (zzlswc@163.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61374178, Grant 61402092, and Grant 61603182, in part by the Fundamental Research Funds for the Central Universities under Grant N171704004, in part by the Online Education Research Fund of MOE Research Center for Online Education, China (Qtone Education) under Grant 2016ZD306, and in part by the Ph.D. Start-Up Foundation of Liaoning Province, China, under Grant 201501141.

ABSTRACT Recently, many chaos-based image encryption algorithms have been proposed. Most of them adopt the traditional confusion-diffusion framework. Multiple encryption rounds or one round of complex encryption is typically executed in these schemes to realize high security. However, these operations will reduce the computational efficiency. To overcome these issues, we propose a novel key-substitution encryption architecture (KSA). Under the proposed KSA, we design a key scheming for updating the initial keys using the plain-image and develop a new substitution method for encrypting various types of images. The simulation results and security analysis demonstrate the superior security and high efficiency of the proposed image encryption algorithm using the KSA (KSA-IEA), which executes only one round of encryption.

INDEX TERMS Encryption architecture, key scheming, plaintext sensitivity, high efficiency, image encryption.

I. INTRODUCTION

In the modern society, digital images have played an increasingly important role because they are easy to understand and provide vivid description. Protecting the security of digital images should be considered because many images contain secret or private information. Various traditional encryption algorithms have been proposed, such as Data Encryption Standard (DES), International Data Encryption Algorithm (IDEA) and Advanced Encryption Standard (AES). However, digital images contain substantial amounts of content, hence, the efficiency of conventional ciphers is low and they are not suitable for encrypting images. Therefore, many image encryption algorithms that use other encryption techniques have been proposed for realizing reasonable encryption performance.

There are various types of image encryption algorithms, for example, algorithms that are based on Latin square [1]–[3], on Sudoku matrix [4], on DNA [5]–[8], on quantum [9]–[13], on waves [14], [15], on electrocardiograph (ECG) signals [16] and on chaos [17]–[26]. Most image ciphers adopt the traditional confusion-diffusion structure that was proposed by Fridrich [27]. Under this structure, confusion is performed to change the pixel positions and the diffusion operation is employed to alter the pixel values. Fig. 1 illustrates this classical encryption architecture.

Chaos-based image ciphers have attracted researchers' attention due to the fundamental characteristics of chaotic systems, such as ergodicity, sensitivity to initial conditions and unpredictability. The existing chaos-based image cryptosystems can be classified into two categories.

In the first category [17]–[21], multiple rounds of confusion-diffusion are executed to realize high security, where the same encryption operation is repeated in each round. In [17], a novel image encryption algorithm is proposed by combining a new two-dimensional Sine Logistic modulation map (2D-SLMM) with a chaotic magic transform (CMT), in which two encryption rounds are performed. Simulation results demonstrate that the proposed

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

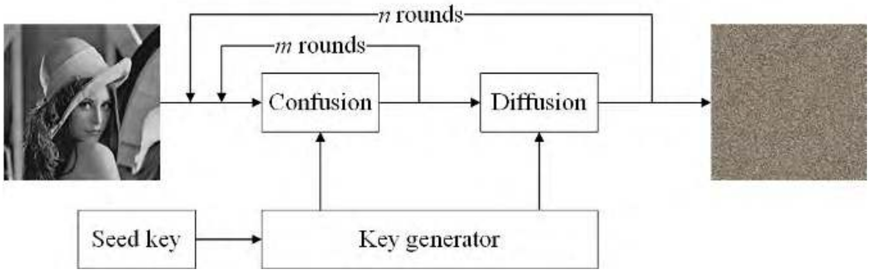


FIGURE 1. Fridrich image encryption architecture.

algorithm has a high security level and can resist various attacks. In [18], an image encryption scheme that is based on the two-dimensional Logistic-adjusted-Sine map (2D-LASM) is presented and two rounds of confusion-diffusion are executed. According to the security analysis, this scheme has strong resistance against various security attacks. In [19], an image encryption algorithm with two rounds of permutation and diffusion operations is proposed, and it is based on a new two-dimensional Logistic-modulated-Sine-coupling-Logistic chaotic map (LSMCL). The results of theoretical analyses and simulations support the security and validity of the proposed algorithm. In [20], three rounds of new digit-level permutations are combined with three rounds of high-speed diffusion operations to encrypt an image. Simulation results demonstrate that this encryption algorithm realizes high security and efficiency. In [21], an image cryptosystem that is based on simultaneous permutation and diffusion is proposed. Under two encryption rounds, the proposed cipher has satisfactory robustness against various types of attacks.

In the encryption algorithms in the second category [22]–[26], only one round of confusion-diffusion is implemented and the security is improved via complex operations. In [22], a novel encryption system that is based on a new one-dimensional chaotic system is proposed, and its excellent performance in resisting various attacks is demonstrated experimentally. In [23], a new image encryption scheme with the complex pre-modular, permutation and diffusion is presented, in which the keystream that is used for encryption depends on the plain-image. The results of numerical experiments and a security analysis demonstrate that this cipher is secure. In [24], chaotic confusion and pixel diffusion are designed based on an improved one-dimensional chaotic system and SHA-256 is employed to generate the initial conditions. The simulation results demonstrate the high encryption performance of the proposed image encryption algorithm. In [25], a novel image encryption algorithm that is based on a new two-dimensional chaotic map is proposed by using bit-level confusion and diffusion simultaneously, in which the initial values of the chaotic system are updated according to the obtained ciphertext. According to the performance analysis, this algorithm realizes satisfactory encryption performance and high efficiency. In [26], an encryption algorithm for color images that is based on a new four-dimensional chaotic system is proposed. The results of simulations and a security analysis demonstrate that the proposed image encryption algorithm performs well in terms of security, robustness and efficiency.

Under the conventional confusion-diffusion framework, multiple encryption rounds or one round of complex encryption is typically executed in these chaos-based image encryption algorithms to realize good encryption performance. However, according to our analysis, they provide inadequate security or have low encryption efficiency. In this paper, we propose a novel key-substitution architecture (KSA)-based image encryption algorithm (KSA-IEA), which executes only one encryption round. The proposed KSA-IEA can avoid the security and efficiency problems that are discussed above.

Our main contributions are as follows:

(1) To overcome the problems of the traditional confusion-diffusion structure, we present a new key-substitution encryption architecture that improves the security and encryption efficiency.
(2) A novel key scheming that is based on weighted summation is designed for enhancing the sensitivity to slight changes in the plain-image and the resistance against known/chosen plaintext attacks.
(3) We propose a new substitution method, in which random grouping, S-box construction, S-box allocation and random substitution are implemented to encrypt the plain-image to realize satisfactory encryption performance.
(4) The proposed image encryption algorithm is evaluated under only one encryption round and its superior encryption performance is demonstrated.

The remainder of this paper is organized as follows: Section II introduces the proposed key-substitution architecture. Section III describes the proposed image encryption algorithm. Section IV presents the simulation results and the security analysis. Section V presents the conclusion of this paper.

II. KEY-SUBSTITUTION ARCHITECTURE (KSA)

In this paper, we propose a novel key-substitution encryption architecture that enhances the security of the

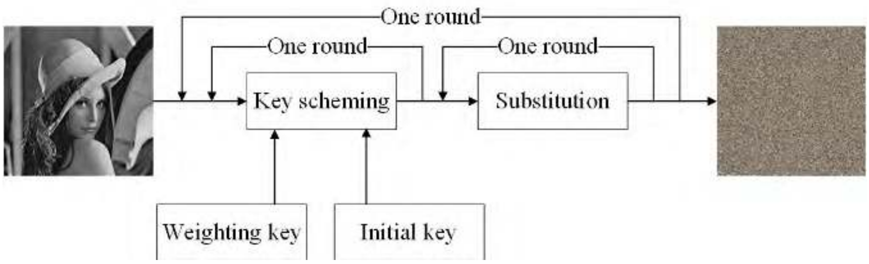


FIGURE 2. Proposed key-substitution encryption architecture.

cryptosystem and reduces the computational complexity. The KSA includes two main stages: key scheming and substitution. Unlike the traditional confusion-diffusion framework, we use a simple key scheming that is based on chaos instead of the diffusion operation to provide sensitivity to changes in the plain-image so that the KSA-based image encryption algorithm can resist known/chosen plaintext attacks. In the substitution phase, the values of all pixels are changed using the chaos-based S-box. The computational efficiency can be enhanced. Under the proposed KSA, one round of encryption can yield satisfactory encryption performance. Fig. 2 illustrates this novel encryption architecture.

A. KEY SCHEMING

Key scheming can establish a close relationship between the plain-image and the initial key by using the plaintext information to update the initial key such that the initial key will be changed according to the plain-image. In the image encryption algorithm based on chaos, the chaotic system is highly sensitive to the initial key. If the initial key is altered as plain-image changes are made to the plain-image, then the corresponding key stream will be changed and, hence, will yield a different cipher-image. Therefore, key scheming can provide plaintext sensitivity for the chaotic image encryption algorithm. The stronger the plaintext sensitivity of the key scheming is, the better the encryption performance of the cryptosystem is. An excellent key scheming can make the cryptosystem highly sensitive to the plain-image with one round of encryption via a simple operation for resisting known/chosen plaintext attacks, hence, multiple rounds of encryption operations or one round of complex encryption operations can be avoided and the encryption efficiency can be enhanced.

The most basic unit of an image is a bit, hence, a plain-image is altered by changing one or more of its bits. However, the bits in the same bit-plane for a plain-image have the same weight and the weights of the bits in different bit-planes are proportional. Therefore, key scheming may provide the lower plaintext sensitivity, for example, via summation over all pixels [28], [29].

We propose a novel key scheming based on weighted summation (KS-WS), which can avoid the problems that are described above. The corresponding equation is as follows:

$$
\begin{aligned}
W_1 \times I \times W_2
&= \begin{bmatrix} w_1 & w_2 & \cdots & w_M \end{bmatrix} \\
&\quad \times \begin{bmatrix} I_1 & I_{M+1} & \cdots & I_{(N-1)M+1} \\ I_2 & I_{M+2} & \cdots & I_{(N-1)M+2} \\ \vdots & \vdots & \ddots & \vdots \\ I_M & I_{2M} & \cdots & I_{MN} \end{bmatrix} \times \begin{bmatrix} w_{M+1} \\ w_{M+2} \\ \vdots \\ w_{M+N} \end{bmatrix} \\
&= S_{WI}, \qquad (1)
\end{aligned}
$$

where I represents the plain-image of size $M \times N$ and S_{WI} is the weighted summation. We employ a chaotic system to construct two different pseudo-random vectors, which are denoted as W_1 of size $1 \times M$ and W_2 of size $N \times 1$. According to Eq. (1), each bit in the plain-image I has a different pseudo-random weight and there is no proportional relationship between the weights of bits. If one or more bits of the plain-image I are changed, the weighted summation S_{WI} will also change. Therefore, the weighted summation S_{WI} can be used to update the initial key to provide strong plaintext sensitivity.

In this paper, the Logistic system [30] is selected and its equation is presented as Eq. (2).

$$z_{n+1} = \mu z_n(1 - z_n), \quad z_n \in (0, 1), \qquad (2)$$

where the Logistic system is chaotic if $\mu \in [3.5699465, 4]$ and the state value satisfies $z \in (0, 1)$.

B. SUBSTITUTION

Substitution has low computational complexity and can change the pixel values. However, the distribution of pixel values remains nonuniform because substitution is equivalent to scrambling the items of the histogram. Thus, traditional substitution cannot provide resistance to statistical analysis attacks. In this paper, we propose a novel substitution method (SM) that overcomes this problem.

The proposed SM consists of three main steps: random grouping, S-box construction and random substitution. First, we use a chaotic system to randomly divide the plain-image into multiple groups for substitution and the encrypted pixels are returned to their original positions. Via this approach, the correlation between adjacent pixels can be eliminated and the encryption order of the pixels is unknown to the

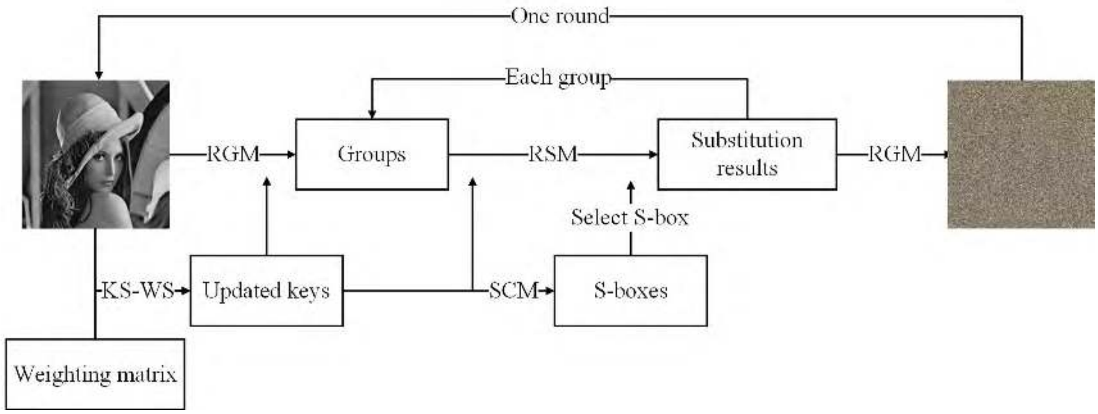


FIGURE 3. Block diagram of the proposed KSA-IEA.

attackers. Second, we construct a chaotic S-box that reduces the storage and transmission space due to the intrinsic properties of chaos. Finally, we randomly allocate the constructed S-boxes to each group to execute the substitution, which can reduce the number of the constructed S-boxes. For realizing the uniform distribution of the cipher-image, a new random substitution method is presented in Eq. (3).

$$c(i) = \begin{cases} S_1(p(i) \oplus S_2(\gamma)), & i = 1 \\ S_1(p(i) \oplus S_2(c(i-1))), & i \neq 1, \end{cases} \tag{3}$$

where $p(i)$ and $c(i)$ represent the ith plaintext and ciphertext, S_1 and S_2 are the allocated S-boxes, and the pseudo-random integer $\gamma \in [0, 255]$ makes $S_2(\gamma)$ pseudo-random. In this random substitution method, each plaintext $p(i)$ is converted into a random value by executing a Xor operation with a pseudo-random value $S_2(\gamma)$ (or $S_2(c(i-1))$) prior to substitution, hence, the output $c(i)$ of the S-box S_1 is also a random value. Thus, we will obtain the cipher-image with a uniform distribution. Via the proposed SM, the image encryption algorithm can realize satisfactory encryption performance.

III. IMAGE ENCRYPTION ALGORITHM USING KEY-SUBSTITUTION ARCHITECTURE (KSA-IEA)

In this paper, we propose a novel KSA-based image encryption algorithm, namely, KSA-IEA, which includes two main methods KS-WS and SM, and executes only one encryption round. In the SM, we design three steps random grouping method (RGM), S-box construction method (SCM) and random substitution method (RSM) for encrypting various types of images. A block diagram of the encryption process is presented in Fig. 3 and the process is described in detail as follows.

A. KEY SCHEMING BASED ON WEIGHTED SUMMATION (KS-WS)

On the basis of Sec. II-A, we design a key scheming, namely, KS-WS, for updating the initial keys using the plain-image to obtain the updated keys, which can provide strong sensitivity to the slight changes in the plain-image. The process for an $M \times N$ grayscale image I is described as follows:

Step 1: Iterate the Logistic system with the weighting key (z_k, μ_k) via Eq. (2) and discard the first 500 values to obtain two pseudo-random weighting vectors: W_1 of size $1 \times M$ and W_2 of size $N \times 1$.

Step 2: Calculate the weighted summation S_{WI} using the weighting vectors W_1, W_2 and the plain-image I via Eq. (1).

Step 3: Update the initial key z_0 with the fractional part of S_{WI} via Eq. (4) to obtain the updated key z.

$$z = mod(z_0 + (S_{WI} - floor(S_{WI})), 1). \tag{4}$$

B. SUBSTITUTION METHOD (SM)

1) RANDOM GROUPING METHOD (RGM)

In a block cipher, the plain-image is often divided into image blocks to be encrypted using the same operation. However, the correlation between adjacent pixels in the image block is high and the pixel grouping is known to the attacker, which will reduce the security of the encryption algorithm. Thus, we design a random grouping method, namely, RGM, for overcoming the above issues. First, we employ chaotic sequences to group the pixels randomly. Second, the current group of pixels is encrypted using the previous group of the ciphertext. Finally, the encrypted pixels are returned to their original positions to produce the final cipher-image. The designed RGM can eliminate the correlation between adjacent pixels. Meanwhile, the pixel grouping and the encryption order of the pixels are both unknown to the third party. Therefore, the security can be enhanced by our RGM. The steps of the proposed RGM are as follows:

Step 1: Update the initial key (z_{01}, μ_{01}) via KS-WS to obtain the updated key (z_1, μ_1).

Step 2: Generate a $1 \times M$ pseudo-random sequence R by iterating the Logistic system with (z_1, μ_1) and discarding the first 500 values. Then, abandon 30 values successively to obtain a $1 \times N$ pseudo-random sequence C and three pseudo-random values, namely, S_r, S_c and D.

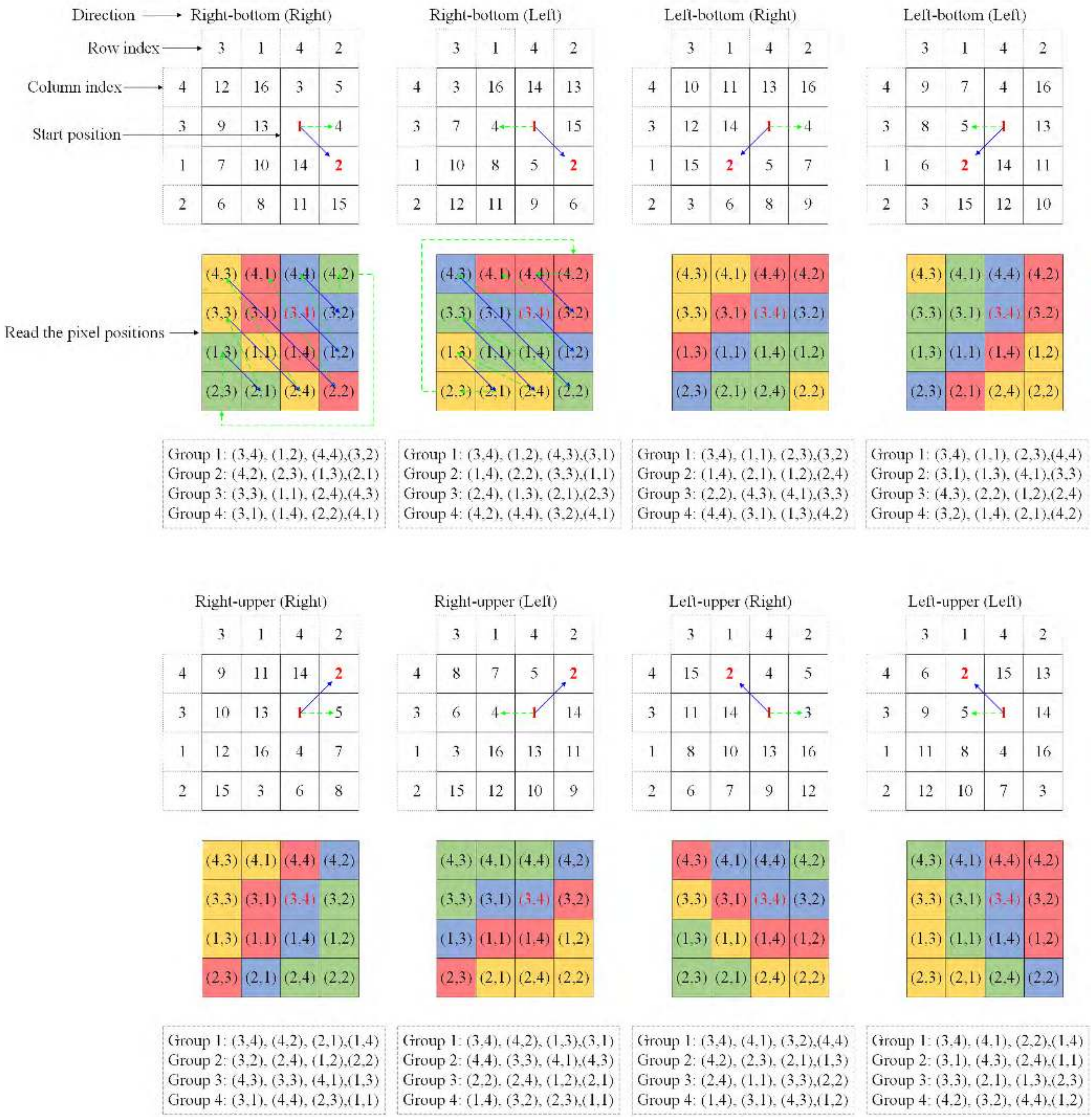


FIGURE 4. Proposed RGM.

Step 3: Sort R and C in ascending order to obtain the row index sequence R' and the column index sequence C'.

Step 4: Calculate the start point position (S'_r, S'_c) and the direction D' via Eq. (5).

$$\begin{cases} S'_r = mod(floor(S_r \times 10^{14}), M) + 1 \\ S'_c = mod(floor(S_c \times 10^{14}), N) + 1 \\ D' = mod(floor(D \times 10^{14}), 8). \end{cases} \tag{5}$$

Step 5: Read the pixel positions according to R', C', (S'_r, S'_c) and D'.

Step 6: Divide all pixels into $G = \min(M, N)$ groups (each group includes $E = \max(M, N)$ pixels and PI represents the grouped plaintext of size $G \times E$) based on the pixel index sequence that was obtained in Step 5.

Fig. 4 illustrates the process of RGM. According to this figure, there are eight directions and the directions yield various random groups. For example, in the first subfigure, the row index and the column index are read from the start position in right-bottom order in the right direction. The detailed read order is shown in this subfigure. According to Step 6, every four pixel indices are grouped together to yield four groups, so the pixels in each group and the encryption orders of the pixels are achieved. Unlike the first subfigure, the read order of the second subfigure is the left direction, so different groups and pixel encryption orders are obtained.

2) S-BOX CONSTRUCTION METHOD (SCM)

In mathematics, an $n \times n$ S-box receives n bits as input and can output n bits. It is a nonlinear mapping S: $\{0, 1\}^n \to \{0, 1\}^n$, where $\{0, 1\}^n$ represents the vector space of n elements from GF(2). GF(2) is the Galois field of two elements, and it is the smallest field. In a block cipher, an S-box as an essential component can provide nonlinear confusion to realize a high level of randomness for the cipher-image. A chaotic S-box is sensitive to the initial key and can reduce the storage and transmission space due to the intrinsic properties of chaos. Therefore, we design a simple and efficient S-box construction method based on chaos, namely, SCM.

In the proposed SCM, we construct 16 S-boxes and the S-boxes are randomly selected for each group, which can strengthen the security of the encryption algorithm and reduce the number of constructed S-boxes. According to the scenario, the number of S-boxes can be adjusted. The process of SCM is described in detail as follows:

Step 1: Update the initial key (z_{02}, μ_{02}) via KS-WS to obtain the updated key (z_2, μ_2).

Step 2: Generate 16 pseudo-random values $\{z_{s_1}, z_{s_2}, \cdots, z_\delta, \cdots, z_{s_{16}}\}$ as the initial keys for constructing the S-boxes by iterating the Logistic system with (z_2, μ_2), where the first 15 values are discarded.

Step 3: Iterate the Logistic system with one of the initial keys that were obtained by Step 2 and discard the first 500 pseudo-random values to produce a chaotic sequence $\hat{S}_1$ of size 1×256.

Step 4: Sort $\hat{S}_1$ in ascending order to obtain the sorted index sequence $\hat{S}'_1 = \hat{S}'_1 - 1$ as the initial 1D S-box.

Step 5: Produce two pseudo-random values, which are denoted as p and q, by abandoning 30 values successively and convert them into two integers, namely, $p', q' \in [0, 15]$, as the parameters of Arnold's cat map via Eq. (6).

$$\begin{cases} p' = mod(floor(p \times 10^{14}), 16) \\ q' = mod(floor(q \times 10^{14}), 16). \end{cases} \tag{6}$$

Step 6: Divide each position of $\hat{S}'_1$ into two 4-bit index values (x_i, y_i) and calculate the new index values (x'_i, y'_i) via Arnold's cat map according to Eq. (7). The elements of $\hat{S}'_1$ are inserted into their new positions to obtain the 2D S-box S^δ.

$$\begin{bmatrix} x'_i \\ y'_i \end{bmatrix} = \begin{bmatrix} 1 & p' \\ q' & p'q'+1 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix} (mod\ 16). \tag{7}$$

Step 7: Repeat Step 3 - Step 6 to obtain 16 S-boxes.

3) RANDOM SUBSTITUTION METHOD (RSM)

According to the introduction of Sec. II-B, we design a random substitution method, namely, RSM. In our RSM, the input value of the S-box is changed to a random value prior to substitution, hence, the output result will be a random value and the pixel distribution of the cipher-image will be uniform, which can resist statistical analysis attacks. Meanwhile, we randomly assign two S-boxes to each group. Via this approach, the diversity of S-box combinations can reduce the number of constructed S-boxes and enhance the security. In the proposed RSM, the current group of pixels is encrypted using the previous group of the ciphertext. The steps of our RSM are as follows:

Step 1: Update the initial key (z_{03}, μ_{03}) via KS-WS to obtain the updated key (z_3, μ_3).

Step 2: Iterate the Logistic system with (z_3, μ_3) to produce $G \times 3$ pseudo-random values P by discarding 10 values successively, where the first 500 values are abandoned to eliminate the transient effect.

Step 3: Obtain three pseudo-random values $\{p_{i,1}, p_{i,2}, p_{i,3}\}$ $(i \in \{1, 2, \cdots, G\})$ from P at a time for calculating two indices of the S-box, namely, $\{\delta_{i,1}, \delta_{i,2}\}$, and a random integer γ_i via Eq. (8), where $CI(i-1, E)$ is the last ciphertext of the previous group.

$$\begin{cases} \delta_{i,1} = \begin{cases} mod(floor(p_{i,1} \times 10^{14}), 16) + 1, & i = 1 \\ mod(mod(floor(p_{i,1} \times 10^{14}), 256) \\ \oplus CI(i-1, E), 16) + 1, & i \neq 1 \end{cases} \\ \delta_{i,2} = \begin{cases} mod(floor(p_{i,2} \times 10^{14}), 16) + 1, & i = 1 \\ mod(mod(floor(p_{i,2} \times 10^{14}), 256) \\ \oplus CI(i-1, E), 16) + 1, & i \neq 1 \end{cases} \\ \gamma_i = \begin{cases} mod(floor(p_{i,3} \times 10^{14}), 256), & i = 1 \\ mod(floor(p_{i,3} \times 10^{14}), 256) \\ \oplus CI(i-1, E), & i \neq 1. \end{cases} \end{cases} \tag{8}$$

Step 4: Substitute the plaintext of the ith group via Eq. (9) to obtain the ciphertext, where the function $f(x)$ converts x into two 4-bit index values, which serve as the inputs of the S-box.

$$CI(i,j) = \begin{cases} S_1^{\delta_{i,1}}(f(PI(i,j) \oplus S_2^{\delta_{i,2}}(f(\gamma_i)))), & i = 1 \\ S_1^{\delta_{i,1}}(f(PI(i,j) \oplus S_2^{\delta_{i,2}}(f(CI(i,j-1))))), & i \neq 1. \end{cases} \tag{9}$$

Step 5: Repeat Step 3 and Step 4 to obtain the ciphertext and return the encrypted pixels to their original positions to obtain the final cipher-image I'.

C. ENCRYPTION PROCESS

The whole encryption process of the proposed KSA-IEA is presented in Algorithm 1.

Algorithm 1 Encryption Process of KSA-IEA

Input: Plain-image I, initial keys (z_k, μ_k), (z_{01}, μ_{01}), (z_{02}, μ_{02}) and (z_{03}, μ_{03}).
Output: Cipher-image I'.

Update all initial keys (z_{01}, μ_{01}), (z_{02}, μ_{02}) and (z_{03}, μ_{03}) to obtain the updated keys (z_1, μ_1), (z_2, μ_2) and (z_3, μ_3) using **KS-WS** with the weighting key (z_k, μ_k);
Divide the plain-image I into $G = \min(M, N)$ groups to obtain the grouped plaintext PI of size $G \times E$ ($E = \max(M, N)$) using **RGM** with (z_1, μ_1);
Construct 16 S-boxes S^{δ} using **SCM** with (z_2, μ_2);
for $i = 1 : G$ **do** % Substitute each group using **RSM** with (z_3, μ_3)
    if $i = 1$ **then** % Calculate two S-box indices $\{\delta_{i,1}, \delta_{i,2}\}$ and a random integer γ_i
        $\delta_{i,1} = mod(floor(p_{i,1} \times 10^{14}), 16) + 1$;
        $\delta_{i,2} = mod(floor(p_{i,2} \times 10^{14}), 16) + 1$;
        $\gamma_i = mod(floor(p_{i,3} \times 10^{14}), 256)$;
    else
        $\delta_{i,1} = mod(mod(floor(p_{i,1} \times 10^{14}), 256) \oplus CI(i-1, E), 16) + 1$;
        $\delta_{i,2} = mod(mod(floor(p_{i,2} \times 10^{14}), 256) \oplus CI(i-1, E), 16) + 1$;
        $\gamma_i = mod(floor(p_{i,3} \times 10^{14}), 256) \oplus CI(i-1, E)$;
    end if
    for $j = 1 : E$ **do** % Substitute each pixel
        if $j = 1$ **then**
            $CI(i, j) = S_1^{\delta_{i,1}}(f(PI(i, j) \oplus S_2^{\delta_{i,2}}(f(\gamma_i))))$;
        else
            $CI(i, j) = S_1^{\delta_{i,1}}(f(PI(i, j) \oplus S_2^{\delta_{i,2}}(f(CI(i, j-1)))))$;
        end if
    end for
end for
Return the pixels of CI to their original positions using **RGM** to obtain the final cipher-image I'.

D. DECRYPTION PROCESS

Decryption is the reverse process of encryption. The updated keys (z_1, μ_1), (z_2, μ_2) and (z_3, μ_3) are transmitted to decrypt the cipher-image I'. First, RGM is executed to obtain the grouped ciphertext. Second, we construct 16 S-boxes via SCM. Then, each group of ciphertext is substituted using RSM to obtain the decrypted pixels. Finally, these pixels are returned to their corresponding positions to obtain the original image I.

E. KSA-IEA FOR COLOR IMAGES

We also design the KSA-IEA for color images. The encryption process is similar to the above process, but the KS-WS is modified to fit the color images. There are three components, namely, I_R, I_G and I_B, in a color plain-image I. The encryption algorithm is as follows:

Step 1: Calculate the weighted summations S_{WI}^R, S_{WI}^G, and S_{WI}^B of the three components I_R, I_G and I_B using the weighting vectors, namely, W_1 and W_2, that are generated by (z_k, μ_k). Then, three pseudo-random values, namely, w_{M+N+1}, w_{M+N+2} and w_{M+N+3}, are multiplied by the three weighted summations in Eq. (10) to obtain the final weighted summation S_{WI}, which is used to update all initial keys to obtain the updated keys, namely, (z_1, μ_1), (z_2, μ_2) and (z_3, μ_3).

$$\begin{bmatrix} w_{M+N+1} & w_{M+N+2} & w_{M+N+3} \end{bmatrix} \times \begin{bmatrix} S_{WI}^R \\ S_{WI}^G \\ S_{WI}^B \end{bmatrix} = S_{WI}. \tag{10}$$

Step 2: Combine the three components I_R, I_G and I_B into a grayscale image $\hat{I}$ of size $\hat{M} \times \hat{N}$.

$$\begin{cases} \hat{M} = 3M, \hat{N} = N, & M \leq N \\ \hat{M} = M, \hat{N} = 3N, & M > N. \end{cases} \tag{11}$$

Step 3: Group the grayscale image $\hat{I}$ via RGM with (z_1, μ_1) and obtain the grouped plaintext PI of size $G \times E$ ($G = \min(\hat{M}, \hat{N})$, $E = \max(\hat{M}, \hat{N})$).

Step 4: Construct 16 S-boxes S_2 via SCM with (z_2, μ_2).

Step 5: Substitute the current group of plaintext using the previous group of ciphertext and the selected S-boxes via RSM and (z_3, μ_3) to obtain the ciphertext CI.

Step 6: Return all encrypted pixels to their original positions to obtain the encrypted grayscale cipher-image $\hat{I}'$.

Step 7: Convert the grayscale cipher-image $\hat{I}'$ to the final color cipher-image I'.

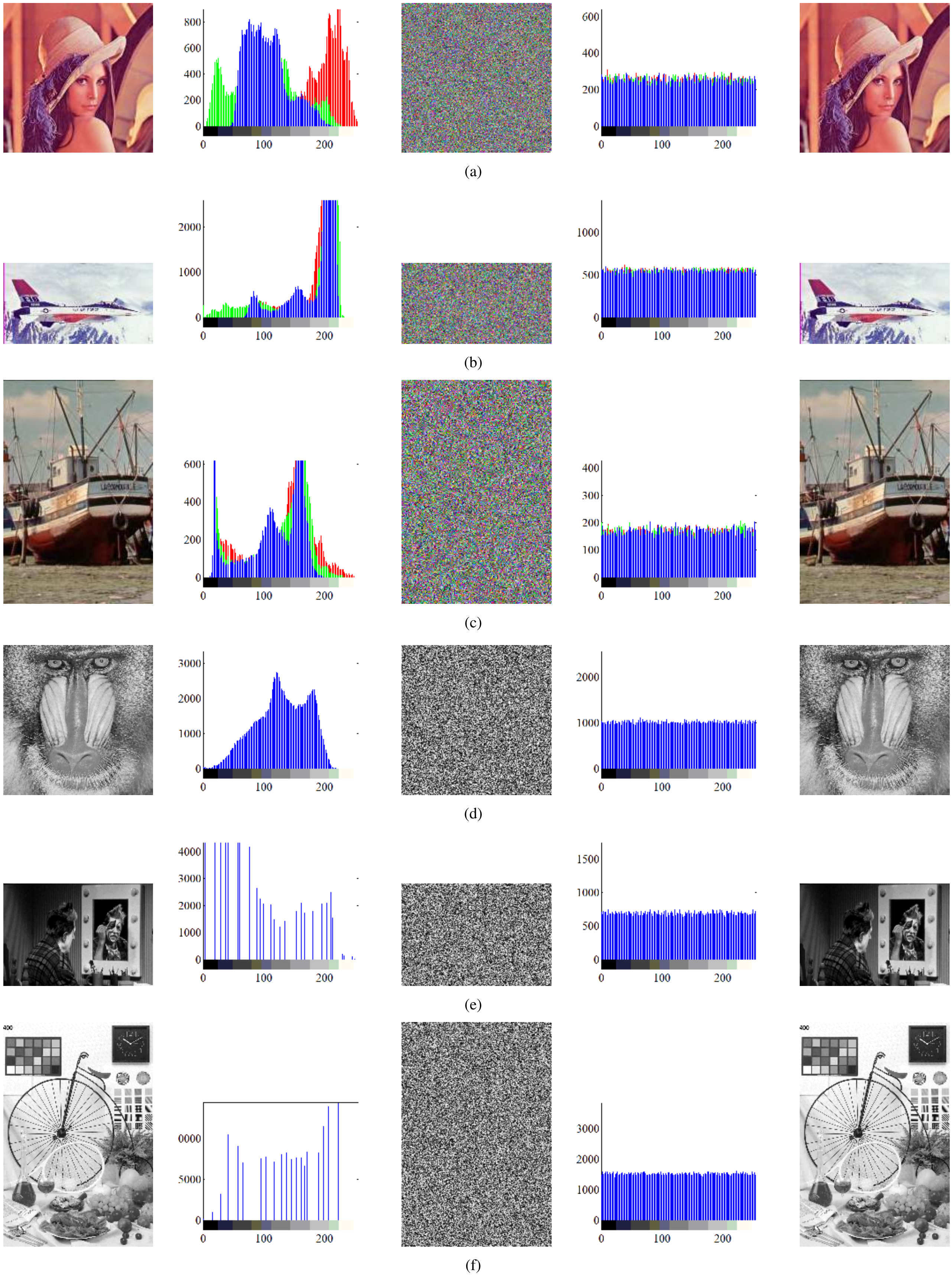


FIGURE 5. Simulation results of our KSA-IEA.

TABLE 1. Encryption time (s) of five image encryption algorithms for grayscale images.

No.	Image	CMT-IEA [17]	LSMCL-IEA [19]	CS-IEA [22]	HF-IEA [24]	KSA-IEA
1	Airplane	1.5002	3.5387	0.8103	16.3491	0.3557
2	Baboon	1.4674	2.7388	0.7683	16.2071	0.3366
3	Barbara	1.4705	2.7521	0.7768	16.1035	0.3388
4	Bridge	1.4473	2.7351	0.8032	16.1704	0.3328
5	Clown	1.4497	2.8157	0.7619	16.3449	0.3313
6	Couple	1.4428	2.7404	0.7682	16.0459	0.3616
7	Elain	1.4980	2.7333	0.8081	16.2090	0.3267
8	Frog	1.4796	2.7560	0.7712	16.2087	0.3368
9	Girlface	1.4373	2.7408	0.7996	16.0758	0.3365
10	Goldhill	1.5115	2.7641	0.7845	16.1205	0.3401
11	Houses	1.4864	2.7689	0.7867	16.1274	0.3408
12	Kiel	1.4524	2.7991	0.7713	16.1079	0.3248
13	Lena	1.5262	2.7113	0.7894	16.2561	0.3503
14	Lighthouse	1.4420	2.8552	0.7944	16.2238	0.3487
15	Tank	1.4535	2.8406	0.7797	16.1171	0.3298
16	Test_pattern	1.4528	2.7649	0.7806	16.1921	0.3221
17	Truck	1.4693	2.7555	0.7714	16.1416	0.3262
18	Trucks	1.4928	2.7714	0.7670	16.1236	0.3447
19	Zelda	1.4987	2.8144	0.8120	16.1240	0.3130
20	Average	1.4725	2.8103	0.7844	16.1710	**0.3367**

The cipher-image I' and the three updated keys (z_1, μ_1), (z_2, μ_2) and (z_3, μ_3) are transmitted to the receiver for decryption. We can decrypt the cipher-image to recover the original color image I via the reverse process of encryption.

IV. SIMULATION AND SECURITY ANALYSIS

A satisfactory image encryption algorithm should have the ability to resist various types of attacks for grayscale and color images. In this section, we present the simulation results of KSA-IEA, which executes only one encryption round, and compare the proposed KSA-IEA with the CMT-based image encryption algorithm (CMT-IEA) [17], LSMCL-based image encryption algorithm (LSMCL-IEA) [19], one-dimensional-chaotic-system-based image encryption algorithm (CS-IEA) [22] and hash-function-based image encryption algorithm (HF-IEA) [24] to analyze the security. In CMT-IEA, the whole encryption framework is carried out two encryption rounds to perform the encryption of the plain-image. In LSMCL-IEA, two rounds of confusion and two rounds of diffusion are implemented successively to obtain the cipher-image. In CS-IEA, a novel one-dimensional chaotic system is utilized in an image encryption algorithm that is based on the confusion-diffusion structure with only one encryption round. In HF-IEA, SHA-256 is combined with chaotic permutation and pixel diffusion to encrypt the plain-image, where the encryption architecture executes only one encryption round. These image encryption algorithms are implemented in MATLAB R2013a on a personal computer with an Intel(R) Core(TM) i7-3667U CPU that runs at 2.50 GHz and 8 GB of RAM. The initial conditions of the proposed KSA-IEA are $z_k = 0.12345678901234$, $z_{01} = 0.23456789012345$, $z_{02} = 0.34567890123456$ and $z_{03} = 0.45678901234567$ and the control parameters are $\mu_k = \mu_{01} = \mu_{02} = \mu_{03} = 3.99999$.

TABLE 2. Encryption time (s) of three image encryption algorithms for color images.

No.	Image	LSMCL-IEA [19]	CS-IEA [22]	KSA-IEA
1	Airplane	8.9564	2.3196	0.8775
2	Baboon	8.5550	2.3231	0.8230
3	House	8.5423	2.3514	0.8557
4	Lena	8.5244	2.3218	0.8825
5	Peppers	8.5384	2.4411	0.8894
6	Seaport	8.5742	2.3146	0.8514
7	Average	8.6151	2.3453	**0.8633**

A. SIMULATION RESULTS

Fig. 5 shows the simulation results (original images and their histograms, cipher-images and their histograms, decrypted images) that were obtained by using KSA-IEA to encrypt various types of plain-images into random-like cipher-images. The tested plain-images include the 512 × 512 Baboon grayscale image, the 350 × 512 Clown grayscale image, the 768 × 512 Bike grayscale image, the 256 × 256 Lena color image, the 276 × 512 Airplane color image, and the 256 × 171 Boat color image. According to Fig. 5, it is difficult to obtain any information of the plain-images from the cipher-images. The pixel distributions of the cipher-images are uniform, hence, the proposed KSA-IEA can resist statistical analysis attacks. The decrypted images that were obtained via lossless encryption algorithm KSA-IEA are consistent with the original images. The simulation results demonstrate that KSA-IEA has satisfactory encryption performance on various types of images.

B. SPEED ANALYSIS

The execution time is an important factor in evaluating an image encryption algorithm. A satisfactory image cipher should have high computational efficiency. We compare the proposed KSA-IEA with CMT-IEA [17], LSMCL-IEA [19],

TABLE 3. NPCR results of five image encryption algorithms for grayscale images.

No.	Image	CMT-IEA [17]	LSMCL-IEA [19]	CS-IEA [22]	HF-IEA [24]	KSA-IEA
1	Airplane	0.996029	0.992165	0.043255	0.995998	0.996056
2	Baboon	0.996067	0.992400	0.043255	0.996063	0.996010
3	Barbara	0.995953	0.992392	0.043255	0.996124	0.996014
4	Bridge	0.996109	0.992154	0.043255	0.995796	0.996262
5	Clown	0.996231	0.992123	0.043255	0.996147	0.996090
6	Couple	0.996094	0.992369	0.043255	0.996311	0.996178
7	Elain	0.996185	0.992093	0.043255	0.995998	0.995895
8	Frog	0.995884	0.992233	0.043255	0.995995	0.996181
9	Girlface	0.996216	0.992195	0.043255	0.996037	0.996117
10	Goldhill	0.996075	0.992150	0.043255	0.995941	0.995914
11	Houses	0.996128	0.992260	0.043255	0.996223	0.996006
12	Kiel	0.996223	0.992066	0.043255	0.996124	0.996010
13	Lena	0.995960	0.992449	0.043255	0.996265	0.995934
14	Lighthouse	0.996342	0.992528	0.043255	0.995846	0.996197
15	Tank	0.996098	0.992502	0.043255	0.996189	0.995956
16	Test_pattern	0.996105	0.992487	0.043255	0.996300	0.995922
17	Truck	0.996132	0.992400	0.043255	0.996269	0.996117
18	Trucks	0.996269	0.992244	0.043255	0.995995	0.996128
19	Zelda	0.996185	0.992400	0.043255	0.995975	0.995956
20	Average	0.996120	0.992295	0.043255	0.996084	**0.996050**

TABLE 4. UACI results of five image encryption algorithms for grayscale images.

No.	Image	CMT-IEA [17]	LSMCL-IEA [19]	CS-IEA [22]	HF-IEA [24]	KSA-IEA
1	Airplane	0.334806	0.334777	0.000170	0.334554	0.335277
2	Baboon	0.334662	0.333732	0.000170	0.335181	0.334449
3	Barbara	0.334633	0.334850	0.000170	0.334954	0.334622
4	Bridge	0.333675	0.334379	0.000170	0.334689	0.334819
5	Clown	0.334216	0.334900	0.000170	0.335805	0.334639
6	Couple	0.334094	0.334897	0.000170	0.334854	0.334031
7	Elain	0.333885	0.334623	0.000170	0.335375	0.334040
8	Frog	0.335095	0.334749	0.000170	0.334982	0.334327
9	Girlface	0.334971	0.334729	0.000170	0.334534	0.334267
10	Goldhill	0.334765	0.335024	0.000170	0.334965	0.334938
11	Houses	0.334839	0.334833	0.000170	0.335072	0.334341
12	Kiel	0.334916	0.334456	0.000170	0.334734	0.335355
13	Lena	0.334536	0.333999	0.000170	0.334971	0.334331
14	Lighthouse	0.334908	0.334833	0.000170	0.335169	0.334862
15	Tank	0.334910	0.334047	0.000170	0.335467	0.334772
16	Test_pattern	0.334046	0.334487	0.000170	0.334614	0.334128
17	Truck	0.334712	0.334403	0.000170	0.335655	0.334213
18	Trucks	0.334838	0.334771	0.000170	0.334792	0.333837
19	Zelda	0.334646	0.334072	0.000170	0.334806	0.334748
20	Average	0.334587	0.334556	0.000170	0.335009	**0.334526**

CS-IEA [22] and HF-IEA [24] in terms of the encryption time for grayscale and color images of size 512 × 512. These image encryption algorithms are implemented in MATLAB R2013a on a personal computer with an Intel(R) Core(TM) i7-3667U CPU that runs at 2.50 GHz and 8 GB of RAM. The evaluated results are presented in Table 1 and Table 2. Our KSA-IEA, which executes only one encryption round, is faster than the compared image encryption algorithms.

C. DIFFERENTIAL ATTACK

We use the Number of Pixels Change Rate (NPCR) and Unified Average Changing Intensity (UACI) to evaluate the ability to resist differential attack. The equations for NPCR and UACI are presented as Eq. (12).

$$\begin{cases} NPCR = \dfrac{\sum_{ij} D(i,j)}{M \times N} \times 100\% \\ UACI = \dfrac{1}{M \times N}\left[\sum_{i,j} \dfrac{|C_1(i,j) - C_2(i,j)|}{255}\right] \times 100\%, \end{cases} \tag{12}$$

where C_1 and C_2 represent two cipher-images that correspond to the same plain-image. If $C_1(i,j) \neq C_2(i,j)$, then $D(i,j) = 1$; otherwise, $D(i,j) = 0$. A satisfactory image encryption algorithm should be sensitive to slight changes in the plain-image to resist known/chosen plaintext attack. Thus, we change the least signification bit (LSB) of a pixel in a

TABLE 5. NPCR and UACI results of three image encryption algorithms for color images.

No.	Image	LSMCL-IEA [19]		CS-IEA [22]		KSA-IEA	
		NPCR	UACI	NPCR	UACI	NPCR	UACI
1	Airplane	0.992193	0.334772	0.043054	0.000671	0.996128	0.334657
2	Baboon	0.992379	0.334990	0.043054	0.000677	0.996113	0.334928
3	House	0.992175	0.334979	0.043054	0.000337	0.995907	0.334755
4	Lena	0.992541	0.334417	0.043054	0.000337	0.996101	0.334745
5	Peppers	0.992305	0.334836	0.043054	0.000678	0.996162	0.334194
6	Seaport	0.992236	0.334719	0.043054	0.001347	0.996033	0.334598
7	Average	0.992305	0.334785	0.043054	0.000674	**0.996074**	**0.334646**

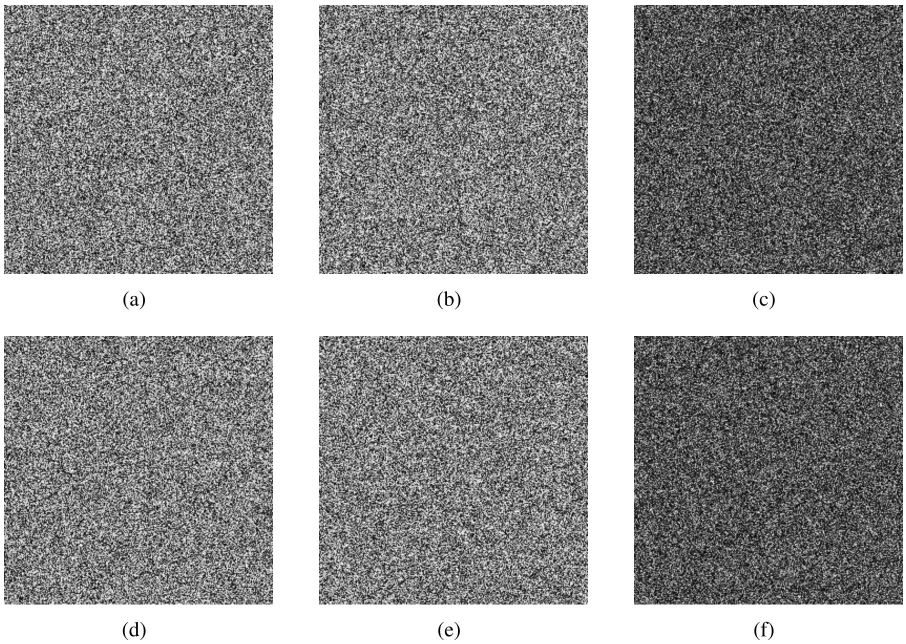


FIGURE 6. Simulation results of our KSA-IEA for special images. (a) Original cipher-image of all white image, (b) modified cipher-image of all white image, (c) difference image between (a) and (b), (d) Original cipher-image of all black image, (e) modified cipher-image of all black image, (f) difference image between (d) and (e).

grayscale image of size 512 × 512 to generate a modified plain-image and encrypt the original plain-image and the modified plain-image using the comparison schemes and our KSA-IEA with the same initial keys. Table 3 and Table 4 list the NPCR and UACI results. As can be seen, the results of the proposed KSA-IEA are close to the ideal values, so this algorithm can provide high plaintext sensitivity. Meanwhile, it has been demonstrated that our KSA-IEA has higher efficiency than the compared encryption schemes. Therefore, the proposed KSA-IEA has superior encryption performance.

We employ the above method to further evaluate the sensitivity to slight changes in color plain-images. Table 5 lists the average NPCR and UACI results of the three components in color images of size 512 × 512. It can be observed that our KSA-IEA is highly sensitive to plaintext and has stronger resistance against differential attack.

To further demonstrate the ability to resist known/chosen plaintext attack, two special images of size 512×512, namely, all black image and all white image, are encrypted by the proposed KSA-IEA. Meanwhile, the LSBs of the first pixels in these two special images are both changed to achieve two modified plain-images, and they are encrypted by our KSA-IEA. Fig. 6 shows the original cipher-images, the modified cipher-images and the difference images. It can be found that the cipher-images of the special images are random, and the original cipher-images are completely different from the modified cipher-images. Thus, the proposed KSA-IEA can resist the known/chosen plaintext attack.

D. CORRELATION ANALYSIS

The neighboring pixels have high correlations due to the high data redundancy in an image. A satisfactory image encryption algorithm should have the ability to destroy the correlations in the horizontal, vertical and diagonal directions. Mathematically, the correlation coefficient is calculated via Eq. (13).

$$\begin{cases} \rho = \dfrac{E[(x-E(x))(y-E(y))]}{D(x)D(y)} \\ E(x) = \dfrac{1}{\omega}\displaystyle\sum_{i=1}^{\omega} x_i \\ D(x) = \dfrac{1}{\omega}\displaystyle\sum_{i=1}^{\omega} [x_i - E(x)]^2, \end{cases} \tag{13}$$

TABLE 6. Correlation coefficients of five image encryption algorithms for grayscale images.

Direction	Plain-image	CMT-IEA [17]	LSMCL-IEA [19]	CS-IEA [22]	HF-IEA [24]	KSA-IEA
Horizontal	0.935766	0.002300	0.001719	0.001746	0.001774	0.001512
Vertical	0.930599	0.002010	0.001358	0.001682	0.001305	0.002145
Diagonal	0.892260	0.001884	0.002162	0.001671	0.001883	0.001520

TABLE 7. Correlation coefficients of three image encryption algorithms for color images.

Component	Direction	Plain-image	LSMCL-IEA [19]	CS-IEA [22]	KSA-IEA
	Horizontal	0.955760	0.0010515	0.000685	0.001286
R	Vertical	0.942115	0.0016172	0.001440	0.001969
	Diagonal	0.919063	0.0011097	0.002112	0.001763
	Horizontal	0.938633	0.0015046	0.001523	0.001697
G	Vertical	0.924148	0.0011315	0.001718	0.001474
	Diagonal	0.890438	0.0014381	0.001484	0.001632
	Horizontal	0.947580	0.0019459	0.001018	0.001321
B	Vertical	0.934551	0.0023904	0.002342	0.001799
	Diagonal	0.903944	0.0017336	0.001541	0.002021
	Horizontal	0.947325	0.0015007	0.001075	0.001435
Average	Vertical	0.933605	0.0017130	0.001833	0.001747
	Diagonal	0.904482	0.0014271	0.001712	0.001805

TABLE 8. Information entropy results of five image encryption algorithms for grayscale images.

No.	Image	CMT-IEA [17]	LSMCL-IEA [19]	CS-IEA [22]	HF-IEA [24]	KSA-IEA
1	Airplane	7.999397	7.999285	7.999340	7.999348	7.999314
2	Baboon	7.999222	7.999235	7.999209	7.999302	7.999320
3	Barbara	7.999305	7.999313	7.999471	7.999254	7.999320
4	Bridge	7.999374	7.999254	7.999252	7.999326	7.999325
5	Clown	7.999229	7.999275	7.999297	7.999318	7.999301
6	Couple	7.999203	7.999383	7.999427	7.999321	7.999338
7	Elain	7.999287	7.999321	7.999298	7.999244	7.999334
8	Frog	7.999169	7.999312	7.999349	7.999342	7.999361
9	Girlface	7.999292	7.999297	7.999197	7.999360	7.999239
10	Goldhill	7.999342	7.999356	7.999270	7.999356	7.999266
11	Houses	7.999412	7.999380	7.999349	7.999302	7.999337
12	Kiel	7.999312	7.999309	7.999208	7.999281	7.999241
13	Lena	7.999346	7.999253	7.999352	7.999294	7.999335
14	Lighthouse	7.999174	7.999447	7.999315	7.999285	7.999375
15	Tank	7.999380	7.999361	7.999314	7.999472	7.999421
16	Test_pattern	7.999379	7.999355	7.999334	7.999304	7.999282
17	Truck	7.999322	7.999301	7.999348	7.999384	7.999256
18	Trucks	7.999377	7.999339	7.999337	7.999273	7.999379
19	Zelda	7.999300	7.999283	7.999377	7.999402	7.999439
20	Average	7.999307	7.999319	7.999318	7.999325	**7.999326**

where x and y represent two adjacent pixel sequences in three directions, $E(x)$ is the expectation of x and $D(x)$ is the variance of x. The larger the correlation coefficient $\rho \in [0, 1]$ is, the higher the correlation between pixels is.

The correlation coefficients along with the horizontal, vertical and diagonal directions are calculated under CMT-IEA [17], LSMCL-IEA [19], CS-IEA [22], HF-IEA [24] and our KSA-IEA. Table 6 and Table 7 list the average correlation coefficients of 19 grayscale images and 6 color images. The coefficients of the cipher-images for the proposed KSA-IEA are close to 0, hence, our KSA-IEA can reduce the correlations of neighboring pixels.

E. INFORMATION ENTROPY

The cipher-image should have high randomness to guarantee high security. The information entropy is a criterion for evaluating the randomness of a cipher-image. The equation is presented as Eq. (14).

$$H(x) = \sum_{i=0}^{MN-1} p(x_i) log \frac{1}{p(x_i)}, \tag{14}$$

where x_i is the ith pixel value in the cipher-image of size $M \times N$ and $p(x_i)$ is the probability of x_i. Table 8 and Table 9 list the information entropy results of four comparable encryption algorithms and our KSA-IEA for grayscale and color images. The information entropy values of the proposed KSA-IEA are

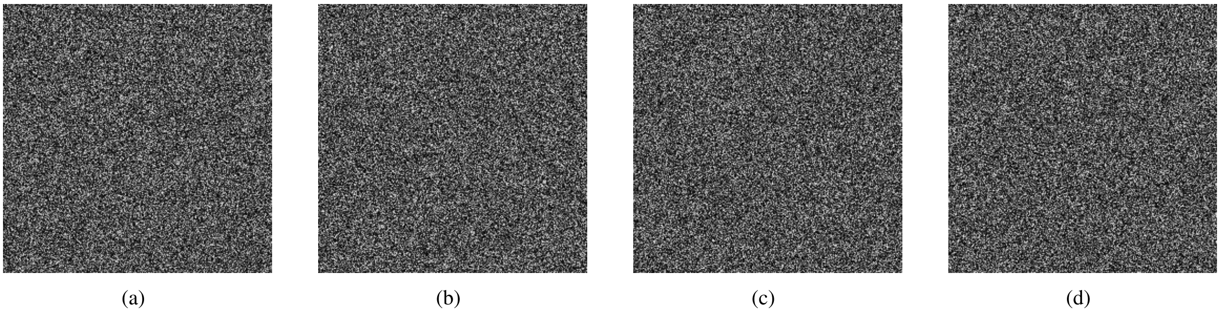


FIGURE 7. Key sensitivity analysis of our KSA-IEA in the encryption process for a grayscale image. (a) z_k, (b) z_{01}, (c) z_{02}, (d) z_{03}.

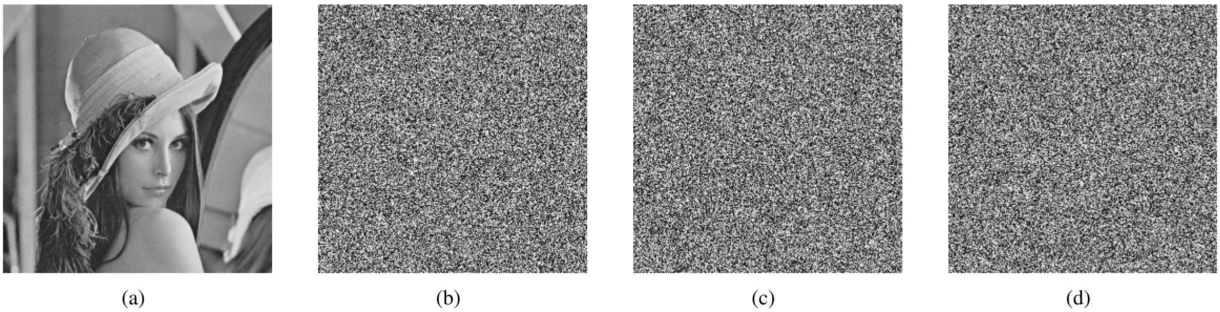


FIGURE 8. Key sensitivity analysis of our KSA-IEA in the decryption process for a grayscale image. (a) Original decrypted image, (b) z_1, (c) z_2, (d) z_3.

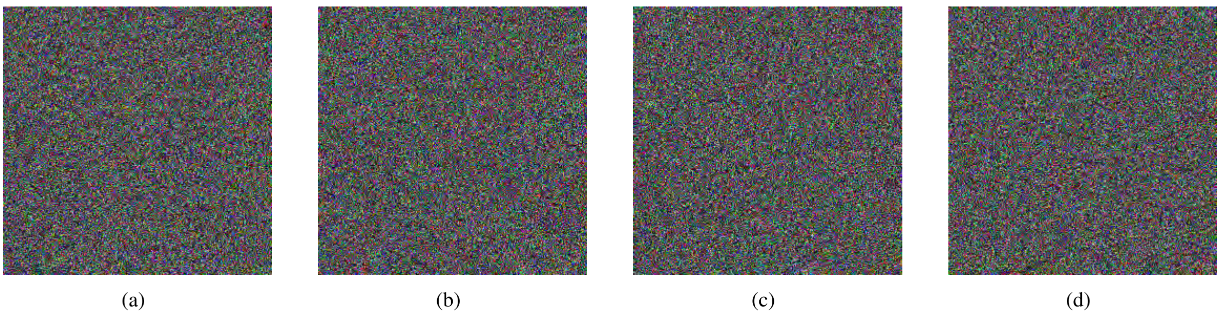


FIGURE 9. Key sensitivity analysis of our KSA-IEA in the encryption process for a color image. (a) z_k, (b) z_{01}, (c) z_{02}, (d) z_{03}.

TABLE 9. Information entropy results of three image encryption algorithms for color images.

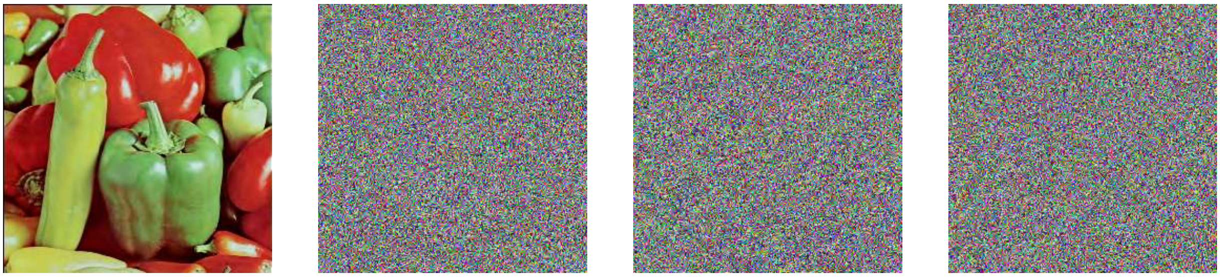
No.	Image	LSMCL-IEA [19]	CS-IEA [22]	KSA-IEA
1	Airplane	7.999269	7.999324	7.999320
2	Baboon	7.999257	7.999372	7.999352
3	House	7.999316	7.999298	7.999292
4	Lena	7.999309	7.999333	7.999349
5	Peppers	7.999332	7.999351	7.999342
6	Seaport	7.999268	7.999263	7.999293
7	Average	7.999292	7.999324	**7.999325**

closer to the ideal value 8. These simulation results demonstrate that the cipher-images of our KSA-IEA are random and cannot leak any useful information to attackers.

F. KEY SPACE AND KEY SENSITIVITY

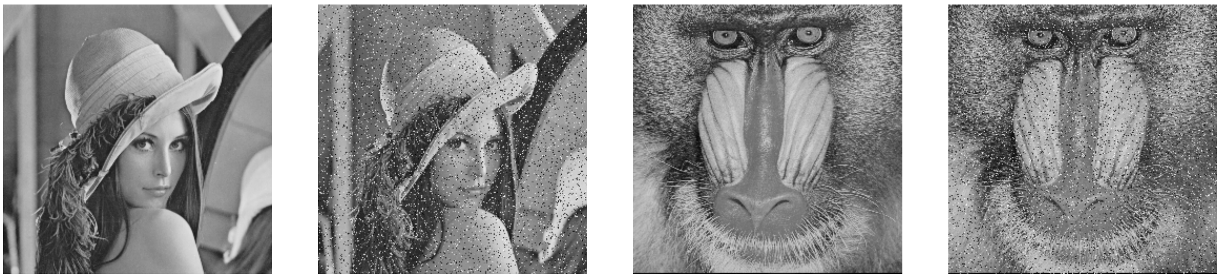
A satisfactory image encryption algorithm should have a sufficiently large key space for resisting brute-force attack. In the proposed KSA-IEA, we encrypt the grayscale/color images with four initial keys. The keys are all of double data type, which stores each real number in 64 bits. According to the IEEE 754-2008 standard and the symmetry of Logistic system, our key space size is at least $2^{124} > 2^{100}$. It is demonstrated that our KSA-IEA performs well in resisting brute-force attack for various images.

The image cryptosystem should be extremely sensitive to slight changes in the initial keys. We evaluate the key sensitivity of the proposed KSA-IEA in the encryption process and the decryption process. First, we use four initial keys with a tiny difference (10^{-14}) to encrypt a plain-image. Fig. 7 shows the difference images. The cipher-images with modified keys are differ completely from the original cipher-images. Then, we use three decryption keys with a tiny difference (10^{-14}) to recover a cipher-image. The original decrypted image and the recovered decrypted images are presented in Fig. 8. The



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

FIGURE 10. **Key sensitivity analysis of our KSA-IEA in the decryption process for a color image. (a) Original decrypted image, (b) z_1, (c) z_2, (d) z_3.**



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

FIGURE 11. **Noise attack analysis of our KSA-IEA. (a) Original plain-image (Lena), (b) decrypted image (Lena), (c) original plain-image (Baboon), (d) decrypted image (Baboon).**

recovered cipher-images are all incorrect. The same experiment is conducted on a color image and the simulation results are exhibited in Fig. 9 and Fig. 10. We came to the same conclusion. Therefore, the proposed KSA-IEA is sensitive to its initial keys in both the encryption and decryption processes for grayscale and color images.

G. NOISE ATTACK

An image encryption algorithm should have the robustness to resist the noise attack to achieve the information of plain-image. In this simulation, we add the Gaussian noises to the cipher-image to receive the plain-image, where the mean of the Gaussian distribution is 0 and the variance is 0.000001. The simulation results are shown in Fig. 11. It can be found that the main information in the original plain-image can be observed from the decrypted image. Therefore, the proposed KSA-IEA is robust and has the resistance against noise attack.

V. CONCLUSION

In this paper, we propose a novel encryption architecture, namely, KSA, that is based on key scheming and substitution for overcoming the low security and low computational efficiency of the traditional confusion-diffusion framework. Under this KSA, a new image encryption algorithm KSA-IEA is presented, which uses only one round of encryption. In this KSA-IEA, a key scheming that is based on weighted summation is designed for enhancing the sensitivity to slight changes in the plain-image. In addition, we also develop a new substitution method that is based on S-boxes for encrypting various types of images. The proposed KSA-IEA is evaluated and is compared with several image encryption schemes under the traditional encryption framework. The simulation results demonstrate that our KSA-IEA has higher security and efficiency.

REFERENCES

[1] 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.

[2] Y. Wu, Y.-C. Zhou, J. P. Noonan, and S. Agaian, "Design of image cipher using latin squares," *Inf. Sci.*, vol. 264, pp. 317–339, Apr. 2014.

[3] H. T. Panduranga, S. K. N. Kumar, and Kiran, "Image encryption based on permutation-substitution using chaotic map and latin square image cipher," *Eur. Phys. J. Special Topics*, vol. 223, pp. 1663–1677, 2014.

[4] Y. Wu, Y. Zhou, S. Agaian, and J. P. Noonan, "2D Sudoku associated bijections for image scrambling," *Inf. Sci.*, vol. 327, pp. 91–109, Jan. 2016.

[5] Y. Zhang, D. Xiao, W. Wen, and K.-W. Wong, "On the security of symmetric ciphers based on DNA coding," *Inf. Sci.*, vol. 289, pp. 254–261, Dec. 2014.

[6] Y. Zhang, Y. Li, W. Wen, Y. Wu, and J.-X. Chen, "Deciphering an image cipher based on 3-cell chaotic map and biological operations," *Nonlinear Dyn.*, vol. 82, pp. 1831–1837, 2015.

[7] A. Belazi, M. Talha, S. Kharbech, and W. Xiang, "Novel medical image encryption scheme based on chaos and DNA encoding," *IEEE Access*, vol. 7, pp. 36667–36681, 2019.

[8] Y. Luo, X. Ouyang, J.-X. Liu, and L.-C. Cao, "An image encryption method based on elliptic curve ElGamal encryption and chaotic systems," *IEEE Access*, vol. 7, pp. 38507–38522, 2019.

[9] X. Liu, D. Xiao, and Y. Xiang, "Quantum image encryption using intra and inter bit permutation based on Logistic map," *IEEE Access*, vol. 7, pp. 6937–6946, 2019.

[10] N. Zhou, X. Yan, H. Liang, X. Tao, and G. Li, "Multi-image encryption scheme based on quantum 3D Arnold transform and scaled Zhongtang chaotic system," *Quantum Inf. Process.*, vol. 17, no. 12, pp. 338–373, 2018.

[11] N. Zhou, W. Chen, X. Yan, and Y. Wang, "Bit-level quantum color image encryption scheme with quantum cross-exchange operation and hyperchaotic system," *Quantum Inf. Process.*, vol. 17, no. 6, pp. 137–160, 2018.
[12] T. Hua, J. Chen, D. Pei, W. Zhang, and N. Zhou, "Quantum image encryption algorithm based on image correlation decomposition," *Int. J. Theor. Phys.*, vol. 54, pp. 526–537, Feb. 2015.
[13] J. Wang, Y.-C. Geng, L. Han, and J.-Q. Liu, "Quantum image encryption algorithm based on quantum key image," *Int. J. Theor. Phys.*, vol. 58, pp. 308–322, Jan. 2019.
[14] G. Ye, H. Zhao, and H. Chai, "Chaotic image encryption algorithm using wave-line permutation and block diffusion," *Nonlinear Dyn.*, vol. 83, pp. 2067–2077, 2016.
[15] X. Huang and G. Ye, "An image encryption algorithm based on irregular wave representation," *Multimed. Tools Appl.*, vol. 77, pp. 2611–2628, Jan. 2018.
[16] G. Ye, K. Jiao, C. Pan, and X. Huang, "An effective framework for chaotic image encryption based on 3D Logistic map," *Secur. Commun. Netw.*, vol. 2018, Oct. 2018, Art. no. 8402578.
[17] Z. Hua, Y. Zhou, C.-M. Pun, and C. L. P. Chen, "2D sine Logistic modulation map for image encryption," *Inf. Sci.*, vol. 297, pp. 80–94, Mar. 2015.
[18] Z. Hua and Y. Zhou, "Image encryption using 2D Logistic-adjusted-Sine map," *Inf. Sci.*, vol. 339, pp. 237–253, Apr. 2016.
[19] H. Zhu, Y. Zhao, and Y. Song, "2D Logistic-modulated-sine-coupling-Logistic chaotic map for image encryption," *IEEE Access*, vol. 7, pp. 14081–14098, 2019.
[20] P. Ping, J. Fan, Y. Mao, F. Xu, and J. Gao, "A chaos based image encryption scheme using digit-level permutation and block diffusion," *IEEE Access*, vol. 6, pp. 67581–67593, 2018.
[21] H. Diab, "An efficient chaotic image cryptosystem based on simultaneous permutation and diffusion operations," *IEEE Access*, vol. 6, pp. 42227–42244, 2018.
[22] C. Pak and L. L. Huang, "A new color image encryption using combination of the 1D chaotic map," *Signal Process.*, vol. 138, pp. 129–137, Sep. 2017.
[23] G. Ye and X. Huang, "An efficient symmetric image encryption algorithm based on an intertwining Logistic map," *Neurocomputing*, vol. 251, pp. 45–53, Aug. 2017.
[24] S. Amina and F. K. Mohamed, "An efficient and secure chaotic cipher algorithm for image content preservation," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 60, pp. 12–32, Jul. 2018.
[25] C. Cao, K. Sun, and W. Liu, "A novel bit-level image encryption algorithm based on 2D-LICM hyperchaotic map," *Signal Process.*, vol. 143, pp. 122–133, Feb. 2018.
[26] X.-J. Tong, M. Zhang, Z. Wang, Y. Liu, H. Xu, and J. Ma, "A fast encryption algorithm of color image based on four-dimensional chaotic system," *J. Vis. Commun. Image Represent.*, vol. 33, pp. 219–234, Nov. 2015.
[27] J. Fridrich, "Symmetric ciphers based on two-dimensional chaotic maps," *Int. J. Bifurcation Chaos*, vol. 8, no. 6, pp. 1259–1284, 1998.
[28] X. Huang and G. Ye, "An image encryption algorithm based on hyperchaos and DNA sequence," *Multimedia Tools Appl.*, vol. 72, no. 1, pp. 57–70, Sep. 2014.
[29] X. Wu, D. Wang, J. Kurths, and H. Kan, "A novel lossless color image encryption scheme using 2D DWT and 6D hyperchaotic system," *Inf. Sci.*, vols. 349–350, pp. 137–153, Jul. 2016.
[30] W. Zhang, H. Yu, Y.-I. Zhao, and Z.-L. Zhu, "Image encryption based on three-dimensional bit matrix permutation," *Signal Process.*, vol. 118, pp. 36–50, Jan. 2016.

ZHILIANG ZHU received the M.S. degree in computer applications and the Ph.D. degree in computer science from Northeastern University, China. His main research interests include information integration, complexity software systems, network coding and communication security, chaos-based digital communications, applications of complex network theories, and cryptography. He has authored and coauthored over 130 international journal papers and 100 conference papers. In addition, he has published five books, including *Introduction to Communication and Program Designing of Visual Basic .NET*. He is a Fellow of the China Institute of Communications. He is a Senior Member of the Chinese Institute of Electronics and the Teaching Guiding Committee for Software Engineering under the Ministry of Education. He is also a recipient of nine academic awards at the national, ministerial, and provincial levels. He has served in different capacities at many international journals and conferences. He currently serves as the Co-Chair for the 1st-10th International Workshop on Chaos-Fractals Theories and Applications.

WEI ZHANG received the Ph.D. degree in computer science and technology from Northeastern University, China, in 2013, where he is currently an Associate Professor with the Software College. His research interests include signal processing and multimedia security.

HAI YU received the B.E. degree in electronic engineering from Jilin University, China, in 1993, and the Ph.D. degree in computer software and theory from Northeastern University, China, in 2006. He is currently an Associate Professor of software engineering with Northeastern University, China. His research interests include complex networks, chaotic encryption, software testing, software refactoring, and software architecture. Moreover, he has served different roles at several international conferences, such as the Associate Chair for the 7th IWCFTA, in 2014, the Program Committee Chair for the 4th IWCFTA, in 2010, the Chair for the Best Paper Award Committee at the 9th International Conference for Young Computer Scientists, in 2008, and a Program Committee Member for the 3rd–10th IWCFTA and the 5th Asia Pacific Workshop on Chaos Control and Synchronization. He currently serves as an Associate Editor for the *International Journal of Bifurcation and Chaos*, a Guest Editor for *Entropy*, and a Guest Editor for the *Journal of Applied Analysis and Computation*. In addition, he was a Lead Guest Editor for the *Mathematical Problems in Engineering*, in 2013.

YANJIE SONG is currently pursuing the Ph.D. degree with Software College, Northeastern University, China. Her main research interests include multimedia encoding and encryption, chaotic encryption, and compressive sensing.

YULI ZHAO received the Ph.D. degree in communication and information systems from Northeastern University, China, in 2013, where she is currently a Lecturer. Her research interest includes applications of complex-network theories to communications.