International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Traitement du Signal
Vol. 41, No. 1, February, 2024, pp. 99-113
Journal homepage: http://iieta.org/journals/ts

Multiple-Image Encryption Using Sine Quadratic Polynomial Mapping and U-Shaped Scanning Techniques

Mehmet Demirtaş

Department of Electrical and Electronics Engineering, Necmettin Erbakan University, Konya 42140, Turkey

Corresponding Author Email: mdemirtas@erbakan.edu.tr

https://doi.org/10.18280/ts.410108

Received: 11 April 2023
Revised: 26 July 2023
Accepted: 6 November 2023
Available online: 29 February 2024

Keywords:
chaotic map, image encryption, multiple-image encryption (MIE), sine quadratic polynomial map (SQPM), U-shaped scanning

ABSTRACT

In the realm of digital image security, the multiple-image encryption (MIE) has garnered increasing attention due to the prevalent dissemination of digital imagery. Responding to this trend, an innovative encryption method has been developed, capable of securing an arbitrary number of images efficiently. This method is underpinned by the newly devised sine quadratic polynomial map (SQPM) and an original space-filling curve technique, termed U-shaped scanning. Extensive analysis, including 2D and 3D phase diagrams, Lyapunov exponents, bifurcation diagrams, and approximate entropy calculations, confirms the SQPM's chaotic properties over a broad spectrum of control parameters. The U-shaped scanning method, novel in its application, facilitates the traversal of every element in a 2D array, irrespective of its dimensions. This method is integral to the permutation phase of the encryption process, where it pre-scrambles input images, and it plays a pivotal role in the diffusion phase through the introduction of U-shaped diffusion. Comprehensive security assessments have been conducted, encompassing secret key analysis, histogram evaluation, correlation assessments, differential analysis, and information entropy measurements. Further scrutiny involves known-plaintext and chosen-plaintext attack resilience, along with visualizations of data loss and noise attack impacts, and execution time analysis across three sets of four images. The results of these security analyses affirm the efficacy of the proposed technique in encrypting multiple images, be they colored or grayscale. This work not only advances the field of image encryption but also introduces novel methodologies with broad applicability in digital image security.

1. INTRODUCTION

With the advancements in digital media technology, a significant surge in image sharing has been observed. Ensuring the security of these shared images is paramount. Image encryption stands out as a highly effective and popular method for providing the required protection [1]. Concurrently, the popularity of MIE algorithms is on the rise, attributed to the extensive sharing of multiple digital images [2]. The primary objective of MIE algorithms is to obscure multiple color or grayscale images from unauthorized access.

The majority of image encryption methodologies documented in the literature focus on securing single color or grayscale images [3]. However, the growing demand for MIE schemes has led to a notable increase in related publications [3-32]. MIE methods are categorized into those encrypting only grayscale images and those capable of encrypting both grayscale and color images. Zarebnia et al. [27] proposed a grayscale MIE algorithm, wherein multiple grayscale images of identical sizes are encrypted by subdividing each input image into subblocks, followed by scrambling and diffusing these subblocks using chaotic values generated by 2D Arnold's cat map and a combined chaotic system. Another approach for encrypting multiple grayscale images is outlined in the study [23], where the scrambling and diffusion stages utilize two cross-coupled piece-wise linear chaotic maps. A method employing the Henon map to generate chaotic parameters for three-dimensional bit scrambling with diffusion operation on input images is discussed in the study [4]. This technique is applicable for encrypting any desired number of grayscale images of the same size. The methods proposed in the studies [2, 3, 13, 17, 25, 26] are also limited to encrypting grayscale images of identical sizes. Similarly, the MIE algorithm presented in the study [24] is capable of encrypting multiple grayscale images of arbitrary sizes. A fundamental limitation in these studies is their applicability solely to grayscale input images. While the extension to color images is feasible by separately encrypting the R, G, and B channels, this approach significantly increases encryption time and computational complexity. Therefore, it is essential that the original method be designed to accommodate encryption of both grayscale and color images without additional complexity.

An additional issue noted in the literature concerning MIE algorithms pertains to the limitation in the number of images they can encrypt. Existing in the literature are double and triple image encryption methods, which fall under the category of MIE algorithms [28, 33-37]. However, these methods are restricted to encrypting only two or three images concurrently. Similarly, certain algorithms are constrained to encrypting a fixed number of images. For example, the MIE algorithm presented in the study [18] employs a cascade modulation chaotic system but is limited to encrypting only three grayscale images simultaneously. This limitation poses a problem in scenarios where a different number of images, such as five, need encryption, rendering the method inapplicable. Another study with a limitation on the number of images that can be encrypted is found in the study [5], where the method requires exactly twelve grayscale images to form three planes, each consisting of four grayscale images. Some MIE schemes amalgamate input images into a 3D cube structure for encryption [10, 21], which inherently limits the flexibility in the number of input images; for instance, creating a cube of size 256×256×256 requires exactly 64 input images of size 512×512. A truly versatile and useful MIE algorithm should possess the capability to encrypt any number of input images concurrently.

Evaluating the effectiveness and security of an MIE algorithm involves simulating specific evaluation metrics [38, 39]. A robust MIE scheme must exhibit resilience against various cryptanalytic attacks, including exhaustive search, statistical attacks, chosen/known-plaintext attacks, entropy attacks, histogram attacks, data-loss attacks, noise attacks, and more. A noticeable gap in the literature is the absence of comprehensive security analyses. For instance, in the studies [2, 17, 19, 20, 24, 30], neither data-loss nor noise analysis is conducted. Similarly, noise attacks are not considered in the studies [5, 23], and data-loss attacks are overlooked in [14]. The omission of both chosen-plaintext and known-plaintext attack analyses is evident in some MIE schemes [2, 11, 16, 17, 29]. Furthermore, the efficiency of hardware implementation, inferred from the total encryption/decryption time, is an essential metric. The execution time of the proposed method is notably absent in the studies [11, 14, 22]. To verify the quality and reliability of an MIE algorithm, all necessary evaluation measures must be thoroughly simulated and documented.

This study introduces significant advancements addressing the previously identified limitations in MIE algorithms. Firstly, an innovative MIE algorithm is proposed, capable of encrypting both color and grayscale images with the flexibility to select an arbitrary number of input images. The efficacy of this algorithm is validated through comprehensive simulation results for diverse image groups, demonstrating its robustness and efficiency. Secondly, this work introduces a novel 1D chaotic map, termed the SQPM. Despite its simplicity, SQPM effectively overcomes the challenges of discontinuity and limited chaotic intervals inherent in classical chaotic maps. It refines the existing sine map by generating continuous chaotic values for a variety of control parameters, thereby enhancing the security of cryptosystems. The control parameters and initial value of SQPM are intricately linked to the external keys, resulting in an expanded keyspace for the proposed MIE algorithm, which contributes significantly to the encryption process's security and robustness. To fortify against potential chosen and known-plaintext attacks, the secure hash algorithm is utilized for calculating SQPM's necessary parameters. Furthermore, the proposed MIE scheme encrypts different grayscale images or various channels of color images simultaneously, ensuring that the pixel values of each image influence the pixel values of all other images, effectively diminishing pixel correlation. An additional innovative aspect of this algorithm is the introduction of U-shaped scanning. This method enables the scanning of all elements in a 2D image of any size, initially applied in the scrambling of input images and subsequently in the diffusion phase. Lastly, the security of the proposed MIE algorithm is rigorously evaluated using essential measures. The experimental results affirm that the proposed MIE scheme is both effective and secure, capable of withstanding various cryptanalytic attacks.

The structure of this paper is as follows: Section 2 introduces SQPM. The SQPM-based MIE algorithm is detailed in Section 3. Section 4 presents the simulation results for the security analyses of the proposed MIE algorithm. The paper is concluded in Section 5.

2. THE PROPOSED CHAOTIC MAP

SQPM represents an innovative one-dimensional chaotic map, ingeniously amalgamating the traditional sine map with a quadratic polynomial. The mathematical representation of SQPM is given in Eq. (1):

$$x_{n+1} = \sin\left(\pi(e^{a+10} + b)(x_n^2 + x_n + c)\right) \quad (1)$$

where, $x_n \in (-1,1)$ denotes the initial value, $a, b, c \in [0, \infty)$ signify the control parameters, and e is Euler's number. As a classical member of the 1D chaotic maps family, the sine map is known for its ability to generate chaotic sequences efficiently, albeit with a low computational cost. However, its major limitation lies in having a single control parameter with an exceedingly narrow chaotic range. The introduction of SQPM marks a significant advancement by expanding the number of control parameters available for image encryption algorithms to three. This enhancement allows SQPM to produce chaotic sequences of substantial complexity while maintaining minimal computational power $\forall\, a, b, c \in [0, \infty)$.

2.1 2D and 3D phase diagrams

The 2D and 3D phase diagrams for the sine map and SQPM are depicted in Figure 1. Figures 1a) and 1d) clearly show that the sine map exhibits a parabolic trajectory, occupying only a limited region of the 2D plane and 3D space. In contrast, SQPM does not adhere to a specific trajectory. Instead, it extensively covers a significant portion of both the 2D plane and 3D space when iterated with varying control parameters, such as $a = b = c = 0$ or $a = 5$, $b = 3$, $c = 15$. This expansive coverage by SQPM illustrates its superior capability in generating random sequences with greater randomness compared to the sine map.

2.2 Bifurcation diagrams

A bifurcation diagram is a powerful tool for visualizing the evolution of a chaotic map's dynamic behavior as its control parameters change. Figure 2 showcases the bifurcation diagrams of both the sine map and SQPM. The sine map transitions into a chaotic regime when its control parameter lies within the range of [0.87, 1]. Notably, even within this chaotic regime, periodic windows are evident, as demonstrated in Figure 2a). The sine map only generates chaotic values across the range of (0,1) when its control parameter equals one. In stark contrast, SQPM exhibits a broader range of dynamic behaviors. As seen in Figures 2b), 2c), and 2d), SQPM consistently maps its input within the range of [-1,1] across all its control parameter values. Unlike the sine map, which is confined to a limited range of control parameters for chaotic behavior, SQPM demonstrates chaotic dynamics $\forall\, a, b, c \in [0, \infty)$ over an expanded parameter range. In essence, the inherent chaotic behavior of the sine map is substantially enhanced through the implementation of SQPM.

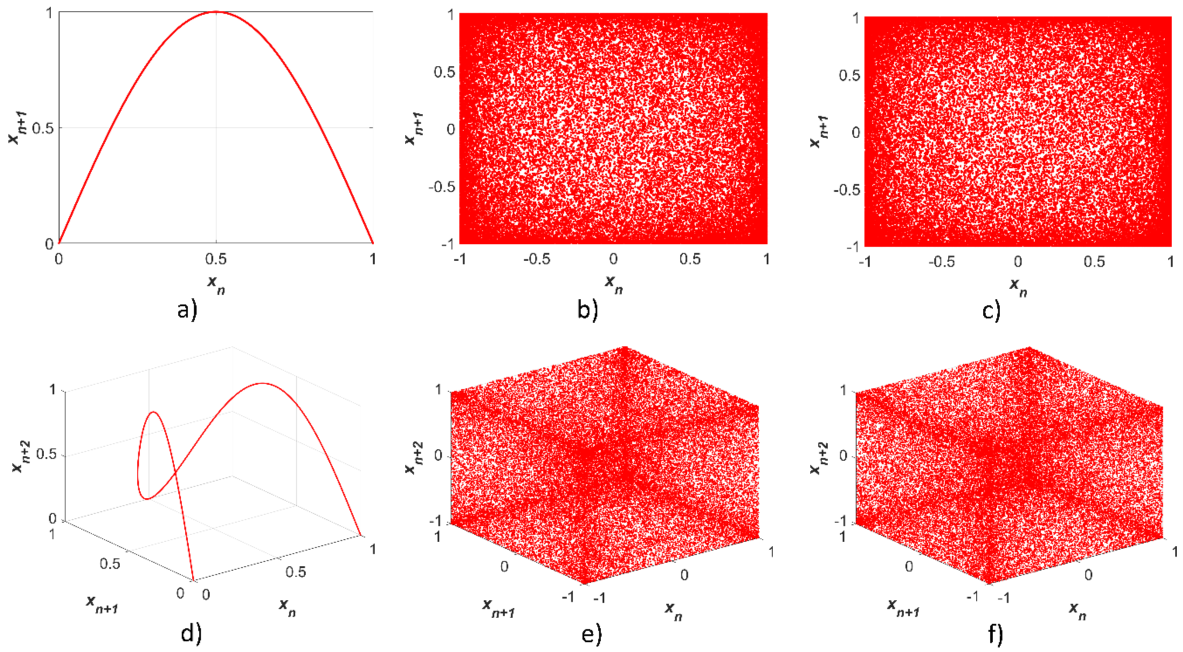

Figure 1. Phase diagrams: a), d) sine map (control parameter is 1); b), e) SQPM (a=b=c=0); c), f) SQPM (a=5, b=3, c=15)

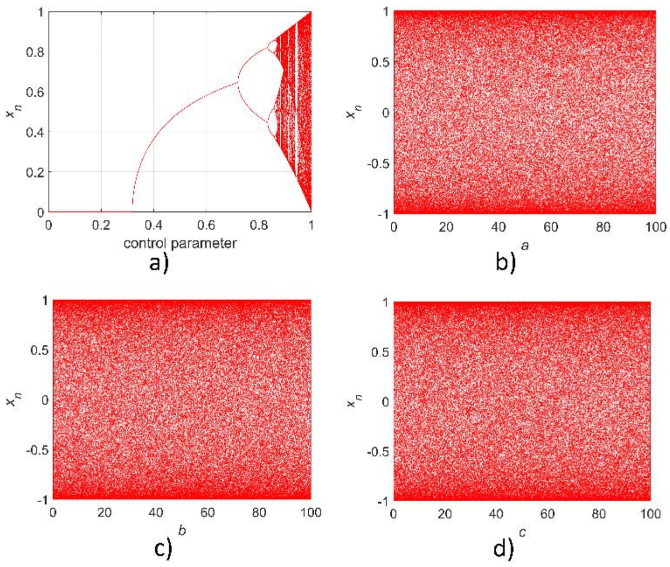

Figure 2. Bifurcation diagrams: a) sine map, b) SQPM ($a \in [0,100], b = c = 0$), c) SQPM ($b \in [0,100], a = c = 0$), d) SQPM ($c \in [0,100], a = b = 0$)

2.3 Lyapunov exponent

The Lyapunov exponent (LE) serves as a crucial measure for quantifying the rate at which two neighboring trajectories, originating from extremely close initial conditions, diverge. A positive LE value signifies chaotic behavior and a high sensitivity to initial conditions. Figure 3a) illustrates the LE values of SQPM in comparison with other chaotic maps such as the logistic map, the sine map, the one-dimensional cosine polynomial map (1-DCP) [40], and the one-dimensional sine-powered chaotic map (1-DSP) [41], plotted against a control parameter. Given that a higher LE value is indicative of superior chaotic performance [42], the LE calculations of SQPM clearly demonstrate its exceptional performance over traditional and other recently developed chaotic maps. Distinctively, SQPM consistently exhibits positive LE values, indicating an absence of non-chaotic behavior. This characteristic is maintained across the board, even with a fixed control parameter a; the LE values of SQPM remain invariably positive for control parameters b and c, as depicted in Figures 3b) and 3c).

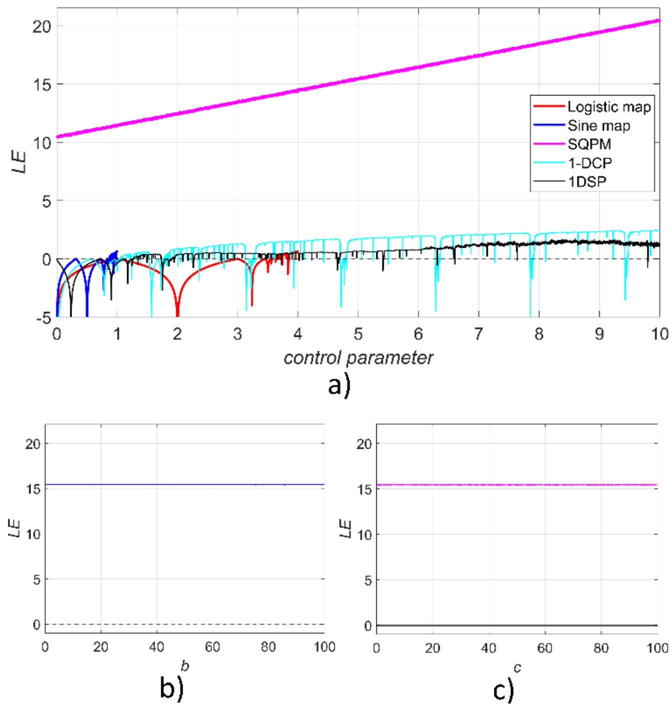

Figure 3. LE values: a) SQPM ($b = c = 0$), 1-DCP, 1DSP ($\beta = 0.3306$) b) SQPM ($a = 5, c = 0$), c) SQPM ($a = 5, b = 0$)

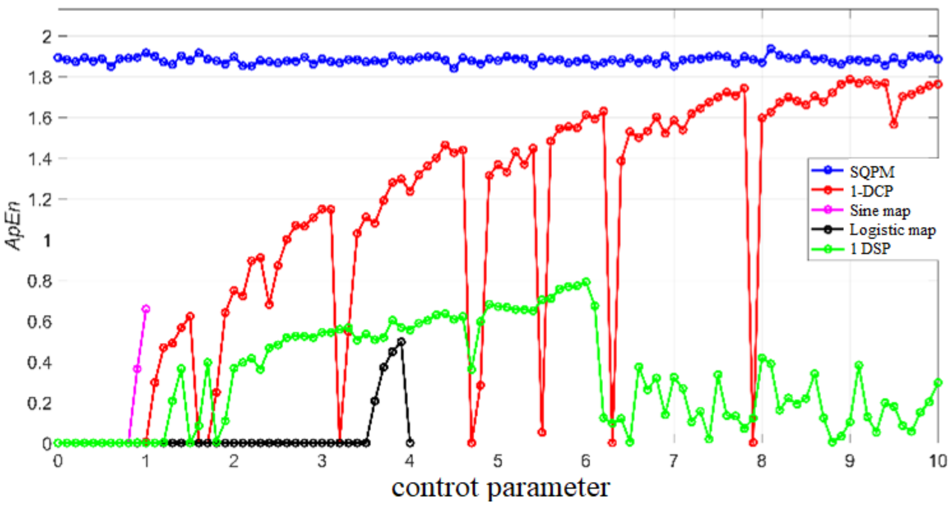

Figure 4. ApEn values: SQPM ($b = c = 0$), 1-DCP, 1DSP ($\beta = 0.3306$)

2.4 Approximate entropy

Approximate Entropy (ApEn) is a statistical method utilized for quantifying the irregularity and complexity of time-series data [43]. While a positive ApEn value does not invariably signify chaos [44], a larger ApEn indicates reduced predictability. Given that ApEn's calculation is contingent upon the data size [45], the chaotic maps under comparison are iterated to produce a sequence of 2000 real numbers each. The resulting ApEn values are graphically represented in Figure 4. The data reveals that SQPM generates sequences with higher levels of unpredictability and complexity, as evidenced by its ApEn values, compared to other maps.

The performance analysis and comparative evaluation with other 1D maps underscore SQPM's superior chaotic performance. The incorporation of three control parameters in SQPM enhances the keyspace for the associated MIE algorithm, thereby bolstering its security against cryptanalytic attacks. Moreover, the simplicity of SQPM's equation facilitates ease of implementation, contributing significantly to the fast performance of the proposed MIE algorithm.

3. SQPM-BASED MIE ALGORITHM

3.1 U-shaped scanning

Space-filling curves are instrumental in scanning every element of a 2D array precisely once, thereby facilitating the creation of a new 1D array. By reshaping the content of this 1D array into another 2D array, the elements of the original array can be effectively scrambled. Hence, space-filling curves are increasingly utilized in the permutation phase of image encryption schemes [46]. Various space-filling curves, such as the Hilbert curve [47, 48], Zigzag transform [49-51], square-wave confusion [52], Y-index curve [53], and L-shaped scanning [54], are employed in image encryption algorithms. This study introduces a novel space-filling curve named U-shaped scanning. Unlike the Hilbert curve, which is limited to scanning square 2D arrays, U-shaped scanning is capable of scanning every element in 2D arrays of any size. This method can be employed to scramble the pixels of an input image. Let I be the input image and I' be the scrambled image, both sharing identical dimensions H×W. U-shaped scanning traverses each element of I, creating distinct 1D sequences $U_{i=1,2,\ldots,\lceil W/2 \rceil}$, each with a length of $2(H-(i-1))+2(W-(i-1))-2$. The final 1D array is formed by concatenating these sequences in reverse order, as depicted in Eq. (2):

$$U = \left[U_{\lceil \frac{W}{2} \rceil} \; U_{\lceil \frac{W}{2} \rceil - 1} \cdots U_2 \; U_1 \right] \tag{2}$$

The 1D array U can be reconfigured into a 2D array I' of dimension×HW. Figure 5 exemplifies the U-shaped scanning process. As demonstrated in the figure, this scanning approach navigates the outer pixels following a U-shaped trajectory, commencing from the pixels in the first row. U-shaped scanning plays a pivotal role in significantly reducing the correlation between adjacent elements of the input images, owing to its application in both permutation and diffusion stages of the image encryption process.

3.2 The proposed algorithm

The proposed encryption method encompasses two primary stages: permutation and diffusion. Initially, U-shaped scanning is employed to pre-scramble the input images. In the case of color images, each channel is scrambled independently to diminish cross-channel correlation. Subsequently, all input images are merged horizontally. The rows and columns of this composite image are then circularly shifted using chaotic arrays generated by SQPM. In the final stage, U-shaped diffusion replaces conventional methods such as row-wise or column-wise substitution, utilizing SQPM to generate the diffusion sequence. Given that SQPM is a map with three control parameters and possesses an extensive chaotic range, it is integral to both permutation and diffusion stages, thereby enhancing the security and keyspace of the proposed MIE method significantly. Figure 6 illustrates the process of the proposed MIE algorithm for input color images.

Consider $I_1, I_2, \ldots, I_k$ as the input images of the proposed MIE algorithm, each sharing the same dimension H×W. These images are concatenated horizontally, regardless of being color or grayscale. The SHA-384 hash value of the amalgamated image, which ensures sensitivity to the plaintext, is selected as the first secret key. This key is subsequently divided into eight subblocks, with each subblock comprising 48 bits.

$$Key = \{K_1, K_2, K_3, K_4, K_5, K_6, K_7, K_8\} \quad (3)$$

The proposed MIE procedure involves distinct steps, utilizing K_1, K_2, K_3, K_4 to generate permutation parameters, while K_5, K_6, K_7, K_8 are employed for creating the diffusion sequence. The procedure is outlined as follows:

Step 1. Apply U-shaped scanning-based scrambling to each input image. For color images, each channel is scrambled individually using U-shaped scanning.

Step 2. Merge the scrambled images or channels horizontally to form a new matrix I_C. The dimensions of I_C are $H_1 \times W_1$, which is equal to $H \times kW$ or $H \times 3kW$ for the grayscale or color input images, respectively.

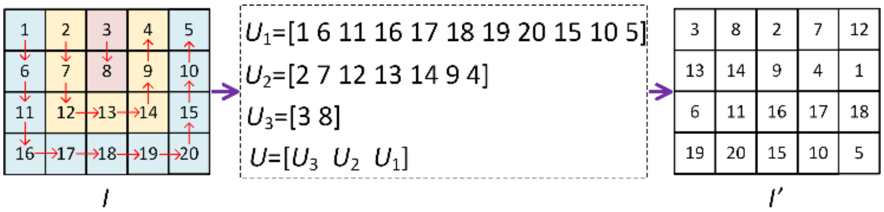

Figure 5. An example of U-shaped scanning and scrambling

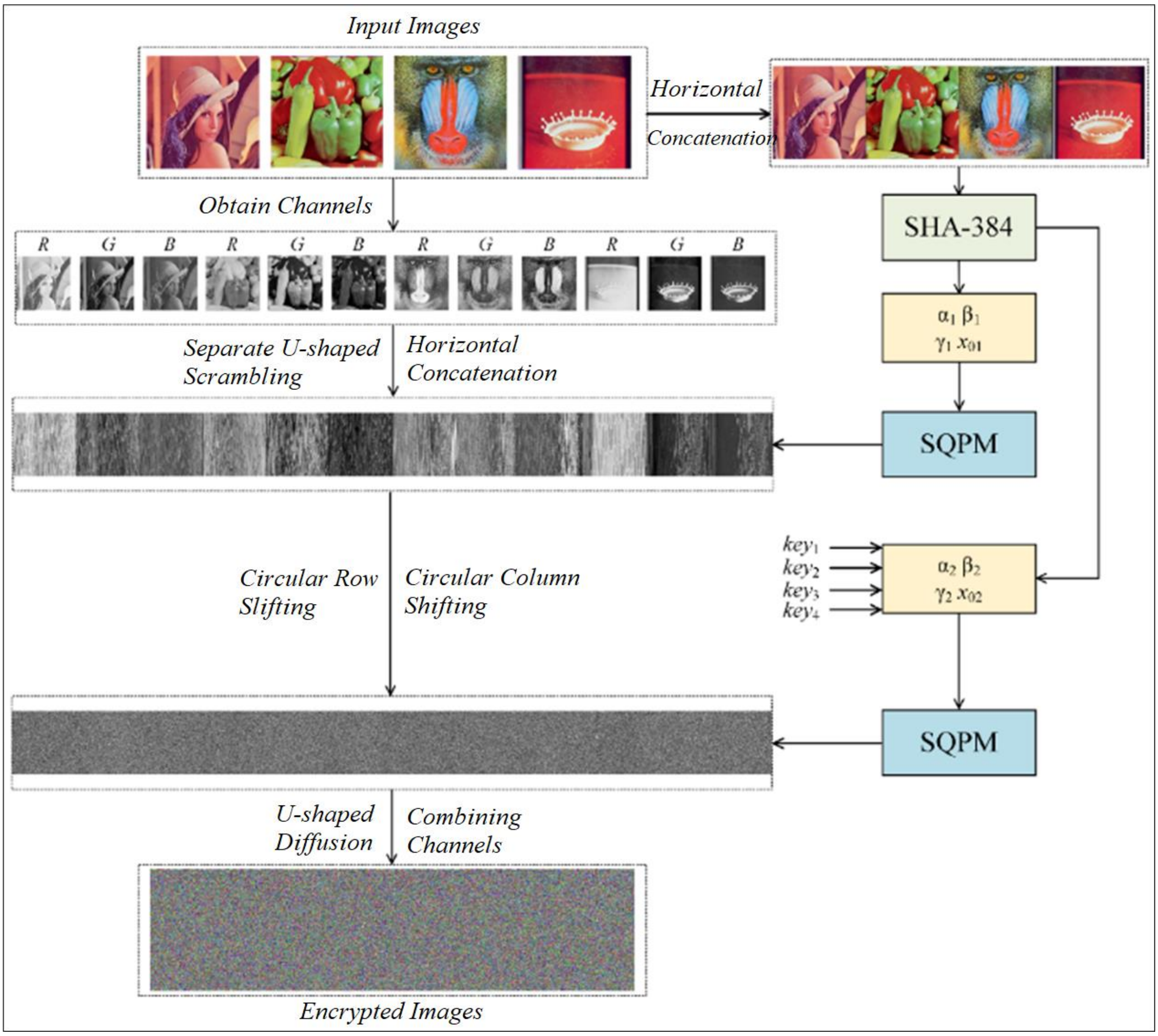

Figure 6. The process of the proposed MIE algorithm

Step 3. Iterate the SQPM $H_1 + W_1 + 500$ times to generate a chaotic sequence X_1. The control parameters and the initial value of SQPM are derived from the K_1, K_2, K_3, K_4 sequences. These sequences are further divided into subblocks of equal lengths: $K_1 = \{k_1, k_2, k_3, k_4, k_5, k_6\}$, $K_2 = \{l_1, l_2, l_3, l_4\}$, $K_3 = \{m_1, m_2\}$. The control parameters a, b, c, as well as the initial value x_{01}, are computed based on the following equations:

$$a = \alpha_1 = \frac{k_{1d}+k_{2d}+k_{3d}+k_{4d}+k_{5d}+k_{6d}}{2^8} \quad (4)$$

$$b = \beta_1 = \frac{(l_1 \oplus l_2)_d + (l_3 \oplus l_4)_d}{2^{10}} \quad (5)$$

$$c = \gamma_1 = nnz(m_1 \oplus m_2) \quad (6)$$

$$x_{01} = \frac{nnz(K_4)}{nz(K_4)} mod\ 1 \quad (7)$$

where, subscript d indicates the conversion from binary to decimal numbers, $nz(\cdot)$ and $nnz(\cdot)$ find the number of zero and nonzero entries in their inputs, respectively.

Step 4. The first 500 elements are removed from the sequence X_1 and the remaining elements are divided into arrays X_2 and X_3, whose lengths are H_1 and W_1, respectively. These arrays are used to calculate the row-shifting array R and the column-shifting array C as follows.

$$R(i) = \begin{cases} (\lceil X_2(i) \times 10^{15} \rceil\ mod\ W_1)\ , & X_2(i) \geq 0 \\ (-\lceil X_2(i) \times 10^{15} \rceil\ mod\ W_1), & X_2(i) < 0 \end{cases} \quad (8)$$

$$C(j) = \begin{cases} (\lceil X_3(j) \times 10^{15} \rceil\ mod\ H_1)\ , & X_3(j) \geq 0 \\ (-\lceil X_3(j) \times 10^{15} \rceil\ mod\ H_1), & X_3(j) < 0 \end{cases} \quad (9)$$

where, $i = 1,2, \ldots, H_1$ and $j = 1,2, \ldots, W_1$.

Step 5. Circularly shift the elements in the rows of matrix I_C by $R(i)$ positions, moving from the first row to the last. If the i-th element of the chaotic sequence is positive, the shift is to the right; if negative, to the left. Following this, perform a circular shift of the elements in the columns by $C(j)$ positions, starting from the last column to the first. Here, columns are shifted upwards for negative j-th elements and downwards for positive ones. This concludes the permutation phase, yielding the resultant image referred to as I_P.

Step 6. Iterate the SQPM $H \times W + 500$ times to produce another chaotic sequence Y. The sequences K_5, K_6 and K_7 are divided into subblocks of equal lengths, denoted as $K_5 = \{n_1, n_2, n_3, n_4\}$, $K_6 = \{o_1, o_2, o_3\}$, $K_7 = \{p_1, p_2\}$, respectively. For the diffusion phase, introduce four new external secret keys: key_1, key_2, key_3, and key_4. Compute the control parameters a_2, b_2, c_2, and the initial value x_{02} of the SQPM using the following equations:

$$a_2 = \alpha_2 + key_1 = \frac{n_{1d}+n_{2d}+n_{3d}+n_{4d}}{2^{12}} + key_1 \quad (10)$$

$$b_2 = \beta_2 + key_2 = \frac{(o_1 \oplus o_2)_d + o_{3d}}{2^{13}} + key_2 \quad (11)$$

$$c_2 = \gamma_2 + key_3 = nz(p_1 \oplus p_2) + key_3 \quad (12)$$

$$x_{02} = \frac{nz(K_8)}{nnz(K_8)} mod\ 1 - key_4 \quad (13)$$

where, $key_{1,2,3} \in [0, \infty)$ and $key_4 \in (0,1)$. The first 500 elements of the sequence Y are discarded and the remaining elements are processed as in Eq. (14) to obtain the diffusion sequence D.

$$D(i) = (\lceil |Y(i)| \times 10^{10} \rceil\ mod\ 256) \quad (14)$$

where, $i = 1,2, \ldots, H \times W$.

Step 7. The image I_P is divided into t images $\{I_{p_1}, I_{p_2}, \ldots, I_{p_t}\}$ ($t = k$ for grayscale images, $t = 3k$ for color images), each with a $H \times W$ dimension. The following operation called U-shaped diffusion is used to substitute the pixels in I_P.

$$E_1 = U_{scan}(I_{p_1}) \oplus D \quad (15)$$

$$E_j = \left(\left(U_{scan}\left(I_{p_j}\right) \oplus D\right) + E_{j-1}\right) mod\ 256 \quad (16)$$

where, $j = 2,3, \ldots, t$, U_{scan} applies U-shaped scanning in its input and generates a 1D array of length $1 \times HW$ as given in Eq. (2).

Step 8. The sequences $E_1, E_2, \ldots, E_t$ are reshaped into $H \times W$ images and concatenated horizontally to form the encrypted image I_E. If the input images are color, the encrypted images are obtained by combining the three sequential images.

3.3 The decryption process

The encrypted image I_E and the secret keys Key, key_1, key_2, key_3, and key_4 are the inputs to the decryption algorithm. If the encrypted image is a color image, R, G, and B channels are extracted and concatenated horizontally. The decryption process of the proposed MIE algorithm is described in the following steps.

Step 1. Using the secret keys and Eqs. (10)-(14), the diffusion sequence D is obtained. The following operations are carried out to obtain the image I_P.

$$I_{p_1} = invU_{scan}(E_1 \oplus D) \quad (17)$$

$$I_{p_j} = invU_{scan}\left((E_j - E_{j-1}) mod\ 256 \oplus D\right) \quad (18)$$

where, $j = 2,3, \ldots, t$ and $invU_{scan}$ implements the inverse U-shaped scanning in its input and generates a 1D array. The sequences $\{I_{p_1}, I_{p_2}, \ldots, I_{p_t}\}$ are reshaped into $H \times W$ images and concatenated horizontally to obtain I_P.

Step 2. The row-shifting array $R(i)$ and the column-shifting array $C(j)$ are calculated using Eqs. (4)-(9) for the de-scrambling phase of I_P. Initially, the elements in the columns are circularly shifted downwards if $C(j)$ is negative, and upwards if $C(j)$ is positive, starting from the last column to the first column ($j = 1,2, \ldots, W_1$). Subsequently, rows are shifted to the left if $R(i)$ is positive, and to the right, if $R(i)$ is negative starting from the first row to the last row ($i = 1,2, \ldots, H_1$). As a result, I_C image is obtained.

Step 3. I_C image is divided into k or $3k$ sub-images with dimensions of $H \times W$ for grayscale or color images, respectively. Inverse U-shaped scanning-based scrambling is applied to each sub-image. Descrambled sub-images are concatenated horizontally. If the original images are color, R, G, and B planes are combined to acquire the input images.

4. SIMULATION RESULTS AND SECURITY ANALYSIS

The security analysis of the proposed MIE algorithm was conducted on a PC equipped with 16 GB RAM and a 2.80 GHz Intel Core i7 processor, utilizing MATLAB 2020a. For the simulations, three distinct groups of images from the USC-SIPI image database [55] were employed. Group 1 comprises the color images "Lena", "Peppers", "Baboon", and "Splash", each of size 512×512. Group 2 includes color images "Female" (4.1.04.tiff), "Couple" (4.1.02.tiff), "House" (4.1.05.tiff), and "Tree" (4.1.06.tiff), all of size 256×256. Group 3 consists of four grayscale images, namely "Lena", "Moon", "Clock", and "Airplane", each with dimensions of 256×256. The encryption outcomes for these three groups of images are illustrated in Figure 7.

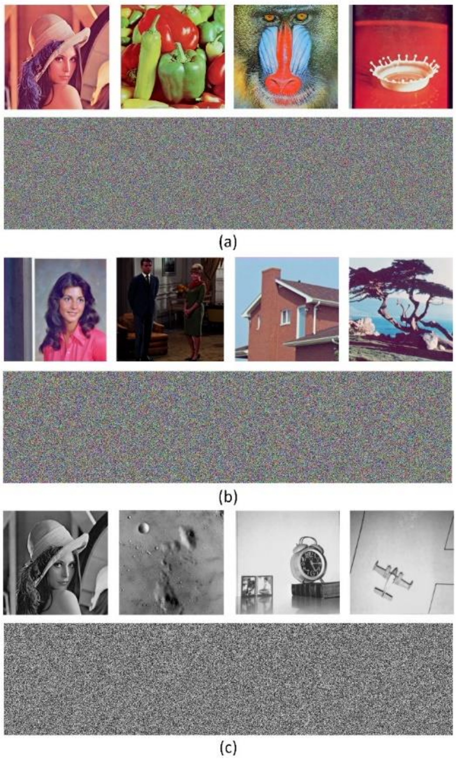

Figure 7. Input images and encrypted versions of a) Group 1 b) Group 2 c) Group 3

4.1 Keyspace and key sensitivity analysis

An MIE algorithm requires a keyspace of 2^{100} or larger to effectively resist exhaustive search attacks [56]. The proposed MIE scheme incorporates five secret keys: Key, key_1, key_2, key_3, and key_4. Key is the first 384-bit long secret key generated by the SHA-384 hash value of the combined input images. The other keys are integral in calculating the control parameters and the initial value for the SQPM in the diffusion phase. The precisions for key_1, key_2, key_3, and key_4 are experimentally set to 10^{-14}, 10^{-14}, 10^{-14}, and 10^{-14}, respectively. Consequently, the total keyspace size is calculated as $2^{384} \times 10^{52} \cong 1.67 \times 2^{556}$, which far exceeds the minimum required keyspace. This substantial keyspace size underscores the robustness of the proposed MIE method against exhaustive search attacks. Table 1 provides a comparative analysis of the keyspace sizes between this study and other MIE algorithms, highlighting the superior performance of the proposed scheme.

Key sensitivity implies that even a minor alteration in any of the secret keys during the decryption phase results in the inability to correctly recover the input images. To assess this, six key sensitivity analyses were conducted, with modifications as detailed in Table 2. For each analysis, a slight change was made to one secret key, keeping the others unaltered. Two key sensitivity tests were performed for each image group, and the decryption outcomes are presented in Figure 8. The results clearly illustrate that any minor deviation in a secret key renders the decrypted images unrecognizable, thereby confirming the high sensitivity of the proposed technique to the secret keys.

Table 1. Keyspace comparison with other MIE algorithms

MIE Algorithm	Keyspace
Proposed method	1.67×2^{556}
Ref. [4]	2^{455}
Ref. [5]	2^{332}
Ref. [7]	2^{555}
Ref. [10]	2^{478}
Ref. [17]	2^{390}
Ref. [18]	$2^{256} + 2$
Ref. [22]	1.55×2^{526}
Ref. [23]	1.245×2^{327}
Ref. [26]	$10^{60} \approx 1.24 \times 2^{199}$
Ref. [30]	2^{332}

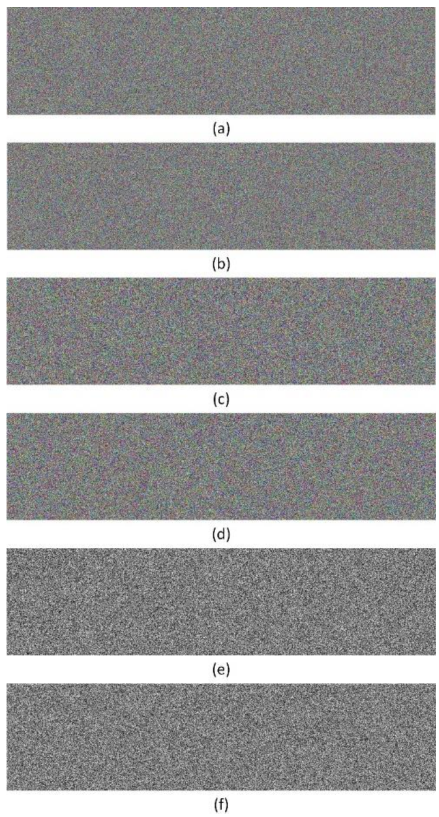

Figure 8. Decryption results. Group 1: a) Key's first bit flipped b) Key's last bit flipped, Group 2: c) $key_1 + 10^{-14}$ d) $key_2 - 10^{-10}$, Group 3: e) $key_3 + 10^{-14}$ f) $key_4 - 10^{-14}$

Table 2. Key sensitivity analysis parameters

Secret Key	Correct Secret Keys	Modification	Tested Images
$\boldsymbol{Key}$	'8387D4EC7F47E1DC9FAEADF5F7DE2D37FF50406F0C371F9 67294D2222D97131033EA16A78D061FC7216E0569E5716C46'	Flip first bit Flip last bit	Group 1
$\boldsymbol{key_1}$	0.5	add 10^{-14}	Group 2
$\boldsymbol{key_2}$	2.5	subtract 10^{-10}	
$\boldsymbol{key_3}$	5.5	add 10^{-14}	Group 3
$\boldsymbol{key_4}$	0.5	subtract 10^{-14}	

4.2 Histogram analysis

Histogram plots are a valuable tool for analyzing the pixel distribution in an image. Typically, the histogram plots of input images exhibit a non-uniform distribution, whereas encrypted images should display a uniform distribution to effectively resist statistical attacks. In the context of an MIE algorithm, particularly for color images, it is crucial that the histograms of the Red (R), Green (G), and Blue (B) channels across all images demonstrate uniform distribution. Figure 9 showcases the histogram graphs for both input and encrypted images in Group 1. As evidenced in the figure, the proposed MIE method successfully ensures a uniform pixel distribution across all channels of the encrypted images.

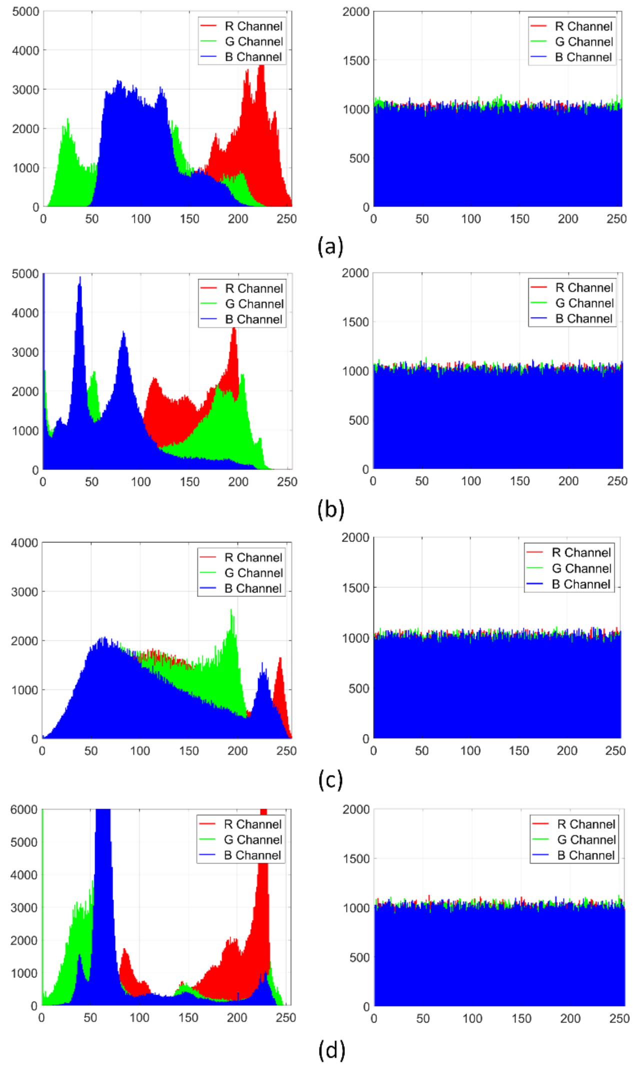

Figure 9. Histogram plots of input and encrypted images in Group 1: a) Lena b) Baboon c) Peppers d) Splash

The uniformity of a histogram graph can be quantitatively assessed by calculating its variance. The variance is computed using the following equation:

$$var = \frac{1}{256^2} \sum_{i=1}^{256} \sum_{j=1}^{256} \frac{(x_i - x_j)^2}{2} \quad (19)$$

where, x_i and x_j represent the total numbers of the i-th and j-th pixels, respectively. To effectively resist statistical attacks, the histogram variance values of encrypted images should be substantially lower than those of the input images. This principle is exemplified in the proposed MIE algorithm, which demonstrates a significant reduction in the histogram variance values for the input images, as detailed in Table 3. For the tested input images, there is an average reduction in variance value by 99.9%. This substantial decrease in variance values strongly indicates the method's capacity to resist statistical attacks.

Table 3. Histogram variance values

Group	Image	Channel	Input Image	Encrypted Image
Group 1	Lena	R	1017335	1093
		G	455719	1861
		B	1377356	1110
	Peppers	R	884298	943
		G	803422	1157
		B	2450373	1041
	Baboon	R	331359	1093
		G	571232	1091
		B	319770	1095
	Splash	R	2422651	1035
		G	3083896	1302
		B	5916965	893
Group 2	Female	R	66969	242
		G	64435	497
		B	113044	303
	Couple	R	210365	276
		G	337858	241
		B	289636	260
	House	R	258577	240
		G	299159	243
		B	394039	269
	Tree	R	81371	282
		G	57009	234
		B	129824	248
Group 3	Lena	-	30666	293
	Moon	-	135688	420
	Clock	-	282062	249
	Airplane	-	220849	248
	Overall Average		807355	652

4.3 Correlation analysis

To evaluate resistance against statistical attacks, it is crucial to conduct correlation analysis in tandem with histogram analysis. Input images typically exhibit strong correlations between adjacent pixels in horizontal (H), vertical (V), and diagonal (D) directions. Additionally, in the case of color images, there exists a cross-channel correlation among the R, G, and B planes [57, 58]. A proficient MIE algorithm should effectively reduce both types of correlations to safeguard against potential statistical attacks. Figure 10 displays the distribution of adjacent pixels in the original and encrypted images from Group 3, along the horizontal, vertical, and diagonal directions. As evidenced in the figure, the proposed MIE algorithm successfully decorrelates the adjacent pixels in all three directions for all images. The correlation between neighboring pixels is quantified using a metric known as the correlation coefficient ($r_{a,b}$), which is defined in Eq. (20).

$$r_{a,b} = \frac{\sum_{i=1}^{K} (a_i - E(a))(b_i - E(b))}{\sqrt{\sum_{i=1}^{K} (a_i - E(a))^2} \sqrt{\sum_{i=1}^{K} (b_i - E(b))^2}} \quad (20)$$

where, a_i and b_i represent the values of adjacent pixels, while K denotes the number of randomly selected pixel pairs. Additionally, $E(a)$ and $E(b)$ are the means of a_i and b_i, respectively. For the purpose of calculating the correlation coefficient values, 10,000 unique pixel pairs were randomly chosen in all directions from both the tested input images and their corresponding encrypted images. The correlation coefficient values thus obtained are presented in Table 4. Notably, the correlation coefficients of the input images are close to 1, indicating a strong correlation. However, the coefficients for the encrypted images are very close to 0, signifying a successful disruption of the horizontal, vertical, and diagonal correlations between neighboring pixels. This disruption is consistently achieved across all channels of the input images by the proposed MIE algorithm.

Table 4. Correlation coefficient values

Group	Image	Channel	Input Image			Encrypted Image		
			H	V	D	H	V	D
Group 1	Lena	R	0.9795	0.9899	0.9680	-0.0076	-0.0085	0.0013
		G	0.9686	0.9814	0.9562	0.0063	-0.0141	0.0172
		B	0.9293	0.9574	0.9176	0.0120	0.0005	-0.0119
	Peppers	R	0.9562	0.9608	0.9152	-0.0062	-0.0074	0.0121
		G	0.9841	0.9797	0.9669	-0.0009	-0.0105	-0.0111
		B	0.9701	0.9704	0.9333	0.0013	0.0062	-0.0044
	Baboon	R	0.9241	0.8687	0.8534	0.0031	0.0108	-0.0044
		G	0.8632	0.7636	0.7499	-0.0094	0.0017	0.0075
		B	0.9011	0.8773	0.8433	-0.0056	0.0148	0.0063
	Splash	R	0.9933	0.9949	0.9892	0.0031	0.0028	-0.0005
		G	0.9830	0.9884	0.9714	0.0011	-0.0049	0.0017
		B	0.9823	0.9773	0.9645	-0.0034	-0.0030	0.0016
Group 2	Female	R	0.9780	0.9870	0.9669	0.0218	-0.0042	0.0239
		G	0.9661	0.9807	0.9508	0.0067	0.0066	-0.0011
		B	0.9530	0.9732	0.9278	-0.0023	0.0083	0.0092
	Couple	R	0.9448	0.9553	0.9174	0.0061	-0.0101	0.0047
		G	0.9268	0.9524	0.9038	0.0171	0.0021	-0.0096
		B	0.9167	0.9391	0.8880	0.0108	0.0037	0.0119
	House	R	0.9657	0.9379	0.9142	-0.0107	-0.0090	0.0001
		G	0.9817	0.9482	0.9353	-0.0055	0.0020	0.0002
		B	0.9825	0.9740	0.9634	0.0030	0.0072	-0.0091
	Tree	R	0.9590	0.9376	0.9131	0.0161	-0.0023	-0.0070
		G	0.9690	0.9467	0.9332	-0.0084	-0.0064	0.0014
		B	0.9602	0.9408	0.9230	0.0030	0.0059	0.0115
Group 3	Lena	-	0.9405	0.9690	0.9175	-0.0001	0.0001	-0.0030
	Moon	-	0.9020	0.9356	0.9045	0.0065	0.0119	0.0169
	Clock	-	0.9533	0.9734	0.9441	-0.0145	0.0035	0.0004
	Airplane	-	0.9569	0.9393	0.8905	0.0102	-0.0028	0.0053

Table 5. Information entropy values

Group	Image	Channel	Input Image	Encrypted Image
Group 1	Lena	R	7.2531	7.9992
		G	7.5940	7.9987
		B	6.9684	7.9992
	Peppers	R	7.3316	7.9994
		G	7.5605	7.9992
		B	7.0196	7.9993
	Baboon	R	7.7067	7.9992
		G	7.4744	7.9992
		B	7.7522	7.9992
	Splash	R	6.9481	7.9993
		G	6.8845	7.9991
		B	6.1265	7.9994
Group 2	Female	R	7.2549	7.9973
		G	7.2704	7.9945
		B	6.7825	7.9967
	Couple	R	6.2499	7.9970
		G	5.9642	7.9973
		B	5.9309	7.9971
	House	R	6.4311	7.9974
		G	6.5389	7.9973
		B	6.2320	7.9970
	Tree	R	7.2104	7.9969
		G	7.4136	7.9974
		B	6.9207	7.9973
Group 3	Lena	-	7.5683	7.9968
	Moon	-	6.7093	7.9954
	Clock	-	6.7057	7.9973
	Airplane	-	6.4523	7.9973

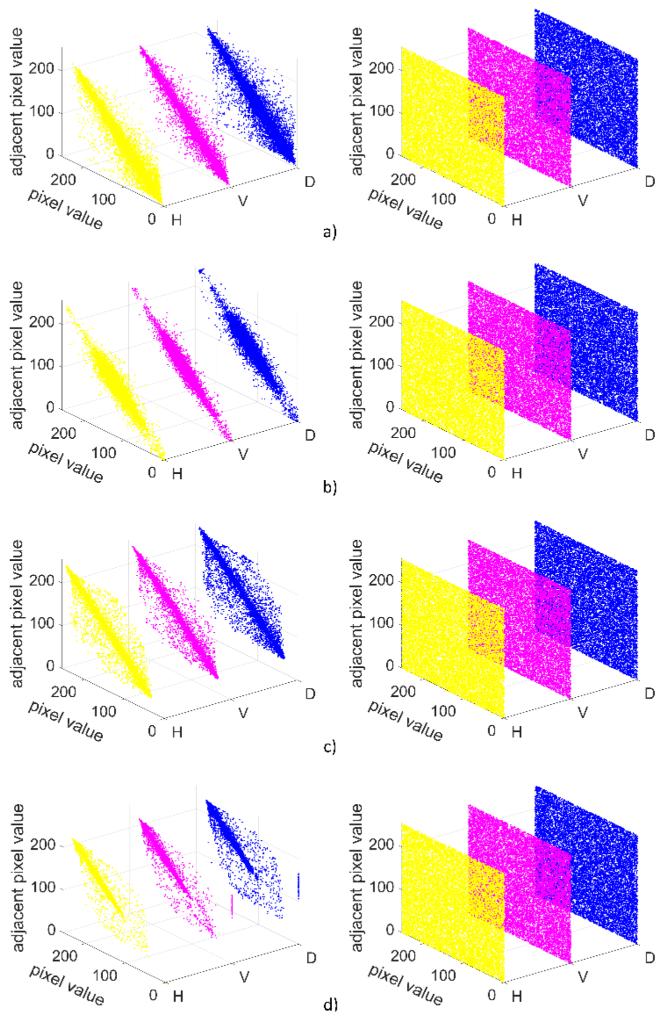

Figure 10. The distribution of adjacent pixels of the input and encrypted images in Group 3: a) Lena b) Moon c) Clock d) Airplane

Table 6. Information entropy (average) comparison

MIE Algorithm	Information Entropy
Proposed method	7.9993
Ref. [2]	7.9992
Ref. [4]	7.99929
Ref. [12]	7.9993
Ref. [17]	7.9993
Ref. [23]	7.9993

4.4 Information entropy analysis

In accordance with information theory, random systems inherently contain more information than deterministic systems. Consequently, the information entropy value of an encrypted image should exceed that of its corresponding input image. For a pixel represented by 8 bits, the information entropy (I) of a grayscale image or a single channel can be computed using the formula:

$$I = \sum_{i=0}^{2^8-1} P(s_i)\frac{1}{\log_2 P(s_i)} \quad (21)$$

where, $s_{i=0,1,..,255}$ is equal to the total number of pixels, and the values of i and $P(s_i)$ can be calculated by dividing s_i by the total number of pixels in an image. Theoretically, when the I value is equal to 8, a random image is obtained. An effective MIE method should yield encrypted images with information entropy values nearing this theoretical maximum. As shown in Table 5, the information entropy values of the encrypted images are greater than 7.99, indicating that the proposed MIE method successfully generates multiple random images/channels. Furthermore, Table 6 compares average information entropy values of the proposed method with several recent MIE algorithms for images of size 512×512. This comparison reveals that the proposed method exhibits similar or superior performance in terms of resisting information entropy attacks compared to other methods in the MIE literature.

4.5 Known-plaintext and chosen-plaintext attack analyses

In scenarios involving known-plaintext attacks (KPAs) and chosen-plaintext attacks (CPAs), an attacker's primary objective is to decipher the secret keys. To counteract these types of attacks, the proposed MIE algorithm incorporates the SHA-384 hash value of the combined input images. Consequently, even if an attacker gains access to both the plaintext and ciphertext, they cannot derive meaningful information from the cryptosystem. The resistance of the proposed MIE method to KPAs and CPAs was evaluated by encrypting five all-black and five all-white images, each 100 × 200 in size, as illustrated in Figure 11. An attacker would be unable to extract any valuable information from the encrypted all-black or all-white images, as these encrypted outputs resemble noise-like, meaningless patterns.

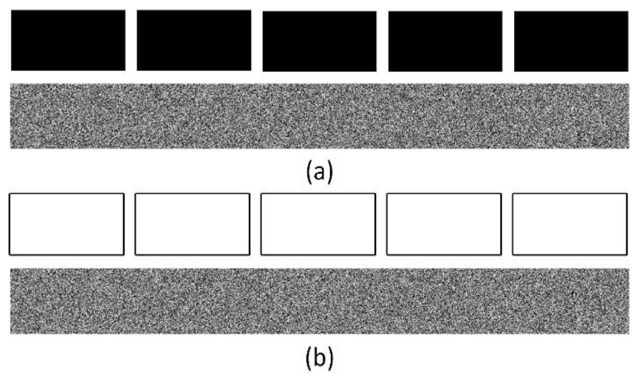

Figure 11. Input images and encrypted versions. a) all-black b) all-white

4.6 Differential attack analysis

Differential attacks focus on uncovering details about the secret key or the plaintext image by analyzing the variations between the input and encrypted output images. For an MIE algorithm to be deemed resistant to differential attacks, it must exhibit a complete change in the output images when any pixel in the input images or channels is altered. To evaluate a cryptosystem's resilience against such attacks, metrics like the Number of Pixel Change Rate (NPCR) and Unified Average Changing Intensity (UACI) are utilized, as defined in the following equations:

$$NPCR = \frac{1}{H\times W}\sum_{j=1}^{W}\sum_{i=1}^{H} D(i,j) \times 100\,\% \quad (22)$$

$$UACI = \frac{1}{H\times W}\sum_{j=1}^{W}\sum_{i=1}^{H}\frac{|E_1(i,j)-E_2(i,j)|}{255} \times 100\,\% \quad (23)$$

$$D(i,j) = \begin{cases} 1, & E_1(i,j) \neq E_2(i,j) \\ 0, & E_1(i,j) = E_2(i,j) \end{cases} \quad (24)$$

where, E_1 and E_2 represent two ciphertext images derived from input images that are identical except for a single pixel variation. As demonstrated in Table 7, an arbitrary input image or channel is selected, and the value of a randomly chosen pixel is slightly altered. Remarkably, even a minor change—such as increasing or decreasing the value of just one pixel in an input image—leads to significant alterations in a large proportion of pixels across all output images in the proposed method. The overall average values of the NPCR and UACI for the tested input images are 99.6002% and 33.4526%, respectively. These averages closely align with the expected NPCR and UACI values of 99.61% and 33.46% [59], thereby confirming the proposed MIE technique's ability to resist differential attacks. Furthermore, Table 8 compares the average NPCR and UACI values of the proposed method with other state-of-the-art MIE algorithms. This comparison demonstrates that the proposed method performs comparably to the leading algorithms in the literature in terms of resisting differential attacks.

Table 7. NPCR and UACI values

Group	Image	Channel	NPCR (%)	UACI (%)
Group 1	Lena	R	99.6090	33.5530
		G	99.6185	33.5717
		B	99.5960	33.4648
	Peppers	R	99.5895	33.5072
		G	99.6006	33.4976
		B	99.5831	33.5054
	Baboon	R	99.6147	33.4679
		G	99.6025	33.4904
		B	99.6265	33.3661
	Splash	R	99.5972	33.5185
		G	99.6147	33.4855
		B	99.6025	33.5051
Group 2	Female	R	99.5804	33.4232
		G	99.6078	32.6229
		B	99.6094	33.2586
	Couple	R	99.6139	33.5591
		G	99.6002	33.4241
		B	99.6063	33.6143
	House	R	99.6338	33.4167
		G	99.6033	33.5439
		B	99.5773	33.4453
	Tree	R	99.5651	33.3014
		G	99.5788	33.6294
		B	99.6032	33.4561
Group 3	Lena	-	99.6002	33.5026
	Moon	-	99.5667	33.5814
	Clock	-	99.5956	33.4971
	Airplane	-	99.6078	33.4639

Table 8. NPCR and UACI (average) comparison

MIE algorithm	NPCR (%)	UACI (%)
Proposed method	99.6002	33.4526
Ref. [4]	99.6052	33.4572
Ref. [7]	99.6085	33.4634
Ref. [8]	99.6143	33.4681
Ref. [12]	99.6142	33.4656
Ref. [13]	99.6060	33.5126
Ref. [18]	99.6289	33.5006

4.7 Data loss and noise attack analysis

During the transmission of multiple images, it is possible that parts of the encrypted image might be cropped or the image could become contaminated with noise. In such scenarios, a robust MIE scheme should ensure that the decrypted images remain visually recognizable. For the images in Groups 2 and 3, decryption results are displayed in Figure 12, where it is assumed that 25% of the data is lost from different corners. The simulation results clearly demonstrate that the decrypted images are identifiable, despite the data loss. Additionally, to simulate potential noise attacks, salt and pepper noise (SPN) with densities of 0.1 and 0.3 is added to the encrypted images across all groups. As evidenced in Figure 13, all decrypted images remain recognizable, though the amount of visual information diminishes with increasing noise density. Consequently, these results affirm that the proposed MIE algorithm is capable of effectively resisting both data loss and noise attacks.

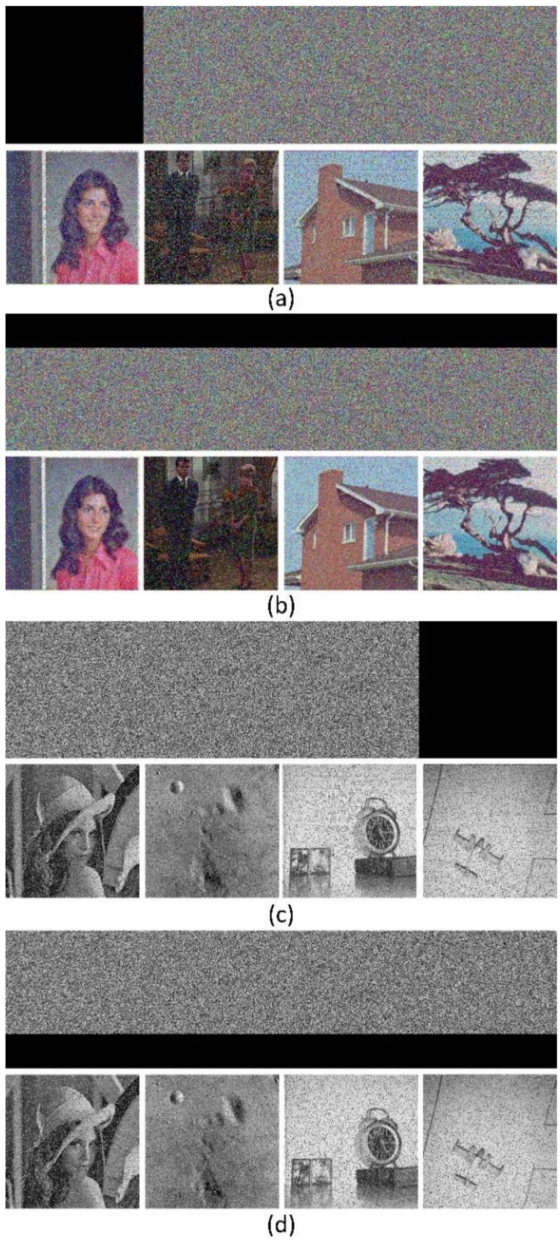

Figure 12. Data loss analysis. a), b) 25% data loss for images in group 2. c), d) 25% data loss for images in group 3

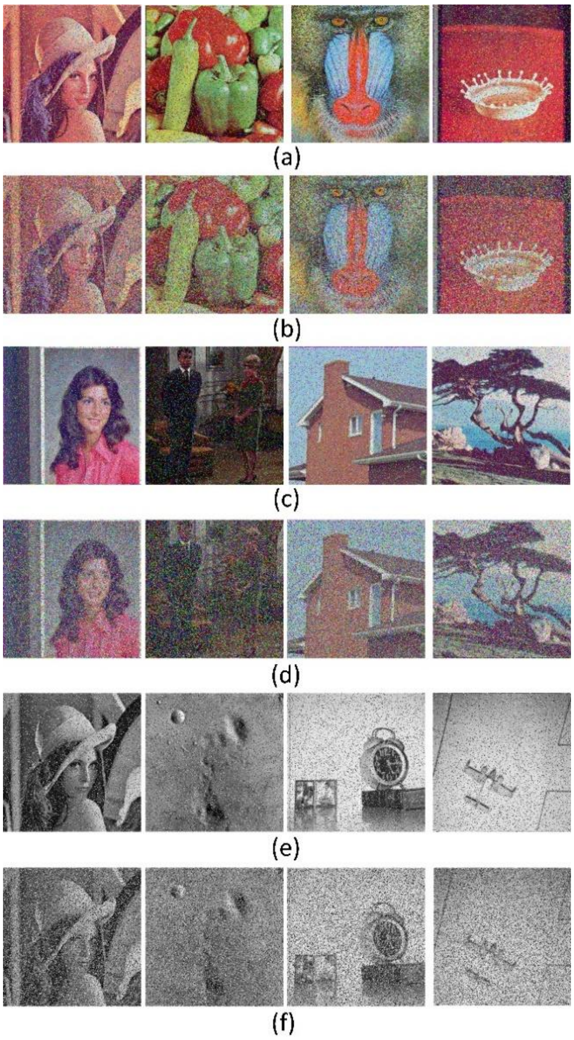

Figure 13. Noise attack analysis. a), b) 0.1 and 0.3 SPN for images in group 1. c), d) 0.1 and 0.3 SPN for images in group 2. e), f) 0.1 and 0.3 SPN for images in group 3

4.8 Encryption and decryption time analysis

Table 9. Encryption and decryption times (in seconds)

Image Type Image Size	Encryption Time	Decryption Time
4 color images 512×512	0.5618	0.4323
4 color images 256×256	0.2615	0.1068
4 grayscale images 256×256	0.1927	0.0615

The suitability of an MIE algorithm for real-time applications can be gauged by assessing the time it takes for encryption and decryption. Table 9 presents the average encryption and decryption times recorded for the proposed method when applied to the tested images. Additionally, Table 10 compares the encryption times of this work with those of various other MIE algorithms. It is important to note that encryption time is influenced not only by the algorithm itself but also by the hardware and software specifications, which are duly listed in the table for a comprehensive understanding. The proposed MIE algorithm demonstrates relatively rapid performance, capable of encrypting four color images of size 512×512 in approximately 0.56 seconds. Hence, its speed makes it a potentially attractive option for real-time applications.

4.9 Future work

The MIE algorithm introduced in this study is designed to encrypt images of identical sizes. Furthermore, it requires that all images within a group be of the same type, either all grayscale or all color. Future research endeavors will focus on developing new algorithms capable of encrypting images of varying types and sizes within a single group, thereby enhancing versatility and applicability.

Table 10. Encryption time (in seconds) comparison

Algorithm	Hardware	Software	Image Type Image Size	Encryption Time
This work	2.80 GHz Intel Core i7 16 GB RAM	MATLAB 2020a	4 color images 512×512	0.5618
			4 color images 256×256	0.2615
			4 grayscale images 256×256	0.1927
Ref. [4]	2.80 GHz Intel Core i7 16 GB RAM	MATLAB 2017b	4 grayscale images 256×256	0.7551
Ref. [8]	2.60 GHz Intel Core i7 16 GB RAM	MATLAB 2016a	4 grayscale images 256×256	0.540
Ref. [13]	M-5Y71@1.20 GHz CPU 8 GB RAM	MATLAB 2016a	4 grayscale images 512×512	1.71
Ref. [18]	N/A	MATLAB 2017	3 grayscale images 256×256	0.9375
Ref. [21]	3.60 GHz Intel Xeon W-2133 CPU 32 GB RAM	Wolfram Mathematica	8 color images 512×512	1.78

5. CONCLUSION

This study introduces a MIE algorithm, anchored in the newly developed chaotic SQPM and the innovative U-shaped scanning space-filling curve. The chaotic and complex behavior of SQPM is extensively validated through phase diagrams, Lyapunov exponents, bifurcation diagrams, and approximate entropy results, demonstrating its effectiveness over a broad range. U-shaped scanning is ingeniously applied for scrambling input images and scanning pixels during the diffusion phase. The algorithm is adept at encrypting an arbitrary number of color or grayscale images.

A notable enhancement in keyspace is achieved through the integration of SQPM, ensuring substantial resistance to brute-force attacks. Moreover, the algorithm exhibits high sensitivity to secret keys, bolstering its security credentials. Rigorous histogram and correlation analyses further affirm the algorithm's capability to effectively counteract statistical attacks. Furthermore, resistance to differential attacks is evidenced by the proximity of average Number of Pixel Change Rate (NPCR) and UACI values to their theoretical counterparts. Analysis under conditions of data loss and noise addition during image transmission reveals that decrypted images remain easily recognizable, highlighting the algorithm's robustness in practical scenarios. Additionally, the speed of encryption and decryption positions this method as a viable candidate for real-time applications.

Future research will focus on extending the algorithm's versatility to encompass encryption of images of diverse types and sizes within the same group, addressing the current limitation of requiring uniform image sizes and types.

REFERENCES

[1] Geetha, S., Punithavathi, P., Infanteena, A.M., Sindhu, S.S.S. (2018). A literature review on image encryption techniques. International Journal of Information Security and Privacy (IJISP), 12(3): 42-83. https://doi.org/10.4018/IJISP.2018070104

[2] Zhang, L., Zhang, X. (2020). Multiple-image encryption algorithm based on bit planes and chaos. Multimedia Tools and Applications, 79: 20753-20771. https://doi.org/10.1007/s11042-020-08835-4

[3] Hanif, M., Naqvi, R.A., Abbas, S., Khan, M.A., Iqbal, N. (2020). A novel and efficient 3D multiple images encryption scheme based on chaotic systems and swapping operations. IEEE Access, 8: 123536-123555. https://doi.org/10.1109/ACCESS.2020.3004536

[4] Demirtaş, M. (2022). A novel multiple grayscale image encryption method based on 3D bit-scrambling and diffusion. Optik, 266: 169624. https://doi.org/10.1016/j.ijleo.2022.169624

[5] Laiphrakpam, D.S., Thingbaijam, R., Singh, K.M., Al Awida, M. (2022). Encrypting multiple images with an enhanced chaotic map. IEEE Access, 10: 87844-87859. https://doi.org/10.1109/ACCESS.2022.3199738

[6] Tang, Y., Shao, Z., Zhao, X., Shang, Y. (2021). Robust multiple color images encryption using discrete Fourier transforms and chaotic map. Signal Processing: Image Communication, 93: 116168. https://doi.org/10.1016/j.image.2021.116168

[7] Wang, X., Liu, H. (2022). Cross-plane multi-image encryption using chaos and blurred pixels. Chaos, Solitons & Fractals, 164: 112586. https://doi.org/10.1016/j.chaos.2022.112586

[8] Wang, X., Wang, Y. (2023). Multiple medical image encryption algorithm based on scrambling of region of interest and diffusion of odd-even interleaved points. Expert Systems with Applications, 213: 118924. https://doi.org/10.1016/j.eswa.2022.118924

[9] Wang, Y., Shang, Y., Shao, Z., Zhang, Y., Coatrieux, G., Ding, H., Liu, T. (2022). Multiple color image encryption based on cascaded quaternion gyrator transforms. Signal Processing: Image Communication, 107: 116793. https://doi.org/10.1016/j.image.2022.116793

[10] Gao, X., Mou, J., Banerjee, S., Cao, Y., Xiong, L., Chen, X. (2022). An effective multiple-image encryption algorithm based on 3D cube and hyperchaotic map. Journal of King Saud University-Computer and Information Sciences, 34(4): 1535-1551. https://doi.org/10.1016/j.jksuci.2022.01.017

[11] Dua, M., Kumar, A., Garg, A., Garg, V. (2022). Multiple image encryption approach using non linear chaotic map and cosine transformation. International Journal of Information Technology, 14(3): 1627-1641. https://doi.org/10.1007/s41870-022-00885-1

[12] Gao, X., Mou, J., Xiong, L., Sha, Y., Yan, H., Cao, Y. (2022). A fast and efficient multiple images encryption based on single-channel encryption and chaotic system. Nonlinear dynamics, 108(1): 613-636. https://doi.org/10.1007/s11071-021-07192-7

[13] Zhang, X., Hu, Y. (2021). Multiple-image encryption algorithm based on the 3D scrambling model and dynamic DNA coding. Optics & Laser Technology, 141: 107073. https://doi.org/10.1016/j.optlastec.2021.107073

[14] ul Haq, T., Shah, T. (2020). Algebra-chaos amalgam and DNA transform based multiple digital image encryption. Journal of Information Security and Applications, 54: 102592. https://doi.org/10.1016/j.jisa.2020.102592

[15] Zhang, W., Zhang, X., Han, S., Wei, X., Wan, X. (2021). Multiple-image encryption based on light-field imaging and gravity model. Optics and Lasers in Engineering, 141: 106565. https://doi.org/10.1016/j.optlaseng.2021.106565

[16] Zhong, H., Li, G. (2022). Multi-image encryption algorithm based on wavelet transform and 3D shuffling scrambling. Multimedia Tools and Applications, 81(17): 24757-24776. https://doi.org/10.1007/s11042-022-12479-x

[17] Zhang, X., Zhang, L. (2022). Multiple-image encryption algorithm based on chaos and gene fusion. Multimedia Tools and Applications, 81(14): 20021-20042. https://doi.org/10.1007/s11042-022-12554-3

[18] Wang, T., Ge, B., Xia, C., Dai, G. (2022). Multi-image encryption algorithm based on cascaded modulation chaotic system and block-scrambling-diffusion. Entropy, 24(8): 1053. https://doi.org/10.3390/e24081053

[19] Paul, A., Kandar, S. (2022). Simultaneous encryption of multiple images using pseudo-random sequences generated by modified Newton-Raphson technique. Multimedia Tools and Applications, 81(10): 14355-14378. https://doi.org/10.1007/s11042-022-12210-w

[20] Bashir, Z., Malik, M.A., Hussain, M., Iqbal, N. (2022). Multiple RGB images encryption algorithm based on elliptic curve, improved Diffie Hellman protocol. Multimedia Tools and Applications, 81(3): 3867-3897. https://doi.org/10.1007/s11042-021-11687-1

[21] Sahasrabuddhe, A., Laiphrakpam, D.S. (2021). Multiple images encryption based on 3D scrambling and hyper-chaotic system. Information Sciences, 550: 252-267. https://doi.org/10.1016/j.ins.2020.10.031

[22] Patro, K.A.K., Acharya, B. (2021). An efficient dual-layer cross-coupled chaotic map security-based multi-image encryption system. Nonlinear Dynamics, 104(3): 2759-2805. https://doi.org/10.1007/s11071-021-06409-z

[23] Patro, K.A.K., Soni, A., Netam, P.K., Acharya, B. (2020). Multiple grayscale image encryption using cross-coupled

chaotic maps. Journal of Information Security and Applications, 52: 102470. https://doi.org/10.1016/j.jisa.2020.102470

[24] Patro, K.A.K., Acharya, B. (2020). A novel multi-dimensional multiple image encryption technique. Multimedia Tools and Applications, 79(19-20): 12959-12994. https://doi.org/10.1007/s11042-019-08470-8

[25] Tang, Z., Song, J., Zhang, X., Sun, R. (2016). Multiple-image encryption with bit-plane decomposition and chaotic maps. Optics and Lasers in Engineering, 80: 1-11. https://doi.org/10.1016/j.optlaseng.2015.12.004

[26] Zhang, X., Wang, X. (2017). Multiple-image encryption algorithm based on mixed image element and permutation. Optics and Lasers in Engineering, 92: 6-16. https://doi.org/10.1016/j.optlaseng.2016.12.005

[27] Zarebnia, M., Pakmanesh, H., Parvaz, R. (2019). A fast multiple-image encryption algorithm based on hybrid chaotic systems for gray scale images. Optik, 179: 761-773. https://doi.org/10.1016/j.ijleo.2018.10.025

[28] Wang, X., Liu, C., Jiang, D. (2021). A novel triple-image encryption and hiding algorithm based on chaos, compressive sensing and 3D DCT. Information Sciences, 574: 505-527. https://doi.org/10.1016/j.ins.2021.06.032

[29] Zhang, X., Wang, X. (2019). Multiple-image encryption algorithm based on DNA encoding and chaotic system. Multimedia Tools and Applications, 78: 7841-7869. https://doi.org/10.1007/s11042-018-6496-1

[30] Hoang, T.M. (2022). A novel design of multiple image encryption using perturbed chaotic map. Multimedia Tools and Applications, 81(18): 26535-26589. https://doi.org/10.1007/s11042-022-12139-0

[31] Patro, K.A.K., Acharya, B. (2018). Secure multi–level permutation operation based multiple colour image encryption. Journal of Information Security and Applications, 40: 111-133. https://doi.org/10.1016/j.jisa.2018.03.006

[32] Xiong, Y., Quan, C., Tay, C.J. (2018). Multiple image encryption scheme based on pixel exchange operation and vector decomposition. Optics and Lasers in Engineering, 101: 113-121. https://doi.org/10.1016/j.optlaseng.2017.10.010

[33] Ye, G., Liu, M., Wu, M. (2022). Double image encryption algorithm based on compressive sensing and elliptic curve. Alexandria Engineering Journal, 61(9): 6785-6795. https://doi.org/10.1016/j.aej.2021.12.023

[34] Gan, Z., Chai, X., Zhang, M., Lu, Y. (2018). A double color image encryption scheme based on three-dimensional Brownian motion. Multimedia Tools and Applications, 77: 27919-27953. https://doi.org/10.1007/s11042-018-5974-9

[35] Chen, Y., Xie, S., Zhang, J. (2022). A novel double image encryption algorithm based on coupled chaotic system. Physica Scripta, 97(6): 065207. https://doi.org/10.1088/1402-4896/ac6d85

[36] Lidong, L., Jiang, D., Wang, X., Zhang, L., Rong, X. (2020). A dynamic triple-image encryption scheme based on chaos, S-box and image compressing. IEEE Access, 8: 210382-210399. https://doi.org/10.1109/ACCESS.2020.3039891

[37] Joshi, A. B., Kumar, D., Gaffar, A., Mishra, D.C. (2020). Triple color image encryption based on 2D multiple parameter fractional discrete Fourier transform and 3D Arnold transform. Optics and Lasers in Engineering, 133: 106139. https://doi.org/10.1016/j.optlaseng.2020.106139

[38] Kaur, M., Kumar, V. (2020). A comprehensive review on image encryption techniques. Archives of Computational Methods in Engineering, 27: 15-43. https://doi.org/10.1007/s11831-018-9298-8

[39] Özkaynak, F. (2018). Brief review on application of nonlinear dynamics in image encryption. Nonlinear Dynamics, 92(2): 305-313. https://doi.org/10.1007/s11071-018-4056-x

[40] Talhaoui, M.Z., Wang, X., Midoun, M.A. (2021). A new one-dimensional cosine polynomial chaotic map and its use in image encryption. The Visual Computer, 37: 541-551. https://doi.org/10.1007/s00371-020-01822-8

[41] Mansouri, A., Wang, X. (2020). A novel one-dimensional sine powered chaotic map and its application in a new image encryption scheme. Information Sciences, 520: 46-62. https://doi.org/10.1016/j.ins.2020.02.008

[42] Wolf, A., Swift, J.B., Swinney, H.L., Vastano, J.A. (1985). Determining Lyapunov exponents from a time series. Physica D: nonlinear phenomena, 16(3): 285-317. https://doi.org/10.1016/0167-2789(85)90011-9

[43] Pincus, S. (1995). Approximate entropy (ApEn) as a complexity measure. Chaos: An Interdisciplinary Journal of Nonlinear Science, 5(1): 110-117. https://doi.org/10.1063/1.166092

[44] Pincus, S.M. (1991). Approximate entropy as a measure of system complexity. Proceedings of the National Academy of Sciences, 88(6): 2297-2301. https://doi.org/10.1073/pnas.88.6.2297

[45] Delgado-Bonal, A., Marshak, A. (2019). Approximate entropy and sample entropy: A comprehensive tutorial. Entropy, 21(6): 541. https://doi.org/10.3390/e21060541

[46] Maniccam, S.S., Bourbakis, N.G. (2004). Image and video encryption using SCAN patterns. Pattern Recognition, 37(4): 725-737. https://doi.org/10.1016/j.patcog.2003.08.011

[47] Shahna, K.U., Mohamed, A. (2020). A novel image encryption scheme using both pixel level and bit level permutation with chaotic map. Applied Soft Computing, 90, 106162. https://doi.org/10.1016/j.asoc.2020.106162

[48] Demirtaş, M. (2022). AFast multiple image encryption algorithm based on hilbert curve and chaotic map. In 2022 Innovations in Intelligent Systems and Applications Conference (ASYU), Antalya, Turkey, pp. 1-5. https://doi.org/10.1109/ASYU56188.2022.9925564

[49] Xingyuan, W., Junjian, Z., Guanghui, C. (2019). An image encryption algorithm based on ZigZag transform and LL compound chaotic system. Optics & Laser Technology, 119: 105581. https://doi.org/10.1016/j.optlastec.2019.105581

[50] Hua, Z., Li, J., Li, Y., Chen, Y. (2021). Image encryption using value-differencing transformation and modified ZigZag transformation. Nonlinear Dynamics, 106: 3583-3599. https://doi.org/10.1007/s11071-021-06941-y

[51] Wang, X., Chen, X. (2021). An image encryption algorithm based on dynamic row scrambling and Zigzag transformation. Chaos, Solitons & Fractals, 147: 110962. https://doi.org/10.1016/j.chaos.2021.110962

[52] Murali, P., Sankaradass, V. (2019). An efficient space filling curve based image encryption. Multimedia Tools and Applications, 78: 2135-2156. https://doi.org/10.1007/s11042-018-6234-8

[53] Niu, Y., Zhang, X. (2020). An effective image encryption

method based on space filling curve and plaintext-related josephus traversal. IEEE Access, 8: 196326-196340. https://doi.org/10.1109/ACCESS.2020.3034666
[54] Wang, X., Chen, Y. (2021). A new chaotic image encryption algorithm based on L-shaped method of dynamic block. Sensing and Imaging, 22: 1-30. https://doi.org/10.1007/s11220-021-00357-z
[55] The USC-SIPI Image Database. https://sipi.usc.edu/database/.
[56] Alvarez, G., Li, S. (2006). Some basic cryptographic requirements for chaos-based cryptosystems. International Journal of Bifurcation and Chaos, 16(8): 2129-2151. https://doi.org/10.1142/S0218127406015970
[57] Demirtaş, M. (2022). A new RGB color image encryption scheme based on cross-channel pixel and bit scrambling using chaos. Optik, 265: 169430. https://doi.org/10.1016/j.ijleo.2022.169430
[58] Yildirim, M. (2021). A color image encryption scheme reducing the correlations between R, G, B components. Optik, 237: 166728. https://doi.org/10.1016/j.ijleo.2021.166728
[59] Li, Y., Wang, C., Chen, H. (2017). A hyper-chaos-based image encryption algorithm using pixel-level permutation and bit-level permutation. Optics and Lasers in Engineering, 90: 238-246. https://doi.org/10.1016/j.optlaseng.2016.10.020