mathematics

Article

Efficient Image Encryption Scheme Using Novel 1D Multiparametric Dynamical Tent Map and Parallel Computing

Achraf Daoui [1], Mohamed Yamni [2], Samia Allaoua Chelloug [3,*], Mudasir Ahmad Wani [4] and Ahmed A. Abd El-Latif [4,5,*]

[1] National School of Applied Sciences, Sidi Mohamed Ben Abdellah-Fez University, Fez 30000, Morocco
[2] Dhar El Mahrez Faculty of Science, Sidi Mohamed Ben Abdellah-Fez University, Fez 30000, Morocco
[3] Department of Information Technology, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, Riyadh 11671, Saudi Arabia
[4] EIAS Data Science Lab, College of Computer and Information Sciences, Prince Sultan University, Riyadh 11586, Saudi Arabia
[5] Department of Mathematics and Computer Science, Faculty of Science, Menoufia University, Shebin El Koom 32511, Egypt
* Correspondence: sachelloug@pnu.edu.sa (S.A.C.); aabdellatif@psu.edu.sa (A.A.A.E.-L.)

Abstract: In order to ensure reliable and secure image exchange, chaotic systems are often considered for their good performance in information security. In this work, we first propose an extended version of a chaotic tent map (TM)—the multiparametric 1D tent map (MTM). The latter contains six control parameters defined over an unlimited range. These parameters strongly influence the MTM output when they are slightly modified by $\mp 10^{-13}$, which makes MTM stronger than the existing TM and other 1D chaotic maps in terms of security key space. Then, this paper proposes a simple, yet powerful method to make uniform the distribution of chaotic sequence values, making the latter suitable for use in cryptosystems. Next, a new image cryptosystem is introduced based on MTM and parallel computing. This computing mode is incorporated to boost the security level of our scheme and to speed up its runtime. Indeed, in only one running round, our encryption scheme generates a security key of space equal to $10^{78\times n}$ with n indicating the number of the available CPU cores. Therefore, the suggested scheme achieves a good trade-off between safety and efficiency. The results of the performed comparisons and numerical experiments indicate on the one hand that MTM exhibits good chaotic characteristics in comparison to its original version. On the other hand, the suggested cryptosystem demonstrates good strength against various kinds of attacks (brute force, statistical, classical, noise, differential, etc.). Furthermore, comparison with similar schemes indicate that the proposed scheme is competitive in terms of execution time and superior in terms of security level.

Keywords: image encryption; tent map; 1D chaotic systems/maps; multiparametric chaotic map; parallel computing; cryptosystem

MSC: 94A08

check for updates

Citation: Daoui, A.; Yamni, M.; Chelloug, S.A.; Wani, M.A.; El-Latif, A.A.A. Efficient Image Encryption Scheme Using Novel 1D Multiparametric Dynamical Tent Map and Parallel Computing. *Mathematics* **2023**, *11*, 1589. https://doi.org/10.3390/math11071589

Academic Editor: Lingfeng Liu

Received: 25 February 2023
Revised: 21 March 2023
Accepted: 22 March 2023
Published: 24 March 2023

1. Introduction

In recent years, communication technologies have made impressive advances, offering easy and large-scale access to information for individuals and organizations around the world. However, the transmission of information through communication channels, especially unsecured ones such as the internet, remains vulnerable to cyberattacks. To counter this risk, scientists are focusing on the development of information security techniques such as steganography [1,2] and encryption [3,4]. These techniques are commonly used to provide secure communication of various data forms, including audio signals [5,6], biomedical signals [7], medical images [7,8], satellite images [9], videos [10], text [11], etc. Images are particularly important in this context, as they serve as a reliable medium for many forms of data that include visual information (text, biosignals, video, barcodes, QR codes, etc.). For this reason, extensive encryption systems have been developed to protect the visual content of images during their communication between internet users.

Several schemes have been implemented for image encryption. The implementation of these schemes is frequently conducted in either the transform domain or in the spatial one. In the transform domain, encryption schemes are usually relying on discrete orthogonal transforms such as discrete Fourier transform (DFT) [12], Meixner moments transform [7], discrete cosine transform (DCT) [13], Hahn moments transform [14], fast Fourier transform (FFT) [15], etc. However, transform-based encryption schemes are generally more expensive in terms of computational and implementation complexity in comparison to spatial-based encryption schemes, which makes them less feasible for real-time applications. To overcome this limitation, encryption systems are frequently designed in the spatial domain due to their high security and low computational complexity. Such schemes can be designed by using DNA encoding [16], elliptic curves [17], fractal sorting matrix [4], Rubik's cube [18], quantum walks [19], etc.

The most popular spatial encryption approaches relies on chaotic systems, which generate chaotic sequences used in image encryption schemes. Chaotic systems can be classified into two groups: (i) one-dimensional (1D) and multidimensional (nD) chaotic systems. The latter are increasingly used for image encryption [20–23] due to their multiple parameters and complex models. However, the implementation of nD chaotic systems is challenging due to the complexity of their models [24]. Regarding 1D chaotic maps/systems, they are popular in image encryption because of their simple structure, making them easy to implement on both software and/or hardware levels [25–28]. However, 1D chaotic systems are limited by some major drawbacks: (i) the limited number of their control parameters that are used as security keys; this problem makes these schemes vulnerable to cyberattacks and can therefore be easily cracked through brute-force attacks; (ii) the limited chaotic ranges of the 1D chaotic systems' control parameters and/or the periodicity of these systems for specific values within their control parameter ranges; and (iii) the nonuniform distribution of the chaotic sequences produced via 1D chaotic systems.

To overcome problem (i), this work proposes a chaotic map called a multiparametric tent map (MTM). The latter is introduced as extension of the well-known 1D tent map (TM). MTM contains six control parameters, whereas its original version (MT) has only one parameter. The proposed model can also solve problem (ii) as the MTM parameters are defined on an infinite interval ($\mathbb{R}$-space). In contrast, the control parameter interval of the original MT is defined on limited interval. To solve problem (iii), a new, simple, yet efficient method is proposed to unify the distribution of chaotic sequences. This method is based on ascending or descending sort of chaotic sequence elements. Then, the index vector associated with the sorted elements is normalized to a given interval.

The implementation of chaos-based encryption schemes usually involves the confusion and diffusion stages. The confusion (permutation) process reduces the correlation between adjacent pixels, and the diffusion phase is used to modify the statistical characteristics of the input image. In general, these phases require the use of long chaotic sequences to achieve high security of the encryption scheme. However, generating large chaotic sequences is a time-consuming process, especially for real-time applications, where execution time is a crucial factor. This problem limits the use of chaotic systems in real-time security application. To speed up the execution time of chaos-based encryption systems, parallel computing strategies can be exploited [29–32]. Indeed, the parallel execution mode can considerably accelerate the encryption speed through the utilization of the full resources of a multicore processor. This type of processor is increasingly used in many recent machines such as smartphones [33], computers [34], Raspberry boards [35], etc. Therefore, the design of encryption schemes that can be run on multicore processors is of obvious interest. Given this fact, the present work includes a novel encryption scheme, which is based on MTM and parallel computing. This computing mode is incorporated to boost the security level of our scheme and to speed up its runtime.

Our scheme consists of three essential phases. The first one consists in the generation of MTM-based chaotic sequences through parallel computing. This phase allows the maximum exploitation of the processor's cores for both reach minimal execution time and maximal security level. The second phase consists in diffusing the pixels of the input image in order to cancel its statistical characteristics. For this purpose, the bitwise XOR operation is performed between the original image and a chaotic matrix produced in the previous phase. The third phase consists in confusing the pixels of the diffused image by employing MTM-based chaotic sequences. This phase considerably increases the security level of the proposed scheme.

This work also presents a simple, yet powerful method to equalize the histogram of chaotic sequences used in image cryptosystems. This method is based on the ascending (or descending) sorting of a chaotic sequence, then normalizing (scaling) the indices of the sorted chaotic sequence values.

The main contributions of this article can be summarized as follows:

(i) the introduction of a new 1D multiparametric tent map (MTM) for improving the chaotic behavior and the key space of the existing 1D tent map.

(ii) the suggestion of a simple yet efficient method to equalize the histogram of chaotic sequence values.

(iii) the introduction of novel encryption scheme based on MTM and parallel computing for fast and secure image communication.

(iv) the proposed parallel encryption algorithm offers a good trade-off between security level and efficiency.

(v) demonstrating the performances of the proposed scheme by providing diverse tests and comparisons with similar schemes.

The rest of the work consists of the following sections. The Section 2 contains related work with discussion focusing on 1D chaotic systems. The Section 3 presents the suggested 1D chaotic map—the MTM. The Section 4 details the design and implementation of the suggested cryptosystem, which is based on MTM with parallel computing. In the Section 5, results of both tests and comparisons are reported, demonstrating the efficiency and superiority of our encryption scheme. The Section 6 concludes the paper and suggests possible extensions of the proposed cryptosystem.

2. Related Work

One-dimensional chaotic systems are frequently exploited in the design of various image encryption schemes. One of the reasons that makes 1D systems widely applied in image encryption schemes is their simple mathematical models, which facilitates their software and hardware implementation.

Table 1 presents a review of recent literature related to image encryption schemes based on 1D chaotic maps. This table also lists some characteristics related to the 1D chaotic systems used: the number of control parameters that are used as security keys in the cryptosystems, the range of the control parameters (limited/infinite), and the distribution of the sequence values generated via the 1D chaotic maps (uniform/not uniform).

Table 1. Literature review of existing 1D chaotic maps applied to image encryption with characteristics related to the used chaotic maps.

Image Encryption Schemes	Used 1D Chaotic Map	Number of Control Parameters	Range of Control Parameters	Chaotic Value Distribution
[36–41]	Logistic map	1	Limited	Not uniform
[42–46]	Sine map	1	Limited	Not uniform
[47–51]	Chebychev map	1	Infinite	Not uniform
[52–55]	Bernoulli map	1	Limited	Not uniform

Table 1. *Cont.*

Image Encryption Schemes	Used 1D Chaotic Map	Number of Control Parameters	Range of Control Parameters	Chaotic Value Distribution
[56–58]	Tent map	1	Limited	Not uniform
[28]	Quadratic map	3	Limited	Not uniform
[59]	q-deformed logistic map	2	Limited	Not uniform
[60]	Modified logistic map	2	Limited	Not uniform
[26,61,62]	Improved 1D maps	1	Limited	Not uniform
Proposed scheme	Multiparametric dynamical tent map (MTM)	6	Infinite	Uniform after histogram equalization

By analyzing the items in Table 1, we can conclude on the one hand that the 1D chaotic maps are widely used in various image encryption schemes, which demonstrates the powerful capability of these maps in the secure image communication. In addition, we can notice that the majority of 1D chaotic systems include only a few of the control parameters (from 1 to 3). Therefore, the size of the security keys is small for 1D chaotic system-based security systems. To increase this size, two strategies can be implemented. The first one involves the generation of several chaotic sequences during the execution of a security system algorithm, and for each generated sequence, a different value of the 1D map parameter is used. This strategy allows us to increase the security level of the algorithms used in security systems. However, when a large chaotic sequences are generated sequentially on a computing machine, the execution time of 1D chaos-based algorithms becomes high. Therefore, this strategy is less efficient, especially for real-time applications. To speed up the process of generating large chaotic sequences, the parallel computing mode can be exploited. Indeed, the use of this mode allows us to generate simultaneously (in a parallel way) several large chaotic sequences by exploiting all the available resources of the computing systems [29,31,32]. Therefore, parallel computing can significantly accelerate the generation of large chaotic sequences in comparison to sequential computing. To exploit the advantages of parallel computing, the encryption system proposed in this work is designed to run on computer systems that support the parallel computing mode.

The second strategy focuses on the use of multiparametric 1D chaotic systems. Typically, this strategy can be achieved via two methods: the first consists in proposing new multiparametric 1D chaotic map models [28]. This method constitutes an interesting field for further research. The second method involves the modification of existing 1D chaotic map models in order to introduce additional control parameters [59,60]. This method has become increasingly interesting and has demonstrated its success in certain security system studies [59,61,62]. In general, the use of 1D multiparametric chaotic maps provides good security for security systems while using minimal chaotic sequences produced by these maps. In this sense, the present work introduces an extended version of the tent map named multiparametric tent map (MTM). Compared to its original version and in comparison with other 1D chaotic maps, MTM has a higher number of parameters. In addition, the parameter ranges of MTM are unlimited. In contrast, the control parameter ranges of most existing 1D systems are limited. These benefits of MTM can be effectively exploited to achieve higher security levels for security systems. Furthermore, from Table 1, we can notice that 1D chaotic systems face a common problem: the nonuniformity of the chaotic sequences generated from such systems. This drawback can be exploited by attackers who try to crack security systems based on statistical analysis. To avoid this drawback, this article presents a simple but effective solution. Our solution is based on the normalization of indices, which are generated by sorting the components of chaotic sequences. The advantages of MTM and the proposed method for the uniform distribution of chaotic sequences are demonstrated in the context of a new encryption scheme. The central idea in designing this scheme is to achieve a strong level of safety while using a minimum number of MTM-based chaotic sequences.

3. Proposed Multiparametric Dynamical Tent Map

The tent map is one of the famous 1D nonlinear dynamical systems that exhibit chaotic behavior. One of the main reasons for the extensive use of 1D TM in security applications is the linear piecewise character of this map [63]. The 1D TM has attracted increasing attention in image encryption applications. Some excellent image encryption schemes are designed based on 1D TM [58,64–66]. However, the main drawback of 1D TM and other 1D chaotic maps is the limited number of their control parameters. The 1D TM contains a single parameter that leads to the occurrence of chaos in a limited interval. This failure makes 1D chaotic map-based encryption schemes highly vulnerable to brute-force cracking attempts by cyberattackers.

Assuming that a 1D chaotic system with multiple control parameters can improve the security level of cryptosystems, a new improved version of 1D TM is proposed in this section—a multiparametric dynamical tent map (MTM). It is worth mentioning that the present framework can be easily applied to other one-dimensional or multidimensional chaotic maps for introducing their multiparametric versions. In this regard, the proposed multiparametric version of a tent map is provided for illustration purposes.

3.1. Tent Map

The classical 1D TM is defined by the following relation [67]:

$$T_{n+1} = \begin{cases} \lambda T_n \text{ for } T_n < 0.5 \\ \lambda(1 - T_n) \text{ for } T_n \geq 0.5 \end{cases} \tag{1}$$

where λ is the control parameter of TM and $0 \leq T_0 \leq 1$ its initial value. For $0 \leq \lambda \leq 2$, TM exhibits a chaotic behavior. Figure 1 shows the bifurcation diagram and the values of Lyapenov exponent (LE) corresponding to TM. The latter is computed by the next relation [37]:

$$LE = \lim_{m \to \infty} \left[\frac{1}{m} \sum_{j=1}^{m} \log_2 \left| \frac{dx_{j+1}}{dx_j} \right| \right] \tag{2}$$

where $\log_2$ is the log base 2 function.

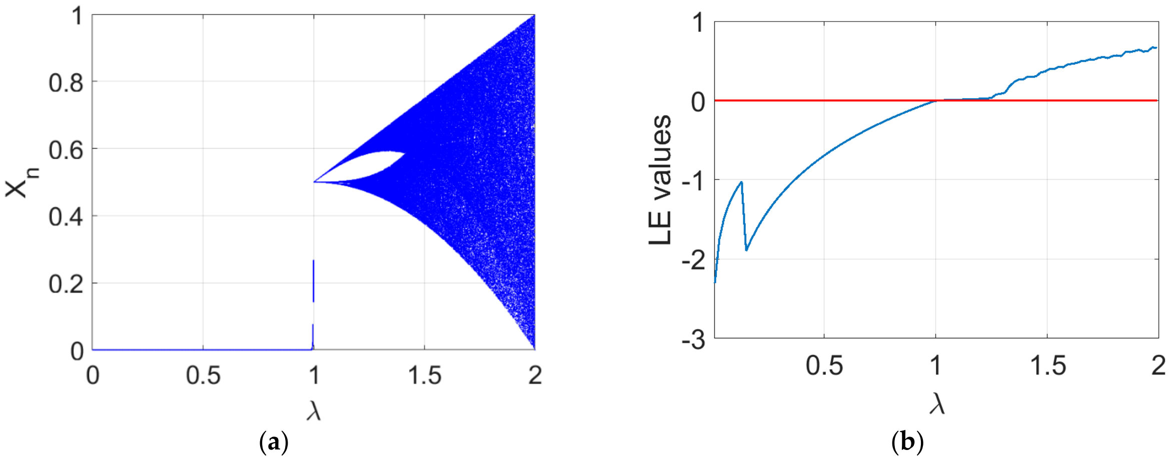


Figure 1. (**a**) Bifurcation diagram and (**b**) LE of *TM* for $\lambda \in [0, 2]$ with $T_0 = 0.5$.

From Figure 1, it can be observed that TM becomes rich in chaotic dynamics when the value of λ tends to 2. However, the "white areas" in the bifurcation diagram and the negative values of LE indicate the lack of the chaotic behavior for particular ranges of λ values. These values should be avoided when designing TM-based security schemes. We can also notice that the interval of λ that leads to a positive value of LE is very limited, which makes TM-based security systems vulnerable to cyberattacks due to their low security level. For significantly improving the chaotic behavior of TM, an extended version of TM is proposed in the next section: the multiparametric tent map (MTM).

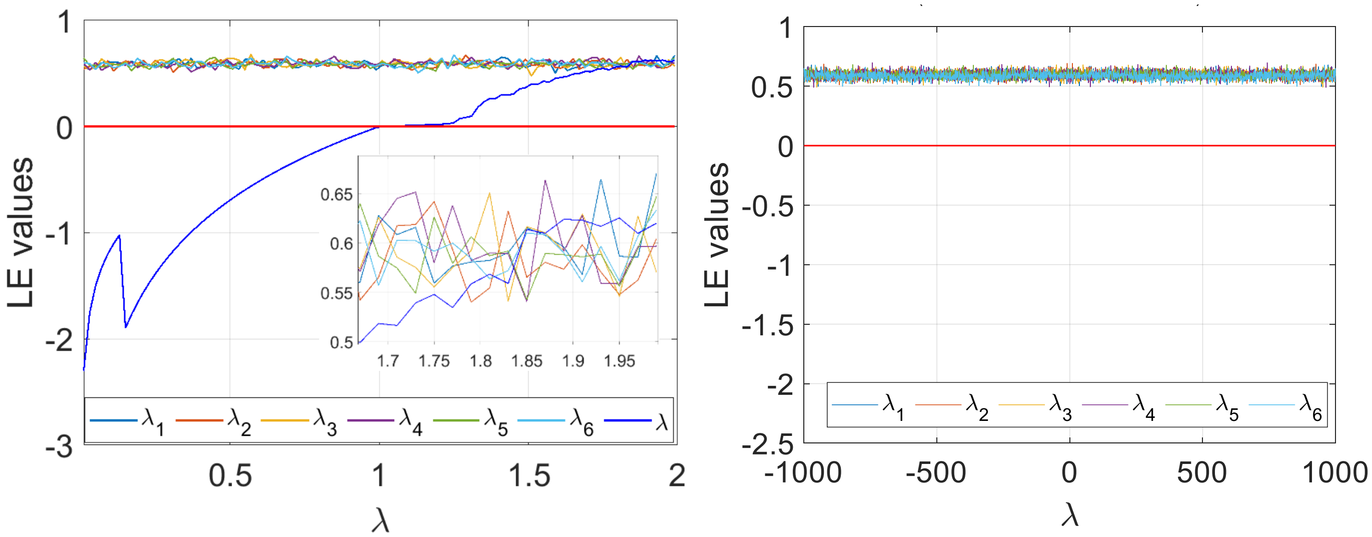
3.2. Proposed MTM Model

Driven by the idea that a 1D chaotic system with large number of parameters can provide high security level to encryption schemes, a new 1D chaotic system with multiple parameters is proposed as extension of the existing TM. The following relation defines the proposed MTM model:

$$X_{n+1} = \begin{cases} (10/7) \times (1 - \alpha|\cos(\lambda_1)| - \alpha|\sin(\lambda_2)| - \alpha|\text{atan}(\lambda_3)|)X_n & \text{for } X_n < 0.7 \\ (10/3) \times (1 - \alpha|\cos(\lambda_4)| - \alpha|\sin(\lambda_5)| - \alpha|\text{atan}(\lambda_6)|)(1 - X_n) & \text{for } X_n \geq 0.7 \end{cases} \quad (3)$$

where $\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5$ and λ_6 represent the MTM control parameters with $cos(x)$, $sin(x)$ and $atan(x)$ denote the trigonometric cosine, sine and arctangent functions, respectively. The variable x of these functions is defined on $\mathbb{R}$ domain. Therefore, a wide range of possibilities is available when selecting the $\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5, \lambda_6$ parameters that are used as security keys. The symbol $|.|$ in Equation (3) denotes the absolute value symbol and α is a constant value that must be selected in the range $\alpha \in [10^{-4}, 10^{-2}]$ to guarantee the chaotic behavior of the MTM and $0 < X_0 \leq 1$ is the initial value of the MTM.

It is worth mentioning that the use of the ratios $10/7$, $10/3$ and the threshold 0.7 in Equation (3) guarantee the chaotic behavior of MTM and ensure both positive and maximal LE values of MTM (see Figure 2). It is also worth mentioning that the incorporation of 6 control parameters in the MTM model is intended to achieve a high security level for MTM-based security schemes (see Sections 5.1 and 5.2).



(**a**) (**b**)

Figure 2. (**a**) LE values corresponding to TM and MTM parameters within the range [0, 2] and (**b**) LE values of MTM in the range [−1000, 1000].

From Equation (3), it can be deduced that the MLM control parameters range is $\mathbb{R}$. This range is very large compared to the original λ-parameter of TM domain and in comparison to other 1D chaotic systems [26,28,59,61,62]. However, it is necessary to examine the chaotic behavior of MTM within the definition domain of its control parameters.

3.3. Bifurcation Diagrams and LE Analysis of MTM

To compare the chaotic behavior of TM and MTM, LE criterion is used in the present test. For this purpose, the control parameter values of TM and MTM are varied in [0, 2]. Then, the result of the present comparison is displayed in Figure 2a. From this figure, it can be observed that LE of MTM remains positive over the entire interval [0, 2] and it remains equal to or greater than LE of the original TM (blue curve). This finding indicates that MTM exhibits better chaotic behavior in comparison to its original version. In the current test, the values of T_0 and α parameters of MTM are set to 0.5 and 10^{-4}, respectively. For TM, its initial value is set to $T_0 = 0.5$. To further demonstrate the chaotic behavior of MTM, the values of LE corresponding to this map are calculated in the range [-1000, 1000]. Subsequently, the obtained LE values are presented in Figure 2b. From this figure, it becomes apparent that LE values remain positive on the full considered interval that indicate the chaotic behavior of MTM.

To further support the results thus far discussed, the graphical bifurcation diagrams are plotted in Figure 3 for the six MTM parameters, which are varied in the range [-1000, 1000]. From this figure, we notice that MTM has bifurcations on this interval for all its six parameters. Therefore, these parameters are suitable for use as security keys when designing MTM-based information security systems.

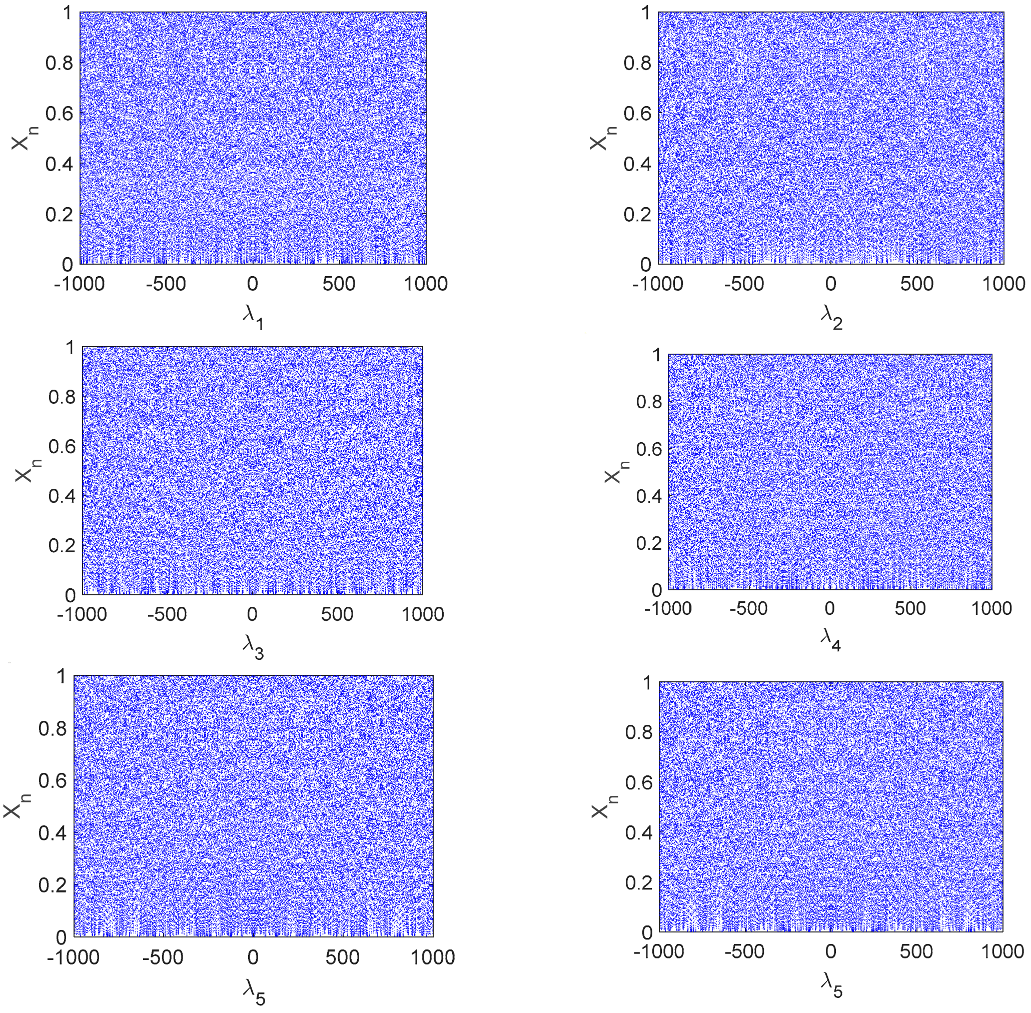
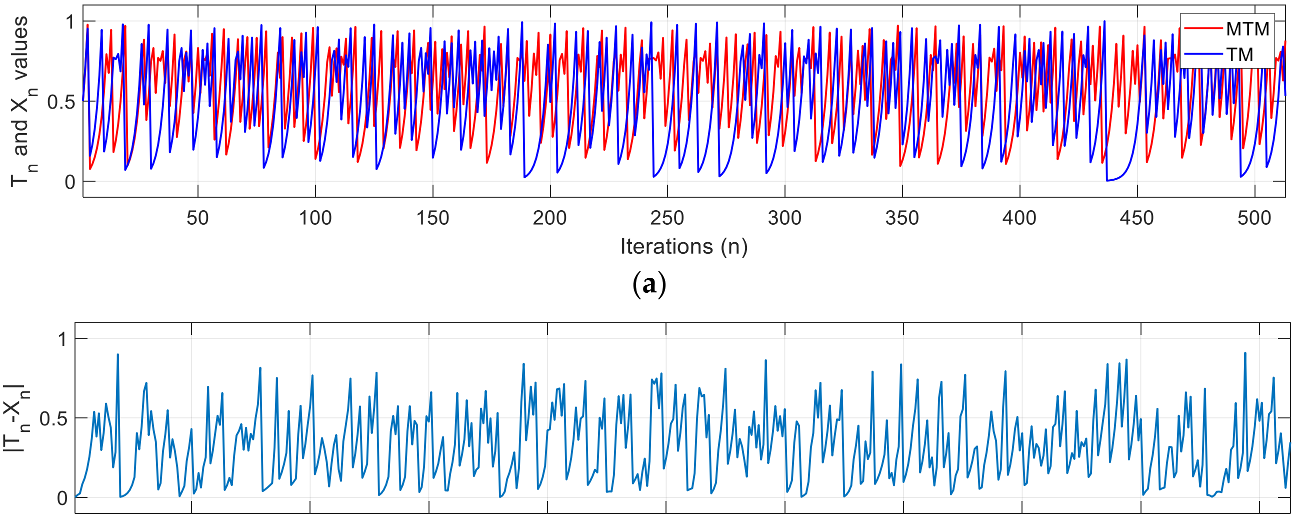


Figure 3. Bifurcation diagrams corresponding to the proposed MTM parameters with $T_0 = 0.5$.

3.4. Time Series and Sensitivity Analysis of MTM Control Parameters

The tests presented in this subsection are performed to analyze the effect of a slight variation of MTM control parameters on the time-series generated by this map. Firstly, two chaotic sequences are generated by TM for $(\lambda, T_0) = (1.9, 0.5)$ and by MTM for $(\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5, \lambda_6, \alpha, X_0) = (1, 1, 1, 1, 1, 1, 10^{-2}, 0.5)$, respectively. These sequences are displayed in Figure 4a and their difference in absolute values is displayed in Figure 4b. From this figure, one can clearly distinguish that TM and MTM generate different chaotic sequences, which validates that MTM can be considered as a new chaotic system.



(**a**)

(**b**)

Figure 4. (**a**) The first 512 iterations of chaotic sequences respectively generated by TM for $(\lambda, T_0) = (1, 0.5)$ and by MTM for $(\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5, \lambda_6, \alpha, X_0) = (1, 1, 1, 1, 1, 1, 10^{-2}, 0.5)$ with (**b**) their difference in absolute values.

In the following test, one examines the effect of a small variation in MLM parameters on the output values of this map. For this purpose, MLM is used to produce a chaotic sequence noted X_n with size L = 512 by using the parameter values listed in the caption of Figure 4. Then, each MLM parameter is modified by a slight variation of the order $\Delta = 10^{-13}$. Next, MLM is used to generate a chaotic sequence noted $X_n{}^*$ as response to the performed variation. The findings of this test are displayed in Figure 5, which indicates that a minor variation of 10^{-13} in one of the MTM parameters leads to a significant fluctuation in the output values of this map. Therefore, MTM is highly sensitive to its six control parameters.

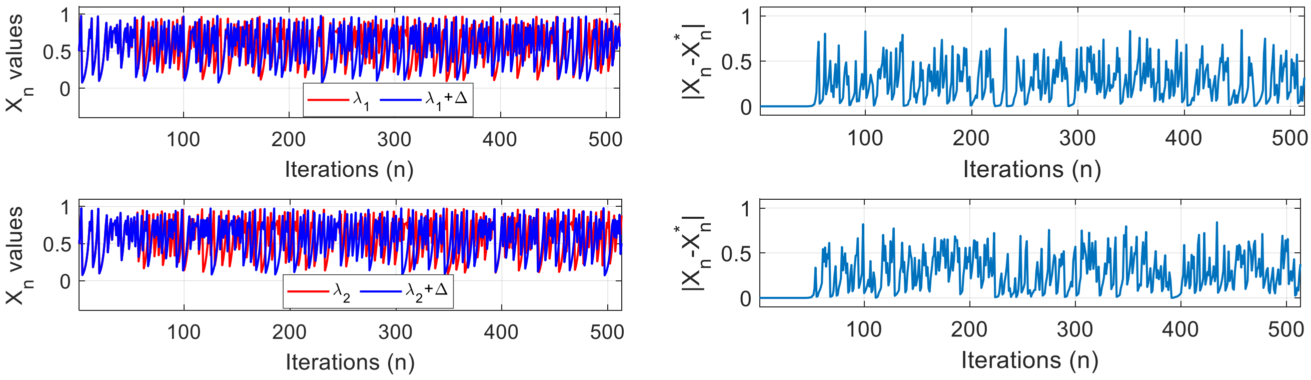


Figure 5. *Cont.*

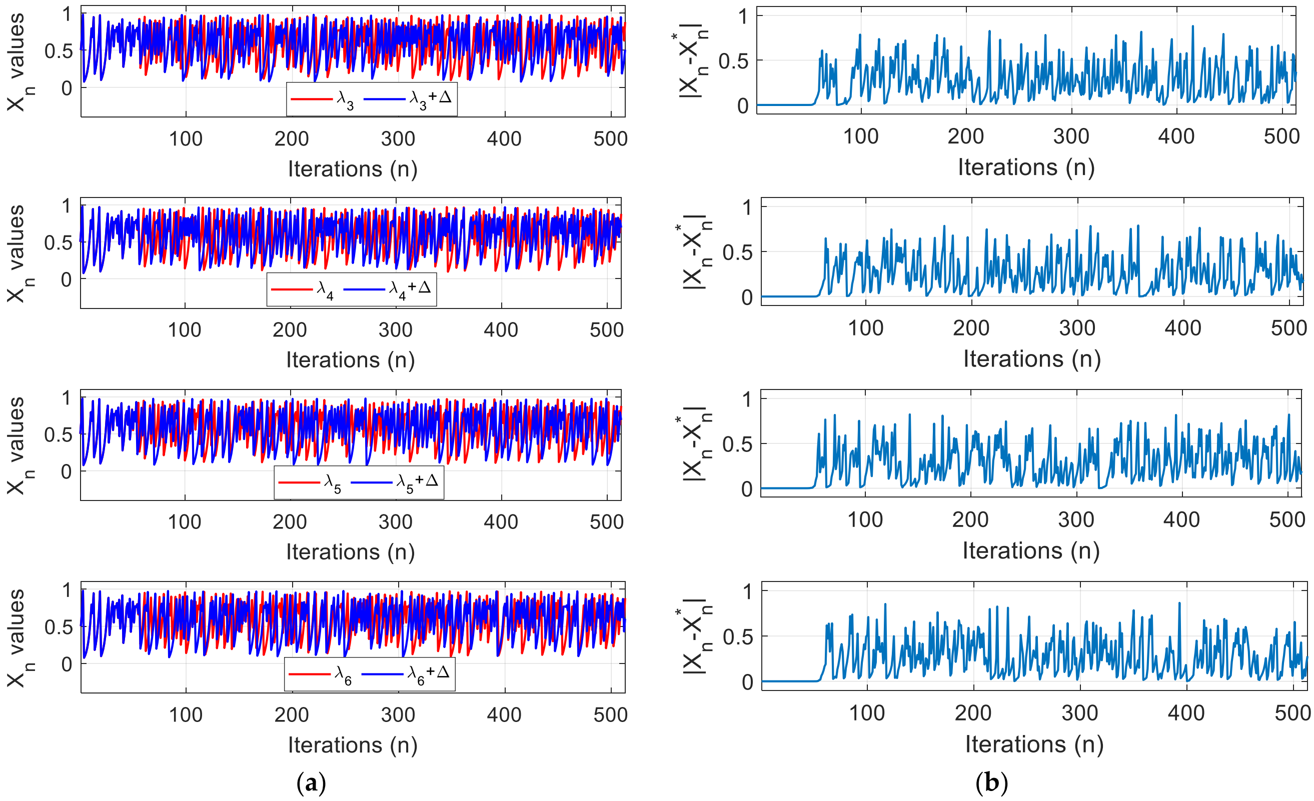


Figure 5. (**a**) Original MTM sequences (red color) and MTM sequences (blue color) generated by following a variation of its control parameters by $\Delta = 10^{-13}$. (**b**) The difference in absolute value between the chaotic sequences.

3.5. Histogram Equalization of MTM Chaotic Sequences

Generally, 1D chaotic systems produce sequence values with nonuniform distribution. This limitation of such systems restricts their application in security systems, particularly for image encryption. Figure 6 shows the distribution of chaotic sequences with different sizes generated by MTM for different values of its control parameters. One can notice from the above figure that the histograms of MTM output values are approximately flat, which means that MTM generates chaotic values with nearly uniform distribution. However, this distribution remains imperfectly uniform. In order to improve its uniformity, Algorithm 1 is involved.

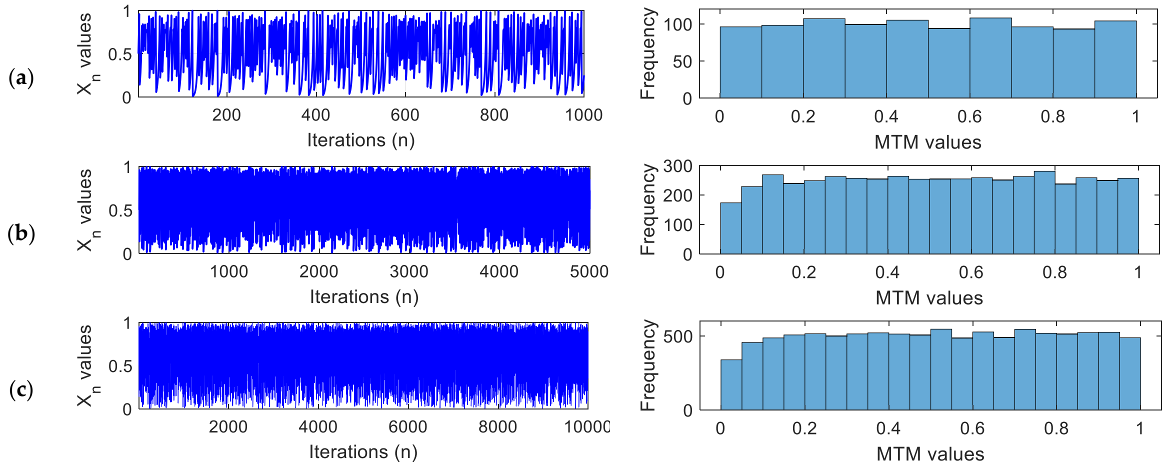


Figure 6. (**a**) Chaotic sequence with different sizes generated by MTM with their histograms with $(\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5, \lambda_6)$ are randomly set to (**a**) $(6, 2, 1, 7, 5, 3)$, (**b**) $(0.5, 2, 8.9, 3.6, 10, 0.1)$, and (**c**) $(1, 0.5, 1, 0, 1000, 0.0002)$, respectively, with $(\alpha, X_0) = (10^{-2}, 0.5)$.

Algorithm 1: Chaotic sequence normalization and histogram equalization

```
            X     Chaotic sequence of length L generated by MTM
   Inputs:  Lb    Lower bound value
            Ub    Upper bound value
   Output:  Y     Normalized chaotic sequence with equalized histogram
        1:  L = length (X)          // Get the length of X sequence
        2:  [IndX] = argsort (X)    // argsort () function [68] returns the indices of the ascending
            sorted elements of the input X
        3:  $A = (Ub - Lb)/(L - 1)$
        4:  $B = (Lb \times L - Ub)/(L - 1)$
        5:  for  i = 1 to L; i++ do
        6:      $Y(i) = IndX(i) \times A + B$
        7:  end for
        8:  return Y
```

After an ascending (or descending) sort of the chaotic sequence X of size L, Algorithm 1 produces a sequence, denoted *IndX*, of chaotic indices of size L. The latter contains integer values in the interval $[1, \ldots, L]$. where each value appears only once, which makes this sequence useful for cryptosystems where the use of chaotic sequences with nonredundant elements is strongly required, especially in the diffusion process, in order to generate an encrypted image with flat histogram. This important property can be effectively exploited in encryption schemes to avoid statistical attacks.

In order to use the *IndX* sequence in the proposed encryption scheme, it is necessary to normalize the values of this sequence in certain intervals: [0–N], [0–M] and [0–255] with $N \times M$ represents the input image size while [0–255] is the range the grayscale image values. For this purpose, each x element in *IndX* sequence is normalized (scaled) into the range $[Lb, Ub]$ according to the following formula:

$$y = (Ub - Lb)\frac{x - \min(IndX)}{\max(IndX) - \min(IndX)} + Lb \tag{4}$$

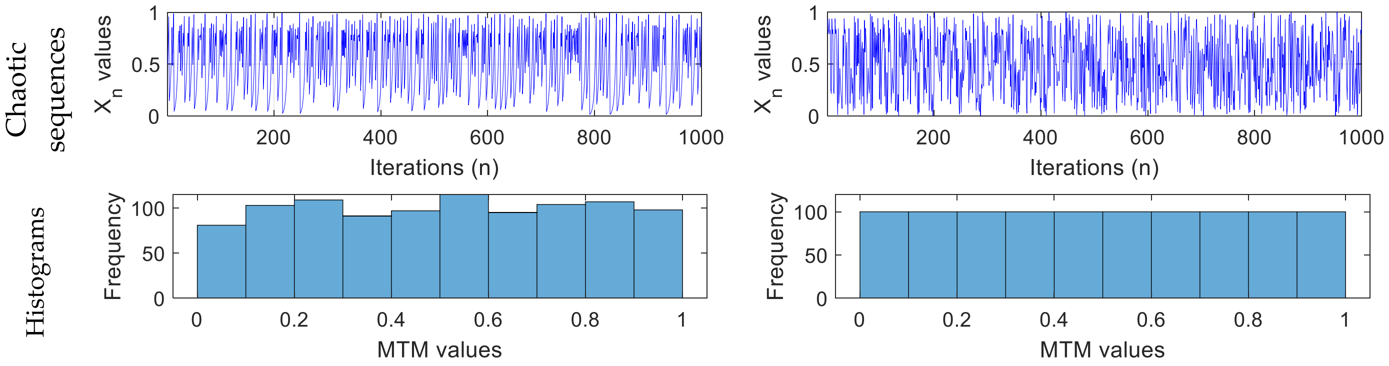
where $\min(IndX) = 1$, $\max(IndX) = L$ represent the minimal and the maximal values in the *IndX* sequence, respectively. Lb and Ub represent respectively the lower and upper bounds of the interval $[Lb, Ub]$, and y is the scaled x value in the range $[Lb, Ub]$.

By expanding Equation (4), it is easy to show that:

$$y = x \times \frac{(Ub - Lb)}{L - 1} + \frac{Lb \times L - Ub}{L - 1} \tag{5}$$

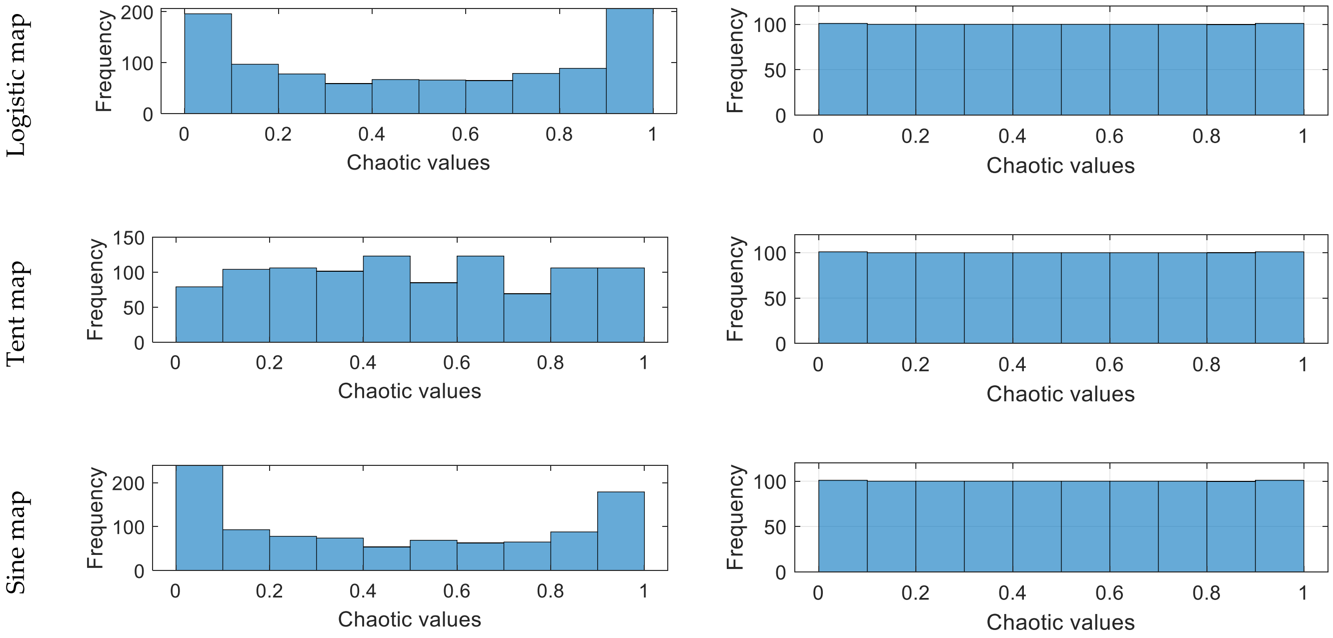
Note that Equation (5) is implemented in lines 3–7 of Algorithm 1.

To test the performance of Algorithm 1, it is used for the histogram equalization of a chaotic sequence of size $L = 1000$ generated by MTM. Figure 7 shows the illustration of original chaotic sequence and its version produced after the histogram equalization using Algorithm 1 with Lb = 0 and Ub = 1. This figure shows on the one hand that the histogram of the original sequence is not uniform. On the other hand, we can observe that following the histogram equalization by using Algorithm 1, the distribution of the chaotic sequence values becomes uniform. Therefore, Algorithm 1 is useful in equalizing the histogram of chaotic sequences generated by MTM or by other 1D chaotic maps. Indeed, Figure 8 shows the histograms of chaotic sequences each of size L = 1000 generated by the logistic map [69], tent map [70] and sine map [71]. Then, the distribution of these sequences is uniformed by Algorithm 1.



(**a**) (**b**)

Figure 7. (**a**) Original chaotic sequence produced via MTM with $(\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5, \lambda_6, \alpha, X_0) = (6, 1, 7, 3, 1, 9, 10^{-2}, 0.5)$ and its (**b**) equalized histogram version generated by MTM and Algorithm 1.



(**a**) (**b**)

Figure 8. (**a**) Histograms of the chaotic sequences generated by logistic map, tent map and sine map, respectively, with (**b**) their histograms after equalizing the distribution of their sequences by using Algorithm 1.

4. Proposed Image Cryptosystem Based on MTM and Parallel Computing

In order to illustrate the practical utility of MTM and Algorithm 1 in information security scenarios, they are utilized in designing a new image encryption scheme. The flowchart of the proposed scheme is presented in Figure 9. This system is implemented by using parallel computing on multiple cores for maximizing their performance in terms of both runtime and security level. The proposed cryptosystem involves three main processes: the generation of the confusion and diffusion keys, the diffusion of the input image, and the confusion of the diffused image. The full details of these processes are presented below.

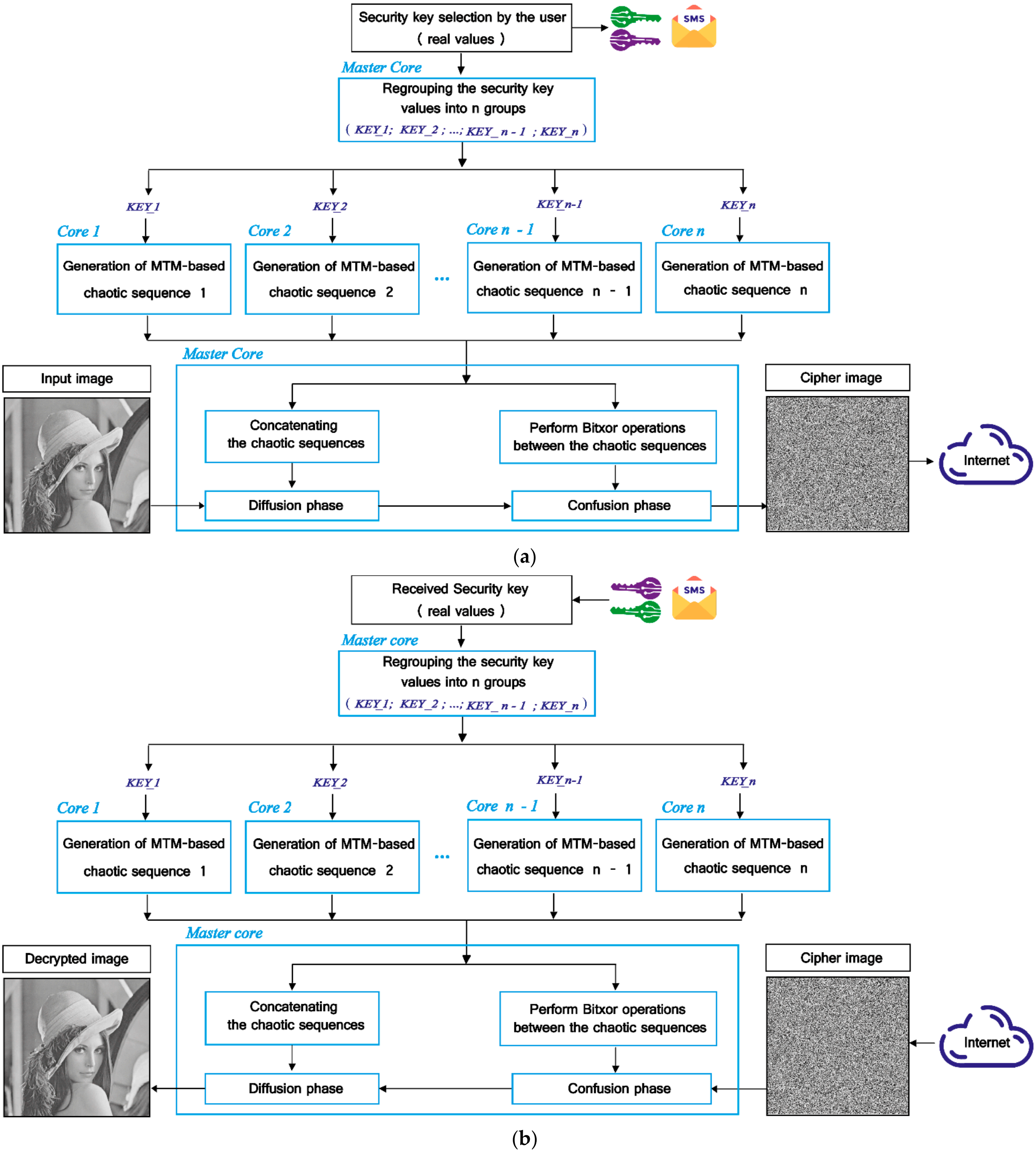

Security key selection by the user
(real values)
SMS
Master Core
Regrouping the security key values into n groups
(KEY_1; KEY_2 ; ...; KEY_ n - 1 ; KEY_n)
KEY_1
KEY_2
KEY_n-1
KEY_n
Core 1
Core 2
Core n - 1
Core n
Generation of MTM-based chaotic sequence 1
Generation of MTM-based chaotic sequence 2
...
Generation of MTM-based chaotic sequence n - 1
Generation of MTM-based chaotic sequence n
Master Core
Input image
Concatenating the chaotic sequences
Perform Bitxor operations between the chaotic sequences
Diffusion phase
Confusion phase
Cipher image
Internet
(a)
Received Security key
(real values)
SMS
Master core
Regrouping the security key values into n groups
(KEY_1; KEY_2 ; ...; KEY_ n - 1 ; KEY_n)
KEY_1
KEY_2
KEY_n-1
KEY_n
Core 1
Core 2
Core n - 1
Core n
Generation of MTM-based chaotic sequence 1
Generation of MTM-based chaotic sequence 2
...
Generation of MTM-based chaotic sequence n - 1
Generation of MTM-based chaotic sequence n
Master core
Decrypted image
Concatenating the chaotic sequences
Perform Bitxor operations between the chaotic sequences
Diffusion phase
Confusion phase
Cipher image
Internet
(b)

Figure 9. Flowchart the (**a**) encryption and (**b**) the decryption phases of the proposed cryptosystem.

4.1. Confusion and Diffusion Key Generation

The first phase of the proposed scheme consists in generating MTM-based chaotic sequences for use in the diffusion and confusion of the input image. For this purpose, the following steps are implemented.

Step 1: In this step, the sender composes a security key noted KEY of $6n$ real values, where n represents the number of the available cores on the computer CPU.

Step 2: In this step, executed at the master core level, the user-entered security key is decomposed into n sub-keys where each one contains 6 real values: $KEY = \{KEY\_1\,;\, KEY\_2\,; \ldots; KEY\_n\}$ with $KEY\_1 = (\lambda_{11}, \lambda_{21}, \lambda_{31}, \lambda_{41}, \lambda_{51}, \lambda_{61})$; $\ldots; KEY\_n = (\lambda_{1n}, \lambda_{2n}, \lambda_{3n}, \lambda_{4n}, \lambda_{5n}, \lambda_{6n})$.

Step 3: This step is executed in parallel mode on the *n* cores of the computing machine. For illustration purposes, one assumes that the following process is executed on Core 1 of the computer's processor.

(a) Use Equation (3) to produce an MTM-based chaotic sequence noted $V1$ with size $N \times M/n$ where $N \times M$ represents the size of the input image and n is the available CPU cores number. In the present process, the six components of KEY_1 are used as MTM control parameters.

(b) Select from *V*1 a sub-vector noted *V_R* of size *L* = *N* with *V_R* = *V*1(1: *N*). The latter will be useful for generating a key for confusing the pixels along the input image rows. Next, use Algorithm 1 to normalize the distribution of *V_R* between *Lb* = 0 and *Ub* = *N*. Then, the output vector values, noted *VR*_1, are rounded to unsigned integers.

(c) Select from *V*1 a sub-vector noted *V_C* of size *L* = *M* with *V_C* = *V*1(*N:N* + *M*). The latter will be useful for generating a key that confuses the pixels along the columns of the plain image. Then, use Algorithm 1 to normalize the distribution of *VC*_1 between *Lb* = 0 and *Ub* = *M*. Next, the output vector values, noted *VC*_1, are rounded to unsigned integers.

(d) Use Algorithm 1 to equalize the histogram distribution of *V*1 sequence values where the bounds of this sequence are set to *Lb* = 0 and *Ub* = 255. Then, the resulting vector values are rounded to unsigned integers coded on 8 bits (uint8), which enables to create *D*1 vector.

Step 4: At the master core level, concatenate the vectors *D*1, … , *Dn* generated by *Core* 1, … , *Core n*, respectively. This procedure leads to the creation of *VD* vector with size $L = N \times M$. Next, reshape the *VD* vector into 2D array denoted *Diff_KEY* of size $N \times M$.

Step 5: In the main core, perform the following *bitxor* operations between *VR*_1, … , *VR_n* vectors produced by core 1, … , core *n*, respectively:

$$\begin{gathered} VR = bitxor(VR\_1, VR\_2) \\ VR = bitxor(VR, VR\_3) \\ \vdots \\ VR = bitxor(VR, VR\_n) \end{gathered} \tag{6}$$

where the *bitxor* function meets the rule:

$$k = bitxor(i, j) \Leftrightarrow i = bitxor(k, j) \text{ with } i, j, k \in \{0, 1\} \tag{7}$$

Step 6: In a similar way to Step 5, generate *VC* vector of size $1 \times M$ as follows:

$$\begin{gathered} VC = bitxor(VC\_1, VC\_2) \\ VC = bitxor(VC, VC\_3) \\ \vdots \\ VC = bitxor(VC, VC\_n) \end{gathered} \tag{8}$$

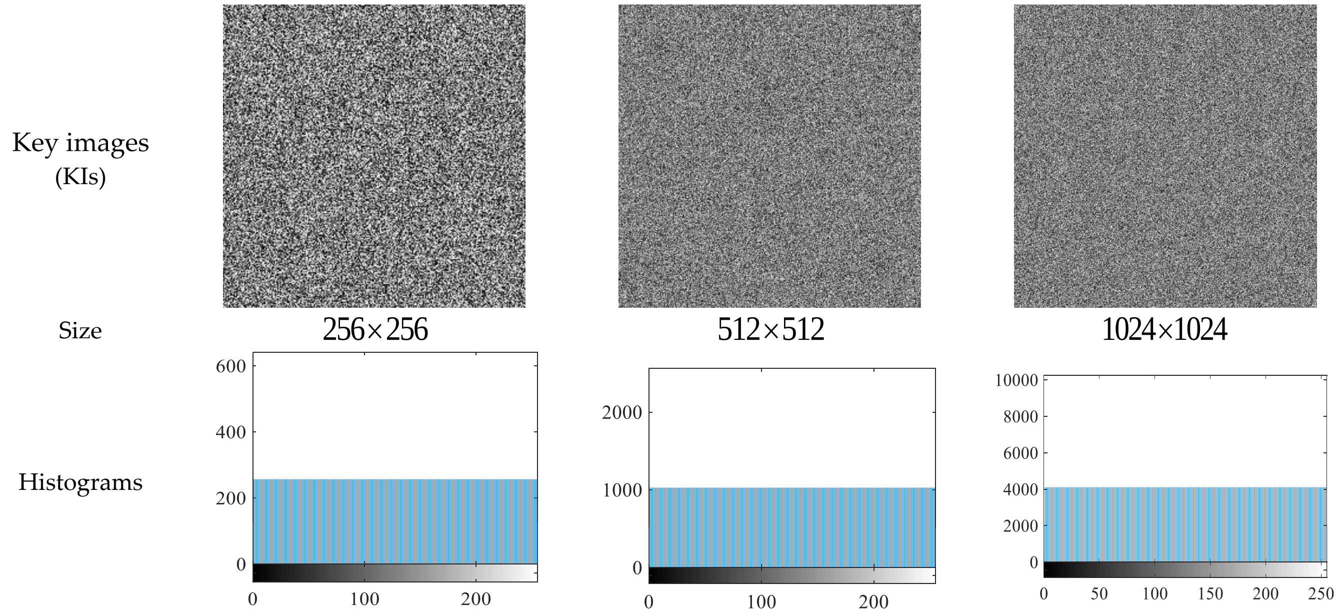
The *VC* vector is constructed to scramble the pixels along the input image columns.

It should be mentioned that steps 5 and 6 generate *VR* and *VC* vectors of size $1 \times N$ and $1 \times M$, respectively. These vectors will then be used in the confusion phase of the input image. These vectors provide logical links between the chaotic sequences derived from each CPU core. In other words, steps 5 and 6 ensure that if an incorrect value is used in the *6n* components of the security key (KEY), the receiver cannot decrypt the ciphered image. Therefore, these steps significantly improve the security level of the proposed cryptosystem.

Figure 10 shows examples of *KEY_Diff* arrays of various sizes produced by the proposed method. For this purpose, a computer equipped with RAM of 4 GB and CPU incorporating four-core technology with speed of 2.40 GHz is used. To perform the parallel computation on the four cores of the machine, the Python package MPI (Message Passing Interface) [72] is used. The latter allows us to run Python applications on machines with multiple processors. From Figure 10, we can notice that the histograms of *KEY_Diff* arrays are uniform, which indicates that the use of such keys in the diffusion process can prevent statistical attacks. In addition, the correlation coefficient (CC) values of *KEY_Diff* are computed along the horizontal, vertical and diagonal directions by using the next formula:

$$CC = \frac{C(I,J)}{\sqrt{V(I)}\sqrt{V(J)}} \tag{9}$$

where $C(I,J)$ is the covariance between two adjacent pixel sequences (I and J) that are selected from the *KEY_Diff* matrix. $V(I)$ and $V(J)$ are the variance in I and J sequences, each of size 8000 samples, respectively. Obviously, if the CC tends to zero, the neighboring I and J values are considered independent. From the results shown in Figure 10, one can observe that the CC values tend towards zero, which confirms the suitability of *Diff_KEY* for use in the next diffusion phase.



Horizontal CC	−0.0022	0.0016	−0.0022
Vertical CC	0.0016	−0.0025	0.0019
Diagonal CC	0.0023	0.0021	0.0016

Figure 10. Key images (KIs) of different sizes generated by the proposed method.

4.2. Diffusion Phase

The current phase allows the modification of the input image's statistical properties in order to avoid crypto analysis statistical attacks. The current phase is obtained by applying the next operation:

$$D = bitxor(In, Diff\_KEY) \tag{10}$$

where *In* refers to the input image of size $N \times M$ and *bitxor()* refers to the bit-wise XOR operation.

To further boost the security of our scheme, the next phase is performed with the aim of re-distributing the pixels within the diffused image.

4.3. Confusion Process

The current phase redistributes the pixels of the diffused image in a pseudo-random manner throughout the entire image area, which reduces the correlation between adjacent pixels. As result, the security level of an encryption system becomes higher and the cryptosystem becomes more resistant to cropping attacks. To this end, the following steps are involved.

Step 1: Use VR vector of size $1 \times N$ to confuse the pixels along the D image rows as follows:

$$C1(i,:) = circ\_shift(D(i,\,:), VR(i)), \quad i = 1, 2, \ldots, N \tag{11}$$

where *circ_shift*(*I*,*j*) designates the circular shifting operation that circularly translates the *I* vector elements by *j* positions [73].

Step 2: Use VC vector of size $1 \times M$ to confuse the pixels along the $C1$ image columns as follows:

$$C2(i,:) = circ\_shift(C1(i,\,:), VC(i)), \quad i = 1, 2, \ldots, M \tag{12}$$

Step 3: To guarantee a maximum distribution of D image pixels, a third round of circular shifting operation is performed for $C2$ rows using the VRC key, which is obtained by the following *bitxor* operation:

$$VRC = bitxor(VC(1:N), VR(1:N)) \text{ with } N \leq M \tag{13}$$

Next, the confusion of $C2$ is performed according to the next circular shifting process:

$$EI = bitxor(C2(i,:), VRC(i)) \text{ with } i = 0, 1, \ldots, N \tag{14}$$

where *EI* represents the encrypted image according to the proposed scheme. Figure 11 shows an illustration of our confusion method applied to 8×8 matrix by using three chaotic sequences.

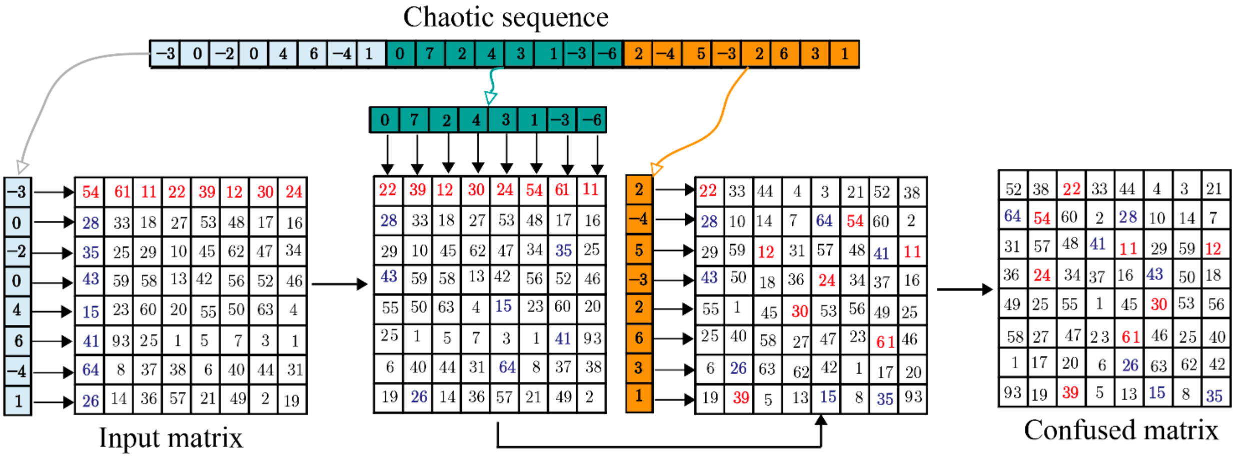
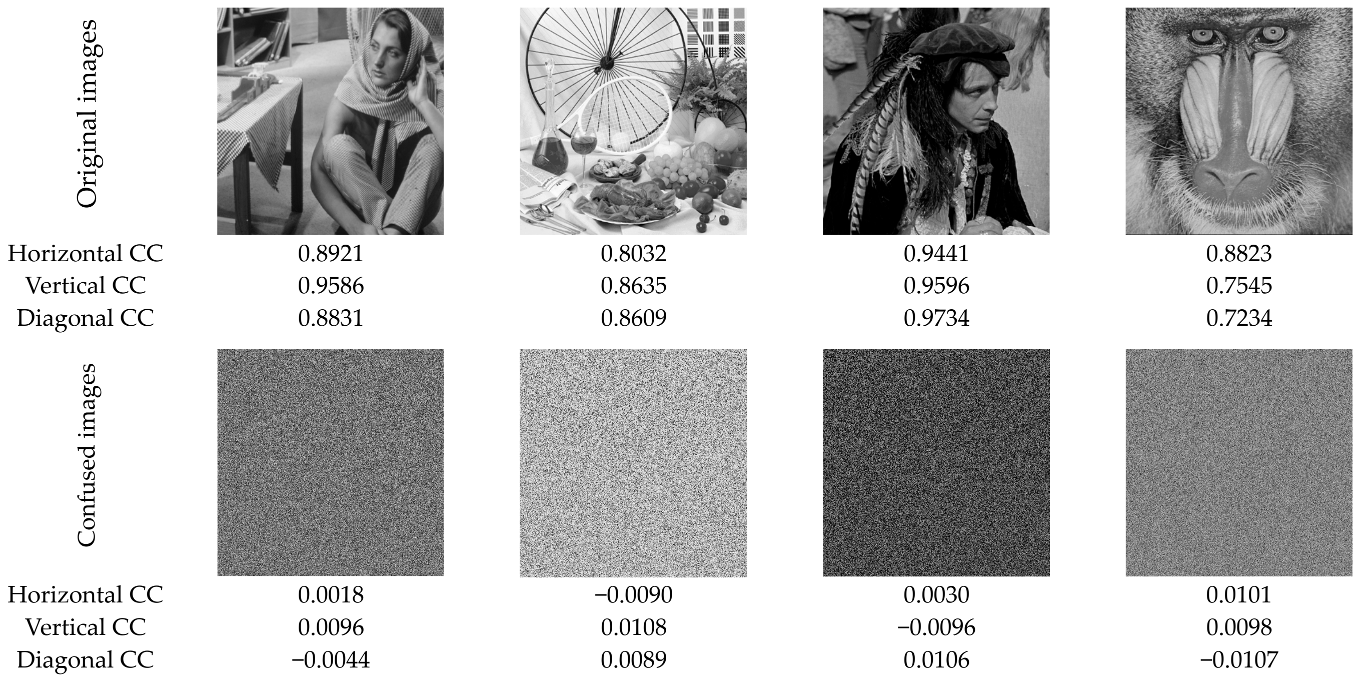


Figure 11. Confusion process of 8×8 matrix rows and columns using a chaotic sequence.

To highlight the performance of the confusion method used, it is employed for confusing standard 512×512 grayscale images selected from the database [74]. Then, the CC values are computed for random 8000 adjacent pixels selected from the original images and its confused versions along three directions: horizontal, vertical, and diagonal. The outcomes of the performed test are plotted in Figure 12. The achieved results indicate that the confusion method used reduces significantly the correlation between adjacent pixels as the CC values tend to zero within the confused images. Moreover, one can notice that the visual information of the original confused fully hides the input images data, which indicates that the confusion method used is efficient.



Horizontal CC	0.8921	0.8032	0.9441	0.8823
Vertical CC	0.9586	0.8635	0.9596	0.7545
Diagonal CC	0.8831	0.8609	0.9734	0.7234
Horizontal CC	0.0018	−0.0090	0.0030	0.0101
Vertical CC	0.0096	0.0108	−0.0096	0.0098
Diagonal CC	−0.0044	0.0089	0.0106	−0.0107

Figure 12. Standard images of sizes 512 × 512 and their confused versions with the corresponding CC values.

To further support the validity of the confusion method used, it is employed for retrieving the visual information from confused cropped test images, as shown in Figure 13. From the latter, it becomes apparent that the visual information of the retrieved images from the confused cropped ones remains recognizable even if the scrambled images are cropped up to 25%. These results demonstrate the validity of the suggested MTM-based confusion method.

After demonstrating the good performance of the suggested confusion and diffusion phases, the following section presents the overall performance of the suggested cryptosystem.

To encrypt/decrypt a color image I of size $3 \times N \times M$ defined in the RGB color space by the proposed cryptosystem, it is necessary to decompose this image into three grayscale level channels (R, G, and B) each of size $N \times M$. Then, each channel is encrypted by the proposed method to obtain its encrypted version. Next, the encrypted channels are concatenated to produce the encrypted color image of size $3 \times N \times M$.

It should be noted from Figure 9 that the proposed encryption scheme is symmetric. Thus, the user must follow the reverse procedure of the encryption phase indicated in Figure 9b for recovering the original image while using the correct security key.

5. Simulation Results with Discussions

The present section covers the various experiments undertaken to validate the good performance of the proposed MTM-based encryption algorithm. It is worth mentioning that all the simulations in this section are performed using a PC equipped with RAM of 4 GB and CPU of 4 cores and frequency equal to 2.40 GHz and four cores. In addition, Python v3.7 is used as programming language, where the Python package MPI (Message Passing Interface) [72] is used to implement the proposed scheme in parallel mode on the four CPU cores.

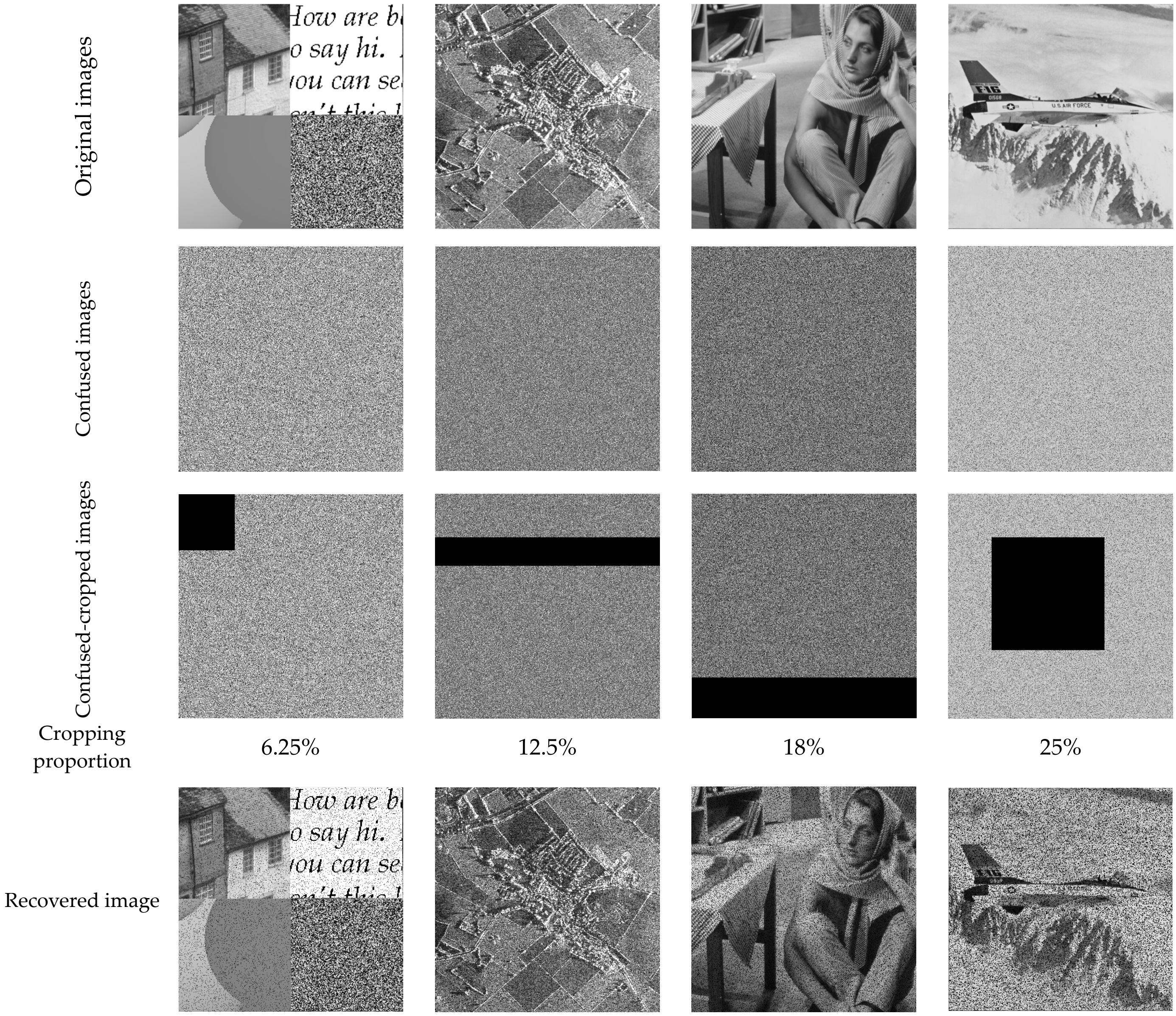


Figure 13. Standard test images of sizes 512 × 512 with its confused, confused cropped, and recovered versions by using the proposed scrambling method.

5.1. Sensitivity Analysis of the Secret Keys

Typically, a high-security image cryptosystem is intended to be extremely sensitive to any variation in its security keys. Indeed, a small variation in any key parameters must lead to a complete failure in recovering the original image from the ciphered one. To test the sensitivity of the proposed scheme to its security keys, the following test is conducted. For this purpose, a set of 512 × 512 test images is selected from the databases [74,75]. These images are then encrypted/decrypted by the proposed algorithm. The security key used in the encryption phase is $KEY = \{KEY\_1;\ KEY\_2;\ KEY\_3;\ KEY\_4\}$. Each KEY_i $(i = 1, \ldots, 4)$ is formed by six real values that represent the MTM control parameters. To simplify the presentation of the proposed scheme outputs, let us select $KEY = KEY\_1 = KEY\_2 = KEY\_3 = KEY\_4 = \{\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5, \lambda_6\} = \{8, 1, 4, 2, 5, 9\}$ with $x_0 = 0.5$ and α= 0.09. Subsequently, during the decryption phase, only one parameter of the latter is varied by $\Delta = 10^{-13}$, and the decrypted images are plotted in Figure 14. From this figure, it appears on the one hand that the original image is successfully decrypted only if a symmetrical security KEY is used in both encryption and decryption phases. On the other hand, the test results demonstrate that any modification by ∓Δ of any parameter of KEY leads to complete failure in recovering the original images. These findings provide a strong indication about the dependence of the proposed encryption scheme on its security KEY.

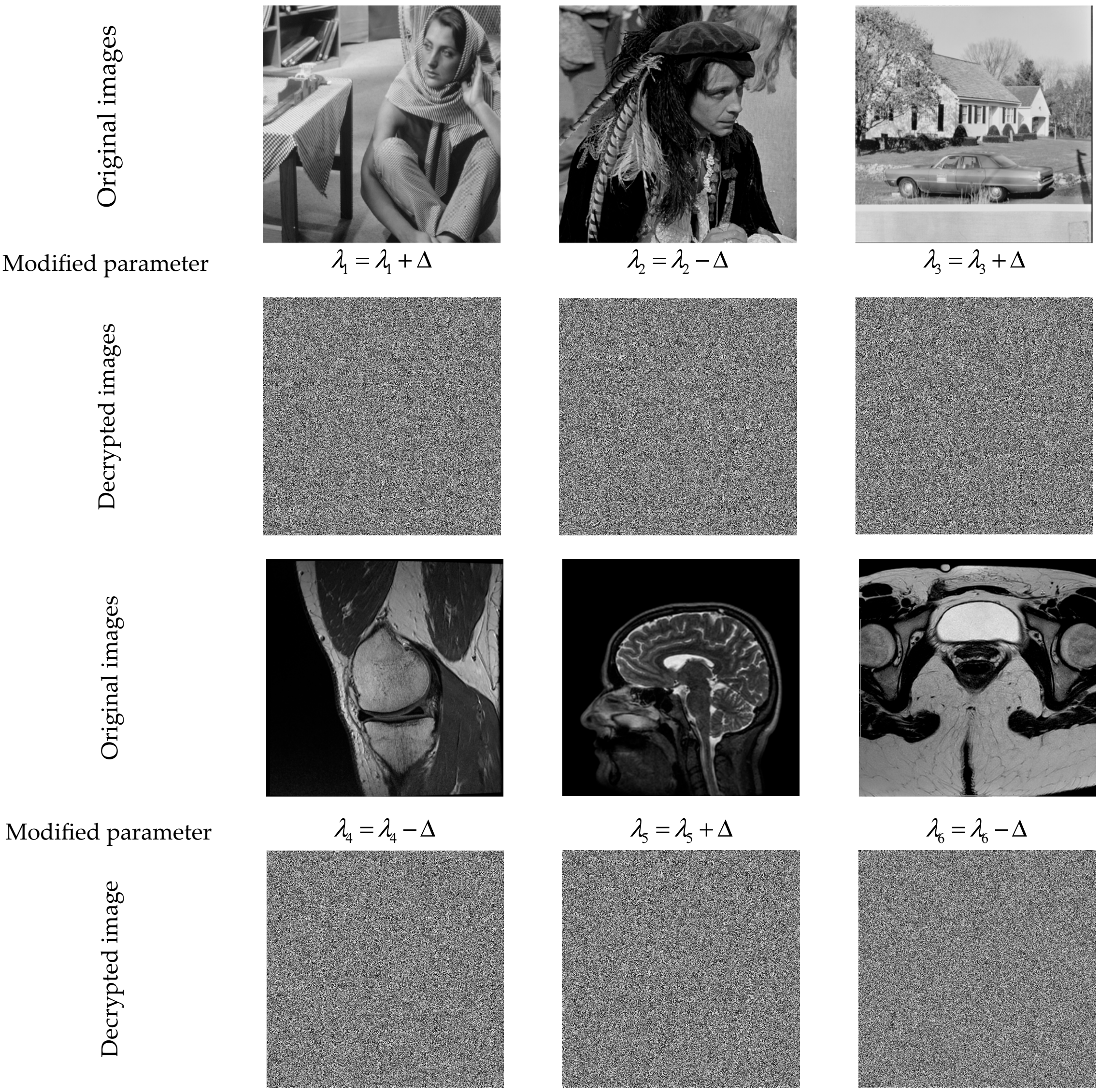

Original images
Modified parameter
$\lambda_1 = \lambda_1 + \Delta$
$\lambda_2 = \lambda_2 - \Delta$
$\lambda_3 = \lambda_3 + \Delta$
Decrypted images
Original images
Modified parameter
$\lambda_4 = \lambda_4 - \Delta$
$\lambda_5 = \lambda_5 + \Delta$
$\lambda_6 = \lambda_6 - \Delta$
Decrypted image

Figure 14. Grayscale test images of size 512 × 512 with its encrypted versions and decrypted ones following a minor modification of KEY parameters by $\mp\Delta = 10^{-13}$.

5.2. Space of Our Scheme's Secret Keys

The security KEY size of our scheme is composed of $6n$ real type values. By considering the accuracy of floating point computing and sensitivity of the KEY parameters (see the previous subsection), the total space of the proposed security scheme key equals $\left(10^{13}\right)^{6n} = 10^{78n}$, with n being the number of the available cores on the machine's CPU. In this work, we use a quad-core CPU (n = 4) of PC, so the key space of the proposed cryptosystem is about $\left(10^{13}\right)^{6\times4} = 10^{312} \cong 2^{1033}$, considering that the sensitivity order of each KEY parameter is $\mp\Delta = 10^{-13}$. Thus, the key space of our scheme far exceeds the minimum recommended key space—2^{100} [76]—to resist brute-force attacks. Table 2 presents a comparison between the proposed encryption scheme and other schemes recently presented [29–32,77–79]. The comparison is performed in terms of the key space. The comparison outcomes indicate the superiority of the proposed scheme over the competing schemes. That is, the proposed scheme can provide more security when communicating images via unsecured communication channels.

Table 2. Comparison in terms of security key space of recent encryption scheme, including the proposed one.

Encryption Scheme	Proposed	Ref. [31]	Ref. [79]	Ref. [77]	Ref. [78]	Ref. [29]	Ref. [32]
Key space	2^{1033}	2^{630}	2^{564}	2^{505}	2^{507}	2^{480}	2^{371}

5.3. Histogram Analysis

The histogram feature provides statistical information concerning the image's content. Thus, an attacker can predict the content of an encrypted image from the analysis of its histogram. In order to hide the statistical information of encrypted images, cryptosystems attempt to generate encrypted images with equalized histograms, which prevent attacks by statistical analysis of encrypted images. To assess the ability of the proposed scheme to withstand statistical attacks, it is used for the encryption of standard and color images selected from the dataset [80]. Figure 15 shows the original test color images with their encrypted versions and the corresponding histograms. From this figure, it can be seen that our scheme is able to produce encrypted images with balanced (flat) histograms, regardless of the visual content of the input images. These results can be explained by the fact that our scheme uses diffusion keys with balanced histograms generated, which ensures the generation of encrypted images of near-fully balanced histograms. The findings of the current test are a strong indication of the robustness of our scheme against statistical attacks.

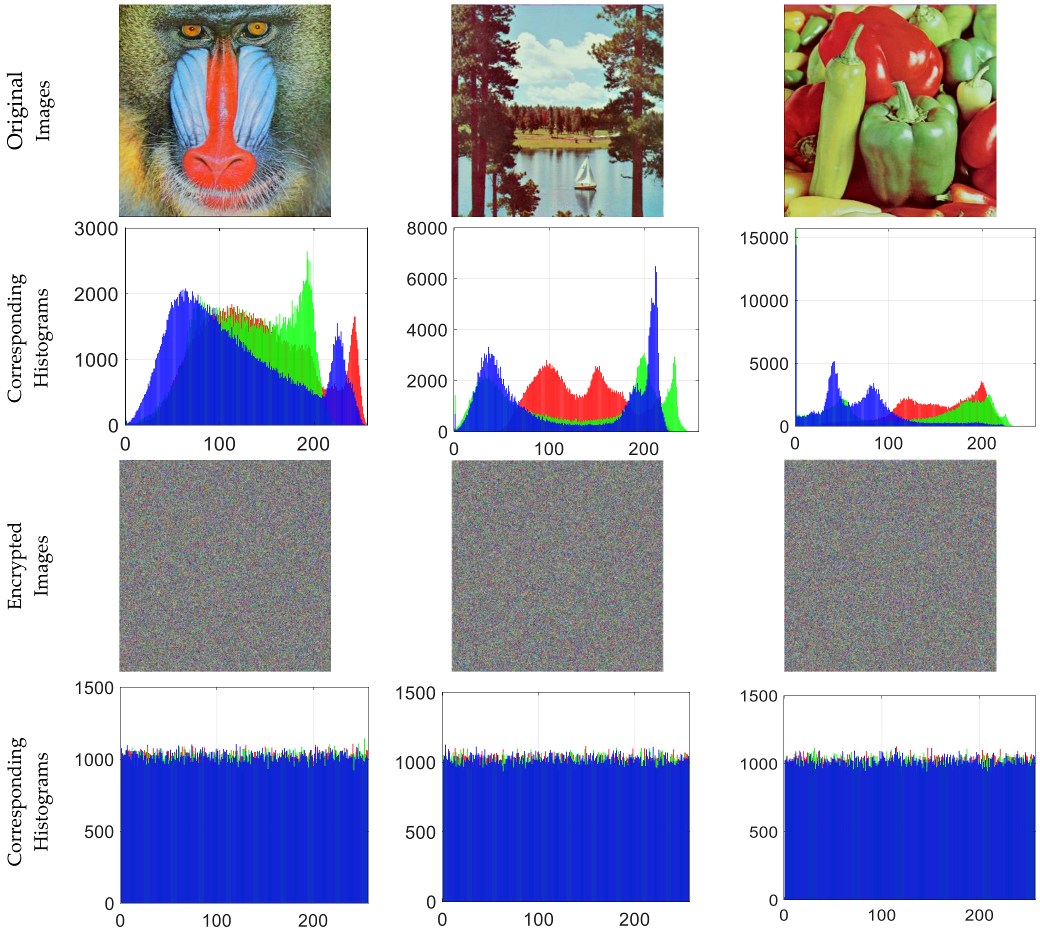
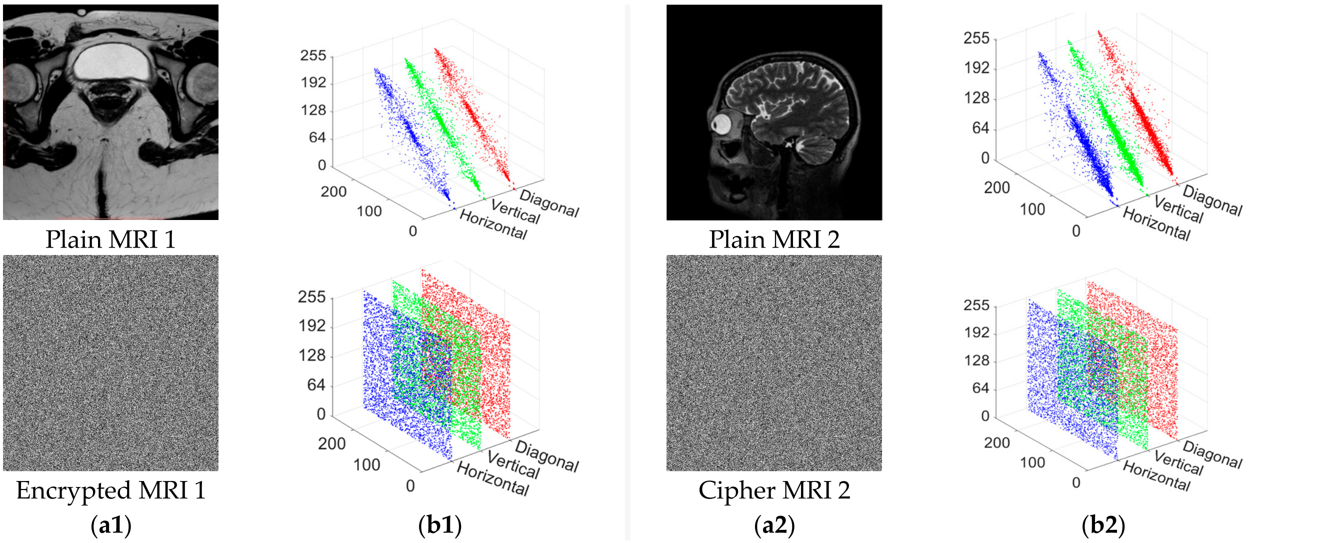


Figure 15. Color test images of size 512 × 512 with their encrypted versions and the corresponding R-, G-, and B-channel histograms.

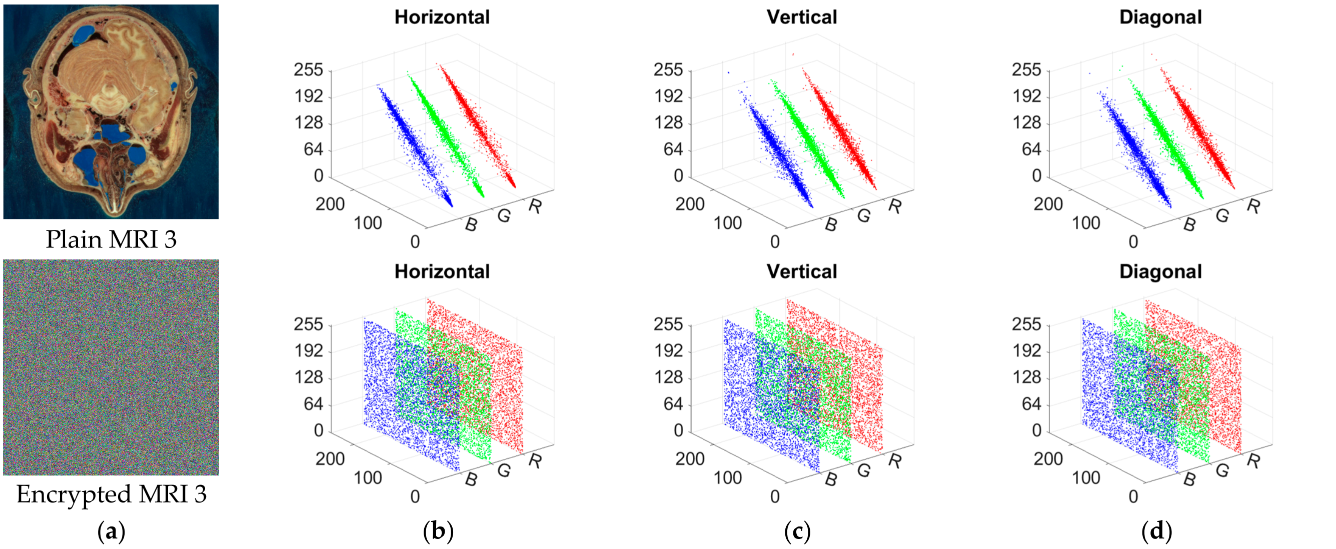
5.4. Correlation Analysis

Naturally, the digital images are captured with certain information redundancy, which causes a high degree of correlation of adjacent values within the captured image. To reduce this correlation, confusion methods are involved in the encryption scheme. A successful encryption scheme is able to significantly reduce the correction of neighboring pixels within a given input image. To assess the capacity of the proposed scheme to greatly reduce correlations between neighboring pixels, 4000 adjacent pixels are arbitrarily selected from the original images and their encrypted versions in three directions: horizontal, vertical, and diagonal. Then, (7) is used to measure the correlation between the selected pixels. Figures 16 and 17 show grayscale and color medical images selected from the databases [75,81], respectively. The CC scores related to the test images and their ciphered version are listed in Table 3. Note that for MRI 3 (Figure 17), the CC are calculated from the averaged CC of the three-color image channels.



(**a1**) (**b1**) (**a2**) (**b2**)

Figure 16. (**a1**,**a2**) Original medical images of size 1024 × 1024 with their encrypted versions and (**b1**,**b2**) the corresponding 4000 pairs of adjacent pixels in the horizontal, vertical, and diagonal directions.



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

Figure 17. (**a**) Original color medical image of size 1024 × 1024 with their encrypted versions and (**b**) the corresponding pairs of 4000 adjacent pixels in the (**b**) horizontal, (**c**) vertical and (**d**) diagonal directions.

Table 3. CC values in the horizontal, vertical and diagonal directions of the original medical images and their encrypted versions.

	MRI 1	Ciphered MRI 1	MRI 2	Ciphered MRI 2	MRI 3	Ciphered MRI 3
Horizontal	0.9613	0.0062	0.9156	0.0051		0.0039
Vertical	0.9150	0.0082	0.9653	−0.0016		−0.0005
Diagonal	0.9482	−0.0072	0.9611	0.0082		−0.0019

The results of the present analysis show on the one hand that the distribution of adjacent pixels in the plain images are around the diagonal, which indicates the presence of strong correlation between the pixels. By contrast, the adjacent pixels in the encrypted images exhibit a flat distribution in the plot, which reflects the weak correlation between the pixels within the ciphered images. Therefore, the proposed encryption scheme is effective in breaking the correlation between the input image pixels.

5.5. Robustness against Differential Attack

When attackers attempt to crack cryptosystems, they can deploy differential attacks. By such attacks, the hacker introduces modifications to the input plaintext image and analyses the effect of performed modifications on the output ciphered image, which may lead to finding a correspondence between the plain image and its encrypted version and thus cracking an encryption system [82]. To evaluate the robustness of an encryption scheme against differential attacks, the criteria of number of pixels change rate (NPCR) and unified average changed intensity (UACI) are usually employed. These metrics are defined below [83]:

$$UACI = 100 \times \frac{\sum_{i,j}\left|I_{i,j} - I'_{i,j}\right|}{255 \times N \times M};\ i = 1, \ldots, N \text{ and } j = 1, \ldots, M \tag{15}$$

$$\begin{array}{c} NPCR = 100 \times \frac{\sum_{i,j} DI_{i,j}}{N \times M} \\ \text{with } DI_{i,j} = \begin{cases} 0 \text{ if } \mathrm{I}_{i,j} = I'_{i,j} \\ 1 \quad \text{Otherwise} \end{cases} \end{array} \tag{16}$$

where $I_{i,j}$ and $I'_{i,j}$ denote the original image and its modified version, respectively.

Since the proposed encryption system is very sensitive to any minor variation by $\delta = 10^{-13}$ of its control parameters, we exploit this property to avoid differential attacks. For this, a small constant is defined (i.e., $\delta = 10^{-11}$). Then, this constant is used to increment one of the proposed system security KEY parameters (i.e., $\lambda_1 = \lambda_1 + \Delta$). This addition is performed at each iteration of the proposed encryption algorithm, resulting in the generation of a unique security key for each iteration of the cryptosystem. That is, the same input image produces different ciphered version for different iterations of the proposed scheme. To test the ability of this method to withstand differential attacks, it is used when encrypting an "Einstein" image of size 256 × 256, "Mandrill" image of size 512 × 512 and {"Pirate", "White", "Black"} images of size 1024 × 1024. These images are encrypted in two consecutive iterations of the proposed cryptosystem, and the achieved results are reported in Figure 18. In accordance with the discussion given in [83], the values of NPCR and UACI obtained by the proposed algorithm indicate the efficiency of our encryption scheme to resist the differential attacks.

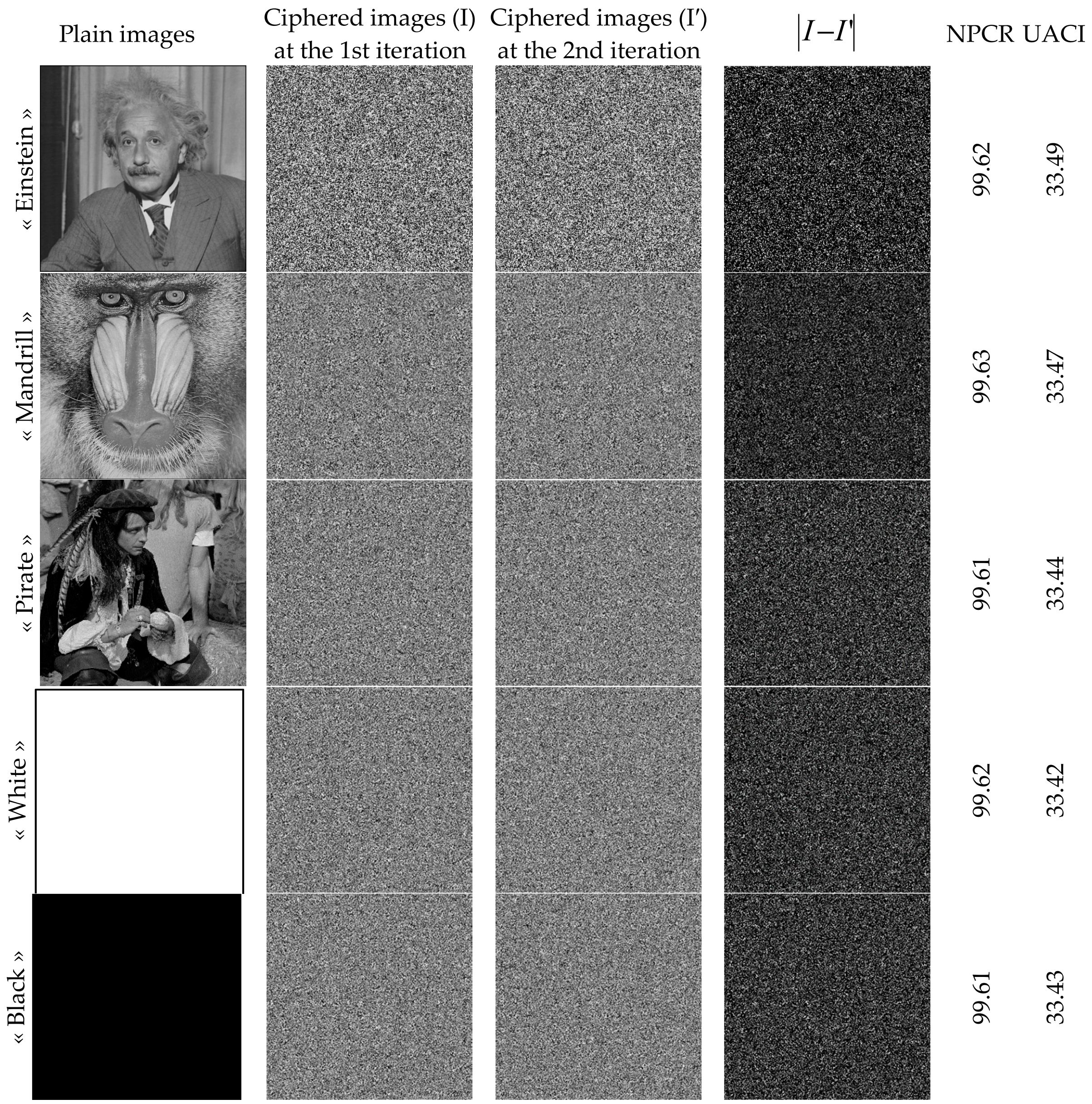


Figure 18. Plain {"Einstein", "Mandrill", "Pirate", "White", "Black"} images with their encrypted versions at two consecutive iterations of the proposed scheme with the corresponding NPCR and UACI values.

5.6. Noise and Data Loss Robustness Analysis

When transmitting encrypted images using communication protocols such as User Datagram Protocol (UDP) [84], it is possible to damage the transmitted images due to noise or data loss. For this purpose, our scheme is tested against noise and data loss effects. To perform the current test, the encrypted "Lena" image is cropped by raising data loss to 50%. Then, the suggested scheme is used to decrypt the attacked image. The outcomes of the current test are shown in Figure 19. These results indicate that the decrypted image retains distinguishable visual features, despite the loss of significant data content from the encrypted image. Thus, the proposed scheme appears strongly resistant against the data loss issue. The test related to the noise effect on the encrypted image is reported in Figure 20. The test findings indicate that the decrypted "Lena" image from its encrypted noised versions is still visually recognizable, indicating that the suggested algorithm maintains its validity in noisy environments.

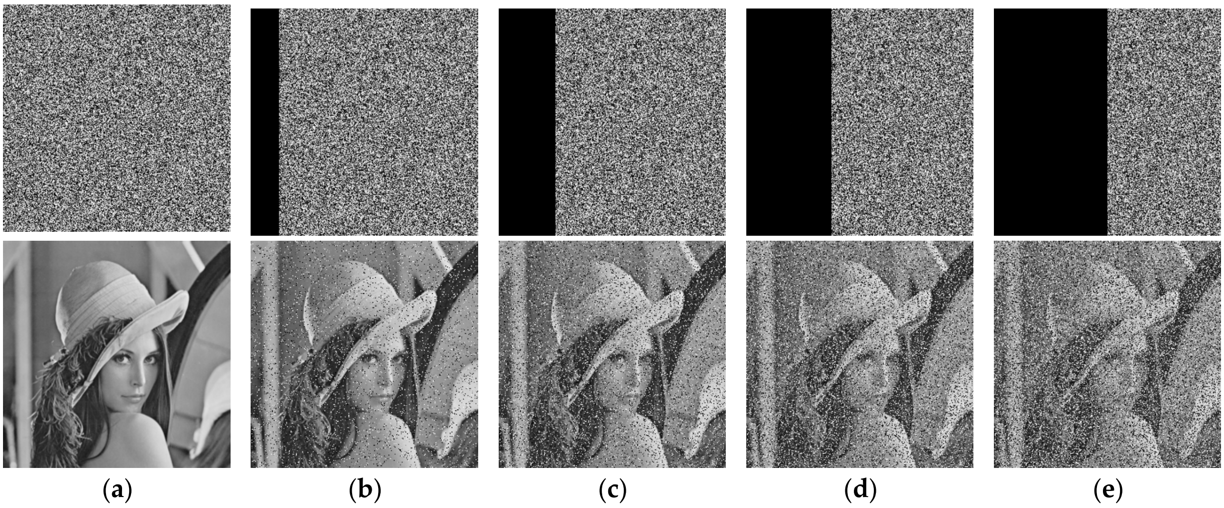


Figure 19. Encrypted "Lena" image in decrypted version after losing (**a**) 0, (**b**) 12.50%, (**c**) 25%, (**d**) 37.50% and (**e**) 50% of the encrypted image data.

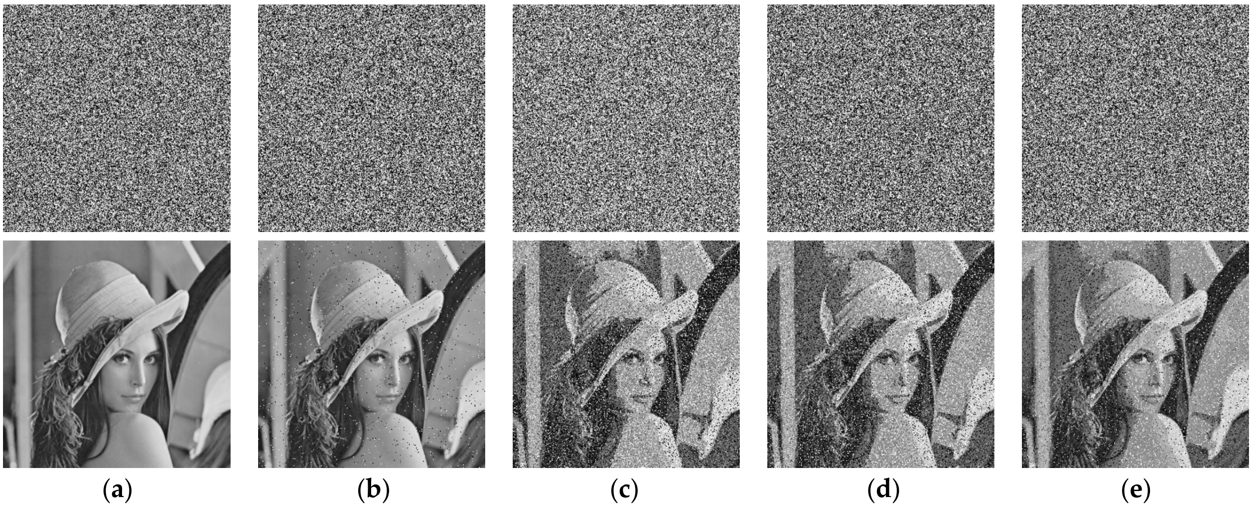


Figure 20. Encrypted "Lena" image and its decrypted versions after: (**a**) 0 % noise addition, (**b**) 3% salt and pepper, (**c**) 3% Gaussian noise, (**d**) 3% speckle noise and (**e**) Poisson noise.

5.7. Randomness Analysis Test

To measure the randomness in an input image, Shannon entropy (E) is widely used as criterion that is defined by:

$$E = -\sum_{i=1}^{r} P(K_i) \log_2 \frac{1}{P(K_i)} \tag{17}$$

where $K_i, i = 0, 1, \dots, r$ are the pixel values with $r = 255$ for grayscale images. $P(K_i)$ represents the probability of K_i symbol. In a grayscale image, all the pixels are randomly distributed in the encrypted image if E = 8.

To perform the present test, we use standard grayscale images shown in Figure 21. These images are then encrypted by the proposed method and the entropy values (E) are calculated for both original images and these encrypted versions. From the results presented in Figure 21, we can notice that the E values are very close to 8, which indicates that the proposed encryption scheme leads to the generation of encrypted images with highly random grayscale values. This result is explained by the fact that the *Diff_Key* generated via our scheme has an equalized histogram.

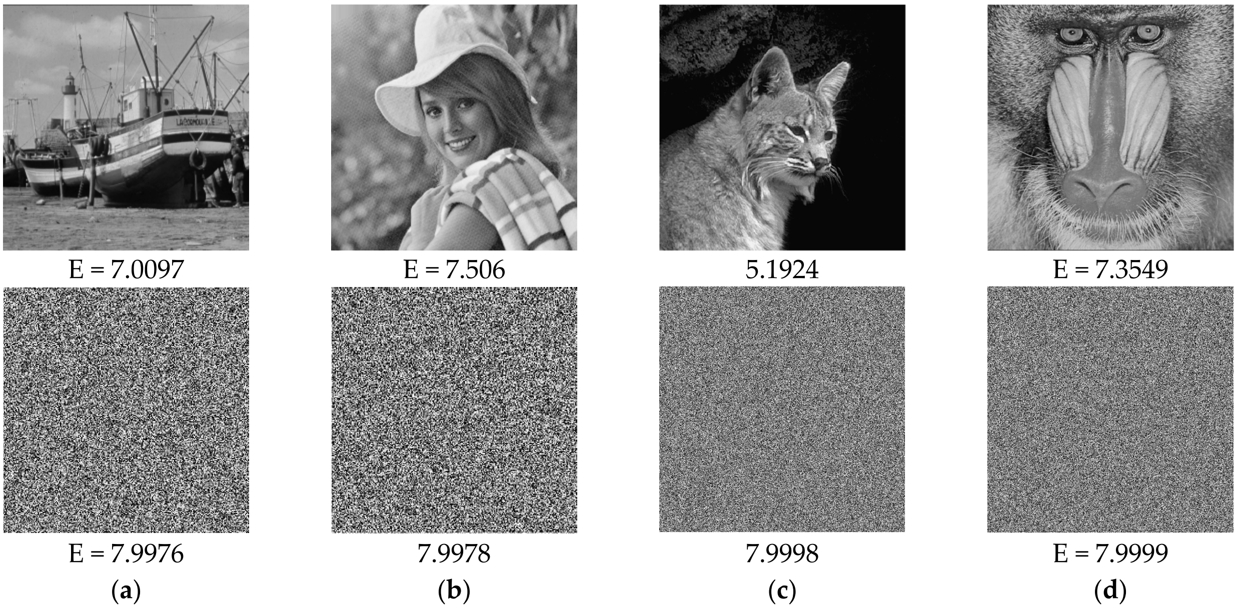


Figure 21. Test images of sizes (**a**,**b**) 256 × 256, (**c**,**d**) 512 × 512 and their encrypted versions through the proposed scheme with the corresponding entropy values.

5.8. Robustness to Classical Attacks

Kerckhoff's principle assumes that the cryptanalyst has all the information about the cryptosystem, with the exception of the security keys (Pareek et al., 2005). Therefore, the robustness of our system is discussed in this section against classical attacks, which can be easily used by a cyberattacker. There are four classical attacks that can be used by an attacker (Wang et al., 2012):

(i). Known plaintext: the hacker disposes of an encrypted string with its plaintext version.
(ii). Cipher text only: the hacker only has a string of encrypted text.
(iii). Chosen plaintext: the hacker is able to access to the encryption process for restricted period. He/she can then randomly use a plaintext to create its ciphered form.
(iv). Chosen cipher text: the hacker can access to the decryption system for a limited period. He/she can then select a random string of cipher text and construct its plaintext format.

Obviously, the most powerful attack is the chosen plaintext attack. Thus, by demonstrating that a cryptographic system is capable of resisting this attack, one can affirm that this system is capable of withstanding the classical attacks (Wang et al., 2012).

Since the proposed system is very sensitive to its control parameters, we adopt a strategy that assigns a unique security key to each input image in order to resist classical attacks. This strategy is summarized as follows:

(i). Define a security KEY to encrypt the first image in a dataset.
(ii). Choose one (or more) parameter(s) of the KEY, and then increment the selected parameter by a small constant value (e.g., $\lambda_1^* = \lambda_1 + \Delta$ with $\Delta = 10^{-13}$) to generate a new security key (KEY*) that is used to encrypt the next image in the dataset, and so on. This strategy creates dynamic security keys, which prevents conventional attacks. For more information regarding this strategy, the reader is referred to [14,85].

5.9. Comparative Analysis

This section provides a comparison between the performance of the proposed encryption scheme with similar parallel-based computing schemes presented in [29,31,32,77,78]. The comparison is conducted in terms of entropy values, CC values and encryption/decryption runtime in seconds. For this purpose, we use two groups of standard grayscale (Figure 22a) and color (Figure 22b) images. Then, the average value of each comparison criterion is computed for the encrypted images, reported in Table 4. It should be mentioned that each value of the comparison criteria is computed 100 times. Then, the average value of the comparison criterion is computed and reported in Table 4.

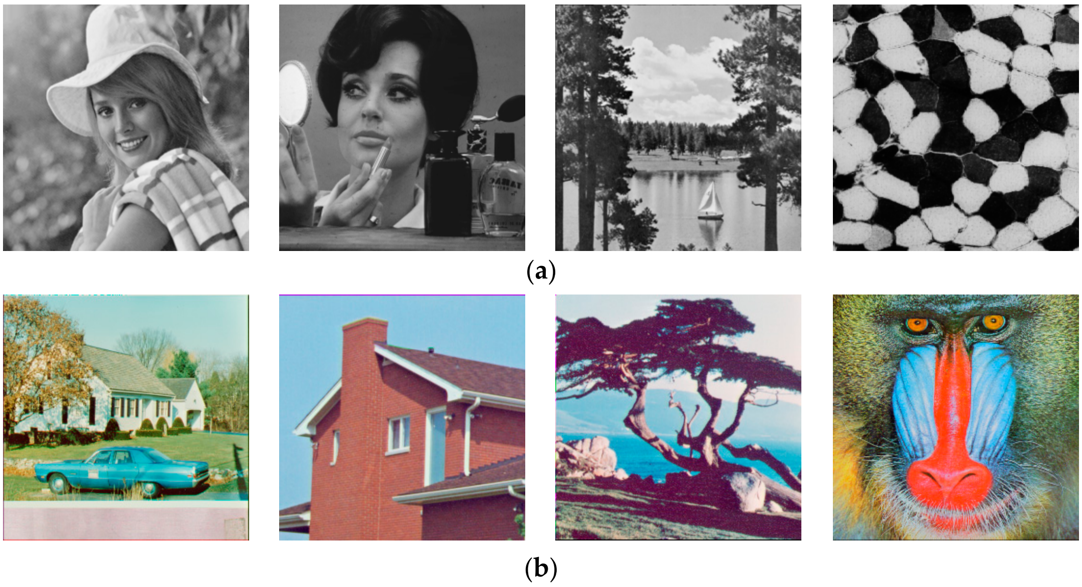


Figure 22. Test (**a**) grayscale and (**b**) color test images of size 512 × 512.

Table 4. Comparison results in terms of the average CC, entropy and encryption/decryption runtime for different encryption schemes including the proposed one.

		Averaged CC (in Absolute Values) Values Along:						Average Entropy (E)		Average Encryption & Decryption Runtime (in Sec.)	
		Horizontal Direction		Vertical Direction		Diagonal Direction					
	Image Group	(a)	(b)	(a)	(b)	(a)	(b)	(a)	(b)	(a)	(b)
Encryption Algorithms	Proposed	0.0019	0.0024	0.0036	0.0028	0.0033	0.0020	7.9998	7.9997	0.4507	1.3533
	Ref. [29]	0.0018	0.0022	0.0041	0.0030	0.0032	0.0022	7.9993	7.9996	0.4602	1.3812
	Ref. [32]	0.0086	0.0074	0.0030	0.0036	0.0032	0.0069	7.9992	7.9993	0.4706	1.4136
	Ref. [77]	0.0066	0.0036	0.0041	0.0040	0.0037	0.0028	7.9926	7.9931	0.3803	1.1406
	Ref. [31]	0.0082	0.0072	0.0072	0.0060	0.0101	0.0088	7.9979	7.9968	0.5103	1.5306
	Ref. [78]	0.0065	0.0061	0.0069	0.0054	0.0062	0.0069	7.9993	7.9992	0.6302	1.8936

The comparison results shown in Table 4 indicate on the one hand that the proposed scheme is competitive with the compared schemes in terms of CC values. The compared schemes, including the proposed one, are able to significantly reduce the correlation between adjacent pixels of the input image as the CC numbers tend to zero. On the other hand, we notice the superiority of the proposed scheme over the compared ones in terms of obtained E values. This result can be explained by the fact that our scheme involves the process of equalizing the distribution of the chaotic sequences used, which ensures a pseudo-random distribution within the encrypted images. On the other hand, the compared schemes ignore this task. The execution time of the proposed scheme is found to be competitive to the compared parallel-based schemes. Consequently, the suggested scheme can be effectively deployed in encryption systems to ensure both fast communication and high security.

6. Conclusions

In this work, an extended version of the existing TM, MTM, was proposed to improve the chaotic behavior of the existing TM. Then, we introduced a simple, yet efficient method to balance the distribution of chaotic sequence values. Next, a novel image encryption scheme was presented using MTM and parallel computing mode. The latter is employed to both speed up the encryption/decryption time and to significantly boost the security level of the offered scheme. Analysis and comparison results have shown on the one hand that the proposed scheme can withstand several types of attacks (brute force, statistical, differential, noise addition, slicing, etc.) On the other hand, the large key size of the proposed scheme ensures its considerable advantage in terms of security level during the communication of encrypted data. In future work, the proposed scheme will be extended to encrypt other kinds of multimedia (volumetric images, videos, medical images, etc.). Furthermore, improvements will be made to make our scheme fully parallel and deployable on hardware boards (i.e., Raspberry Pi, Arduino, FPGA, etc.).

Author Contributions: Conceptualization, A.D. and A.A.A.E.-L.; Methodology, A.D. and A.A.A.E.-L.; Formal analysis, A.D. and M.Y.; Investigation, S.A.C. and M.A.W.; Resources, S.A.C. and M.A.W.; Data curation, M.A.W.; Writing—review & editing, A.D., M.Y., S.A.C. and A.A.A.E.-L.; Visualization, M.Y.; Supervision, A.A.A.E.-L.; Project administration, M.A.W.; Funding acquisition, S.A.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by Princess Nourah bint Abdulrahman University Researchers Supporting Project (PNURSP2023R239), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia. This work was also supported by the EIAS Data Science Lab, College of Computer and Information Sciences, Prince Sultan University, Riyadh, Saudi Arabia.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are available from the first author (Achraf Daoui, achraf.daoui@usmba.ac.ma) upon reasonable request.

Acknowledgments: The authors would like to acknowledge the support of the Deanship of Scientific Research at Princess Nourah bint Abdulrahman University. This work was supported by Princess Nourah bint Abdulrahman University Researchers Supporting Project (PNURSP2023R239), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia. This work was also supported by the EIAS Data Science Lab, College of Computer and Information Sciences, Prince Sultan University, Riyadh, Saudi Arabia.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Mandal, P.C.; Mukherjee, I.; Paul, G.; Chatterji, B.N. Digital Image Steganography: A Literature Survey. *Inf. Sci.* **2022**, *609*, 1451–1488. [CrossRef]
2. Liao, X.; Yin, J.; Chen, M.; Qin, Z. Adaptive Payload Distribution in Multiple Images Steganography Based on Image Texture Features. *IEEE Trans. Dependable Secur. Comput.* **2022**, *19*, 897–911. [CrossRef]
3. Hua, Z.; Zhou, Y.; Huang, H. Cosine-Transform-Based Chaotic System for Image Encryption. *Inf. Sci.* **2019**, *480*, 403–419. [CrossRef]
4. Xian, Y.; Wang, X. Fractal Sorting Matrix and Its Application on Chaotic Image Encryption. *Inf. Sci.* **2021**, *547*, 1154–1169. [CrossRef]
5. Kordov, K. A Novel Audio Encryption Algorithm with Permutation-Substitution Architecture. *Electronics* **2019**, *8*, 530. [CrossRef]
6. Daoui, A.; Karmouni, H.; Sayyouri, M.; Qjidaa, H. Efficient Methods for Signal Processing Using Charlier Moments and Artificial Bee Colony Algorithm. *Circuits Syst. Signal Process.* **2021**, *41*, 166–195. [CrossRef]
7. Daoui, A.; Yamni, M.; Karmouni, H.; Sayyouri, M.; Qjidaa, H.; Ahmad, M.; Abd El-Latif, A.A. Biomedical Multimedia Encryption by Fractional-Order Meixner Polynomials Map and Quaternion Fractional-Order Meixner Moments. *IEEE Access* **2022**, *10*, 102599–102617. [CrossRef]
8. Wu, Y.; Zhang, L.; Berretti, S.; Wan, S. Medical Image Encryption by Content-Aware DNA Computing for Secure Healthcare. *IEEE Trans. Ind. Inform.* **2023**, *19*, 2089–2098. [CrossRef]

9. Naim, M.; Pacha, A.A.; Serief, C. A Novel Satellite Image Encryption Algorithm Based on Hyperchaotic Systems and Josephus Problem. *Adv. Space Res.* **2021**, *67*, 2077–2103. [CrossRef]
10. Song, X.-H.; Wang, H.-Q.; Venegas-Andraca, S.E.; Abd El-Latif, A.A. Quantum Video Encryption Based on Qubit-Planes Controlled-XOR Operations and Improved Logistic Map. *Phys. Stat. Mech. Its Appl.* **2020**, *537*, 122660. [CrossRef]
11. Rajesh, S.; Paul, V.; Menon, V.G.; Khosravi, M.R. A Secure and Efficient Lightweight Symmetric Encryption Scheme for Transfer of Text Files between Embedded IoT Devices. *Symmetry* **2019**, *11*, 293. [CrossRef]
12. Joshi, A.B.; Kumar, D.; Gaffar, A.; Mishra, D.C. Triple Color Image Encryption Based on 2D Multiple Parameter Fractional Discrete Fourier Transform and 3D Arnold Transform. *Opt. Lasers Eng.* **2020**, *133*, 106139. [CrossRef]
13. Wang, X.; Liu, C.; Jiang, D. A Novel Triple-Image Encryption and Hiding Algorithm Based on Chaos, Compressive Sensing and 3D DCT. *Inf. Sci.* **2021**, *574*, 505–527. [CrossRef]
14. Daoui, A.; Yamni, M.; Karmouni, H.; Sayyouri, M.; Qjidaa, H.; Ahmad, M.; El-Latif, A.A.A. Color Stereo Image Encryption and Local Zero-Watermarking Schemes Using Octonion Hahn Moments and Modified Henon Map. *J. King Saud Univ. Comput. Inf. Sci.* **2022**, *34*, 8927–8954. [CrossRef]
15. Wen, H.; Wu, J.; Ma, L.; Liu, Z.; Lin, Y.; Zhou, L.; Jian, H.; Lin, W.; Liu, L.; Zheng, T.; et al. Secure Optical Image Communication Using Double Random Transformation and Memristive Chaos. *IEEE Photonics J.* **2023**, *15*, 1–11. [CrossRef]
16. Zhang, C.; Zhang, W.; Chen, C.; He, X.; Qiu, K. Physical-Enhanced Secure Strategy for OFDMA-PON Using Chaos and Deoxyribonucleic Acid Encoding. *J. Light. Technol.* **2018**, *36*, 1706–1712. [CrossRef]
17. Hayat, U.; Azam, N.A. A Novel Image Encryption Scheme Based on an Elliptic Curve. *Signal Process.* **2019**, *155*, 391–402. [CrossRef]
18. Chen, L.; Yin, H.; Yuan, L.; Machado, J.A.T.; Wu, R.; Alam, Z. Double Color Image Encryption Based on Fractional Order Discrete Improved Henon Map and Rubik's Cube Transform. *Signal Process. Image Commun.* **2021**, *97*, 116363. [CrossRef]
19. Abd-El-Atty, B.; Iliyasu, A.M.; Alanezi, A.; Abd El-latif, A.A. Optical Image Encryption Based on Quantum Walks. *Opt. Lasers Eng.* **2021**, *138*, 106403. [CrossRef]
20. Nestor, T.; De Dieu, N.J.; Jacques, K.; Yves, E.J.; Iliyasu, A.M.; Abd El-Latif, A.A. A Multidimensional Hyperjerk Oscillator: Dynamics Analysis, Analogue and Embedded Systems Implementation, and Its Application as a Cryptosystem. *Sensors* **2019**, *20*, 83. [CrossRef]
21. Vaidyanathan, S.; Sambas, A.; Tlelo-Cuautle, E.; El-Latif, A.A.A.; Abd-El-Atty, B.; Guillén-Fernández, O.; Benkouider, K.; Mohamed, M.A.; Mamat, M.; Ibrahim, M.A.H. A New 4-D Multi-Stable Hyperchaotic System with No Balance Point: Bifurcation Analysis, Circuit Simulation, FPGA Realization and Image Cryptosystem. *IEEE Access* **2021**, *9*, 144555–144573. [CrossRef]
22. Benkouider, K.; Vaidyanathan, S.; Sambas, A.; Tlelo-Cuautle, E.; El-Latif, A.A.A.; Abd-El-Atty, B.; Bermudez-Marquez, C.F.; Sulaiman, I.M.; Awwal, A.M.; Kumam, P. A New 5-D Multistable Hyperchaotic System with Three Positive Lyapunov Exponents: Bifurcation Analysis, Circuit Design, FPGA Realization and Image Encryption. *IEEE Access* **2022**, *10*, 90111–90132. [CrossRef]
23. Gao, X.; Mou, J.; Xiong, L.; Sha, Y.; Yan, H.; Cao, Y. A Fast and Efficient Multiple Images Encryption Based on Single-Channel Encryption and Chaotic System. *Nonlinear Dyn.* **2022**, *108*, 613–636. [CrossRef]
24. Zhou, Y.; Bao, L.; Chen, C.L.P. A New 1D Chaotic System for Image Encryption. *Signal Process.* **2014**, *97*, 172–182. [CrossRef]
25. Dhall, S.; Pal, S.K.; Sharma, K. Cryptanalysis of Image Encryption Scheme Based on a New 1D Chaotic System. *Signal Process.* **2018**, *146*, 22–32. [CrossRef]
26. Pak, C.; An, K.; Jang, P.; Kim, J.; Kim, S. A Novel Bit-Level Color Image Encryption Using Improved 1D Chaotic Map. *Multimed. Tools Appl.* **2019**, *78*, 12027–12042. [CrossRef]
27. Midoun, M.A.; Wang, X.; Talhaoui, M.Z. A Sensitive Dynamic Mutual Encryption System Based on a New 1D Chaotic Map. *Opt. Lasers Eng.* **2021**, *139*, 106485. [CrossRef]
28. Liu, L.; Wang, J. A Cluster of 1D Quadratic Chaotic Map and Its Applications in Image Encryption. *Math. Comput. Simul.* **2023**, *204*, 89–114. [CrossRef]
29. Wang, X.; Feng, L.; Zhao, H. Fast Image Encryption Algorithm Based on Parallel Computing System. *Inf. Sci.* **2019**, *486*, 340–358. [CrossRef]
30. Wang, H.; Xiao, D.; Li, M.; Xiang, Y.; Li, X. A Visually Secure Image Encryption Scheme Based on Parallel Compressive Sensing. *Signal Process.* **2019**, *155*, 218–232. [CrossRef]
31. Yavuz, E. A New Parallel Processing Architecture for Accelerating Image Encryption Based on Chaos. *J. Inf. Secur. Appl.* **2021**, *63*, 103056. [CrossRef]
32. Song, W.; Fu, C.; Zheng, Y.; Tie, M.; Liu, J.; Chen, J. A Parallel Image Encryption Algorithm Using Intra Bitplane Scrambling. *Math. Comput. Simul.* **2023**, *204*, 71–88. [CrossRef]
33. Li, S.; Mishra, S. Optimizing Power Consumption in Multicore Smartphones. *J. Parallel Distrib. Comput.* **2016**, *95*, 124–137. [CrossRef]
34. Salami, B.; Noori, H.; Naghibzadeh, M. Fairness-Aware Energy Efficient Scheduling on Heterogeneous Multi-Core Processors. *IEEE Trans. Comput.* **2021**, *70*, 72–82. [CrossRef]
35. Daoui, A.; Yamni, M.; Karmouni, H.; Sayyouri, M.; Qjidaa, H.; Motahhir, S.; Jamil, O.; El-Shafai, W.; Algarni, A.D.; Soliman, N.F.; et al. Efficient Biomedical Signal Security Algorithm for Smart Internet of Medical Things (IoMTs) Applications. *Electronics* **2022**, *11*, 3867. [CrossRef]
36. Liansheng, S.; Cong, D.; Xiao, Z.; Ailing, T.; Anand, A. Double-Image Encryption Based on Interference and Logistic Map under the Framework of Double Random Phase Encoding. *Opt. Lasers Eng.* **2019**, *122*, 113–122. [CrossRef]

37. Zhang, G.; Ding, W.; Li, L. Image Encryption Algorithm Based on Tent Delay-Sine Cascade with Logistic Map. *Symmetry* **2020**, *12*, 355. [CrossRef]
38. Liu, X.; Xiao, D.; Liu, C. Three-Level Quantum Image Encryption Based on Arnold Transform and Logistic Map. *Quantum Inf. Process.* **2021**, *20*, 23. [CrossRef]
39. Zareai, D.; Balafar, M.; Feizi Derakhshi, M.R. A New Grayscale Image Encryption Algorithm Composed of Logistic Mapping, Arnold Cat, and Image Blocking. *Multimed. Tools Appl.* **2021**, *80*, 18317–18344. [CrossRef]
40. Kumar, M.; Gupta, P. A New Medical Image Encryption Algorithm Based on the 1D Logistic Map Associated with Pseudo-Random Numbers. *Multimed. Tools Appl.* **2021**, *80*, 1–27. [CrossRef]
41. Arif, J.; Khan, M.A.; Ghaleb, B.; Ahmad, J.; Munir, A.; Rashid, U.; Al-Dubai, A.Y. A Novel Chaotic Permutation-Substitution Image Encryption Scheme Based on Logistic Map and Random Substitution. *IEEE Access* **2022**, *10*, 12966–12982. [CrossRef]
42. Sangavi, V.; Thangavel, P. An Exotic Multi-Dimensional Conceptualization for Medical Image Encryption Exerting Rossler System and Sine Map. *J. Inf. Secur. Appl.* **2020**, *55*, 102626. [CrossRef]
43. Liu, J.; Wang, Y.; Liu, Z.; Zhu, H. A Chaotic Image Encryption Algorithm Based on Coupled Piecewise Sine Map and Sensitive Diffusion Structure. *Nonlinear Dyn.* **2021**, *104*, 4615–4633. [CrossRef]
44. Liu, Y.; Qin, Z.; Liao, X.; Wu, J. Cryptanalysis and Enhancement of an Image Encryption Scheme Based on a 1-D Coupled Sine Map. *Nonlinear Dyn.* **2020**, *100*, 2917–2931. [CrossRef]
45. Wang, M.; Wang, X.; Wang, C.; Zhou, S.; Xia, Z.; Li, Q. Color Image Encryption Based on 2D Enhanced Hyperchaotic Logistic-Sine Map and Two-Way Josephus Traversing. *Digit. Signal Process.* **2022**, *132*, 103818. [CrossRef]
46. Shao, S.; Li, J.; Shao, P.; Xu, G. Chaotic Image Encryption Using Piecewise-Logistic-Sine Map. *IEEE Access* **2023**, 1. [CrossRef]
47. Shakiba, A. A Novel Randomized One-Dimensional Chaotic Chebyshev Mapping for Chosen Plaintext Attack Secure Image Encryption with a Novel Chaotic Breadth First Traversal. *Multimed. Tools Appl.* **2019**, *78*, 34773–34799. [CrossRef]
48. Abd-El-Atty, B.; Iliyasu, A.M.; Abd El-Latif, A.A. A Multi-Image Cryptosystem Using Quantum Walks and Chebyshev Map. *Complexity* **2021**, *2021*, e9424469. [CrossRef]
49. Khan, M.; Alanazi, A.S.; Khan, L.S.; Hussain, I. An Efficient Image Encryption Scheme Based on Fractal Tromino and Chebyshev Polynomial. *Complex Intell. Syst.* **2021**, *7*, 2751–2764. [CrossRef]
50. Huang, S.; Jiang, D.; Wang, Q.; Guo, M.; Huang, L.; Li, W.; Cai, S. High-Quality Visually Secure Image Cryptosystem Using Improved Chebyshev Map and 2D Compressive Sensing Model. *Chaos Solitons Fractals* **2022**, *163*, 112584. [CrossRef]
51. Gupta, M.; Gupta, K.K.; Shukla, P.K. Session Key Based Novel Lightweight Image Encryption Algorithm Using a Hybrid of Chebyshev Chaotic Map and Crossover. *Multimed. Tools Appl.* **2021**, *80*, 33843–33863. [CrossRef]
52. Zhang, W.; Zhu, Z.; Yu, H. A Symmetric Image Encryption Algorithm Based on a Coupled Logistic–Bernoulli Map and Cellular Automata Diffusion Strategy. *Entropy* **2019**, *21*, 504. [CrossRef] [PubMed]
53. Gu, Z.; Li, H.; Khan, S.; Deng, L.; Du, X.; Guizani, M.; Tian, Z. IEPSBP: A Cost-Efficient Image Encryption Algorithm Based on Parallel Chaotic System for Green IoT. *IEEE Trans. Green Commun. Netw.* **2022**, *6*, 89–106. [CrossRef]
54. Yang, C.; Pan, P.; Ding, Q. Image Encryption Scheme Based on Mixed Chaotic Bernoulli Measurement Matrix Block Compressive Sensing. *Entropy* **2022**, *24*, 273. [CrossRef] [PubMed]
55. Alexan, W.; Elkandoz, M.; Mashaly, M.; Azab, E.; Aboshousha, A. Color Image Encryption Through Chaos and KAA Map. *IEEE Access* **2023**, *11*, 11541–11554. [CrossRef]
56. Yoosefian Dezfuli Nezhad, S.; Safdarian, N.; Hoseini Zadeh, S.A. New Method for Fingerprint Images Encryption Using DNA Sequence and Chaotic Tent Map. *Optik* **2020**, *224*, 165661. [CrossRef]
57. Li, C.; Luo, G.; Qin, K.; Li, C. An Image Encryption Scheme Based on Chaotic Tent Map. *Nonlinear Dyn.* **2017**, *87*, 127–133. [CrossRef]
58. Naskar, P.K.; Bhattacharyya, S.; Nandy, D.; Chaudhuri, A. A Robust Image Encryption Scheme Using Chaotic Tent Map and Cellular Automata. *Nonlinear Dyn.* **2020**, *100*, 2877–2898. [CrossRef]
59. Muñoz-Guillermo, M. Image Encryption Using Q-Deformed Logistic Map. *Inf. Sci.* **2021**, *552*, 352–364. [CrossRef]
60. Daoui, A.; Karmouni, H.; El ogri, O.; Sayyouri, M.; Qjidaa, H. Robust Image Encryption and Zero-Watermarking Scheme Using SCA and Modified Logistic Map. *Expert Syst. Appl.* **2022**, *190*, 116193. [CrossRef]
61. Han, C. An Image Encryption Algorithm Based on Modified Logistic Chaotic Map. *Optik* **2019**, *181*, 779–785. [CrossRef]
62. Belazi, A.; Kharbech, S.; Aslam, M.N.; Talha, M.; Xiang, W.; Iliyasu, A.M.; El-Latif, A.A.A. Improved Sine-Tangent Chaotic Map with Application in Medical Images Encryption. *J. Inf. Secur. Appl.* **2022**, *66*, 103131. [CrossRef]
63. Nagaraj, N. The Unreasonable Effectiveness of the Chaotic Tent Map in Engineering Applications. *Chaos Theory Appl.* **2022**, *4*, 197–204. [CrossRef]
64. Mondal, B.; Singh, S.; Kumar, P. A Secure Image Encryption Scheme Based on Cellular Automata and Chaotic Skew Tent Map. *J. Inf. Secur. Appl.* **2019**, *45*, 117–130. [CrossRef]
65. Ramasamy, P.; Ranganathan, V.; Kadry, S.; Damaševičius, R.; Blažauskas, T. An Image Encryption Scheme Based on Block Scrambling, Modified Zigzag Transformation and Key Generation Using Enhanced Logistic—Tent Map. *Entropy* **2019**, *21*, 656. [CrossRef] [PubMed]
66. Sneha, P.S.; Sankar, S.; Kumar, A.S. A Chaotic Colour Image Encryption Scheme Combining Walsh–Hadamard Transform and Arnold–Tent Maps. *J. Ambient Intell. Humaniz. Comput.* **2020**, *11*, 1289–1308. [CrossRef]
67. Kanso, A. Self-Shrinking Chaotic Stream Ciphers. *Commun. Nonlinear Sci. Numer. Simul.* **2011**, *16*, 822–836. [CrossRef]

68. Numpy. Argsort—NumPy v1.24 Manual. Available online: https://numpy.org/doc/stable/reference/generated/numpy.argsort.html#numpy.argsort (accessed on 16 March 2023).
69. Parah, S.A.; Loan, N.A.; Shah, A.A.; Sheikh, J.A.; Bhat, G.M. A New Secure and Robust Watermarking Technique Based on Logistic Map and Modification of DC Coefficient. *Nonlinear Dyn.* **2018**, *93*, 1933–1951. [CrossRef]
70. Ma, B.; Chang, L.; Wang, C.; Li, J.; Wang, X.; Shi, Y.-Q. Robust Image Watermarking Using Invariant Accurate Polar Harmonic Fourier Moments and Chaotic Mapping. *Signal Process.* **2020**, *172*, 107544. [CrossRef]
71. Belazi, A.; El-Latif, A.A.A. A Simple yet Efficient S-Box Method Based on Chaotic Sine Map. *Optik* **2017**, *130*, 1438–1444. [CrossRef]
72. Dalcin, L.; Fang, Y.-L.L. Mpi4py: Status Update After 12 Years of Development. *Comput. Sci. Eng.* **2021**, *23*, 47–54. [CrossRef]
73. Shift Array Circularly-MATLAB Circshift. Available online: https://www.mathworks.com/help/matlab/ref/circshift.html (accessed on 4 March 2022).
74. Dataset of Standard 512x512 Grayscale Test Images. Available online: https://ccia.ugr.es/cvg/CG/base.htm (accessed on 22 January 2023).
75. 3.0T GE Discovery 750W MRI Scanner Images | Magnetic Resonance Research Facility. Available online: https://medicine.uiowa.edu/mri/30t-ge-discovery-750w-mri-scanner-images (accessed on 11 March 2022).
76. Alvarez, G.; Li, S. Some Basic Cryptographic Requirements for Chaos-Based Cryptosystems. *Int. J. Bifurc. Chaos* **2006**, *16*, 2129–2151. [CrossRef]
77. Rostami, M.J.; Shahba, A.; Saryazdi, S.; Nezamabadi-pour, H. A Novel Parallel Image Encryption with Chaotic Windows Based on Logistic Map. *Comput. Electr. Eng.* **2017**, *62*, 384–400. [CrossRef]
78. Abbas, A.M.; Alharbi, A.A.; Ibrahim, S. A Novel Parallelizable Chaotic Image Encryption Scheme Based on Elliptic Curves. *IEEE Access* **2021**, *9*, 54978–54991. [CrossRef]
79. Luo, Y.; Ouyang, X.; Liu, J.; Cao, L. An Image Encryption Method Based on Elliptic Curve Elgamal Encryption and Chaotic Systems. *IEEE Access* **2019**, *7*, 38507–38522. [CrossRef]
80. CVG-UGR-Image Database. Available online: https://ccia.ugr.es/cvg/dbimagenes/ (accessed on 14 February 2023).
81. NIH Clinical Center Releases Dataset of 32,000 CT Images. Available online: https://www.nih.gov/news-events/news-releases/nih-clinical-center-releases-dataset-32000-ct-images (accessed on 19 August 2021).
82. Mansouri, A.; Wang, X. A Novel One-Dimensional Sine Powered Chaotic Map and Its Application in a New Image Encryption Scheme. *Inf. Sci.* **2020**, *520*, 46–62. [CrossRef]
83. Wu, Y. NPCR and UACI Randomness Tests for Image Encryption. *Cyber J. J. Sel. Areas Telecommun.* **2011**, *1*, 31–38.
84. Mahmoodi Khaniabadi, S.; Javadpour, A.; Gheisari, M.; Zhang, W.; Liu, Y.; Sangaiah, A.K. An Intelligent Sustainable Efficient Transmission Internet Protocol to Switch between User Datagram Protocol and Transmission Control Protocol in IoT Computing. *Expert Syst.* **2022**, e13129. [CrossRef]
85. Daoui, A.; Karmouni, H.; Sayyouri, M.; Qjidaa, H.; Maaroufi, M.; Alami, B. New Robust Method for Image Copyright Protection Using Histogram Features and Sine Cosine Algorithm. *Expert Syst. Appl.* **2021**, *177*, 114978. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.