Multimedia Tools and Applications (2024) 83:12985–13028
https://doi.org/10.1007/s11042-023-15880-2

A novel structure of fast and efficient multiple image encryption

Thang Manh Hoang[1]

Received: 18 January 2022 / Revised: 23 March 2023 / Accepted: 22 May 2023 /
Published online: 2 July 2023

Abstract

A huge volume of image data is created every day, and it requires a fast and efficient encryption to keep them confidential. A chaos-based encryption is considered as the most suitable one for image encryption, and multiple image encryption is one of approaches to achieve the fast and efficient performance. However, the existing methods of multiple image encryption is with a lack of diffusion effect, inefficiency in using random number generated by chaotic map, and low speed. In this paper, a novel structure of chaos-based encryption is proposed to encrypt multiple images at the same time, in which the permutation and diffusion are integrated and they share the same chaotic map. The exclusive-OR operation is chosen for calculation and data manipulation during encryption. Therefore, the proposed structure allows to improve the efficiency and to reduce the time consumption for the encryption. In addition, the chaotic map is perturbed frequently and its dynamics is dependent on the content of images. It creates the dynamical session key, so the proposed structure can resist from the types of chosen-plaintext and chosen-ciphertext attacks. Two exemplar ciphers employing the proposed structure are demonstrated with the use of Logistic and Standard maps. The simulation results will be analysed and compared with those of existing methods to show the feasibility and effectiveness of the proposed structure of multiple image encryption.

Keywords Digital perturbed chaos · Perturbed chaotic map (PCM) · Chaos-based image encryption · Multiple image encryption (MIE)

1 Introduction

Since chaos was discovered by E. Lorenz[35], it has been explored and developed in many fields of science and engineering [44]. One of prominent applications of chaos in information engineering is the chaos-based cryptography [6, 16, 28, 31]. In fact, a chaos-based cryptography utilizes the complexity of dynamics [27] rather than that of number theory and algebra

✉ Thang Manh Hoang
thang.hoangmanh@hust.edu.vn

[1] School of Electrical and Electronic Engineering, and Vietnam-Japan International Institute for Science of Technology, Hanoi University of Science and Technology, 1 Dai Co Viet,
Hai Ba Trung district, Hanoi, Vietnam

as in the conventional cryptography. So far, many chaos-based image cryptosystems were proposed with various ranges of aspects, those employs from simple chaotic maps such as the Logistic map [30] to highly complicated ones such as fractional-order hyperchaotic system [22], from simple algorithms [6] to complicated ones e.g., quantum method [62, 63], etc.

Nowadays the image data is massively created by the technological advances in image acquisition. Among them, there are a high volume of image data such as medical images needed to keep confidential, so it requires to have fast and efficient algorithms. For decades, chaos-based image encryption has been researched because it provides the advantages with simple implementation and high performance for the bulk data like images and multimedia data. On aspects of performance, there are three main approaches to have a fast and efficient chaos-based image encryption, i.e., suitable selection of plaintext for encryption (e.g. selective encryption), computational optimization of encryption algorithms, and structural optimization of encryption. Firstly, the selective encryption is to consider to encrypt only part of image data what significantly contributes to the visual structure [4, 7, 9, 26, 29, 43, 53, 58]. However, the context of high security requires to encrypt all image data. Secondly, the computational optimization for the encryption algorithm is to choose the simple operators with low computation cost, switching mechanisms (e.g. DNA sequence [13, 29, 48, 55], look-up tables[10, 24]), or dealing with blocks of pixels [8, 15, 18, 54, 59]. Among them, the simplest way to enhance the speed of encryption algorithms is to choose the operators with low computational cost such exclusive-OR (XOR) and modulus in the digital platform, etc., for both the chaotic map and the ciphertext computation. However, most of existing encryption algorithms employ complicated operators such as multiplier, addition, cyclic shift, or even exponent. Thirdly, the structural optimization for an encryption algorithm is to arrange the entities in a cryptosystem in order to have high speed encryption. For example, the combination of permutation and diffusion (CPD) in the configuration of cryptosystem allows to reduce the number of access times to the data in the memory during encryption [10, 12, 14, 50]. However, all of existing encryption algorithms with the CPD were designed to encrypt a single image, so the efficiency of encryption is limited.

In addition, a chaotic map works as a pseudo-random number generator for chaos-based cryptosystems. Therefore, a fast and efficient encryption algorithm is achieved if pseudo-random numbers are used efficiently. In other words, for a certain number of bits generated by a chaotic map, the more pixels are encrypted, the more efficient cipher is. Multiple image encryption (MIE) is one of approaches to improve the efficiency for the chaos-based image encryption. Recently, there are several works on the MIE using chaos, e.g. [13, 23, 25, 32, 36, 37, 39, 40, 42, 45, 46, 48, 49, 51, 57–61, 63]. However, there are flaws in such the MIEs, i.e., a lack of diffusion effect in the encryption in [36, 40, 46, 48, 57–60, 60, 61], inefficiency in using bits generated by the chaotic map with the permutation separated from the diffusion [37–40, 46, 48, 56–58, 61, 63], choosing complicated computational operations [40, 46, 48, 56, 57, 60, 61, 63]. In addition, all of existing algorithms of MIE are designed for single round of encryption, so it can be broken more easily than that with multiple rounds of encryption [3, 20].

On aspects of security, Gonzalo Alvarez et. al. [1, 2] suggested that the dynamics of a chaotic map must be complicated enough to assure the security. One of methods to achieve complicated dynamics of a chaotic map is to make the chaotic map perturbed. It is proved that a perturbed chaotic map (PCM) has dynamics more complicated than that of original one [33, 47]. However, all of existing methods employ algebraic operations for the perturbation, so it takes large time consumption during iteration of PCMs. Besides, most of existing algorithms of MIE use the static session key, in which the session key is constant during encryption.

Although the initial values of chaotic systems are calculated from the image content in terms of hash values as in [37–40, 56–58, 61], the session key is not changed during encryption, or it is static. In contrast, the dynamical session key is changed during encryption. In fact, a chaos-based cryptosystem with the dynamical session key can resist from the types of chosen-plaintext and chosen-ciphertext attacks. One of methods to have the dynamical session key is that the dynamics of chaotic map is involved by the image content during encryption as in [12, 17, 19, 21, 34, 41].

Overall, the existing algorithms of MIE are with a lack of diffusion effect, inefficiency in using random number generated by chaotic map, and low speed. In this paper, a novel structure of chaos-based encryption is proposed to encrypt multiple images at the same time, in which the permutation and diffusion are integrated and they share the same chaotic map. The XOR operation is chosen for calculation and data manipulation during encryption. Therefore, the proposed structure allows to improve the efficiency and to reduce the time consumption for the encryption. In addition, the chaotic map is perturbed frequently and its dynamics is dependent on the content of images. It creates the dynamical session key, so the proposed structure can resist from the types of chosen-plaintext and known-plaintext attacks. Two exemplar ciphers employing the proposed structure are demonstrated with the use of Logistic and Standard maps. The simulation results will be analysed and compared with those of existing methods to show the feasibility and effectiveness of the proposed structure of MIE.

This paper contributes the followings.

1. To gain high speed and efficiency: the permutation and diffusion is integrated; only one chaotic map is required; and only the XOR operation is used in both the perturbation of chaotic map and the encryption equations of pixels.
2. To achieve high security: the dynamical session key is generated by a chaotic map with non-stationary dynamics by means of perturbation; and the image content is involved in the generation of session key by means of the chaotic dynamics during encryption.

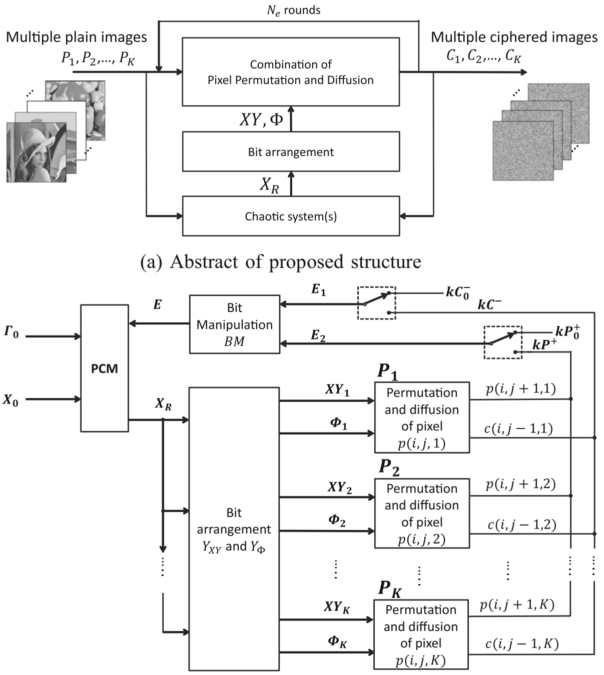
The rest of paper is organized as follows. The configuration and operation of the proposed structure are presented in Section 2. The specific example illustrates in Section 3 with details about the simulation result, the statistical and security analyses, and the comparison with the results of existing methods. Section 4 presents the discussion and conclusion of the work.

2 Proposed structure of MIE

The configuration of the proposed structure of chaos-based MIE is illustrated in Fig. 1. Figure 1(a) displays the model to integrate the permutation and diffusion, in which only one PCM is used. Figure 1(b) shows the detail of the structure that the PCM provides chaotic values for the encryption and perturbed by pixel values. Pixels are shuffled and diffused one by one in every image.

2.1 Perturbed chaotic map

Here, the perturbed chaotic map [21] is employed for the proposed structure. Figure 2 illustrates the configuration of PCM, in which Δ_{X_n} and Δ_{Γ_n} are perturbation amounts applying to state variables and control parameters, respectively. There, D and G are the number of dimensions and that of control parameters, respectively. k_1 is the length of the input bit



(a) Abstract of proposed structure

(b) Detail of proposed structure

Fig. 1 Proposed structure and its configuration

sequence E, and R is the number of chaotic iterations at which values of state variables are read for the encryption. Equations for a generic PCM are expressed as

$$\begin{cases} X_{n+1} &= F(\hat{X}_n, \hat{\Gamma}_n), \\ \hat{X}_n &= X_n \oplus \Delta_{X_n}, \\ \hat{\Gamma}_n &= \Gamma_0 \oplus \Delta_{\Gamma_n}, \end{cases} \tag{1}$$

where $F(.)$ is the chaotic function; X_n, $\hat{X}_n$, and $\hat{\Gamma}_n$ are the vectors of state variables, perturbed state variables, and perturbed control parameters, respectively; X_0 and Γ_0 are initial values

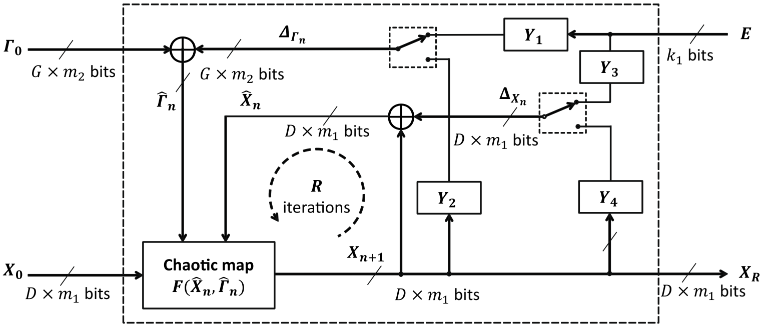


Fig. 2 The structure of PCM

of state variable and control parameter, respectively. Assumed that the PCM is implemented on the digital platform, so values of X_n and Γ_n are represented in the format of fixed-point number of m_1 and m_2 bits, respectively. The perturbation amounts are constructed by

$$\Delta_{\Gamma_n} = \begin{cases} Y_1 \circ E & \text{for } n = 0; \\ Y_2 \circ X_n & \text{for } 1 \leq n \leq R, \end{cases} \tag{2}$$

and

$$\Delta_{X_n} = \begin{cases} Y_3 \circ E & \text{for } n = 0; \\ Y_4 \circ X_n & \text{for } 1 \leq n \leq R, \end{cases} \tag{3}$$

where n is current number of chaotic iterations; E is the external source represented by k_1 bits and it can be constructed by bits of either initial values (kC_0^- and kP_0^+) or current values of pixels ($p(i, j+1, k)$ and $c(i, j-1, k)$, $k = 1..K$); the rules of bit arrangement Y_i, $i = 1..4$, are to construct the perturbation amounts; and $\circ$ is the operator of bit arrangement.

In the operation, the PCM is firstly initialized by Γ_0 and X_0, then it iterates R times to produce chaotic values X_R for the encryption. It requires that values of $\hat{\Gamma}_n$ and $\hat{X}_n$ are always in the ranges such that the PCM exhibits chaotic behavior.

2.2 Bit arrangement

The bit arrangement rule Y is to construct new bit sequences from a given bit sequence. Assumed that A and B are bit matrices with the sizes $I_A \times J_A$ and $I_B \times J_B$, i.e., $A = [a_{ij}]_{1 \leq i \leq I_A,\ 1 \leq j \leq J_A}$ and $B = [b_{ij}]_{1 \leq i \leq I_B,\ 1 \leq j \leq J_B}$, and with $a_{ij}, b_{ij} \in \{0, 1\}$. Bit matrix A is constructed from bits of matrix B by using the bit arrangement rule Y as

$$A = Y \circ B. \tag{4}$$

Here, the rule of bit arrangement is represented by an array of 2-tuples $Y = [(y_{ij}^{(r)}, y_{ij}^{(c)})]_{1 \leq i \leq I_A,\ 1 \leq j \leq J_A}$, in which $y_{ij}^{(r)}$ and $y_{ij}^{(c)}$ are row and column indexes of matrix B with $y_{ij}^{(r)} \in [1, I_B]$ and $y_{ij}^{(c)} \in [1, J_B]$. In fact, bits of A come from those of B as $a_{ij} = b_{y_{ij}^{(r)} y_{ij}^{(c)}}$. As special cases, bits in A can be deliberately fixed at the logic ‘0’ or ‘1’, and in those cases the 2-tuple $(y_{ij}^{(r)}, y_{ij}^{(c)})$ is denoted by B_0 and B_1 for the logic ‘0’ and ‘1’, respectively.

In this work, the bit matrix B is a representation of chaotic values of X_n. Therefore, the bit matrix A can be used for constructing I_A bit sequences. Each bit sequence is a concatenation of bits in the same row of matrix A, i.e. $A_i = ||_{j=1}^{J_A} a_{i,j}$, where $||$ denotes for the bit concatenation.

For example, the chaotic value $X_R = (3.4162, 1.1963)$ is represented in the format of fixed-point number with 18 bits as a whole and 16 bits for the fractional part (denoted as $\langle 18, 16 \rangle$). The value of X_R in binary is

$$X_{bin} = \begin{pmatrix} 11.0110101010001010 \\ 01.0011001001000010 \end{pmatrix}. \tag{5}$$

Figure 3 illustrates the operation of bit arrangement in Eq. (4). The bit matrix B is a matrix representation of X_{bin}. As a result, the bit matrix A is obtained, and the bit sequences $A_1 = 1011$ and $A_2 = 1010$ are concatenation of bits in rows of A as described above.

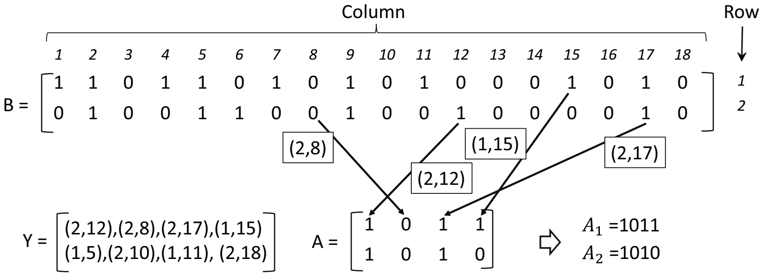


Fig. 3 Example of bit arrangement

2.3 Bit manipulation

In this work, the block Bit manipulation (BM) in Fig. 1b performs combining two bit sequences E_1 and E_2, so as to have a bit sequence E. Let us denote $|.|$ be the function returning the length of bit sequence, and $\Updownarrow$ be the bit-interleaving concatenation. In fact, the bit-interleaving concatenation can be any rule of bit interleaving of two bit sequences. There are three cases of relative length of E, E_1 and E_2. If $|E_1| + |E_2| = |E|$, the BM is simply $E = E_1 \Updownarrow E_2$. For two other cases of relative lengths, i.e., $|E_1| + |E_2| > |E|$ and $|E_1| + |E_2| < |E|$, in this work, the bit sequence E can be obtained by two simple procedures as followings. Respectively, T and T_i are denoted for the resultant bit sequence of interleaving and portion of T after division, respectively.

- Case 1: For $|E_1| + |E_2| > |E|$
  Step 1: Concatenate two bit sequences $T = E_1 \Updownarrow E_2$.
  Step 2: Find integer n such that $(n-1) * |E| < |T| \leq n * |E|$.
  Step 3: Separate sequence T into n portions T_i, $i = \{1...n\}$, such that $|T_i| = |E|$ for $i = 1..n-1$ and the last portion T_n is with the length $|T_n| \leq |E|$.
  Step 4: Pad $|E| - |T_n|$ bit zeros into sequence T_n.
  Step 5: Perform XORing the bits at the same position in the sequences as $E = \oplus_{i=1}^{n} T_i$.
- Case 2: For $|E_1| + |E_2| \leq |E|$
  Step 1: Find integer m such that $(m-1) * (|E_1| + |E_2|) < |E| \leq m * (|E_1| + |E_2|)$.
  Step 2: Concatenate m times $E_1 \Updownarrow E_2$ to get bit sequence T.
  Now, the relative length is $|T| \geq |E|$. At this point, Steps 2 to 5 in Case 1 are performed on bit sequence T to obtain bit sequence E.

2.4 Permutation and diffusion for a pair of pixels

The permutation and diffusion are integrated in order to thoroughly exploit bits generated by the PCM. Assumed that the MIE performs encrypting K images at the same time and all images are with the same size. Let us denote 3-tuples (i, j, k) and (i', j', k') be the coordinates of source and destination pixels, respectively, with $i, i' \in [1, M]$, $j, j' \in [1, N]$ and $k, k' \in [1, K]$. Specifically, $p(i, j, k)$ and $p(i', j', k')$ are the source plain pixel and destination pixel of images P_k and $P_{k'}$, respectively. Correspondingly, $c(i, j, k)$ and $c(i', j', k')$ are ciphered pixels of $p(i, j, k)$ and $p(i', j', k')$, respectively. Also, $\Phi_k = [\phi_{2,k}, \phi_{1,k}]$ $(k = 1...K)$ is denoted for the vector of values for the diffusion, where its members are constructed from bits of chaotic values. The permutation and diffusion for the encryption are carried out for every pair of pixels in K images by four steps as followings.

Step 1: For source pixels $p(i, j, k)$, $k = 1...K$, calculate the coordinates of destination pixels, $XY = \{XY_k | k = 1...K\}$, and the vectors of values for the diffusion $\Phi = \{\Phi_k | k = 1...K\}$ as

$$\begin{cases} XY = Y_{XY} \circ X_R, \\ \Phi \quad = Y_\Phi \circ X_R, \end{cases} \tag{6}$$

where X_R is the vector of chaotic variables in binary what is generated by the PCM after R iterations; $Y_{XY} = [Y_{XY_K}, Y_{XY_{K-1}}, ..., Y_{XY_1}]^T$ and $Y_\Phi = [Y_{\Phi_K}, Y_{\Phi_{K-1}}, ..., Y_{\Phi_1}]^T$ are the rules of bit arrangements to extract bits from X_R. The member of Y_{XY} and Y_Φ are $Y_{XY_k} = [Y_i \;\; Y_j \;\; Y_k]^T$ and $Y_{\Phi_k} = [Y_{\phi_{2,k}} \;\; Y_{\phi_{1,k}}]^T$ are to construct the coordinate of destination pixels $XY_k = (i', j', k')$ for the permutation and random values $\Phi_k = (\phi_{2,k}, \phi_{1,k})$ for the diffusion. Respectively, the coordinate of destination pixels XY_k and random values Φ_k are calculated by

$$\begin{cases} i' = Y_i \circ X_R, \\ j' = Y_j \circ X_R, \\ k' = Y_k \circ X_R, \end{cases} \tag{7}$$

and

$$\begin{cases} \phi_{2,k} = Y_{\phi_2} \circ X_R, \\ \phi_{1,k} = Y_{\phi_1} \circ X_R. \end{cases} \tag{8}$$

Step 2: Compute the ciphered values for both source and destination pixels by XORing as

$$\begin{cases} c(i, j, k) \quad = p(i, j, k) \oplus c(i, j-1, k) \oplus \phi_{1,k}, \\ c(i', j', k') = p(i', j', k') \oplus \phi_{2,k}. \end{cases} \tag{9}$$

Step 3: Permute ciphered source and destination pixels as

$$\begin{cases} temp \qquad = c(i, j, k), \\ c(i, j, k) \quad = c(i', j', k'), \\ c(i', j', k') = temp, \end{cases} \tag{10}$$

where $temp$ is a temporary variable. For the context of MIE, a pixel of an image can be permuted with another pixel in either the same image (intra-image permutation when $k' = k$) or another image (inter-image when $k' \neq k$) as shown in Fig. 4.

For the inverse permutation and inverse diffusion of a pair of pixels, the order to compute values of recovered plain pixels is inverse in compared with that in the encryption. Specifically, the decryption algorithm is as

Step 1: For source pixels $p(i, j, k)$, $k = 1...K$, calculate the coordinates of destination pixels, $XY = \{XY_k | k = 1...K\}$, and the vectors of values for the diffusion $\Phi = \{\Phi_k | k = 1...K\}$ exactly identical to those given in (6)-(8).

Step 2: Permute ciphered source and destination pixels as

$$\begin{cases} temp \qquad = c(i, j, k), \\ c(i, j, k) \quad = c(i', j', k'), \\ c(i', j', k') = temp. \end{cases} \tag{11}$$

Step 3: Compute the recovered plaintext values for both source and destination pixels as

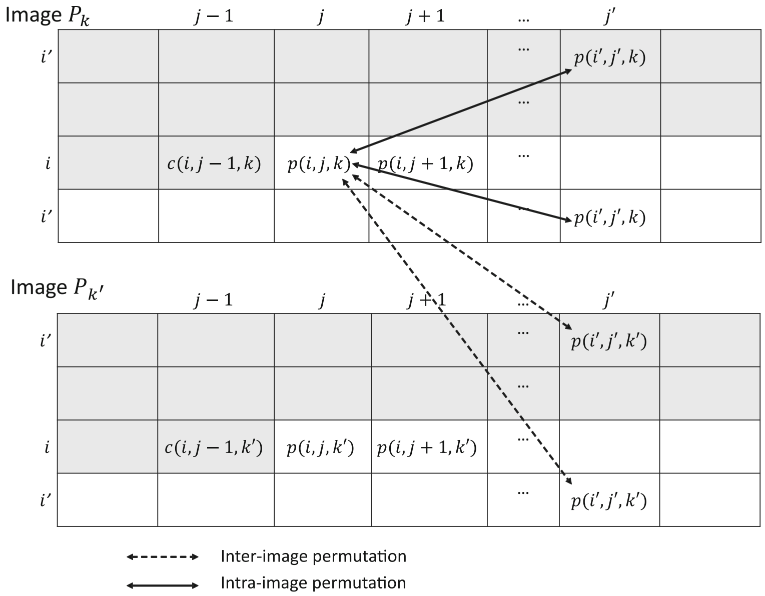


Fig. 4 All possible destination pixels $p(i', j', k')$ for source pixel $p(i, j, k)$ in image P_k

$$\begin{cases} p(i, j, k) & = c(i, j, k) \oplus c(i, j-1, k) \oplus \phi_{1,k}, \\ p(i', j', k') & = c(i', j', k') \oplus \phi_{2,k}. \end{cases} \tag{12}$$

2.5 Operation of encryption algorithm

The proposed structure in Fig. 1(b) operates with the flowchart as illustrated in Fig. 5(a). There are three phases, i.e., chaos iteration, calculation of coordinates and values, and diffusion and permutation.

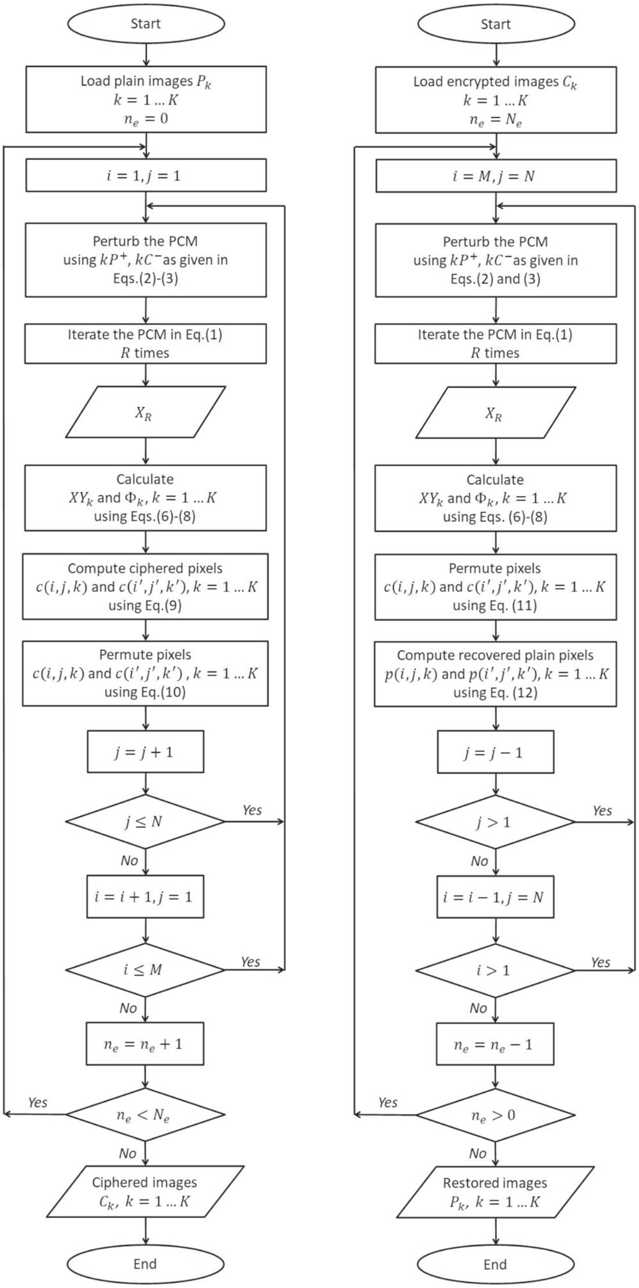
At first, the PCM in (1) is iterated R times with its inputs Γ_0, X_0, and E. Γ_0 and X_0 are interfered by E at every iteration number $n = 0$ and by the feedback for $1 \leq n \leq R$ as given in (2) and (3). The perturbation amount E is the result of bit manipulation with inputs E_1 and E_2 as

$$E_1 = \begin{cases} ||_{k=1}^{K} c_{0,k} & \text{for } (i, j) = (1, 1), k = 1..K \text{ and } n_e = 1; \\ ||_{k=1}^{K} c(i, j-1, k) & \text{for } (i, j) \neq (1, 1), k = 1..K \text{ and } 2 \leq n_e \leq N_e, \end{cases} \tag{13}$$

and

$$E_2 = \begin{cases} ||_{k=1}^{K} p_{0,k} & \text{for } (i, j) = (M, N), k = 1..K \text{ and } n_e = N_e; \\ ||_{k=1}^{K} p(i, j+1, k) & \text{for } (i, j) \neq (M, N), k = 1..K \text{ and } 1 \leq n_e < N_e, \end{cases} \tag{14}$$

where $||$ is the bit concatenation; n_e is number of encryption rounds; $p(i, j+1, k)$ and $c(i, j-1, k)$ are neighbor plain and neighbor ciphered pixels of the current one $p(i, j, k)$ in image k as depicted in Fig. 4; $p_{0,k}$ and $c_{0,k}$ are initial values for image k, and $kC_0^- = \{c(0, k)|k = 1...K\}$ and $kP_0^- = \{c(0, k)|k = 1...K\}$ are considered as part of the secret key. As illustrated in Figs. 1(b) and 5, $kP^+ = \{p(i, j+1, k)|k = 1...K\}$ and $kC^- = \{c(i, j-1, k)|k = 1...K\}$ are vectors of neighbor plain pixels and neighbor ciphered pixels from K images.


Start
Load plain images P_k
$k = 1 \ldots K$
$n_e = 0$
$i = 1, j = 1$
Perturb the PCM using kP^+, kC^-as given in Eqs.(2)-(3)
Iterate the PCM in Eq.(1) R times
X_R
Calculate XY_k and Φ_k, $k = 1 \ldots K$ using Eqs.(6)-(8)
Compute ciphered pixels $c(i,j,k)$ and $c(i',j',k')$, $k = 1 \ldots K$ using Eq.(9)
Permute pixels $c(i,j,k)$ and $c(i',j',k')$, $k = 1 \ldots K$ using Eq.(10)
$j = j + 1$
$j \leq N$
Yes
No
$i = i + 1, j = 1$
$i \leq M$
Yes
No
$n_e = n_e + 1$
$n_e < N_e$
Yes
No
Ciphered images C_k, $k = 1 \ldots K$
End
Start
Load encrypted images C_k
$k = 1 \ldots K$
$n_e = N_e$
$i = M, j = N$
Perturb the PCM using kP^+, kC^-as given in Eqs.(2) and (3)
Iterate the PCM in Eq.(1) R times
X_R
Calculate XY_k and Φ_k, $k = 1 \ldots K$ using Eqs. (6)-(8)
Permute pixels $c(i,j,k)$ and $c(i',j',k')$, $k = 1 \ldots K$ using Eq. (11)
Compute recovered plain pixels $p(i,j,k)$ and $p(i',j',k')$, $k = 1 \ldots K$ using Eq. (12)
$j = j - 1$
$j > 1$
Yes
No
$i = i - 1, j = N$
$i > 1$
Yes
No
$n_e = n_e - 1$
$n_e > 0$
Yes
No
Restored images P_k, $k = 1 \ldots K$
End

(a) Encryption (b) Decryption

Fig. 5 The flowchart of encryption and decryption algorithms

From (2)-(3) and (13)-(14), the dynamics of PCM is involved by the content of plain and intermediate ciphered images by means of perturbation.

In the second phase, the coordinates of K destination pixels are computed as Step 1 in Subsection 2.4. In the third phase, the diffusion and permutation are performed as Steps 2 to 4 in Subsection 2.4.

In each encryption round, the source pixels from K images are scanned and processed sequentially from left to right and top to bottom of images. The encryption as a whole are repeated N_e times.

According to the procedure as described above, the configuration of the decryption is identical to that of the encryption as illustrated in Fig. 1(b), but it is performed in reverse order. Specifically, the order of pixels is reverse in compared with that in the encryption, i.e. pixels from bottom to top and right to left. The flowchart of decryption algorithm is shown in Fig. 5(b).

Remarks for the design criteria:

There are some important points in the proposed structure when it is employed in the design of a cryptosystem. Those are related to the criteria in choosing values of parameters for the design.

1. For the selection of chaotic map model: In fact, any chaotic map can be used for the proposed structure. However, the number of computational operations of a encryption algorithm is dependent on the complexity of chaotic map's model. Therefore, the criteria to choose a chaotic map model in the proposed structure is that the number of dimensions is as small as possible, but the number of flippable bits is large enough to construct the perturbation amounts, and other values for the encryption. Anyways, the number of bits representing for the fractional portion must greater than 32 to avoid the deterioration of dynamics of chaotic map.
2. For the bit arrangement rules Y: There are two types of bit arrangement in the proposed structure, i.e., Y_i within the PCM and the set of Y_{XY} and Y_Φ for the permutation and diffusion. Firstly, Y_i ($i = 1..4$) are to construct the perturbation amounts to the PCM. The criteria for Y_i is that the PCM must work in chaotic behavior. So, they are chosen so that some bits at specific positions of values of control parameters and of state variables are fixed at the logic '0' or '1'. Secondly, Y_{XY} and Y_Φ are to induce for coordinates of pixels in the permutation as well as for random values in the diffusion. In fact, Y_{XY} and Y_Φ should be chosen so that pixels at the same coordinates should be shuffled with those at different destination coordinates and random values have a uniform distribution. Therefore, it is suggested in [21] that the bits at positions at least the fifth and beyond after the decimal point should be used for both the types of bit arrangements.
3. For the number of iterations R: In the proposed structure, the PCM is iterated R times before chaotic value X_R is used for the encryption. So, the value of R is chosen as small as possible to save the encryption time. In fact, if the PCM is set up to work in chaotic behavior, the value of R should be in the range of 1 to 5 is acceptable.

Next, the exemplar simulation using the proposed structure is carried out, then, the security analysis is shown.

3 Exemplar simulation

Let us consider the example with the use of two well-known chaotic maps, i.e. Logistic and Standard maps. Values of state variables and control parameters are represented in the format

Table 1 The bit patterns and value ranges of perturbed Logistic map

Parameter	Format	# flippable bits	Bit pattern	Value range
$\gamma_n^{(1)}$	$\langle 34, 32 \rangle$	30	11.11xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	[3.75,4.0)
$x_n^{(1)}$	$\langle 33, 32 \rangle$	31	0.xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1	$[2^{-32}$,1)

of fixed-point number. To avoid the degradation of chaotic orbits, the fraction portion of the chaotic value must be at least 32 bits. Here, the bit patterns for the state variables and control parameters are designated so that the PCMs exhibit the chaotic behavior.

3.1 Chosen PCMs

3.1.1 The perturbed Logistic map

The perturbed Logistic map is expressed by

$$x_{n+1}^{(1)} = \hat{\gamma}_n^{(1)} \hat{x}_n^{(1)} (1 - \hat{x}_n^{(1)}), \tag{15}$$

where $\gamma_n^{(1)}$ and $x_n^{(1)}$ are the control parameter and the state variable, respectively; and perturbed state variable and control parameter are $\hat{\gamma}_n^{(1)}$ and $\hat{x}_n^{(1)}$. The bit patterns for values of state variable and control parameter are chosen as given in Table 1. Notably, there are some bits being kept constant at '0' or '1' while 'x' is denoted for bits whose state can be flippable. The bit pattern of $x_n^{(1)}$ with bit '1' at the rightmost is to ensure that the value of state variable is never got stuck at the unstable fixed point, i.e., 0.0.

The rules of bit arrangements are chosen for the perturbed Logistic map as given in Table 2.

3.1.2 The perturbed standard map

The perturbed Standard map is chosen as

$$\begin{bmatrix} x_{n+1}^{(2)} \\ x_{n+1}^{(1)} \end{bmatrix} = MOD\left(\begin{bmatrix} \hat{x}_n^{(2)} + \hat{\gamma}_n^{(1)} sin(\hat{x}_n^{(2)} + \hat{x}_n^{(1)}) \\ \hat{x}_n^{(1)} + \hat{x}_n^{(2)} \end{bmatrix}, 2\pi \right) \tag{16}$$

Table 2 Bit arrangements in the perturbed Logistic map

Y_i	Bit arrangement
xY_1	$[B_0 B_0 B_0 B_0$(1,59)(1,33)(1,9)(1,10)(1,17)(1,52)(1,16)(1,49)(1,15)(1,57)(1,23)(1,12)(1,19)(1,39) (1,31)(1,25)(1,51)(1,37)(1,35)(1,56)(1,18)(1,46)(1,45)(1,24)(1,36)(1,3)(1,2)(1,34)(1,48)(1,58)]
Y_2	$[B_0 B_0 B_0 B_0$(1,6)(1,4)(1,20)(1,13)(1,15)(1,32)(1,18)(1,9)(1,7)(1,22)(1,29)(1,16)(1,19)(1,31)(1,26) (1,17)(1,21)(1,10)(1,2)(1,30)(1,27)(1,3)(1,8)(1,12)(1,14)(1,11)(1,23)(1,5)(1,24)(1,25)]
Y_3	$[B_0$(1,7)(1,38)(1,28)(1,1)(1,22)(1,11)(1,50)(1,21)(1,40)(1,8)(1,42)(1,20)(1,44)(1,47)(1,53)(1,29) (1,5)(1,14)(1,60)(1,6)(1,55)(1,32)(1,61)(1,4)(1,30)(1,13)(1,54)(1,26)(1,43)(1,41)(1,27)$B_0]$
Y_4	$[B_0$(1,10)(1,9)(1,2)(1,12)(1,4)(1,14)(1,30)(1,27)(1,5)(1,29)(1,24)(1,16)(1,17)(1,13)(1,31)(1,23) (1,15)(1,20)(1,11)(1,7)(1,8)(1,22)(1,26)(1,19)(1,28)(1,6)(1,25)(1,32)(1,3)(1,21)(1,18)$B_0]$

Table 3 The bit patterns and value ranges of perturbed Standard map

Parameter	Format	# flippable bits	Bit pattern	Value range
$\hat{\gamma}_n^{(1)}$	$\langle 35, 32\rangle$	34	xx1.xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	[1.0,8.0)
$\hat{x}_n^{(2)}$	$\langle 35, 32\rangle$	34	xxx.xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1	$[2^{-32}$,$2\pi)$
$\hat{x}_n^{(1)}$	$\langle 35, 32\rangle$	34	xxx.xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx1	$[2^{-32}$,$2\pi)$

where, the vectors of state variables and control parameter are $X_n = [x_n^{(2)} \;\; x_n^{(1)}]^T$ and $\Gamma_n = \gamma_n^{(1)}$, respectively. Accordingly, the vectors of perturbed state variables and control parameter are $\hat{X}_n = [\hat{x}_n^{(2)} \;\; \hat{x}_n^{(1)}]^T$ and $\hat{\Gamma}_n = \hat{\gamma}_n^{(1)}$.

The bit patterns for values of state variables and control parameter of the perturbed Standard map are chosen as shown in Table 3. Bit ‘1’ in the bit pattern of the control parameter makes the value range discontinued in four sub-ranges, i.e., [1.0, 2.0), [3.0, 4.0), [5.0, 6.0) and [7.0, 8.0). It is noted that the right-most bits ‘1’ in the bit patterns of state variables are to avoid the fixed point at (0, 0) of chaotic attractor.

The rules of bit arrangements are chosen for the perturbed Standard map as given in Table 4.

3.2 Chosen values of parameters for MIE

In this example, a set of 8 images ($K = 8$) with the size of $M = 256$, $N = 256$ are encrypted at the same time, i.e. Lena, Cameraman, House, Boat, Clock, Black and White as in the first column of Fig. 6. Each pixel is represented in 8-bit grayscale, so E_1 and E_2 are of 64 bits constructed by kC^- and kP^+ as given in (13)–(14). The required number of bits representing for E is dependent on the PCM for the MIE. As shown in Tables 1 and 3, the number of flippable bits in the value representation of control parameters and state variables is 61 and 102 bits for the perturbed Logistic and Standard maps, respectively. The rules of bit arrangements for Y_i ($i = 1..4$) of the perturbed Logistic and Standard maps are shown in Tables 2 and 4, respectively.

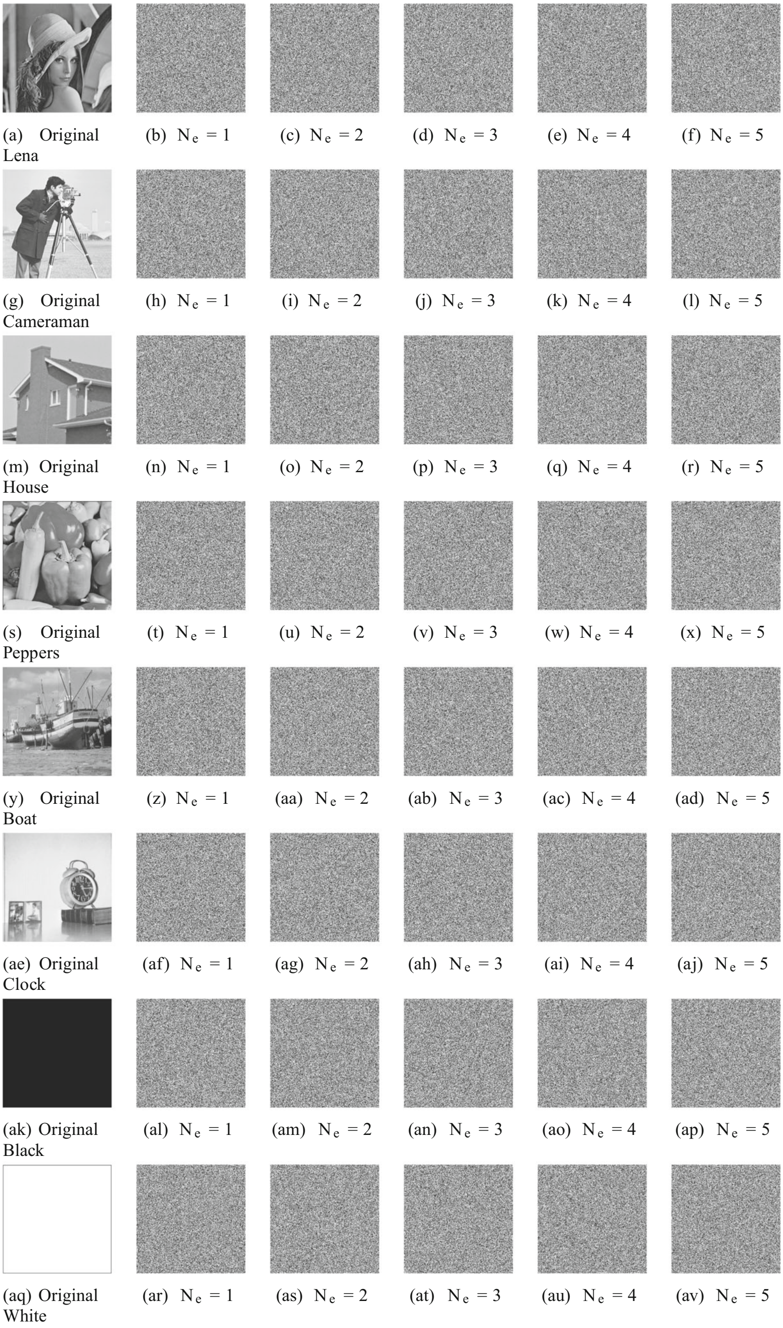
The adopted values for the initial conditions of the perturbed Logistic and Standard maps are shown in Table 5. The value for kC_0^- and kP_0^+ is chosen as in Table 6. Tables 7 and 8 show the bit arrangements $Y_{XY_k} = [Y_i \;\; Y_j \;\; Y_k]^T$ and $Y_\Phi = [Y_{\phi_1} \;\; Y_{\phi_2}]^T$ for inducing the coordinates of destination pixels (i', j', k') for the permutation and $\Phi = [\phi_1 \;\; \phi_2]$ for the diffusion in each of PCMs. It is noted that the inter-image permutation is applied in the simulation.

3.3 Simulation results

The encrypted images using the proposed structure are carried out with the use of perturbed Logistic and Standard maps for the fixed number of chaotic iterations $R = 5$ and various number of encryption rounds $N_e = 1..5$. To save the space, only those with the perturbed Logistic map are shown in Fig. 6, and those look like random images. Next, the quantitative estimation will be shown in the statistical analysis.

Table 4 Bit arrangements in the perturbed Standard map

Y_i	Bit arrangement
Y_1	[(1,45)(1,32)B_0(1,95)(1,44)(1,19)(1,92)(1,101)(1,46)(1,11)(1,27)(1,42)(1,63)(1,26)(1,64)(1,74)(1,22)(1,12)
	(1,33)(1,35)(1,49)(1,56)(1,7)(1,28)(1,84)(1,3)(1,96)(1,77)(1,54)(1,62)(1,24)(1,52)(1,99)(1,60)(1,57)]
Y_2	[(2,7)(2,15)B_0(2,13)(1,8)(1,17)(2,21)(1,11)(2,28)(2,22)(2,30)(2,17)(2,1)(1,9)(2,34)(1,29)(2,12)(1,5)(1,30)
	(2,24)(1,6)(1,23)(2,18)(2,32)(2,11)(1,4)(2,9)(1,21)(1,2)(1,12)(2,27)(1,26)(2,8)(1,10)(1,1)]
Y_3	$\left[\begin{matrix}(1, 87)(1, 81)(1, 51)(1, 15)(1, 20)(1, 88)(1, 1)(1, 43)(1, 16)(1, 100)(1, 68)(1, 41)(1, 39)(1, 6)(1, 67)(1, 5)(1, 13)\\(1, 21)(1, 53)(1, 70)(1, 75)(1, 40)(1, 38)(1, 102)(1, 4)(1, 90)(1, 91)(1, 82)(1, 9)(1, 25)(1, 36)(1, 72)(1, 14)(1, 78)\end{matrix}\right.$
	$\left.\begin{matrix}(1, 58)(1, 18)(1, 86)(1, 85)(1, 73)(1, 23)(1, 66)(1, 47)(1, 98)(1, 59)(1, 80)(1, 30)(1, 34)(1, 97)(1, 8)(1, 29)(1, 37)B_0\\(1, 10)(1, 69)(1, 55)(1, 83)(1, 76)(1, 93)(1, 89)(1, 31)(1, 71)(1, 17)(1, 2)(1, 79)(1, 50)(1, 48)(1, 94)(1, 61)(1, 65)B_0\end{matrix}\right]$
Y_4	$\left[\begin{matrix}(2, 31)(1, 31)(2, 11)(2, 8)(1, 33)(1, 13)(2, 28)(1, 15)(2, 2)(2, 27)(2, 6)(2, 14)(1, 28)(2, 21)(2, 9)(2, 32)(2, 4)(1, 4)\\(2, 5)(1, 21)(2, 12)(1, 25)(1, 12)(2, 30)(1, 6)(2, 34)(1, 19)(1, 20)(1, 27)(2, 3)(2, 19)(2, 15)(1, 2)(1, 26)(2, 23)(1, 5)\end{matrix}\right.$
	$\left.\begin{matrix}(1, 22)(1, 7)(1, 34)(2, 13)(2, 24)(2, 1)(1, 30)(2, 29)(2, 10)(1, 14)(2, 17)(1, 29)(1, 3)(2, 18)(2, 22)(1, 9)B_0\\(1, 11)(1, 17)(1, 32)(1, 24)(1, 23)(2, 7)(2, 25)(1, 8)(1, 1)(2, 20)(2, 16)(2, 26)(2, 33)(1, 18)(1, 16)(1, 10)B_0\end{matrix}\right]$



(a) Original Lena (b) $N_e = 1$ (c) $N_e = 2$ (d) $N_e = 3$ (e) $N_e = 4$ (f) $N_e = 5$

(g) Original Cameraman (h) $N_e = 1$ (i) $N_e = 2$ (j) $N_e = 3$ (k) $N_e = 4$ (l) $N_e = 5$

(m) Original House (n) $N_e = 1$ (o) $N_e = 2$ (p) $N_e = 3$ (q) $N_e = 4$ (r) $N_e = 5$

(s) Original Peppers (t) $N_e = 1$ (u) $N_e = 2$ (v) $N_e = 3$ (w) $N_e = 4$ (x) $N_e = 5$

(y) Original Boat (z) $N_e = 1$ (aa) $N_e = 2$ (ab) $N_e = 3$ (ac) $N_e = 4$ (ad) $N_e = 5$

(ae) Original Clock (af) $N_e = 1$ (ag) $N_e = 2$ (ah) $N_e = 3$ (ai) $N_e = 4$ (aj) $N_e = 5$

(ak) Original Black (al) $N_e = 1$ (am) $N_e = 2$ (an) $N_e = 3$ (ao) $N_e = 4$ (ap) $N_e = 5$

(aq) Original White (ar) $N_e = 1$ (as) $N_e = 2$ (at) $N_e = 3$ (au) $N_e = 4$ (av) $N_e = 5$

Fig. 6 Encrypted images using the perturbed Logistic map with $R = 5$ and various number of encryption rounds $N_e = 1..5$

Table 5 Initial values for simulation

PCM	Parameter/ State variable	Value	Bit pattern
Logistic map	$\gamma_0^{(1)}$	3.7599	11.11000010100010001100111001110000
	$x_0^{(1)}$	0.5599	0.10001111010101011001101100111101
Standard map	$\gamma_0^{(1)}$	1.2299	001.00111010110110101011100111110101
	$x_0^{(2)}$	1.2299	001.00111010110110101011100111110101
	$x_0^{(1)}$	4.5599	100.10001111010101011001101100111101

3.4 Statistical analyses

In order to confirm the feasibility of the proposed structure, the statistical analyses are considered for the simulation results. Histogram, information entropy and correlation of two adjacent pixels are measured for the encrypted images. In the context of multiple images, the average values are determined for the effectiveness of the MIE.

3.4.1 Histogram analysis

The distribution of pixel values of an image can be analysed and further analysis for the histogram can be measured by means of χ^2. For a 8-bit grayscale image, χ^2 is calculated by

$$\chi^2 = \sum_{i=0}^{255} \frac{(O_i - E_i)^2}{E_i}, \tag{17}$$

Table 6 Initial values of kC_0^- and kP_0^+ for simulation

Parameter		Value
kC_0^-	$c_{0,8}$	30
	$c_{0,7}$	111
	$c_{0,6}$	130
	$c_{0,5}$	165
	$c_{0,4}$	231
	$c_{0,3}$	140
	$c_{0,2}$	73
	$c_{0,1}$	9
kP_0^+	$p_{0,8}$	250
	$p_{0,7}$	176
	$p_{0,6}$	25
	$p_{0,5}$	141
	$p_{0,4}$	222
	$p_{0,3}$	39
	$p_{0,2}$	147
	$p_{0,1}$	215

Table 7 Bit arrangements for permutation and diffusion of the MIE using the perturbed Logistic map

	Bit arrangements
Y_i	$\begin{bmatrix} (1,2) & (1,6) & (1,23) & (1,6) & (1,3) & (1,25) & (1,14) & (1,15) \\ (1,5) & (1,18) & (1,16) & (1,29) & (1,10) & (1,13) & (1,17) & (1,25) \\ (1,3) & (1,10) & (1,14) & (1,28) & (1,32) & (1,22) & (1,2) & (1,2) \\ (1,26) & (1,32) & (1,24) & (1,22) & (1,2) & (1,10) & (1,21) & (1,19) \\ (1,13) & (1,15) & (1,25) & (1,21) & (1,15) & (1,3) & (1,30) & (1,31) \\ (1,31) & (1,11) & (1,20) & (1,7) & (1,13) & (1,5) & (1,8) & (1,26) \\ (1,9) & (1,17) & (1,30) & (1,12) & (1,31) & (1,7) & (1,15) & (1,12) \\ (1,4) & (1,27) & (1,8) & (1,19) & (1,24) & (1,26) & (1,11) & (1,30) \end{bmatrix}$
Y_j	$\begin{bmatrix} (1,19) & (1,9) & (1,5) & (1,25) & (1,16) & (1,31) & (1,5) & (1,30) \\ (1,30) & (1,18) & (1,12) & (1,11) & (1,5) & (1,25) & (1,6) & (1,18) \\ (1,30) & (1,9) & (1,16) & (1,11) & (1,30) & (1,28) & (1,4) & (1,26) \\ (1,22) & (1,24) & (1,20) & (1,23) & (1,10) & (1,26) & (1,8) & (1,26) \\ (1,12) & (1,20) & (1,28) & (1,14) & (1,32) & (1,22) & (1,23) & (1,9) \\ (1,31) & (1,25) & (1,19) & (1,25) & (1,14) & (1,4) & (1,23) & (1,15) \\ (1,28) & (1,18) & (1,21) & (1,32) & (1,28) & (1,6) & (1,13) & (1,14) \\ (1,14) & (1,29) & (1,7) & (1,11) & (1,12) & (1,8) & (1,27) & (1,32) \end{bmatrix}$
Y_k	$\begin{bmatrix} (1,19) & (1,16) & (1,4) \\ (1,28) & (1,18) & (1,14) \\ (1,4) & (1,15) & (1,19) \\ (1,3) & (1,24) & (1,14) \\ (1,24) & (1,5) & (1,3) \\ (1,26) & (1,12) & (1,9) \\ (1,15) & (1,13) & (1,8) \\ (1,20) & (1,25) & (1,21) \end{bmatrix}$
Y_{ϕ_1}	$\begin{bmatrix} (1,25) & (1,32) & (1,27) & (1,30) & (1,20) & (1,17) & (1,6) & (1,16) \\ (1,10) & (1,7) & (1,16) & (1,21) & (1,8) & (1,29) & (1,17) & (1,25) \\ (1,17) & (1,5) & (1,11) & (1,18) & (1,19) & (1,2) & (1,16) & (1,14) \\ (1,14) & (1,2) & (1,4) & (1,28) & (1,3) & (1,13) & (1,5) & (1,18) \\ (1,23) & (1,30) & (1,13) & (1,8) & (1,22) & (1,20) & (1,4) & (1,29) \\ (1,15) & (1,19) & (1,9) & (1,24) & (1,23) & (1,22) & (1,30) & (1,3) \\ (1,29) & (1,26) & (1,22) & (1,6) & (1,28) & (1,9) & (1,16) & (1,28) \\ (1,12) & (1,3) & (1,31) & (1,20) & (1,6) & (1,21) & (1,8) & (1,11) \end{bmatrix}$
Y_{ϕ_2}	$\begin{bmatrix} (1,30) & (1,17) & (1,27) & (1,16) & (1,23) & (1,12) & (1,17) & (1,16) \\ (1,8) & (1,10) & (1,18) & (1,25) & (1,19) & (1,9) & (1,2) & (1,15) \\ (1,31) & (1,17) & (1,10) & (1,25) & (1,18) & (1,11) & (1,32) & (1,22) \\ (1,7) & (1,32) & (1,25) & (1,2) & (1,5) & (1,13) & (1,25) & (1,22) \\ (1,4) & (1,31) & (1,11) & (1,30) & (1,32) & (1,17) & (1,2) & (1,10) \\ (1,26) & (1,24) & (1,20) & (1,27) & (1,30) & (1,16) & (1,19) & (1,18) \\ (1,5) & (1,18) & (1,11) & (1,26) & (1,31) & (1,4) & (1,2) & (1,13) \\ (1,22) & (1,9) & (1,18) & (1,23) & (1,4) & (1,10) & (1,6) & (1,23) \end{bmatrix}$

where the expected occurrence frequency E_i for the image with the size of $M \times N$ is $\frac{M*N}{256}$; the observed occurrence frequency O_i is the number of pixels with the value i. Here, the hypothesis test is accepted if $\chi^2 \leq \chi^2_\alpha(255)$; α is the significance level. Here, it is chosen as $\alpha = 0.05$, so $\chi^2_{0.05}(255) = 293.247$; it means that the histogram is considered as a uniform distribution if $\chi^2 \leq 293.247$.

Table 9 shows the χ^2-test results for the original and the ciphered images with various number of encryption rounds. For $N_e \geq 3$, the histogram of all individual encrypted images meets the condition of uniform distribution, i.e., ($\chi^2 \leq \chi^2_{0.05}(255)$). Overall, the average values of χ^2 tests for the histogram analysis also indicate that the uniform distribution is obtained with $N_e \geq 2$.

Table 8 Bit arrangements for permutation and diffusion of the MIE using the perturbed Standard map

Bit arrangements	
Y_i	$\begin{bmatrix} (1,34) & (1,32) & (2,7) & (1,5) & (2,31) & (2,33) & (2,32) & (2,12) \\ (1,12) & (1,6) & (2,18) & (2,16) & (1,22) & (2,11) & (1,1) & (1,7) \\ (2,21) & (2,2) & (1,11) & (2,5) & (1,21) & (2,6) & (1,18) & (2,17) \\ (1,24) & (1,4) & (1,30) & (1,27) & (2,27) & (2,14) & (1,3) & (2,13) \\ (1,13) & (2,24) & (2,25) & (2,22) & (1,15) & (2,3) & (2,4) & (1,33) \\ (1,10) & (1,14) & (2,29) & (1,19) & (1,23) & (2,34) & (2,15) & (2,23) \\ (1,9) & (1,25) & (1,28) & (1,8) & (2,1) & (1,26) & (2,19) & (2,8) \\ (2,30) & (2,10) & (1,2) & (2,26) & (1,17) & (1,16) & (2,20) & (2,9) \end{bmatrix}$
Y_j	$\begin{bmatrix} (1,17) & (1,7) & (2,1) & (2,9) & (2,14) & (1,5) & (1,21) & (2,30) \\ (1,17) & (2,4) & (2,9) & (1,13) & (2,18) & (2,23) & (2,9) & (1,12) \\ (1,29) & (2,23) & (2,16) & (2,2) & (1,8) & (2,23) & (2,33) & (1,28) \\ (1,27) & (2,7) & (1,4) & (1,2) & (1,32) & (2,6) & (1,25) & (2,34) \\ (2,28) & (1,26) & (1,1) & (1,22) & (2,30) & (2,32) & (1,8) & (2,28) \\ (1,29) & (1,26) & (2,17) & (2,34) & (1,28) & (2,26) & (2,3) & (2,1) \\ (1,31) & (2,27) & (2,11) & (2,1) & (2,16) & (1,7) & (1,2) & (2,25) \\ (1,20) & (1,20) & (2,22) & (2,26) & (1,1) & (1,10) & (1,11) & (1,22) \end{bmatrix}$
Y_k	$\begin{bmatrix} (1,11) & (1,31) & (2,30) \\ (2,20) & (2,18) & (2,34) \\ (2,16) & (2,8) & (1,32) \\ (1,20) & (2,33) & (2,24) \\ (2,3) & (2,11) & (1,24) \\ (2,8) & (1,31) & (2,11) \\ (1,7) & (2,17) & (1,27) \\ (1,7) & (2,18) & (1,25) \end{bmatrix}$
Y_{ϕ_1}	$\begin{bmatrix} (1,29) & (2,18) & (1,18) & (2,31) & (2,32) & (2,27) & (1,25) & (1,26) \\ (1,22) & (2,1) & (1,2) & (1,28) & (2,30) & (1,1) & (2,26) & (2,16) \\ (1,30) & (2,3) & (2,28) & (1,31) & (1,19) & (1,4) & (1,14) & (2,5) \\ (1,23) & (1,13) & (1,16) & (1,33) & (2,8) & (1,24) & (2,14) & (2,10) \\ (1,27) & (2,13) & (1,34) & (1,21) & (2,24) & (2,15) & (2,17) & (2,29) \\ (1,7) & (2,23) & (1,17) & (1,32) & (1,20) & (1,12) & (2,33) & (1,21) \\ (2,25) & (1,9) & (2,11) & (2,6) & (1,15) & (2,9) & (1,5) & (2,2) \\ (2,34) & (2,7) & (2,20) & (1,6) & (2,19) & (2,22) & (1,11) & (2,4) \end{bmatrix}$
Y_{ϕ_2}	$\begin{bmatrix} (2,12) & (2,25) & (2,4) & (1,22) & (2,12) & (2,33) & (1,8) & (1,25) \\ (2,16) & (2,27) & (2,11) & (2,17) & (2,19) & (1,23) & (2,18) & (1,24) \\ (2,34) & (2,1) & (1,28) & (1,19) & (1,17) & (1,20) & (1,10) & (1,11) \\ (2,15) & (2,19) & (1,5) & (1,26) & (2,34) & (1,20) & (1,29) & (1,26) \\ (1,10) & (1,1) & (2,33) & (1,33) & (2,1) & (2,3) & (1,3) & (2,30) \\ (2,8) & (1,21) & (2,6) & (2,25) & (1,18) & (1,24) & (2,24) & (2,21) \\ (1,3) & (1,28) & (1,6) & (1,31) & (2,2) & (1,7) & (2,4) & (1,12) \\ (2,12) & (2,4) & (1,9) & (2,21) & (1,31) & (2,13) & (2,17) & (2,15) \end{bmatrix}$

3.4.2 Information entropy

Information entropy (IE) of an image indicates the probability of pixel value v_i, $p(v_i)$, for a 8-bit grayscale image and it is computed by

$$IE = \sum_{i=0}^{255} p(v_i) log_2 \frac{1}{p(v_i)} \quad \text{(bits)}. \tag{18}$$

It is expected that encrypted images have IE as close to the ideal value, i.e., 8 bits, as possible. Table 10 displays the information entropy of original and encrypted images with various number of encryption rounds. IE of individual encrypted images and the averages

Table 9 Histogram analysis for χ^2 Test

PCM	N_e	χ^2 Test ($\chi^2_{0.05}(255) = 293.247$)								
		Lena	Camer- amen	House	Peppers	Boat	Clock	Black	White	On average
Logistic	Orig.	30578	161272	299789	36778	100675	282062	16711680	16711680	4291814
	1	281.922	254.258	259.875	273.883	262.250	258.219	*1062.477*	*2392.203*	*630.636*
	2	238.211	252.125	208.617	257.438	237.180	252.172	245.367	251.742	242.856
	3	238.375	245.875	264.758	241.383	275.664	234.813	243.320	259.594	250.473
	4	267.672	277.516	249.273	260.406	275.359	227.414	256.852	268.805	260.412
	5	252.133	256.773	236.805	245.398	259.133	239.484	274.234	244.641	251.075
Standard	1	236.383	241.453	248.883	206.836	249.727	232.078	245.930	*5033.102*	*836.799*
	2	259.336	233.156	263.609	272.781	264.000	*302.359*	232.109	*346.734*	271.761
	3	238.336	260.984	267.953	282.203	253.281	239.352	267.023	235.664	255.600
	4	234.406	261.313	268.492	243.914	257.492	257.117	258.156	226.477	250.921
	5	237.430	293.011	284.930	240.664	285.102	275.016	239.664	265.258	265.563

Table 10 Information Entropy

PCM	N_e	Information Entropy (bits) Lena	Cameramen	House	Peppers	Boat	Clock	Black	White	On average
Logistic	Original	7.5691	6.9046	6.4971	7.5327	7.1587	6.7057	0	0	5.2960
	1	7.9969	7.9972	7.9972	7.9970	7.9971	7.9972	7.9883	7.9751	7.9933
	2	7.9974	7.9972	7.9977	7.9972	7.9974	7.9972	7.9973	7.9972	7.9973
	3	7.9974	7.9973	7.9971	7.9973	7.9970	7.9974	7.9973	7.9971	7.9972
	4	7.9970	7.9969	7.9973	7.9971	7.9970	7.9975	7.9972	7.9970	7.9971
Standard	5	7.9972	7.9972	7.9974	7.9973	7.9971	7.9974	7.9970	7.9973	7.9972
	1	7.9974	7.9973	7.9973	7.9977	7.9972	7.9974	7.9973	7.9440	7.9907
	2	7.9971	7.9974	7.9971	7.9970	7.9971	7.9967	7.9975	7.9962	7.9970
	3	7.9974	7.9971	7.9970	7.9969	7.9972	7.9974	7.9971	7.9974	7.9972
	4	7.9974	7.9971	7.9970	7.9973	7.9972	7.9972	7.9972	7.9975	7.9972
	5	7.9974	7.9967	7.9969	7.9973	7.9969	7.9970	7.9974	7.9971	7.9971

are very close to the ideal value, 8 bits, for any number of encryption rounds. For $N_e \geq 2$, the entropy is greater than 7.9962 and its average is 7.9972.

3.4.3 Correlation of two adjacent pixels

The correlation of two adjacent pixels can be measured by the correlation coefficient $\rho_{X,Y}$. For the grayscale image, the correlation coefficient is considered for pairs of adjacent pixels in three directions, i.e., horizontal, vertical and diagonal. An image with lower absolute values of correlation coefficients is more random in pixel values, or less visual structure. The equation for the Pearson correlation coefficient of two sequences X and Y is

$$\rho_{X,Y} = \frac{\sum_{i=1}^{N_{pair}} (x_i - \overline{X})(y_i - \overline{Y})}{\sqrt{\left(\sum_{i=1}^{N_{pair}} (x_i - \overline{X})^2\right)\left(\sum_{i=1}^{N_{pair}} (y_i - \overline{Y})^2\right)}}, \tag{19}$$

where x_i and y_i are values of adjacent pixels in the sequences X and Y, respectively; N_{pair} is the number of adjacent pixel pairs from an image; $\overline{X}$ and $\overline{Y}$ are the means of X and Y, respectively. The pairs of adjacent pixels are chosen in three directions, i.e., horizontal, vertical and diagonal.

Respectively, Tables 11, 12 and 13 show the correlation coefficients in horizontal, vertical and diagonal of original and ciphered images with various number of encryption rounds. Obviously, the correlation coefficients of encrypted images are very small and significantly less than those of original images in every direction. It means that the visual structure of original images are completely removed in the ciphered images. The average values of correlation coefficients are also relatively small and those fluctuate around zero regardless of number of encryption rounds with both PCMs.

3.5 Security analyses

Below is the security analyses based on the space of secret key, the sensitivity of secret key, and the sensity of plaintext. It is noted from the tables that if values are displayed in italic, those are not passed the random test.

3.5.1 Space of secret key

In the proposed structure, the secret key consists of the initial values of Γ_0 and X_0, as well as those of kC_0^- and kP_0^+. For the initial values of PCMs, only flippable bits in the state variables and control parameters are counted for the space of secret key. The number of bits for kC_0^- and kP_0^+ is dependent on the number of images K and number of bits representing for a pixel.

According to Tables 1 and 3, the number of flippable bits in the initial values of Logistic and Standard maps is 61 and 102 bits, respectively. Also, the number of bits representing for initial values of kC_0^- and kP_0^+ is 128 bits for eight images with each pixel of 8-bit grayscale. So, the space of secret key of the cryptosystems is 2^{189} and 2^{230} for Logistic and Standard maps, respectively. With these numbers of key space, the exemplar cryptosystems are secured with modern computers.

Table 11 Correlation coefficients of horizontal direction

PCM	N_e	ρ of horizontal direction								
		Lena	Camer- amen	House	Peppers	Boat	Clock	Black	White	On average
Logistic	Original	0.93998	0.91957	0.97807	0.94777	0.92684	0.95649	NaN	NaN	0.94479
	1	-0.00274	0.01514	-0.00433	-0.00539	0.00286	0.00438	-0.00669	-0.02085	-0.00220
	2	-0.00178	-0.00699	0.00770	-0.00233	-0.00884	0.00463	-0.00508	0.00020	-0.00156
	3	-0.00614	-0.00299	-0.00368	0.00117	-0.00017	0.00480	-0.00164	0.00222	-0.00080
	4	0.00160	0.00297	-0.00103	0.00048	0.00201	0.00272	0.00320	0.00068	0.00158
Standard	5	0.00446	0.00155	0.00187	-0.00655	-0.00509	-0.00194	-0.00091	0.00469	-0.00024
	1	0.00431	-0.00391	0.00290	0.00523	-0.00454	-0.00069	-0.00105	0.04612	0.00605
	2	0.00825	0.00246	-0.00497	0.00221	-0.00497	0.00009	0.00458	-0.00481	0.00035
	3	-0.00019	-0.00335	0.00477	-0.00707	-0.00482	0.00235	-0.00803	-0.00267	-0.00238
	4	0.00338	-0.00275	0.00131	0.00638	-0.00069	-0.00201	0.00182	0.00104	0.00106
	5	-0.00010	0.00028	-0.00832	0.00005	-0.00822	-0.00176	-0.00501	-0.00590	-0.00362

Table 12 Correlation coefficients of vertical direction

PCM	N_e	ρ of vertical direction								
		Lena	Camer- amen	House	Peppers	Boat	Clock	Black	White	On average
Logistic	Original	0.96934	0.95494	0.96528	0.94819	0.94519	0.97408	NaN	NaN	0.95950
	1	-0.00296	-0.01474	0.00565	0.00038	-0.00762	-0.00121	0.00137	0.03704	0.00224
	2	-0.00224	-0.00261	0.00514	0.00424	0.00126	0.00028	-0.00313	-0.00269	0.00003
	3	0.00158	0.00596	-0.00692	-0.00194	0.00184	0.00207	0.00459	0.00213	0.00117
	4	0.00575	0.00251	0.00012	-0.00378	0.00733	-0.00142	0.00048	-0.00394	0.00088
Standard	5	0.00049	0.00420	0.00432	0.00470	0.00385	0.00556	-0.00411	-0.00579	0.00165
	1	-0.00434	-0.00391	-0.00312	-0.00541	0.00205	-0.00534	-0.00143	*-0.05612*	-0.00970
	2	0.00109	-0.00186	-0.00322	-0.00060	-0.00178	-0.00250	-0.00071	0.00179	-0.00097
	3	0.00062	0.00597	0.00545	-0.00568	0.00035	0.00174	0.00122	0.00129	0.00137
	4	-0.00293	0.00233	-0.00175	0.00678	-0.00187	0.00491	-0.00118	-0.00011	0.00077
	5	0.00250	0.00178	-0.00615	0.00286	0.00020	-0.00059	0.00004	0.00403	0.00058

Table 13 Correlation coefficients of diagonal direction

PCM	N_e	ρ of diagonal direction								
		Lena	Camer- amen	House	Peppers	Boat	Clock	Black	White	On average
Logistic	Original	0.91793	0.89619	0.94835	0.90359	0.88334	0.93893	NaN	NaN	0.91472
	1	0.00436	-0.01174	0.00247	0.00052	-0.00083	-0.00280	0.00253	-0.02088	-0.00329
	2	-0.00127	0.00153	-0.00278	-0.00311	-0.00418	-0.00189	0.00505	-0.00614	-0.00160
	3	-0.00063	-0.01042	-0.00343	0.00080	-0.00227	0.00305	-0.00492	-0.00050	-0.00229
	4	0.00353	-0.00167	-0.00211	-0.00944	-0.00464	-0.00259	0.00044	-0.00032	-0.00210
Standard	5	-0.00032	0.00055	-0.00127	-0.00103	-0.00816	-0.00244	-0.00271	0.00077	-0.00183
	1	0.00439	-0.00229	-0.00056	0.00050	-0.00558	-0.00372	-0.00289	-0.04404	-0.00677
	2	-0.00652	-0.00348	-0.00140	0.00057	-0.00722	-0.00388	-0.00444	-0.00616	-0.00407
	3	-0.00133	-0.00797	0.00188	-0.00542	0.00536	-0.00073	0.00246	0.00659	0.00011
	4	-0.00147	0.00479	-0.00489	0.00004	-0.00099	0.00727	0.00264	-0.00854	-0.00014
	5	-0.00216	-0.00131	0.00516	-0.00115	0.00231	-0.00088	-0.00188	-0.00063	-0.00007

3.5.2 Sensitivity of secret key

The sensitivity of secret key can be measured for the difference between two versions of ciphertexts being encrypted by two secret keys. Among two secret keys, one secret key is obtained with little modification to the other one. The number of pixels change rate ($NPCR$) and unified averaged changed intensity ($UACI$) are used for evaluating the sensitivity of secret key.

Let us call C_1 and C_2 be ciphertexts obtained by encrypting the same plaintext using two secret keys S and S', respectively. The modified secret key is $S' = S + \Delta_S$. Here, Δ_S is the tolerance of one certain element of secret key, and Δ_S is smallest value but larger than zero. Here, Tables 14 and 15 show original and modified secret keys for the simulation of sensitivity of secret key. Tables 16 and 17 display the tolerance of secret key Δ_S for the values of initial state variables, control parameters, plaintexts and ciphertexts. In order to measure the sensitivity of small changes in the secret key, the simulation is carried out for each tolerance individually while the others are kept intact as defined in Tables 5 and 6.

Let us consider the difference between two images C_1 and C_2. Firstly, the difference function between two values a and b is $Difp(a, b)$ defined by

$$Difp(a,b) = \begin{cases} 1, \text{ for } a \neq b; \\ 0, \text{ for } a = b. \end{cases} \tag{20}$$

Secondly, the difference between two images C_1 and C_2 is considered by the difference for every pair of pixels at the same position (i, j), i.e., $C_1(i, j)$ and $C_2(i, j)$ for $i = 1..M$ and $j = 1..N$. The $NPCR$ and $UACI$ are measured by

$$NPCR = \frac{\sum_{i,j} Difp(C_1(i,j), C_2(i,j))}{M \times N} \times 100\%, \tag{21}$$

and

$$UACI = \frac{1}{M \times N} \left[\sum_{i,j} \frac{|C_1(i,j) - C_2(i,j)|}{255} \right] \times 100\%. \tag{22}$$

Table 14 The original value of secret key and its modified versions for the perturbed Logistic map

Value of secret key	Elements of secret key			
S	$x_0^{(1)}$	$\gamma_0^{(1)}$	kP_0	kC_0
$S + \Delta S_{x_0^{(1)}}$	$x_0^{(1)} + \Delta_{x_0^{(1)}}$	$\gamma_0^{(1)}$	kP_0	kC_0
$S - \Delta S_{x_0^{(1)}}$	$x_0^{(1)} - \Delta_{x_0^{(1)}}$	$\gamma_0^{(1)}$	kP_0	kC_0
$S + \Delta S_{\gamma_0^{(1)}}$	$x_0^{(1)}$	$\gamma_0^{(1)} + \Delta_{\gamma_0^{(1)}}$	kP_0	kC_0
$S - \Delta S_{\gamma_0^{(1)}}$	$x_0^{(1)}$	$\gamma_0^{(1)} - \Delta_{\gamma_0^{(1)}}$	kP_0	kC_0
$S + \Delta S_{kP_0}$	$x_0^{(1)}$	$\gamma_0^{(1)}$	$p_{0,1} + \Delta_{p_{0,1}}$	kC_0
$S - \Delta S_{kP_0}$	$x_0^{(1)}$	$\gamma_0^{(1)}$	$p_{0,1} + \Delta_{p_{0,1}}$	kC_0
$S + \Delta S_{kC_0}$	$x_0^{(1)}$	$\gamma_0^{(1)}$	kP_0	$c_{0,1} + \Delta_{c_{0,1}}$
$S - \Delta S_{kC_0}$	$x_0^{(1)}$	$\gamma_0^{(1)}$	kP_0	$c_{0,1} - \Delta_{c_{0,1}}$

Table 15 The original value of secret key and its modified versions for the perturbed Standard map

Value of secret key	Elements of secret key				
S	$x_0^{(1)}$	$x_0^{(2)}$	$\gamma_0^{(1)}$	kP_0	kC_0
$S+\Delta S_{x_0^{(1)}}$	$x_0^{(1)}+\Delta_{x_0^{(1)}}$	$x_0^{(2)}$	$\gamma_0^{(1)}$	kP_0	kC_0
$S-\Delta S_{x_0^{(1)}}$	$x_0^{(1)}-\Delta_{x_0^{(1)}}$	$x_0^{(2)}$	$\gamma_0^{(1)}$	kP_0	kC_0
$S+\Delta S_{x_0^{(2)}}$	$x_0^{(1)}$	$x_0^{(2)}+\Delta_{x_0^{(2)}}$	$\gamma_0^{(1)}$	kP_0	kC_0
$S-\Delta S_{x_0^{(2)}}$	$x_0^{(1)}$	$x_0^{(2)}-\Delta_{x_0^{(2)}}$	$\gamma_0^{(1)}$	kP_0	kC_0
$S+\Delta S_{\gamma_0^{(1)}}$	$x_0^{(1)}$	$x_0^{(2)}$	$\gamma_0^{(1)}+\Delta_{\gamma_0^{(1)}}$	kP_0	kC_0
$S-\Delta S_{\gamma_0^{(1)}}$	$x_0^{(1)}$	$x_0^{(2)}$	$\gamma_0^{(1)}-\Delta_{\gamma_0^{(1)}}$	kP_0	kC_0
$S+\Delta S_{kP_0}$	$x_0^{(1)}$	$x_0^{(2)}$	$\gamma_0^{(1)}$	$p_{0,1}+\Delta_{p_{0,1}}$	kC_0
$S-\Delta S_{kP_0}$	$x_0^{(1)}$	$x_0^{(2)}$	$\gamma_0^{(1)}$	$p_{0,1}-\Delta_{p_{0,1}}$	kC_0
$S+\Delta S_{kC_0}$	$x_0^{(1)}$	$x_0^{(2)}$	$\gamma_0^{(1)}$	kP_0	$c_{0,1}+\Delta_{c_{0,1}}$
$S-\Delta S_{kC_0}$	$x_0^{(1)}$	$x_0^{(2)}$	$\gamma_0^{(1)}$	kP_0	$c_{0,1}-\Delta_{c_{0,1}}$

In this work, $NPCR$ and $UACI$ are tested for the randomness in response to the small change in the secret key with a significance level α for 8-bit grayscale images of the size 256×256 as presented in [52]. The randomness tests are passed if $NRCP > NRCP_\alpha^*$ and $UACI_\alpha^{*-} < UACI < UACI_\alpha^{*+}$. The critical values at $\alpha = 0.05$ for both $NPCR$ and $UACI$ are $NRCP_{0.05}^* = 99.5693\%$, $UACI_{0.05}^{*-} = 33.2824\%$ and $UACI_{0.05}^{*+} = 33.6447\%$.

Tables 18 and 20 present $NPCR$ for the sensitivity of secret key with the use of perturbed Logistic and Standard maps, respectively. It is almost insensitive to a small change in kP_0^+. For the rest of state variables and control parameters, more than 98.639% and 99.300% pixels of ciphertexts are changed due to the tolerance S' in the modified secret key in the perturbed Logistic and Standard maps, respectively. Except for kP_0^+, all individual and averaged values of $NPCR$ are passed the random test (or greater than $NRCP_\alpha^*$) for $N_e \geq 2$. In other words, the cryptosystems employing the proposed structure using the perturbed Logistic and Cat, Standard maps are with high sensitivity to the secret key.

Table 16 Tolerance in the value of state variables and control parameters for $NPCR$ and $UACI$

PCM	Parameter/State variable	Value	Bit pattern of ΔS
Logistic map	$\Delta_{\gamma_0^{(1)}}$	$+2^{-32}$	0.00000000000000000000000000000001
	$\Delta_{x_0^{(1)}}$	$+2^{-31}$	0.00000000000000000000000000000010
Standard map	$\Delta_{\gamma_0^{(1)}}$	-2^{-32}	0.00000000000000000000000000000001
	$\Delta_{x_0^{(2)}}$	$+2^{-31}$	0.00000000000000000000000000000010
	$\Delta_{x_0^{(1)}}$	$+2^{-31}$	0.00000000000000000000000000000010

Table 17 Tolerance in the value of kC_0^- and kP_0^+ for $NPCR$ and $UACI$

Param		Value
$\Delta_{kC_0^-}$	$\Delta_{c_{0,8}}$	0
	$\Delta_{c_{0,7}}$	0
	$\Delta_{c_{0,6}}$	0
	$\Delta_{c_{0,5}}$	0
	$\Delta_{c_{0,4}}$	0
	$\Delta_{c_{0,3}}$	0
	$\Delta_{c_{0,2}}$	0
	$\Delta_{c_{0,1}}$	1
$\Delta_{kP_0^+}$	$\Delta_{p_{0,8}}$	0
	$\Delta_{p_{0,7}}$	0
	$\Delta_{p_{0,6}}$	0
	$\Delta_{p_{0,5}}$	0
	$\Delta_{p_{0,4}}$	0
	$\Delta_{p_{0,3}}$	0
	$\Delta_{p_{0,2}}$	0
	$\Delta_{p_{0,1}}$	1

In addition, the change in the intensity $UACI$ for the sensitivity of secret key with the use of perturbed Logistic and Standard maps for each element of secret key is also seen in Tables 19 and 21, respectively. Similar to the $NCPR$, for $N_e \geq 2$, all values of $UACI$ and its averages are passed the random tests, or the values of $UACI$ are within in the range of (33.2824%,33.6447%) with $\alpha = 0.05$ for all of perturbed Logistic and Standard maps.

As seen from Tables 18, 19, 20, and 21 for both $NPCR$ and $UACI$, the cryptosystem using any PCMs is almost insensitivity to kP_0^+. According to (14), the value of kP_0^+ is only used for the last pixels of plain images in the last round of encryption. As given in Table 17, the tolerance is occurred for the last pixel of only one of eight images. This makes a few pixels related to the tolerance of kP_0^+ changed in the final round of encryption. However, the encryption is highly sensitive to kC_0^-. That is because the encryption is the forward direction of pixel scanning. Therefore, the decryption is the reverse direction, so it will be highly sensitive to kP_0^+ and insensitive to kC_0^-. The sensitivity of kC_0^- and kP_0^+ is significant, but it is asymmetry in the encryption and decryption.

3.5.3 Sensitivity of plaintext

A cryptosystem can resist from the types of known-plaintext and chosen-plaintext attacks if its sensitivity of plaintext is significant. In general, the original image is with little modification to become the modified plain image. Both the original and modified plain images are encrypted using the same value of secret key to produce two ciphered images. Sensitivity of the plaintext is obtained by means of statistical comparison between such two ciphered images. In this work, due to the encryption of multiple images at the same time, a set of modified images for analysis consists of one modified image and other original ones. The modified image is chosen alternatively among eight original images. Therefore, there are eight sets of modified plain images as listed in Table 22. Encryption is carried out for the set of original images

Table 18 Sensitivity of the secret key by means of $NPCR$ with the perturbed Logistic map

N_e	Images	$NPCR$ (%) for the sensitivity on			
		$x_0^{(1)}$	$\gamma_0^{(1)}$	kC_0^-	kP_0^+
1	Lena	99.586	99.622	99.614	0
	Cameraman	99.586	99.586	99.614	0
	House	99.582	99.586	99.608	0
	Peppers	99.582	99.643	99.612	0
	Boat	99.596	99.580	*99.547*	0
	Clock	99.590	99.620	*99.548*	0
	Black	99.541	99.467	*99.544*	0
	White	*98.766*	*98.639*	*98.766*	0
	On average	99.479	99.468	99.482	0
2	Lena	99.623	99.626	99.583	0
	Cameraman	99.640	99.667	99.570	0
	House	99.620	99.605	99.629	0
	Peppers	99.619	99.593	99.605	0
	Boat	99.579	99.586	99.596	0
	Clock	99.637	99.594	99.619	0
	Black	99.603	99.593	99.600	0
	White	99.622	99.641	99.611	0
	On average	99.618	99.613	99.602	0
3	Lena	99.623	99.599	99.609	0
	Cameraman	99.625	99.603	99.648	0
	House	99.579	99.615	99.619	0
	Peppers	99.625	99.577	99.585	0
	Boat	99.637	99.605	99.600	0
	Clock	99.615	99.606	99.594	0
	Black	99.631	99.626	99.612	0
	White	99.623	99.706	99.629	0
	On average	99.620	99.617	99.612	0
4	Lena	99.609	99.631	99.603	0
	Cameraman	99.599	99.637	99.622	0
	House	99.594	99.612	99.594	0
	Peppers	99.594	99.596	99.638	0
	Boat	99.654	99.648	99.597	0
	Clock	99.611	99.654	99.617	0
	Black	99.640	99.583	99.623	0
	White	99.594	99.612	99.605	0
	On average	99.612	99.622	99.612	0
5	Lena	99.594	99.598	99.588	0.005
	Cameraman	99.617	99.583	99.617	0.005
	House	99.622	99.612	99.583	0.005
	Peppers	99.631	99.599	99.640	0.005
	Boat	99.649	99.619	99.597	0.005
	Clock	99.629	99.609	99.597	0.005

Table 18 continued

N_e	Images	$NPCR$ (%) for the sensitivity on			
		$x_0^{(1)}$	$\gamma_0^{(1)}$	kC_0^-	kP_0^+
	Black	99.626	99.608	99.628	0.005
	White	99.623	99.615	99.625	0.005
	On average	99.624	99.605	99.609	0.005

Table 19 Sensitivity of the secret key by means of $UACI$ with the perturbed Logistic map

N_e	Images	$UACI$ (%) for the sensitivity on			
		$x_0^{(1)}$	$\gamma_0^{(1)}$	kC_0^-	kP_0^+
1	Lena	33.570	33.621	33.426	0
	Cameraman	*33.143*	*33.208*	*33.006*	0
	House	33.385	33.336	33.308	0
	Peppers	33.317	33.539	33.388	0
	Boat	33.377	33.526	33.359	0
	Clock	33.354	33.292	33.421	0
	Black	33.430	33.346	33.553	0
	White	*32.311*	*32.147*	*32.430*	0
	On average	33.236	33.252	33.236	0
2	Lena	33.512	33.486	33.427	0
	Cameraman	33.391	33.564	33.423	0
	House	33.415	33.613	33.525	0
	Peppers	33.486	33.541	33.609	0
	Boat	33.532	33.589	33.556	0
	Clock	33.365	33.515	33.468	0
	Black	33.483	33.391	33.466	0
	White	33.508	33.613	33.532	0
	On average	33.462	33.539	33.501	0
3	Lena	33.604	33.619	33.556	0
	Cameraman	33.564	33.635	33.359	0
	House	33.420	33.342	33.377	0
	Peppers	33.490	33.496	33.514	0
	Boat	33.427	33.288	33.417	0
	Clock	33.377	33.357	33.536	0
	Black	33.617	33.467	33.399	0
	White	33.627	33.553	33.440	0
	On average	33.516	33.470	33.450	0
4	Lena	33.534	33.304	33.577	0
	Cameraman	33.621	33.432	33.503	0
	House	33.373	33.525	33.569	0
	Peppers	33.563	33.539	33.323	0
	Boat	33.527	33.474	33.447	0
	Clock	33.413	33.400	33.454	0
	Black	33.443	33.360	33.374	0
	White	33.379	33.433	33.502	0

Table 19 continued

N_e	Images	$UACI$ (%) for the sensitivity on			
		$x_0^{(1)}$	$\gamma_0^{(1)}$	kC_0^-	kP_0^+
	On average	33.482	33.433	33.469	0
5	Lena	33.289	33.445	33.580	0.002
	Cameraman	33.536	33.505	33.333	0.001
	House	33.423	33.559	33.550	0.002
	Peppers	33.468	33.434	33.468	0.001
	Boat	33.465	33.622	33.591	0.002
	Clock	33.371	33.347	33.348	0.002
	Black	33.611	33.508	33.430	0.002
	White	33.639	33.300	33.455	0.002
	On average	33.475	33.465	33.469	0.002

Table 20 Sensitivity of the secret key by means of $NPCR$ with the perturbed Standard map

N_e	Images	$NPCR$ (%) for the sensitivity on				
		$x_0^{(2)}$	$x_0^{(1)}$	$\gamma_0^{(1)}$	kC_0^-	kP_0^+
1	Lena	99.611	99.582	99.615	99.608	0
	Cameraman	99.625	99.649	99.649	99.603	0
	House	99.576	99.623	99.599	99.612	0
	Peppers	99.623	99.596	99.629	99.635	0
	Boat	99.600	99.583	99.612	99.596	0
	Clock	99.594	99.637	*99.565*	99.594	0
	Black	99.593	99.615	99.629	99.625	0
	White	*99.411*	*99.333*	*99.300*	*99.336*	0
	On average	99.579	99.577	99.575	99.576	0
2	Lena	99.606	99.617	99.617	99.605	0
	Cameraman	99.615	99.644	99.602	99.620	0
	House	99.603	99.667	99.585	99.614	0
	Peppers	99.588	99.603	99.599	99.590	0
	Boat	99.609	99.594	99.635	99.614	0
	Clock	99.640	99.619	99.586	99.614	0
	Black	99.582	99.609	99.609	99.605	0
	White	99.599	99.601	99.594	99.637	0
	On average	99.605	99.619	99.603	99.612	0
3	Lena	99.576	99.628	99.612	99.661	0
	Cameraman	99.605	99.580	99.612	99.609	0
	House	99.628	99.579	99.648	99.657	0
	Peppers	99.641	99.606	99.609	99.628	0
	Boat	99.619	99.594	99.615	99.593	0
	Clock	99.622	99.628	99.579	99.641	0
	Black	99.646	99.648	99.617	99.623	0
	White	99.593	99.625	99.609	99.611	0
	On average	99.616	99.611	99.613	99.628	0
4	Lena	99.635	99.585	99.614	99.599	0

Table 20 continued

N_e	Images	$NPCR$ (%) for the sensitivity on				
		$x_0^{(2)}$	$x_0^{(1)}$	$\gamma_0^{(1)}$	kC_0^-	kP_0^+
	Cameraman	99.612	99.597	99.611	99.599	0
	House	99.628	99.602	99.594	99.603	0
	Peppers	99.617	99.606	99.614	99.611	0
	Boat	99.608	99.608	99.573	99.640	0
	Clock	99.612	99.652	99.610	99.597	0
	Black	99.586	99.593	99.649	99.637	0
	White	99.594	99.652	99.641	99.614	0
	On average	99.612	99.612	99.613	99.613	0
5	Lena	99.606	99.614	99.608	99.622	0.005
	Cameraman	99.623	99.652	99.628	99.591	0.005
	House	99.576	99.611	99.640	99.597	0.005
	Peppers	99.579	99.622	99.614	99.600	0.005
	Boat	99.615	99.591	99.582	99.635	0.005
	Clock	99.615	99.599	99.619	99.594	0.005
	Black	99.634	99.625	99.612	99.612	0.005
	White	99.594	99.599	99.608	99.619	0.005
	On average	99.605	99.614	99.614	99.609	0.005

Table 21 Sensitivity of the secret key by means of $UACI$ with the perturbed Standard map

N_e	Images	$UACI$ (%) for the sensitivity on				
		$x_0^{(2)}$	$x_0^{(1)}$	$\gamma_0^{(1)}$	kC_0^-	kP_0^+
1	Lena	33.474	33.425	33.409	*33.252*	0
	Cameraman	33.355	33.375	33.550	33.364	0
	House	33.434	33.587	33.558	33.363	0
	Peppers	33.358	33.484	33.491	33.466	0
	Boat	33.454	33.423	33.464	*33.263*	0
	Clock	33.341	33.368	33.348	33.442	0
	Black	33.424	33.496	33.500	33.520	0
	White	*32.583*	*32.416*	*32.517*	*32.659*	0
	On average	33.303	33.322	33.355	33.291	0
2	Lena	33.553	33.445	33.340	33.510	0
	Cameraman	33.466	33.422	33.440	33.418	0
	House	33.436	33.535	33.476	33.495	0
	Peppers	33.625	33.450	33.318	33.346	0
	Boat	33.464	33.595	33.564	33.506	0
	Clock	33.361	33.607	33.556	33.484	0
	Black	33.346	33.371	33.342	33.382	0
	White	33.622	33.496	33.559	33.614	0
	On average	33.484	33.490	33.449	33.469	0
3	Lena	33.431	33.617	33.474	33.571	0
	Cameraman	33.546	33.406	33.441	33.390	0
	House	33.411	33.452	33.331	33.531	0

Table 21 continued

N_e	Images	$UACI$ (%) for the sensitivity on				
		$x_0^{(2)}$	$x_0^{(1)}$	$y_0^{(1)}$	kC_0^-	kP_0^+
	Peppers	33.475	33.374	33.407	33.420	0
	Boat	33.301	33.363	33.394	33.409	0
	Clock	33.523	33.493	33.519	33.421	0
	Black	33.445	33.478	33.624	33.621	0
	White	33.613	33.549	33.420	33.532	0
	On average	33.468	33.467	33.451	33.487	0
4	Lena	33.297	33.447	33.349	33.369	0
	Cameraman	33.478	33.471	33.589	33.440	0
	House	33.377	33.526	33.464	33.382	0
	Peppers	33.530	33.426	33.304	33.367	0
	Boat	33.330	33.562	33.535	33.421	0
	Clock	33.561	33.420	33.491	33.378	0
	Black	33.334	33.449	33.377	33.296	0
	White	33.506	33.588	33.432	33.448	0
	On average	33.427	33.486	33.443	33.388	0
5	Lena	33.578	33.497	33.586	33.555	0.000
	Cameraman	33.513	33.458	33.499	33.396	0.001
	House	33.427	33.446	33.633	33.423	0.002
	Peppers	33.506	33.371	33.383	33.366	0.002
	Boat	33.373	33.306	33.509	33.503	0.002
	Clock	33.346	33.439	33.500	33.432	0.003
	Black	33.521	33.490	33.469	33.561	0.001
	White	33.431	33.547	33.438	33.464	0.002
	On average	33.462	33.444	33.502	33.463	0.002

and eight sets of modified plain images separately for analysis. To analyze the sensitivity of plaintext on a certain plain image, every pair of ciphered images obtained by encrypting two sets (original and modified images) are compared reciprocally. Then, average values are calculated for every pair of sets for comparison.

Here, the modification is made to only one pixel to get a modified image. Because the direction of pixel scanning in the encryption is left to right and top to bottom, the last pixels of original images, $p(255, 255, k)$ for $k = 1..K$, are chosen to modify for each sets. The chosen pixels are either added 1 to if their values are less than 255 or subtracted 1 from if their values are equal to 255. With the direction of pixel scanning, the modification of the last pixel makes less affect to other pixels in the first encryption round.

The sensitivity of plaintext is measured by means of $NPCR$ and $UACI$. The randomness test is also examined for the uniform distribution similar to that in Subsection 3.5.2. Tables 23, 24, 25, and 26 show the values of $NCPR$ and $UACI$ and its averages using the perturbed Logistic and Standard maps, respectively. In the first round of encryption, all randomness tests for both $NPCR$ and $UACI$ are not passed for all cases of chaotic maps. From the second round of encryption and beyond, all values of $NPCR$ and $UACI$ are passed the randomness tests. Importantly, all the average values of $NPCR$ and $UACI$ are also satisfied the condition to pass the randomness tests, i.e. $NPCR_{avg} > 99.5693$ and $UACI_{avg} \in (33.2824, 33.6447)$.

Table 22 Original and modified images for the sensitivity of plaintext

Sets of images	Images							
Original	Lena	Cameraman	House	Peppers	Boat	Clock	Black	White
Modified Set 1	Modified Lena	Cameraman	House	Peppers	Boat	Clock	Black	White
Modified Set 2	Lena	Modified Cameraman	House	Peppers	Boat	Clock	Black	White
Modified Set 3	Lena	Cameraman	Modified House	Peppers	Boat	Clock	Black	White
Modified Set 4	Lena	Cameraman	House	Modified Peppers	Boat	Clock	Black	White
Modified Set 5	Lena	Cameraman	House	Peppers	Modified Boat	Clock	Black	White
Modified Set 6	Lena	Cameraman	House	Peppers	Boat	Modified Clock	Black	White
Modified Set 7	Lena	Cameraman	House	Peppers	Boat	Clock	Modified Black	White
Modified Set 8	Lena	Cameraman	House	Peppers	Boat	Clock	Black	Modified White

Table 23 Sensitivity of the plaintext by means of $NPCR$ using the perturbed Logistic map

		$NPCR$ (%) for the sensitivity on							
N_e	Images	Lena	Cameramen	House	Peppers	Boat	Clock	Black	White
1	Lena	*94.800*	*42.320*	*95.139*	*94.650*	*85.585*	*0.009*	*82.217*	*95.183*
	Cameraman	*95.454*	*49.461*	*95.811*	*95.357*	*87.558*	*0.009*	*84.744*	*95.818*
	House	*94.279*	*36.421*	*94.666*	*94.098*	*84.074*	*0.009*	*80.406*	*94.704*
	Peppers	*94.409*	*38.445*	*94.785*	*94.324*	*84.575*	*0.008*	*81.049*	*94.839*
	Boat	*95.474*	*49.672*	*95.874*	*95.396*	*87.607*	*0.009*	*84.686*	*95.891*
	Clock	*97.044*	*58.965*	*97.429*	*96.938*	*92.409*	*0.009*	*89.560*	*97.559*
	Black	*94.704*	*42.332*	*95.052*	*94.553*	*85.484*	*0.009*	*82.181*	*95.097*
	White	*94.566*	*41.788*	*94.832*	*94.493*	*85.170*	*0.009*	*82.039*	*95.001*
	On average	*95.091*	*44.926*	*95.449*	*94.976*	*86.558*	*0.009*	*83.360*	*95.512*
2	Lena	99.612	99.611	99.591	99.623	99.573	99.596	99.606	99.590
	Cameraman	99.611	99.612	99.622	99.620	99.595	99.606	99.643	99.585
	House	99.619	99.615	99.593	99.608	99.574	99.641	99.655	99.617
	Peppers	99.623	99.615	99.661	99.620	99.606	99.605	99.594	99.597
	Boat	99.615	99.620	99.588	99.608	99.612	99.590	99.648	99.667
	Clock	99.577	99.629	99.608	99.606	99.628	99.588	99.661	99.594
	Black	99.648	99.586	99.673	99.637	99.666	99.603	99.603	99.632
	White	99.632	99.623	99.577	99.596	99.614	99.641	99.634	99.588
	On average	99.617	99.614	99.614	99.615	99.609	99.609	99.631	99.609
3	Lena	99.600	99.619	99.617	99.597	99.573	99.628	99.614	99.612
	Cameraman	99.593	99.580	99.628	99.579	99.617	99.591	99.605	99.600
	House	99.652	99.619	99.594	99.608	99.611	99.606	99.596	99.605
	Peppers	99.590	99.583	99.629	99.583	99.640	99.629	99.609	99.597
	Boat	99.609	99.628	99.617	99.596	99.600	99.591	99.623	99.632
	Clock	99.608	99.637	99.603	99.600	99.571	99.597	99.588	99.609
	Black	99.599	99.629	99.628	99.596	99.591	99.673	99.606	99.637
	White	99.652	99.590	99.626	99.620	99.605	99.670	99.596	99.663
	On average	99.613	99.611	99.618	99.597	99.601	99.623	99.605	99.619
4	Lena	99.604	99.596	99.623	99.654	99.590	99.631	99.599	99.605
	Cameraman	99.596	99.620	99.617	99.605	99.600	99.599	99.626	99.596
	House	99.579	99.620	99.619	99.609	99.628	99.632	99.608	99.615
	Peppers	99.617	99.596	99.577	99.614	99.615	99.625	99.628	99.579
	Boat	99.629	99.588	99.626	99.570	99.631	99.657	99.639	99.620
	Clock	99.593	99.591	99.588	99.577	99.588	99.635	99.596	99.638
	Black	99.609	99.651	99.574	99.634	99.603	99.577	99.593	99.603
	White	99.615	99.626	99.599	99.609	99.582	99.620	99.620	99.583
	On average	99.605	99.611	99.603	99.609	99.605	99.622	99.614	99.605
5	Lena	99.643	99.648	99.614	99.606	99.606	99.574	99.577	99.620
	Cameraman	99.582	99.603	99.614	99.596	99.600	99.571	99.635	99.591
	House	99.628	99.585	99.583	99.602	99.600	99.582	99.599	99.609
	Peppers	99.605	99.649	99.617	99.634	99.597	99.609	99.573	99.580
	Boat	99.635	99.631	99.628	99.605	99.640	99.597	99.629	99.588

Table 23 continued

		$NPCR$ (%) for the sensitivity on							
N_e	Images	Lena	Cameramen	House	Peppers	Boat	Clock	Black	White
	Clock	99.591	99.632	99.617	99.634	99.628	99.620	99.605	99.606
	Black	99.652	99.672	99.590	99.605	99.596	99.602	99.635	99.588
	White	99.636	99.603	99.656	99.612	99.629	99.629	99.623	99.660
	On average	99.622	99.628	99.615	99.612	99.612	99.598	99.610	99.605

Table 24 Sensitivity of the plaintext by means of $UACI$ using the perturbed Logistic map

		$UACI$ (%) for the sensitivity on							
N_e	Images	Lena	Cameramen	House	Peppers	Boat	Clock	Black	White
1	Lena	*31.908*	*14.224*	*31.918*	*31.781*	*28.779*	*0.003*	*27.661*	*32.028*
	Cameraman	*31.855*	*16.258*	*31.920*	*31.510*	*28.999*	*0.004*	*28.198*	*31.854*
	House	*31.689*	*12.105*	*31.763*	*31.448*	*28.203*	*0.003*	*26.968*	*31.681*
	Peppers	*31.620*	*12.817*	*31.874*	*31.552*	*28.409*	*0.004*	*27.064*	*31.941*
	Boat	*32.091*	*16.668*	*32.207*	*32.080*	*29.359*	*0.003*	*28.181*	*32.201*
	Clock	*32.659*	*19.689*	*32.604*	*32.475*	*31.037*	*0.003*	*29.978*	*32.784*
	Black	*31.823*	*14.212*	*31.869*	*31.646*	*28.815*	*0.002*	*27.614*	*31.966*
	White	*30.918*	*13.065*	*30.897*	*30.786*	*27.820*	*0.004*	*26.603*	*31.025*
	On average	*31.820*	*14.880*	*31.882*	*31.660*	*28.928*	*0.003*	*27.783*	*31.935*
2	Lena	33.458	33.333	33.418	33.604	33.508	33.378	33.471	33.513
	Cameraman	33.439	33.432	33.626	33.488	33.453	33.447	33.618	33.534
	House	33.550	33.570	33.498	33.412	33.495	33.424	33.596	33.521
	Peppers	33.384	33.410	33.485	33.565	33.623	33.561	33.458	33.482
	Boat	33.528	33.320	33.548	33.463	33.395	33.556	33.480	33.547
	Clock	33.327	33.415	33.381	33.356	33.386	33.505	33.456	33.463
	Black	33.469	33.482	33.387	33.407	33.468	33.429	33.454	33.466
	White	33.352	33.623	33.473	33.382	33.408	33.429	33.608	33.511
	On average	33.438	33.448	33.477	33.460	33.467	33.466	33.518	33.505
3	Lena	33.442	33.376	33.630	33.516	33.630	33.580	33.369	33.610
	Cameraman	33.367	33.513	33.502	33.465	33.533	33.598	33.596	33.428
	House	33.432	33.376	33.396	33.514	33.476	33.401	33.403	33.395
	Peppers	33.591	33.613	33.318	33.587	33.401	33.470	33.483	33.539
	Boat	33.482	33.512	33.432	33.469	33.473	33.286	33.433	33.372
	Clock	33.460	33.572	33.360	33.580	33.424	33.351	33.513	33.417
	Black	33.525	33.598	33.424	33.339	33.480	33.430	33.495	33.364
	White	33.417	33.556	33.373	33.516	33.413	33.563	33.455	33.469
	On average	33.465	33.515	33.429	33.498	33.479	33.460	33.468	33.449
4	Lena	33.518	33.417	33.570	33.597	33.617	33.434	33.496	33.601
	Cameraman	33.396	33.467	33.516	33.535	33.453	33.482	33.490	33.302
	House	33.509	33.541	33.418	33.571	33.468	33.465	33.394	33.507
	Peppers	33.420	33.602	33.472	33.368	33.497	33.334	33.368	33.414
	Boat	33.488	33.446	33.391	33.536	33.394	33.463	33.481	33.577

Table 24 continued

N_e	Images	$UACI$ (%) for the sensitivity on Lena	Cameramen	House	Peppers	Boat	Clock	Black	White
	Clock	33.501	33.499	33.622	33.349	33.525	33.481	33.334	33.518
	Black	33.446	33.400	33.471	33.535	33.529	33.491	33.561	33.475
	White	33.489	33.451	33.458	33.514	33.523	33.361	33.408	33.536
	On average	33.471	33.478	33.490	33.501	33.501	33.439	33.442	33.491
5	Lena	33.472	33.524	33.602	33.588	33.424	33.495	33.314	33.377
	Cameraman	33.384	33.452	33.508	33.367	33.444	33.423	33.465	33.550
	House	33.524	33.430	33.482	33.393	33.478	33.535	33.317	33.356
	Peppers	33.506	33.487	33.500	33.494	33.589	33.516	33.380	33.377
	Boat	33.609	33.389	33.466	33.487	33.445	33.505	33.564	33.446
	Clock	33.490	33.343	33.399	33.365	33.354	33.502	33.419	33.354
	Black	33.526	33.463	33.305	33.436	33.339	33.478	33.478	33.493
	White	33.576	33.441	33.533	33.604	33.426	33.409	33.611	33.505
	On average	33.511	33.441	33.474	33.467	33.437	33.483	33.444	33.432

Table 25 Sensitivity of the plaintext by means of $NPCR$ using the perturbed Standard map

N_e	Images	$NPCR$ (%) for the sensitivity on Lena	Cameramen	House	Peppers	Boat	Clock	Black	White
1	Lena	*88.344*	*93.297*	*68.153*	*94.684*	*5.080*	*98.892*	*0.009*	*0.009*
	Cameraman	*91.721*	*95.140*	*76.227*	*96.167*	*5.150*	*99.176*	*0.009*	*0.009*
	House	*88.341*	*93.253*	*68.074*	*94.688*	*5.034*	*98.970*	*0.009*	*0.009*
	Peppers	*86.403*	*92.166*	*63.039*	*93.866*	*4.741*	*98.817*	*0.009*	*0.009*
	Boat	*91.676*	*95.181*	*75.928*	*96.167*	*5.135*	*99.121*	*0.008*	*0.008*
	Clock	*91.632*	*95.171*	*76.118*	*96.173*	*5.116*	*99.132*	*0.009*	*0.009*
	Black	*88.194*	*93.265*	*67.233*	*94.664*	*4.999*	*98.956*	*0.009*	*0.008*
	White	*91.446*	*94.971*	*75.986*	*95.966*	*5.124*	*98.894*	*0.009*	*0.009*
	On average	*89.720*	*94.056*	*71.345*	*95.297*	*5.047*	*98.995*	*0.009*	*0.009*
2	Lena	99.591	99.611	99.605	99.611	99.588	99.612	99.608	99.614
	Cameraman	99.588	99.597	99.615	99.643	99.631	99.652	99.617	99.600
	House	99.619	99.596	99.574	99.619	99.590	99.622	99.577	99.629
	Peppers	99.576	99.642	99.626	99.631	99.602	99.629	99.602	99.590
	Boat	99.611	99.608	99.608	99.597	99.660	99.654	99.582	99.615
	Clock	99.608	99.617	99.612	99.625	99.623	99.629	99.593	99.608
	Black	99.663	99.597	99.643	99.612	99.611	99.611	99.628	99.628
	White	99.629	99.628	99.580	99.632	99.600	99.614	99.597	99.649
	On average	99.611	99.612	99.608	99.621	99.613	99.628	99.601	99.617
3	Lena	99.600	99.576	99.631	99.594	99.628	99.615	99.603	99.660
	Cameraman	99.603	99.612	99.594	99.605	99.570	99.588	99.612	99.580
	House	99.588	99.615	99.680	99.637	99.611	99.612	99.615	99.625
	Peppers	99.638	99.579	99.596	99.626	99.617	99.615	99.603	99.628
	Boat	99.605	99.667	99.582	99.590	99.597	99.631	99.662	99.643

Table 25 continued

		$NPCR$ (%) for the sensitivity on							
N_e	Images	Lena	Cameramen	House	Peppers	Boat	Clock	Black	White
	Clock	99.646	99.606	99.652	99.622	99.612	99.609	99.625	99.643
	Black	99.657	99.629	99.640	99.626	99.606	99.609	99.620	99.619
	White	99.590	99.614	99.626	99.619	99.582	99.609	99.629	99.614
	On average	99.616	99.612	99.625	99.615	99.603	99.611	99.621	99.627
4	Lena	99.591	99.664	99.673	99.664	99.612	99.663	99.599	99.590
	Cameraman	99.600	99.590	99.596	99.605	99.622	99.596	99.599	99.606
	House	99.612	99.603	99.652	99.609	99.665	99.591	99.579	99.602
	Peppers	99.619	99.602	99.583	99.571	99.634	99.635	99.626	99.606
	Boat	99.571	99.614	99.609	99.580	99.605	99.628	99.638	99.593
	Clock	99.649	99.603	99.638	99.594	99.611	99.664	99.628	99.617
	Black	99.612	99.612	99.611	99.617	99.628	99.614	99.634	99.620
	White	99.614	99.574	99.570	99.628	99.669	99.608	99.617	99.608
	On average	99.609	99.608	99.617	99.609	99.631	99.625	99.615	99.605
5	Lena	99.606	99.603	99.582	99.660	99.580	99.626	99.640	99.623
	Cameraman	99.626	99.594	99.579	99.641	99.629	99.591	99.625	99.631
	House	99.634	99.570	99.620	99.625	99.632	99.612	99.594	99.590
	Peppers	99.596	99.634	99.580	99.640	99.662	99.615	99.588	99.612
	Boat	99.608	99.588	99.620	99.632	99.637	99.590	99.594	99.594
	Clock	99.590	99.605	99.648	99.586	99.620	99.582	99.664	99.620
	Black	99.600	99.579	99.588	99.586	99.668	99.588	99.603	99.594
	White	99.628	99.635	99.614	99.622	99.590	99.573	99.615	99.603
	On average	99.611	99.601	99.604	99.624	99.627	99.597	99.615	99.608

Table 26 Sensitivity of the plaintext by means of $UACI$ using the perturbed Standard map

		$UACI$ (%) for the sensitivity on							
N_e	Images	Lena	Cameramen	House	Peppers	Boat	Clock	Black	White
1	Lena	*29.782*	*31.267*	*23.008*	*31.703*	*1.703*	*33.174*	*0.003*	*0.005*
	Cameraman	*30.784*	*32.085*	*25.667*	*32.356*	*1.750*	33.442	*0.003*	*0.003*
	House	*29.590*	*31.301*	*22.861*	*31.799*	*1.720*	*33.141*	*0.003*	*0.003*
	Peppers	*29.077*	*30.906*	*21.199*	*31.456*	*1.622*	*33.237*	*0.002*	*0.003*
	Boat	*30.529*	*31.743*	*25.392*	*32.128*	*1.739*	*33.201*	*0.003*	*0.002*
	Clock	*30.907*	*31.735*	*25.454*	*32.153*	*1.718*	*33.183*	*0.003*	*0.004*
	Black	*29.691*	*31.297*	*22.578*	*31.802*	*1.688*	33.318	*0.003*	*0.003*
	White	*29.972*	*31.146*	*24.577*	*31.236*	*1.595*	*32.360*	*0.003*	*0.001*
	On average	*30.042*	*31.435*	*23.842*	*31.829*	*1.692*	*33.132*	*0.003*	*0.003*
2	Lena	33.500	33.576	33.408	33.581	33.334	33.380	33.393	33.379
	Cameraman	33.514	33.350	33.471	33.460	33.378	33.506	33.468	33.512
	House	33.513	33.292	33.523	33.442	33.496	33.445	33.528	33.435
	Peppers	33.414	33.473	33.391	33.574	33.372	33.403	33.435	33.523
	Boat	33.511	33.623	33.603	33.505	33.520	33.466	33.558	33.502

Table 26 continued

		$UACI$ (%) for the sensitivity on							
N_e	Images	Lena	Cameramen	House	Peppers	Boat	Clock	Black	White
	Clock	33.501	33.520	33.387	33.536	33.536	33.586	33.375	33.407
	Black	33.304	33.294	33.491	33.375	33.524	33.363	33.323	33.551
	White	33.530	33.434	33.564	33.484	33.458	33.451	33.566	33.537
	On average	33.473	33.445	33.480	33.495	33.452	33.450	33.456	33.481
3	Lena	33.568	33.535	33.578	33.629	33.637	33.587	33.563	33.555
	Cameraman	33.480	33.389	33.508	33.603	33.440	33.520	33.622	33.496
	House	33.348	33.518	33.414	33.580	33.530	33.482	33.544	33.500
	Peppers	33.299	33.340	33.337	33.546	33.538	33.412	33.307	33.397
	Boat	33.562	33.391	33.344	33.537	33.405	33.478	33.336	33.497
	Clock	33.431	33.573	33.386	33.484	33.511	33.381	33.520	33.504
	Black	33.415	33.560	33.468	33.626	33.511	33.378	33.429	33.482
	White	33.354	33.302	33.404	33.542	33.330	33.553	33.521	33.423
	On average	33.432	33.451	33.430	33.568	33.488	33.474	33.480	33.482
4	Lena	33.471	33.483	33.485	33.493	33.532	33.527	33.519	33.438
	Cameraman	33.644	33.427	33.543	33.400	33.398	33.461	33.420	33.573
	House	33.382	33.610	33.551	33.452	33.518	33.493	33.581	33.618
	Peppers	33.494	33.643	33.495	33.530	33.453	33.448	33.408	33.512
	Boat	33.608	33.483	33.418	33.528	33.562	33.357	33.550	33.521
	Clock	33.550	33.379	33.345	33.435	33.546	33.487	33.410	33.453
	Black	33.428	33.523	33.481	33.535	33.338	33.377	33.462	33.461
	White	33.388	33.545	33.420	33.570	33.493	33.629	33.290	33.609
	On average	33.496	33.512	33.467	33.493	33.480	33.472	33.455	33.523
5	Lena	33.485	33.482	33.418	33.494	33.512	33.491	33.591	33.572
	Cameraman	33.371	33.549	33.490	33.451	33.457	33.588	33.465	33.391
	House	33.445	33.392	33.420	33.341	33.487	33.549	33.561	33.505
	Peppers	33.522	33.528	33.424	33.479	33.321	33.520	33.450	33.455
	Boat	33.512	33.522	33.621	33.595	33.534	33.457	33.493	33.569
	Clock	33.608	33.425	33.494	33.499	33.404	33.594	33.430	33.553
	Black	33.567	33.453	33.370	33.414	33.511	33.443	33.435	33.393
	White	33.475	33.461	33.472	33.545	33.565	33.399	33.386	33.465
	On average	33.498	33.477	33.464	33.477	33.474	33.505	33.476	33.488

In summary, the sensitivity of plaintext is significant for the perturbed Logistic and Standard maps for $N_e \geq 2$. In fact, that must be higher in the first round of encryption if the modified pixels are as close as the first pixels of images, i.e. $p(1, 1, k)$ for $k = 1..K$.

3.6 Comparison with existing methods of MIE

3.6.1 The statistical results and security

In this section, the results obtained from the exemplar simulation using the proposed structure are compared with those of existing, recent methods of MIE, in terms of statistical analysis and security analysis. Here, the values obtained on the exemplar simulation at $N_e \geq 2$ are used the comparison as shown in Table 27. In fact, the effectiveness of the proposed structure can be seen via the comparison that most of the results from the examples using the proposed structure are almost comparable to those from the existing methods, except for the space of secret key. Even though the comparable result is obtained in the exemplar simulation, the cryptosystem requires number of encryption rounds $N_e \geq 3$ because of requirements of statistical analyses in Subsection 3.4. In practice, the space of secret key of chaos-based cryptosystem in these examples can be easily extended by increasing the number of bits representing for the fractional part in the fixed-point number for the value of state variables and control parameters. More information about the space of secret key and efficiency of the proposed structure will be discussed in Section 4.

3.6.2 The computational cost

It is noted that all the existing methods of MIE were designed for a single round of encryption. For a fair comparison, the required computational resource of the proposed structure is considered for an individual round of encryption. Here, the computational resource is in terms of the number of operations as well as the amount of memory.

Table 28 presents the required computational resource of the proposed structure and existing methods of MIE. Here, only the significant number of operations and the significant amount of memory in byte that are dependent on the size of inputs are retained. Overall, the required amounts of computational resource for the proposed structure are less than most of existing methods of MIE. In fact, the number of operations for the chaotic iterations in the proposed structure is greater than that of [40] and [39], but the number of operations for others is significantly less than those in the mentioned works. Besides, the proposed structure requires the memory space almost equivalent to that of existing methods of MIE. In other words, the proposed structure with lower number of operations offers higher speed and more efficient in compared with the existing methods of MIE.

4 Discussion and Conclusion

The proposed structure can be employed to design chaos-based cryptosystems of MIE. The proposed structure provides the structural and cryptographic advantages over the existing methods. For the structural advantages, the proposed model accepts any model of chaotic map, and it requires only a single chaotic map. Besides, the permutation and diffusion are integrated in processing K pixels at the same time, and the perturbation to the chaotic map and data manipulation are performed by the XOR operation. As a consequence, a crypstosystem employing the proposed structure requires less computational resource in compared with the existing methods of MIE.

For the cryptographic advantages, the proposed structure also provides the elastic key space and the content-dependent encryption. Firstly, because the values of state variables and control parameters are represented in the format of fixed-point number, so the key space can be extended by increasing the number of bits in the fractional portions. The key space can also be enlarged by increasing the number of pixels kC_0^- and kP_0^+ in the secret key. In that case, more than one neighboring pixel is used in the diffusion process in (9) instead of single neighbor pixel in the exemplar simulation. Secondly, the perturbation amounts to the chaotic map is constructed with the involvement of the image content by means of E as

Table 27 Comparison with other methods of MIE

Method	Statistical analysis			Security analysis		
	Quantitative histogram analysis	Information Entropy	Correlation of adjacent pixels	Space of secret key	Sensitivity of secret key	Sensitivity of plaintext (or differential attack)
Proposed method (using Logistic map)	all passed the χ^2-test with $N_e \geq 2$	≥ 7.9971	H:-0.0003 V: 0.0009 D:-0.002	$\approx 7.8464 * 10^{56}$ (2^{189})	NPCR: [99.601, 99.622] UACI: [33.433,33.539]	NPCR: [99.593, 99.631] UACI: [33.426,33.522]
Proposed method (using Standard map)	all passed the χ^2-testwith $N_e \geq 2$	≥ 7.9970	H: -0.0011 V: 0.0004 D:-0.001	$\approx 1.7254 * 10^{69}$ (2^{230})	NPCR: [99.601, 99.628] UACI: [33.388,33.506]	NPCR: [99.596, 99.628] UACI: [33.425,33.581]
[5]	V/D	7.999	H= -0.00265 V=0.00704 D=0.00323	1.3408×10^{154} (2^{512})	V/D	NPCR: 99.608 UACI: 33.454
[48]	V/D	7.9984	H=0.00719 V=-0.00020 D=0.00056	3.2768×10^{73} (2^{245})	N/A	NPCR: 99.636 UACI: 33.393
[36]	V/D	7.9993	H=0.00187 V=0.00030 D=0.00031	10^{165} ($\approx 2^{548}$)	Cdr: [99.591,99.648] NPCR: N/A UACI: N/A	NPCR: 99.638 UACI: 33.299
[37]	V/D	7.9993	H=-0.00058 V=-0.00029 D=-0.00031	3.4×10^{128} ($2^{427} bit$)	NPCR: [93.750, 99.668] UACI: [29.015,36.476]	NPCR: 99.616 UACI: 33.443
[39]	V/D	7.9978	H=-0.00124 V=-0.00151 D=0.0005	3.4×10^{128}(2^{427})	NPCR: 99.621 UACI: 33.479	NPCR: 99.617 UACI: 33.459
[11]	V/D	7.996	H= 0.00133 V=0.00253 D=0.00026	1.1579×10^{77} (2^{256})	V/D	NPCR: 99.610 UACI: 33.562

Table 27 continued

Method	Statistical analysis			Security analysis		
	Quantitative histogram analysis	Information Entropy	Correlation of adjacent pixels	Space of secret key	Sensitivity of secret key	Sensitivity of plaintext (or differential attack)
[40]	V/D	7.9997	H= -0.0036 V=0.0016 D=0.0058	1.34×10^{154} ($\approx 2^{512}$)	V/D	NPCR: 99.615 UACI: 33.47
[46]	V/D	7.9992	H=-0.02775 V=0.03057 D=0.00385	7.37×10^{134} ($\approx 2^{448}$)	V/D	N/A
[56]	V/D	7.9995	H=-0.00364 V=0.00262 D=0.00124	10^{195} ($\approx 2^{648}$)	V/D	NPCR: 99.613 UACI: 33.466
[58]	V/D	7.9992	H= 0.00092 V=-0.00102 D=0.000525	$10^{135} \times (8k)!+$ ($\approx 2^{248} \times (8k)!$)	N/A	NPCR: 99.5 UACI: 33.48
[59]	V/D	7.8102	N/A	10^{56} ($\approx 2^{186}$)	N/A	N/A
[61]	V/D	7.9993	H= -0.00089 V=0.00189 D=0.00071	10^{160} ($\approx 2^{232}$)	N/A	NPCR: 99.660 UACI: 33.650
[60]	V/D	7.9993	H=0.00030 V=0.00263 D=0.00412	10^{60} ($\approx 2^{199}$)	Cdr = 99.778 NPCR: N/A UACI: N/A	NPCR: 99.620 UACI: 33.500
[57]	V/D	7.9993	H= -0.00092 V=-0.0005 D=0.00035	10^{56} ($\approx 2^{186}$)	V/D	NPCR: 99.62 UACI: 33.46
[63]	V/D	7.9941	H=0.00137 V=0.00117 D=-0.00370	$1.96 \times 10^{56}(2^{1}87)$ ($\approx 2^{187}$)	V/D	N/A

Table 28 Comparison of the computational cost with other methods of MIE

Methods	Number of operations for		Memory required (in byte)
	Chaotic iterations	Others	
The proposed structure	RMN	$6K$	$34K + KMN + 2Klog_2(KMN)$
[38]	$3\left(M_1 + N_1\right) + 2max\left(M_1, N_1\right)$	$8M_1N_1 + 3max\left(M_1, N_1\right)\left(max\left(M_1, N_1\right) - 1\right)$	$M_1N_1 + M_1 + N_1 + 8max\left(M_1, N_1\right) + max\left(M_1, N_1\right)\frac{M_1log_2M_1+N_1log_2N_1}{8}$
[40]	$4M$	$24M + 9MN + 3KMN + \frac{3M(M-1)}{2}$	$32M + 4KMN$
[56]	$4K + M + 3MN$	$M + 2N + 13KMN + K\left(13 - N\right) + \frac{3MN(MN-1)}{2}$	$5K\left(8 + log_2 4K\right) + MN\left(8 + K + log_2 MN\right) + 4M + max\left(M, N\right)$
[58]	$5MN$	$\left(9 + 12K + 192K^2\right)MN + 4\left(MN\right)^2$	$MN\left(22 + K + \frac{log_2(MN)}{8}\right) + Klog_2\left(8K\right)$
[60]	$KM_yN_y + MN$	$\left(3 + K\right)MN + \frac{3KM_yN_y\left(KM_yN_y-1\right)}{2}$	$MN(5 + K) + 4KM_yN_y + \frac{KM_yN_y*log_2\left(KM_yN_y\right)}{8}$
[61]	$M + N + MN$	$4M + 4N + 8MN + 3K\left(M + N\right) + 5KMN$	$M + N + K + MN(1 + K)$
[57]	$4MN + 4KMN$	$2MN + 6\left(KMN\right)^2 + 60KMN$	$9MN + KMN\left(54 + \frac{log_2(KMN)}{2}\right)$

It is noted that:
$KMN = M_1N_1$;
$M_y = M/s, N_y = N/s, s < min(M, N), mod(M, s) = 0, mod(N, s) = 0$

in (1)-(3), in other words, the encryption is dependent on the image content. The benefit of the content-dependent encryption is that it can resist from the types of chosen-plaintext and known-plaintext attacks. In addition, the diffusion effect is obtained not only via (9) directly but also via the dynamics of chaotic map indirectly by the perturbation.

In conclusion, the example and simulation results shows the feasibility and effectiveness of the proposed structure for fast and efficient cryptosystems. In the future work, specific cryptosystems using above-mentioned chaotic maps will be implemented in hardware for applications.

Declarations

Conflicts of interest The authors declare that they have no conflict of interest.

References

1. Alvarez G, Li S (2006) Some basic cryptographic requirements for chaos-based cryptosystems. Journal of Bifurcation and Chaos 16(08):2129–2151
2. Alvarez G, Amigó JM, Arroyo D, Li S (2011) Lessons Learnt from the Cryptanalysis of Chaos-Based Ciphers, Springer Berlin Heidelberg, Berlin, Heidelberg pp 257–295
3. Arroyo D, Diaz J, Rodriguez F (2013) Cryptanalysis of a one round chaos-based substitution permutation network. Signal Processing 93(5):1358–1364
4. Ayoup AM, Hussein AH, Attia MAA (2016) Efficient selective image encryption. Multimedia Tools and Applications 75(24):17171–17186
5. Banik A, Shamsi Z, Laiphrakpam DS (2019) An encryption scheme for securing multiple medical images. J Inf Sec Appl 49:102398
6. Baptista MS (1998) Cryptography with chaos. Physics Letters A 240(1):50–54
7. Bhatnagar G, Wu QMJ (2012) Selective image encryption based on pixels of interest and singular value decomposition. Digital Signal Processing 22(4):648–663
8. Chai X, Gan Z, Zhang M (2017) A fast chaos-based image encryption scheme with a novel plain image-related swapping block permutation and block diffusion. Multimed Tools Appl vol 76 p 15561–15585
9. Cheng S, Wang L, Ao N, Han Q (2020) A selective video encryption scheme based on coding characteristics. Symmetry 12(3):332
10. Cheng P, Yang H, Wei P (2015) A fast image encryption algorithm based on chaotic map and lookup table. Nonlinear Dynnamics vol 79 p 2121–2131
11. Deepak M, Ashwin V, Amutha R (2014) A new multistage multiple image encryption using a combination of chaotic block cipher and iterative fractional Fourier transform. In: 2014 First International Conference on Networks Soft Computing (ICNSC2014), pp 360–364
12. Diab H (2018) An efficient chaotic image cryptosystem based on simultaneous permutation and diffusion operations. IEEE Access vol 6 p 42227–42244
13. Enayatifar R, Guimarães FG, Siarry P (2019) Index-based permutation-diffusion in multiple-image encryption using DNA sequence. Optics and Lasers in Engineering vol 115 p 131–140
14. Farajallah M, Assad SE, Deforges O (2016) Fast and secure chaos-based cryptosystem for images. International Journal of Bifurcation and Chaos 26(02):1650021
15. Fouda JAE, Effa JY, Sabat SL, Ali M (2014) A fast chaotic block cipher for image encryption. Communications in Nonlinear Science and Numerical Simulation 19(3):578–588
16. Fridrich J (1998) Symmetric ciphers based on two-dimensional chaotic maps. International Journal of Bifurcation and Chaos 08(06):1259–1284
17. Gayathri J, Subashini S (2018) A spatiotemporal chaotic image encryption scheme based on self adaptive model and dynamic keystream fetching technique. Multimed Tools Appl vol 77 p 24751–24787
18. Hanis S, Amutha R (2019) A fast double-keyed authenticated image encryption scheme using an improved chaotic map and a butterfly-like structure. Nonlinear Dynamics vol 95 p 421–432
19. Hoang TM (2021) Perturbed chaotic map with varying number of iterations and application in image encryption. In: 2020 IEEE Eighth International Conference on Communications and Electronics (ICCE), pp 413–418

20. Hoang TM, Thanh HX (2018) Cryptanalysis and security improvement for a symmetric color image encryption algorithm. Optik vol 155 p 366–383
21. Hoang TM, Assad SE (2020) Novel models of image permutation and diffusion based on perturbed digital chaos. Entropy 22(5):548
22. Hosny KM, Kamal ST, Darwish MM (2022) Novel encryption for color images using fractional-order hyperchaotic system. J of Ambient Intell Humanized Comput 13(2):973–988
23. Jahangir S, Shah T (2021) A novel multiple color image encryption scheme based on algebra $m(2, f2[u]/<u^8>)$ and chaotic map. J Inf Sec Appl vol 59 p 102831
24. Jx Chen, Zhu Zl FuC, Lb Zhang, Zhang Y (2015) An efficient image encryption scheme using lookup table-based confusion and diffusion. Nonlinear Dynamics 81(3):1151–1166
25. Karawia A (2018) Encryption algorithm of multiple-image using mixed image elements and two dimensional chaotic economic map. Entropy vol 20 p 801
26. Khan J, Ahmad J (2019) Chaos based efficient selective image encryption. Multidimensional Systems and Signal Processing vol 30 p 943–961
27. Kocarev L (2001) Chaos-based cryptography: A brief overview. IEEE Circuits and Systems Magazine 1(3):6–21
28. Kocarev L, Lian S (2011) Chaos-based Cryptography. Springer
29. Kulsoom A, Xiao D, ur Rehman A, Abbas SA (2016) An efficient and noise resistive selective image encryption scheme for gray images based on chaotic maps and DNA complementary rules. Multimedia Tools and Applications 75(1):1–23
30. Kumar M, Gupta P (2021) A new medical image encryption algorithm based on the 1D Logistic map associated with pseudo-random numbers. Multimedia Tools and Applications 80(12):18941–18967
31. Kumar M, Saxena A, Vuppala SS (2020) A Survey on Chaos Based Image Encryption Techniques, Springer International Publishing, Cham pp 1–26
32. Li X, Meng X, Wang Y, Yang X, Yin Y, Peng X, He W, Dong G, Chen H (2017) Secret shared multiple-image encryption based on row scanning compressive ghost imaging and phase retrieval in the Fresnel domain. Optics and Lasers in Engineering vol 96 p 7–16
33. Liu L, Liu B, Hu H, Miao S (2018) Reducing the dynamical degradation by bi-coupling digital chaotic maps. International Journal of Bifurcation and Chaos 28(05):1850059
34. Liu W, Sun K, Zhu C (2016) A fast image encryption algorithm based on chaotic map. Optics and Lasers in Engineering vol 84 p 26–36
35. Lorenz EN (1963) Deterministic nonperiodic flow. J Atmos Sci 20(2):130–141
36. Malik DS, Shah T (2020) Color multiple image encryption scheme based on 3D-chaotic maps. Mathematics and Computers in Simulation vol 178 p 646–666
37. Patro KAK, Acharya B (2018) Secure multi-level permutation operation based multiple colour image encryption. J Inf Sec Appl vol 40 p 111–133
38. Patro KAK, Soni A, Netam PK, Acharya B (2020) Multiple grayscale image encryption using cross-coupled chaotic maps. J Inf Sec Appl vol 52 p 102470
39. Patro KAK, Acharya B (2020) A novel multi-dimensional multiple image encryption technique. Multimedia Tools and Applications 79(19):12959–12994
40. Sahasrabuddhe A, Laiphrakpam DS (2020) Multiple images encryption based on 3D scrambling and hyper-chaotic system. Inf Sci
41. Shen Q, Liu W (2017) A novel digital image encryption algorithm based on orbit variation of phase diagram. Int J of Bifurcation Chaos 27(13):1750204
42. Situ G, Zhang J (2006) Position multiplexing for multiple-image encryption. J Opt A Pure Appl Opt 8(5):391–397
43. Som S, Kotal A, Mitra A, Palit S, Chaudhuri BB (2014) A chaos based partial image encryption scheme. In: 2014 2nd International Conference on Business and Information Management (ICBIM), pp 58–63
44. Strogatz S (2015) Nonlinear Dynamics and Chaos?: with Applications to Physics, Biology, Chemistry, and Engineering. Westview Press
45. Sui LS, Duan KK, Liang JL, Zhang ZQ, Meng HN (2014) Asymmetric multiple-image encryption based on coupled logistic maps in fractional Fourier transform domain. Opt Lasers Eng vol 62
46. Tang Z, Song J, Zhang X, Sun R (2016) Multiple-image encryption with bit-plane decomposition and chaotic maps. Opt Lasers Eng vol 80 p 1–11
47. Tao S, Ruli W, Yixun Y (1998) Perturbance-based algorithm to expand cycle length of chaotic key stream. Electronics Letters 34(1):873–874
48. ul Haq T, Shah T, (2020) Algebra-chaos amalgam and DNA transform based multiple digital image encryption. J Inf Sec Appl vol 54 p 102592
49. Wang X, Zhao D (2011) Multiple-image encryption based on nonlinear amplitude-truncation and phase-truncation in Fourier domain. Opt Commun 284(1):148–152

50. Wang Y, Wong KW, Liao X, Chen G (2011) A new chaos-based fast image encryption algorithm. Applied Soft Computing 11(1):514–522
51. Wang Y, Quan C, Tay C (2014) Nonlinear multiple-image encryption based on mixture retrieval algorithm in fresnel domain. Opt Commun vol 330 p 91–98
52. Wu Y, Noonan JP, Agaian S (2011) NPCR and UACI randomness tests for image encryption. In: Cyber Journals: Multidisciplinary Journals in Science and Technology, Journal of Selected Areas in Telecommunications (JSAT) 1(2):31 – 38
53. Xiang T, Wong KW, Liao X (2007) Selective image encryption using a spatiotemporal chaotic system. Chaos vol 17 p 023115
54. Xiao JT, Wang Z, Zhang M, Liu Y, Xu H, Ma J (2015) An image encryption algorithm based on the perturbed high-dimensional chaotic map. Nonlinear Dyn 80(3):1493–1508
55. Yang Z, Liang D, Ding D, Hu Y (2021) Dynamic analysis of fractional-order memristive chaotic system with time delay and its application in color image encryption based on DNA encoding. The European Physical Journal Special Topics vol 230 p 1785–1803
56. Zarebnia M, Pakmanesh H, Parvaz R (2019) A fast multiple-image encryption algorithm based on hybrid chaotic systems for gray scale images. Optik vol 179 p 761–773
57. Zhang X, Wang X (2019) Multiple-image encryption algorithm based on DNA encoding and chaotic system. Multimedia Tools and Applications 78(6):7841–7869
58. Zhang L, Zhang X (2020) Multiple-image encryption algorithm based on bit planes and chaos. Multimedia Tools and Applications 79(29):20753–20771
59. Zhang X, Wang X (2017) Multiple-image encryption algorithm based on mixed image element and chaos. Comput Electr Eng vol 62 p 401–413
60. Zhang X, Wang X (2017) Multiple-image encryption algorithm based on mixed image element and permutation. Opt Lasers in Eng vol 92 p 6–16
61. Zhang X, Wang X (2018) Multiple-image encryption algorithm based on the 3D permutation model and chaotic system. Symmetry vol 10 p 660
62. Zhou NR, Huang LX, Gong LH, Zeng QW (2020) Novel quantum image compression and encryption algorithm based on QWT and 3D hyper-chaotic henon map. Quantum Inf Process 19(9):284
63. Zhou N, Yan X, Liang H (2018) Multi-image encryption scheme based on quantum 3D Arnold transform and scaled Zhongtang chaotic system. Quantum Inf Process vol 17 p 338

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.