Signal Processing 160 (2019) 45–58

Contents lists available at ScienceDirect

Signal Processing

journal homepage: www.elsevier.com/locate/sigpro

A new hybrid digital chaotic system with applications in image encryption

Moatsum Alawida[a], Azman Samsudin[a], Je Sen Teh[a,*], Rami S. Alkhawaldeh[b]

[a] School of Computer Sciences, Universiti Sains Malaysia (USM), Gelugor 11800, Penang, Malaysia
[b] Department of Computer Information Systems, The University of Jordan, Aqaba 77110, Jordan

ARTICLE INFO

Article history:
Received 4 October 2018
Revised 7 February 2019
Accepted 15 February 2019
Available online 18 February 2019

Keywords:
Chaos theory
Chaotic map
Cryptography
Image cipher
Image encryption
Perturbation

ABSTRACT

Image encryption provides confidentiality to images by transforming them into unrecognizable forms that resemble white noise. Digital chaos is one of the areas used to design image ciphers due to its many desirable properties such as ergodicity, initial state sensitivity and unpredictability. This paper proposes a new image cipher based on the perturbation of a new hybrid chaotic system. The proposed hybrid system cascades and combines two chaotic maps as part of a new chaotification method. Chaotic performance evaluation indicates that the new hybridization method has better complexity, higher sensitivity, and a larger chaotic parameter range as compared to other recently proposed chaotification methods. The new chaotic system is then used in the design of a new image cipher. Both confusion and diffusion properties are attained based on perturbing the chaotic states and system parameters, which are then involved in pixel shuffling and substitution operations respectively. Security and performance analysis indicates that the proposed scheme is highly resistant to various cryptanalytic attacks, is statistically superior and more secure than previously proposed chaos-based image ciphers.

1. Introduction

With the speedy growth of communication technology, it has become vital to protect private data from unauthorized access or attackers. The amount of digital images being circulated through social media or instant messaging is at an all-time high due to the increased use of mobile phones with digital cameras. Thus, the confidentiality of these images is of upmost importance. Image encryption is used to protect the confidentiality of image data while being transferred over an insecure channel such as the internet. Compared to text or binary data, images have additional characteristics such as higher correlation between pixels and large data capacity that need to be taken into consideration when selecting a suitable cipher. Thus, conventional encryption algorithms such as DES, 3DES and AES are not suitable for digital images because they are inefficient for real time messaging [1]. As a result, many image-specific encryption algorithms have been proposed in recent years based on various fields that include cellular automata [2,3], wavelet transform [4,5], compressing sensing [6], selective encryption [7], DNA coding [8,9] and chaos [10–14].

Digital chaos can be divided into two broad categories, high-dimensional (HD) and one-dimensional (1D) maps [15]. HD chaotic maps are more secure than 1D counterparts due to their large keyspace, highly dynamical system behaviour, complex attractor and high ergodicity. However, the high computational complexity of HD chaotic maps makes them costly to implement on hardware and software. In contrast, 1D chaotic maps are simple functions that have low computational costs. Chaotic maps such as the logistic map have been used in various cryptographic applications [14,16], due to the in-depth studies that have been performed on its chaotic properties [17,18]. Despite these advantages, applying 1D chaotic maps directly into a cryptographic algorithm leads to security vulnerabilities. Therefore, 1D chaotic maps should be enhanced in terms of their statistical characteristics before being used for cryptographic applications.

Many chaotification methods have been introduced to overcome 1D chaotic defects, ranging from cascading chaotic systems [19,20], pseudorandom perturbation [21], switching systems [22,23], combination systems based on modular operation [24–27]. These methods enhance an existing 1D chaotic map by increasing its chaotic parameter range and chaotic complexity. However, the chaotic parameter ranges still include periodic regions, which need to be avoided especially when designing secure cryptosystems. Moreover, sensitivity to initial conditions and/or control parameters is still low in some of these recently proposed methods

* Corresponding author.
E-mail address: jesen_teh@usm.my (J.S. Teh).

https://doi.org/10.1016/j.sigpro.2019.02.016
0165-1684/

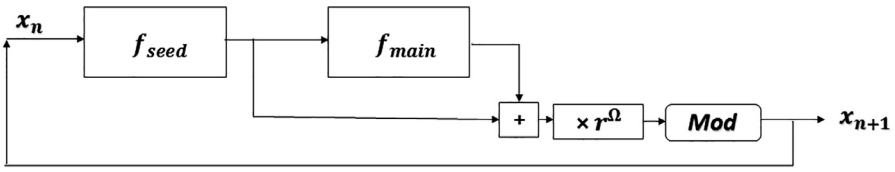

Fig. 1. The structure of the proposed chaotic systems.

[28]. Each of these existing chaotification methods come with their own set of limitations or drawbacks. Firstly, cascading chaotic systems is able to achieve a large Lyapunov exponent (LE) only when the two cascaded maps have large positive LE values. Some cascading models attempt to use more chaotic maps to achieve good chaotic behaviours [20]. However, the use of more chaotic maps will lead to higher computational complexity. Also, perturbation methods only focus on prolonging a cycle length by negating the effect of dynamic degradation in digital chaos. On the other hand, switching systems produce new chaotic characteristics based on a rule, whereby selection of a suitable rule that improves chaotic behaviour is a complex process. The chaotification method that combines two or more chaotic states from different maps by using the modulo operation results in simple patterns that are easy to estimate because the results of the modulo operation are merely the difference or the sum of the two underlying chaotic states.

As described by Shannon, a secure encryption algorithm must consist of two main properties, diffusion and confusion [29]. Diffusion ensures that any change in the plaintext can be spread out at to at least half of the ciphertext whereas the confusion phase is to hide the relationship between key and the ciphertext. Many image encryption schemes have been proposed based on alternating between operations that provide diffusion and confusion properties [5,12,30,31]. Operations such as pixel permutation or shuffling contribute towards confusion [32] whereas pixel substitution is commonly used for its diffusion property [11]. The intrinsic characteristics of digital chaos such as nonlinearity, ergodicity and high sensitivity to initial conditions can be utilized to fulfil both of these security notions.

Generally, the most permutation-based algorithms can be classified into three groups: (1) permutation that are based on discretized chaotic maps such as the Arnold cat map and Henon map [33,34], where the chaotic states were discretized to the size of the image and used to calculate new positions for pixels. However, the discretized maps suffer from periodicity, leading to a cycle after a specific number of iterations. (2) Bit-shifting and permutation-based algorithm that uses chaotic states to scramble bit values for all pixels before permutation is performed. Thus, the pixel positions and values are both modified [33,35]. However, it has a high computational cost to achieve a desired level of security, whereby the statistical information such as bit distribution and correlation are still not modified after many rounds. (3) Permutation based on sorting chaotic states [36,37], where the indexes of sorted chaotic states are used to change pixel positions. Sorting-based permutation mainly depends on the underlying chaotic characteristics to randomize the image permutation. Chaotic maps with low complexity leads to a more predictable permutation process with repeated patterns and strong correlation, and vice versa.

In diffusion rounds, the values of current pixels usually influence subsequent pixels or blocks in a linear order [26,35,38]. This implies that effect of diffusion is stronger for earlier pixels because they will be diffused throughout the image as compared to the final pixels. Hence, most image encryption algorithms will repeat the diffusion rounds to also diffuse the values of the final pixels to the earlier ones. The diffusion process uses chaotic states (which are usually directly or indirectly affected by pixel values) to modify the values of other pixels in order to achieve uniform distribution. Therefore, high randomness of chaotic states and strong diffusive properties are desired for chaos-based image encryption.

In order to overcome these challenges, this paper proposes a novel image encryption based on perturbing the chaotic states and control parameters of new hybrid maps by using plainimage pixels. Firstly, a new hybrid chaotic system will utilize cascade and combination methods to formulate new chaotic maps. The proposed hybridization method generates new 1D chaotic maps that retains the phase space of its underlying maps. Thus, the new maps also be used as direct replacements for other chaos-based cryptographic algorithms. The resulting 1D maps have better chaotic complexity, uniformly distributed LEs, wide chaotic parameter ranges, and uniform distributions. These maps depict highly chaotic behaviour for all parameter settings. In other words, the selection of any control parameter will lead to optimal chaotic behaviour because there are no chaotic regions that outperform the rest. Secondly, the perturbation-based image encryption uses plaintext pixels in both the confusion and diffusion rounds to overcome drawbacks of traditional chaos-based confusion and diffusion architectures. Both the confusion and diffusion operations use the pixel values from one-half of the image to encrypt the second half. The reverse is then performed using the second half to encrypt the first half. Chaotic state perturbation is used in confusion phase to permute the pixels, whereas system parameter perturbation is used in diffusion phase to modify the pixel values. Experimental results indicate that the new chaotic maps generated by the hybrid chaotic system has better chaotic complexity as compared to its underlying chaotic maps and also other chaotification methods. In addition, the proposed image cipher is also shown to be highly efficient, statistically secure, and resistant to cryptanalysis.

The remaining sections of this paper are organized as follows: Section 2 discusses the proposed chaotic system and introduces two new 1D hybrid chaotic maps. Security analysis of these maps are also provided in this section. Section 3 provides a detailed explanation of the proposed encryption algorithm followed by an analysis of its security and performance in Section 4. The paper is finally concluded in Section 5.

2. Proposed chaotic system

2.1. Chaotic system description

The proposed chaotic system is as shown in Fig. 1 where the + and $\times$ symbols are floating point addition and multiplication respectively, while ***Mod*** is a modulo 1 operation. The r^{Ω} multiplier is used for better distribution of the state variables in the phase space. Selection of the exponent value, Ω is a trade-off between chaoticity and computational speed.

The basic concept of the proposed system is the use of one chaotic map to perturb another based on the following generic equation,

$$x_{n+1} = ((f_{\text{main}}(f_{\text{seed}}(x_n)) + f_{\text{seed}}(x_n))r^{\Omega}) \ mod \ 1, \tag{1}$$

where f_{main} and f_{seed} are both classical 1D chaotic maps. The result of calculating f_{seed} is used as an input to f_{main}. Despite only

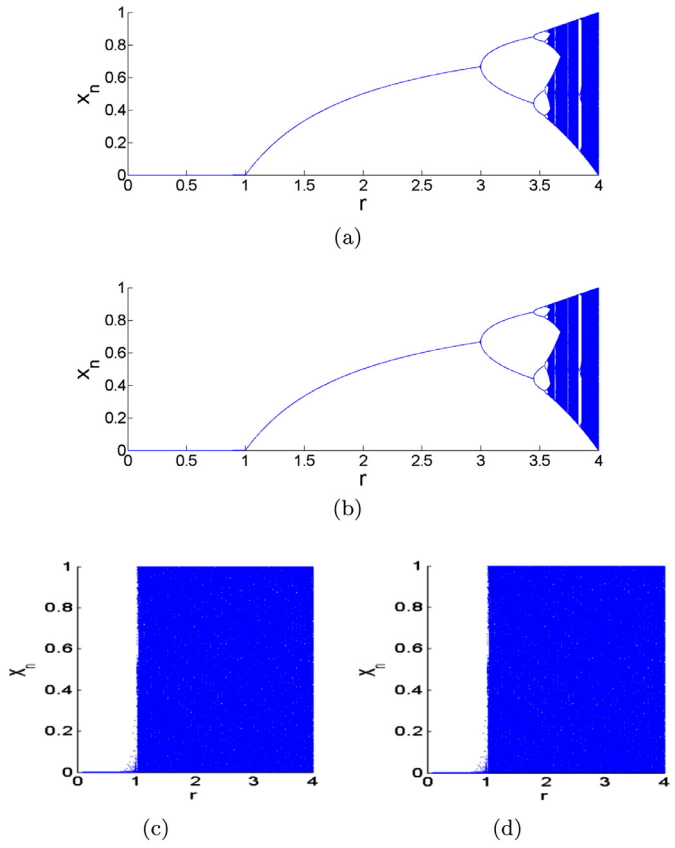

Fig. 2. Bifurcation plots of the (a) logistic map, (b) sine map, (c) TLTS map, (d) TSTS map.

iterating two chaotic maps, the output of the proposed chaotic system is influenced by the dynamical properties of three different maps (the main map, seed map, and modified seed map). Thus, the chaotic characteristics of the proposed system is more complex than methods that only combine two chaotic maps [19], while having lower computational complexity as compared to chaotic systems that utilize three maps [20]. Based on this generic description of the proposed chaotic system, we develop two hybrid chaotic maps using the tent, logistic and sine maps. These classical maps are defined as follows:

$$x_{n+1} = f_1(x, r) = r x_n (1 - x_n) \tag{2}$$

$$x_{n+1} = f_S(x, r) = \frac{r}{4}\sin(\pi x_n) \tag{3}$$

$$x_{n+1} = f_T(x, r) = \begin{cases} \frac{r}{2}x_n & \text{if } x_n < 0.5 \\ \frac{r}{2}(1 - x_n) & \text{if } x_n \geq 0.5, \end{cases} \tag{4}$$

where r is the control parameter that falls in the range of (0,4). Classical chaotic maps suffer from a limited chaotic range which in turn leads to a limited keyspace if the control parameter is used as part of the secret key. It is also widely known that the logistic and sine maps have windows of periodic behaviour and are susceptible to parameter estimation attacks [39]. Fig. 2(a) and (b) shows the bifurcation diagram of the classical chaotic maps which depict chaoticity as r approaches 4. The iteration function diagram for both the logistic and sine maps are shown in Fig. 5(a) which depicts their unimodal behaviour.

To overcome the inherent weaknesses of these classical maps, the tent map is used as the seed map, f_{seed} whereas the logistic and sine maps are used separately as the main map, f_{main} for the proposed system (Eq. (1)). These new hybrid maps are henceforth referred to as the Tent–Logistic–Tent system (TLTS) and the Tent–Sine–Tent system (TSTS), defined mathematically defined as

$$x_{n+1} = f_{\text{TLT}}(x_n, r) = \begin{cases} \left(\frac{r^2}{2}x_n\left(1 - \frac{r}{2}x_n\right) + \frac{r}{2}x_n\right)r^{14} \bmod 1, & \text{if } x_n < 0.5 \\ \left(\frac{r^2}{2}(1 - x_n)\left(1 - \frac{r}{2}(1 - x_n)\right)\right. & \\ \quad \left. + \frac{r}{2}(1 - x_n)\right)r^{14} \bmod 1, & \\ & \text{if } x_n \geq 0.5 \end{cases} \tag{5}$$

$$x_{n+1} = f_{\text{TST}}(x_n, r) = \begin{cases} \left(\frac{r}{4}\sin\left(\pi\frac{r}{2}x_n\right) + \frac{r}{2}x_n\right)r^{14} \bmod 1, & \text{if } x_n < 0.5 \\ \left(\frac{r}{4}\sin\left(\pi\frac{r}{2}(1 - x_n)\right) + \frac{r}{2}(1 - x_n)\right)r^{14} \bmod 1, & \text{if } x_n \geq 0.5, \end{cases} \tag{6}$$

where the exponent value of the multiplier, $\Omega = 14$ was selected experimentally to provide a balance between optimal chaotic behaviour and speed.

2.2. Chaotic system analysis

The bifurcation diagram of TLTS and TSTS are shown in Fig. 2(c) and (d) respectively, whereby both maps show fully chaotic behaviour without windows of periodicity in the region of $r \in (1.05, 4)$. Also, Fig. 3 shows that the LE values of the new hybrid maps are positive after $r > 1.05$, and are greater than the classical maps as well as other proposed chaotic systems (Tent–Logistic map and double-sine map from [19] and Logistic–Tent system from [25]). A positive LE value indicates chaotic behaviour whereas higher LE values indicate higher sensitivity to slight changes in initial value and/or system parameters. This is due to the increased rate of divergence of the system variables. This result demonstrates the capability of the proposed hybridization method in improving the chaotic behaviour of its underlying 1D map.

Next, the Shannon entropy (SE) is calculated for the attractors of the newly proposed maps to compare their ergodicity properties against their classical counterparts. Non-ergodicity leads to negative characteristics such as unvisited regions of the attractor or convergence to a specific region of the attractor after a certain number of iterations. In other words, a chaos-based encryption algorithm based on a chaotic map that depicts poor ergodicity will be susceptible to statistical attacks. Fig. 4 depicts the SE values of the proposed hybrid chaotic maps which are close to the ideal value of 10. This is calculated by dividing the attractor into 2^{10} sub-regions then computing the SE values for control parameters ranging between 0 and 4. After $r > 1.05$ the SE value approaches 10 which means the trajectories visit all sub-regions 2^{10} equally.

The iteration function diagram is also used to observe the behaviour of the proposed maps' chaotic trajectories in the phase space. Fig. 5(b) and (c) shows that both the TLTS and TSTS have complex iteration function diagrams as compared to their underlying classical maps which are unimodal in nature. The attractor of the hybrid systems are complex, indicating their capability to resist different types of attacks such as phase space reconstruction and parameter identification techniques.

In short, the new hybrid chaotic maps have desirable characteristics such as a large phase space, high ergodicity, and high sensitivity to slight changes in initial conditions and/or control parameters. These characteristics are analogous to the requirements of encryption algorithms. In addition, these maps preserves the original structure of the classical maps in terms of their parameter range. Therefore, they can directly replace logistic or sine maps in any existing encryption algorithm to improve their security.

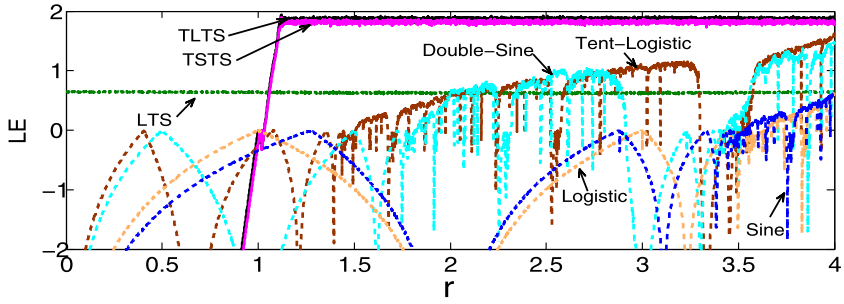

Fig. 3. Lyapunov exponent values of the new chaotic maps, logistic map, sine map, Tent–Logistic map and double-sine map [19], and Logistic–Tent system [25].

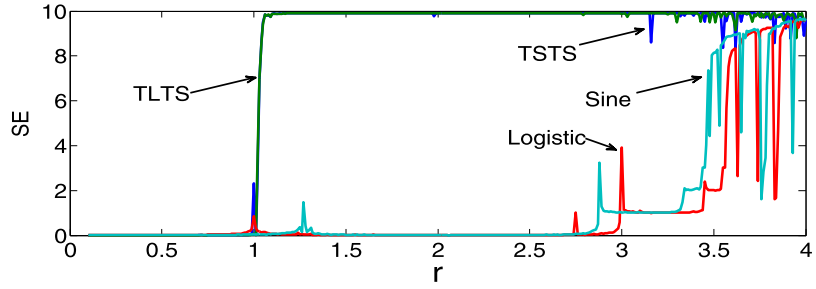

Fig. 4. Shannon entropy values of the new chaotic maps and classical maps.

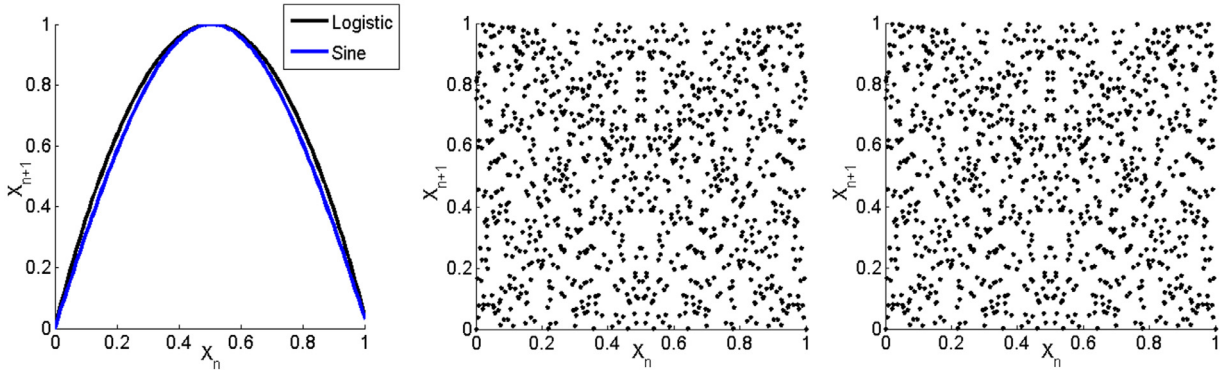

Fig. 5. The iteration function diagram of the (a) classical chaotic maps, $r = 4$; (b) TLTS, $r = 3$; (c) TSTS, $r = 3$.

3. New image encryption scheme

Using the TLTS and TSTS maps, this section focuses on designing a new chaos-based image encryption scheme. The initial conditions and control parameters (key states) are extracted from the secret key and used to produce data sequences from the proposed maps. Two permutation (confusion) and diffusion phases are used to completely encrypt an image. Both the permutation and diffusion operations are designed to use chaotic states and plainimage data to change pixel positions and substitute pixel values respectively, resulting in a noise-like cipherimage. The structure of encryption and decryption processes are shown in Fig. 6. Before the first round, a plainimage transformed into a 1D array by reading pixel values in a column-wise manner. Then, this array is divided into two halves for processing, where the first and second halves have lengths of $\lfloor \frac{MN}{2} \rfloor$ and $\lceil \frac{MN}{2} \rceil$ respectively. After completing the first round an intermediate ciphertext is obtained, which is then transformed into a 1D array for the second round of encryption by reading pixel values in a row-wise manner. Upon completion of the second round, the final cipherimage is obtained. The mathematical operations involved in the encryption algorithm are summarized below:

- $\oplus$: Bitwise exclusive-OR operation
- $\times$: Floating point multiplication
- $mod(x, y)$: x modulo y operation
- $\lceil x \rceil$: Smallest integer greater than or equal to x
- $\lfloor x \rfloor$: Largest integer less than or equal to x

The proposed algorithm also utilizes fixed point representation for real numbers, whereby $U(j, k)$ indicates that j and k bits are used to represent the integer and fractional portions of a real number respectively. The following equation is used to transform the a $U(0, k)$ fixed point number into its respective floating point value

$$NR(x) = \sum_{i=1}^{k} \frac{b_i}{2^i}, \tag{7}$$

where b_i is the value of bit i of the fixed point representation of x.

3.1. Secret key generation

To thwart brute force attacks, the recommended keyspace for encryption algorithms should be larger than 2^{128}. As shown in Fig. 3, TLTS and TSTS depict chaoticity for chaotic parameter values in the range of [1.05, 4]. For the proposed cipher, the chaotic

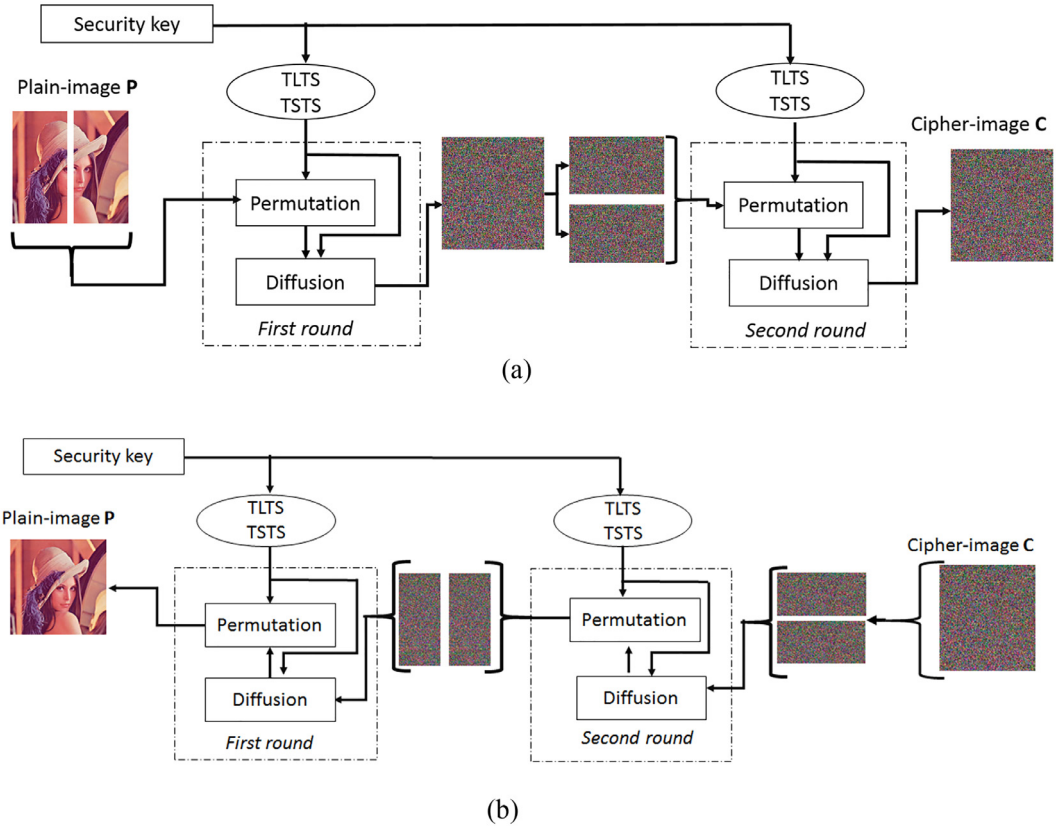

Fig. 6. The structure of the (a) encryption process and (b) decryption process.

behaviour of the maps directly affect its diffusion and confusion properties. Thus, the secret key has been designed to be based only on the chaotic maps to maximize its effect on their chaotic dynamics, and thus provide higher security. The secret key consists 312 bits that represented by hexadecimal numbers. It is used to generate key states, which are used initialize the initial conditions and control parameters of the chaotic maps. By using $U(0, 52)$ fixed point representation and then converting the result into a floating point number, key states can be generated based on Algorithm 1, where $\{x_0, y_0, r_1, r_2\}$ are used for the permutation phase whereas $\{r_1, r_2, \dot{x_0}, \dot{y_0}\}$ are used for the diffusion phase. As an example, we set the secret key (represented in hexadecimal) to be

$$K = 4_1 4_2 4_3 4_4 4_5 \ldots 4_{75} 4_{76} 4_{77} 4_{78}.$$

where '4' is represented as 0100 in binary. The resulting key states after processing are {3.1305, 2.3679, 0.6484, 0.2968, 0.4436, 0.8872} for $\{r_1, r_2, \dot{x_0}, \dot{y_0}, x_0, y_0\}$ respectively. This simple example depicts the capability of the key generation to produce key states that are randomized despite using a non-random key.

Algorithm 1: The generation of key states.

Data: Input: Secret key K with length of 312 bits.
Result: Output: Key states $\{x_0, y_0, r_1, r_2, \dot{x_1}, \dot{y_1}\}$.

L = 52;
$x_0 = \sum_{i=1}^{L} \frac{k_i}{2^i}$;
$y_0 = (\sum_{i=L+1}^{2L} \frac{k_i}{2^{i-1}} + x_0)\ mod\ 1$;
$r_1 = ((\sum_{i=2L+1}^{3L} \frac{k_i}{2^{i-2*L}} + y_0)10^{14})\ mod\ 2.95 + 1.05$;
$r_2 = ((\sum_{i=3L+1}^{4L} \frac{k_i}{2^{i-3*L}} + r_1)10^{14})\ mod\ 2.95 + 1.05$;
$\dot{x_0} = (\sum_{i=4L+1}^{5L} \frac{k_i}{2^{i-4*L}} + r_2)\ mod\ 1$;
$\dot{y_0} = (\sum_{i=5L+1}^{6L} \frac{k_i}{2^{i-5*L}} + \dot{x_0})\ mod\ 1$;
$r_1 = ((x_0 + y_0 + r_1 + r_2 + \dot{x_0} + \dot{y_0})10^{14})\ mod\ 2.95 + 1.05$;
$r_2 = ((x_0 + y_0 + r_1 + r_2 + \dot{x_0} + \dot{y_0})10^{14})\ mod\ 2.95 + 1.05$;
$\dot{x_0} = (x_0 + y_0 + r_1 + r_2 + \dot{x_0} + \dot{y_0})\ mod\ 1$;
$\dot{y_0} = (x_0 + y_0 + r_1 + r_2 + \dot{x_0} + \dot{y_0})\ mod\ 1$;
$x_0 = (x_0 + y_0 + r_1 + r_2 + \dot{x_0} + \dot{y_0})\ mod\ 1$;
$y_0 = (x_0 + y_0 + r_1 + r_2 + \dot{x_0} + \dot{y_0})\ mod\ 1$;

3.2. Permutation phase

The permutation phase is a two-pass phase performed to provide the confusion property. In the first pass, the permutation process utilizes the pixel values from the first half to permute the second half, whereas the second pass performs the reverse. The final pixel positions are sufficiently shuffled after applying this two-pass process. The basic idea behind the permutation phase is to use the pixel values from one-half of the plainimage to perturb the iterations of the modified map. The final chaotic state after all the perturbed iterations are performed will be used to generate a permutation pattern. The first and second pass use the TLTS and TSTS maps respectively.

One pixel value is used to perturb chaotic state before each iteration. This perturbed state will then be further modified by the next pixel value before the next iteration is processed. This is repeated until all pixels in the current half are used for perturbation. The final chaotic state will then be used as a new initial value to iterate the chaotic map. The chaotic map is iterated to generate a data sequence that is equal to the size of the other half of the plainimage. The index value of each element in the data sequence is associated with the location of each pixel in this half.

Next, the data sequence is sorted in ascending order. The new index values of the sorted components are used as the new pixel

locations. As an example, if the first element in the data sequence is sorted to be the last element in the data sequence, the first pixel is now relocated to the last position of the image. Algorithm 2

Algorithm 2: Permutation process.

Data: Image halves (P_1 and P_2 with lengths K_1 and K_2, respectively) and initial conditions (x_0, y_0) and control parameter (r_1, r_2) for TLTS and TSTS.
Result: Permutated image halves (P_1 and P_2)

for j = *1 to 2* **do**
  if j = *1* **then**
    $L = K_1$;
  else
    $L = K_2$;
  end
  for i = *1 to* L **do**
    if j = *1* **then**
      $x_{i-1} = (x_{i-1} + \mathbf{NR}(P_{1,i}))\ mod\ 1$;
      $x_i = \mathbf{TLTS}(x_{i-1}, r_1)$;
    else
      $y_{i-1} = (y_{i-1} + \mathbf{NR}(P_{2,i}))\ mod\ 1$;
      $y_i = \mathbf{TSTS}(y_{i-1}, r_1)$;
    end
  end
  if j = *1* **then**
    $m_0 = x_L$;
  else
    $m_0 = y_L$;
  end
  for i = *1 to* L **do**
    if j = *1* **then**
      $m_i = \mathbf{TLTS}(m_{i-1}, r_1)$;
    else
      $m_i = \mathbf{TSTS}(m_{i-1}, r_2)$;
    end
  end
  I = Index (Sort (m)) with ascending order and obtain index I and m', where $m' = m_I$;
  if j = *1* **then**
    $P_2 = permute(P_2, m_I)$
  else
    $P_1 = permute(P_1, m_I)$
  end
end

details the exact steps involved in the permutation phase whereas Fig. 7 provides an actual example of how the permutation process shuffles the pixel locations.

3.3. Diffusion phase

Diffusion property is generally achieved via nonlinear transformation of the plainimage. The value of a cipherimage's pixels should be highly sensitive to changes of even a single bit of any of the plainimage's pixels. The probability that the cipherimage pixel bits change when a single bit in the plainimage is toggled should be approximately 50% to indicate strong diffusion property. Similar to the permutation phase, the diffusion phase will also utilize halves of the plainimage/intermediate cipherimage and the two hybrid chaotic maps to perform nonlinear transforms of the pixel values.

Instead of modifying all pixels from each half as a whole, a pixel from the second half of the image is used to perturb the control parameter of the TLTS map using modular operation, then the result of iterating TLTS, $\dot{x}$ is used to encrypt pixels from the first half via XOR operation. This process is immediately repeated by using a pixel from the first half to perturb the control parameter of the TSTS map, then the result of iterating TSTS, $\dot{y}$ is used to encrypt a pixel in the second half. The diffusion phase differs from the permutation phase such that pixels from both sides will be modified by one another in a single pass.

If the size of the two halves are not equal, the pixels from the final row/column in the first half will be used to encrypt the final two rows/columns in the second half. Algorithm 3 provides

Algorithm 3: Diffusion process.

Data: Image (P_1 and P_2), L length of P_1, initial conditions $(\dot{x_0}, \dot{y_0})$ and control parameter (r_1, r_2) for TLTS and TSTS.
Result: Diffusion image (P_1 and P_2)

for i = *1 to* L **do**
  $r_1'' = (r_1 + NR(P_{2,i}))\ mod\ 2.95 + 1.05$;
  $\dot{x_i} = \mathbf{TLTS}(\dot{x_{i-1}}, r_1'')$;
  $P_{1,i} = (\dot{x_i} \times 2^{32})\ mod\ 256 \bigoplus P_{1,i}$;
  $r_2'' = (r_2 + NR(P_{1,i}))\ mod\ 2.95 + 1.05$;
  $\dot{y_i} = \mathbf{TSTS}(\dot{y_{i-1}}, r_2'')$;
  $P_{2,i} = (\dot{y_i} \times 2^{32})\ mod\ 256 \bigoplus P_{2,i}$;
if $L \neq Length(P_2)$ **then**
  $P_{2,L+1} = (\dot{y_L} \times 2^{32})\ mod\ 256 \bigoplus P_{2,L+1}$;

the detailed steps for the diffusion phase whereas Fig. 8 depicts the diffusion phase when two halves are not equal. Encryption of a plainimage is complete after performing two rounds of permutation and diffusion on an image that is first divided into two halves vertically, then horizontally in the second round.

3.4. Discussion

The new enhanced chaotic maps (TLTS and TSTS) have better chaotic complexity and widely chaotic parameter range than counterparts. The improved chaotic properties indirectly strengthens the proposed chaos-based cipher against attacks such as parameter estimation. The proposed encryption scheme leverages the highly chaotic behaviour of the proposed maps and plainimage pixel values to achieve both confusion and diffusion properties. The proposed encryption scheme has the following advantages (which will be supported by security and performance evaluation results in Section 4:

- The scheme performs encryption based on both the secret key and plainimage data to ensure that both confusion and diffusion requirements are fulfilled. Thus, the proposed scheme is able to resist well-known cryptanalytic attacks like differential, chosen-plaintext, and statistical attacks.
- Due to how each round divides the plainimage and how the permutation phases are designed, the proposed cipher ensures that each pixel value will affect the encryption of all remaining pixel values regardless of its position in the image.
- Due to the design of the key generation algorithm and the encryption phases, any slight change to the secret key will result in a large change in the key states and correspondingly, the cipherimage.
- Chaos-based permutation algorithms generally produce the same permutation pattern for all plainimages for a specific key. In this paper, the permutation is unique to each plainimage because the chaotic states have been perturbed by plainimage pixels prior to generating the permutation pattern.

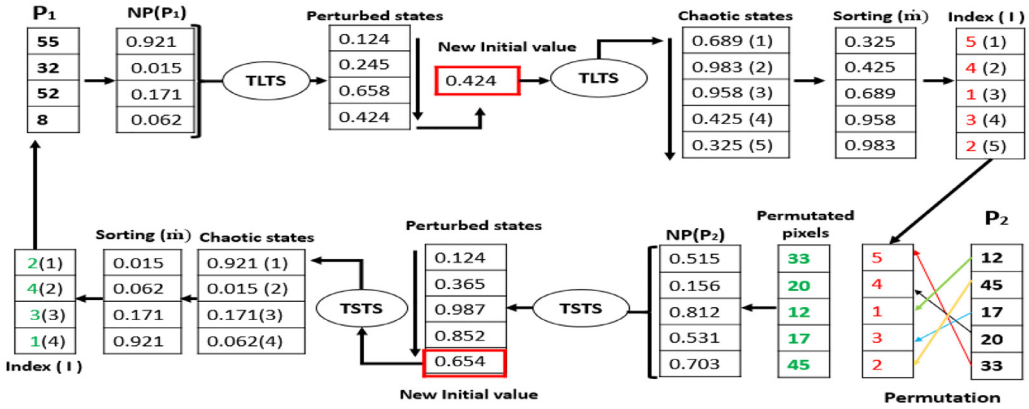

Fig. 7. An example of permutation algorithm.

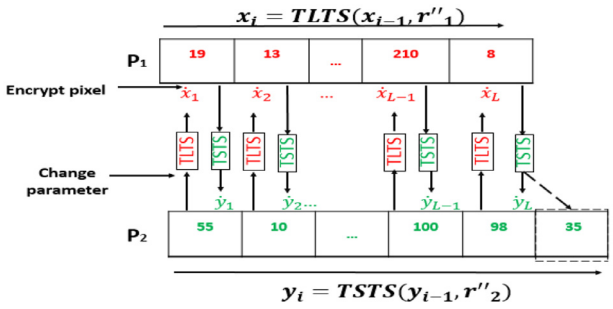

Fig. 8. An example of diffusion algorithm.

However, the proposed image encryption is not without drawbacks. Due to its immense sensitivity to changes in the plaintext, it is susceptible to noise and cropping attacks. This is due to the nature of the proposed cipher that depends heavily on the plainimage and chaotic states to perform permutation. Thus, any changes in pixel positions of the cipherimage will result in an entirely different permutation that will prevent the image from being decrypted successfully. This can be exploited by attackers to prevent legitimate receivers from successfully recovering the plainimage. This issue can be addressed in future improvements to the cipher in terms of its robustness against noise introduced into the cipherimage.

4. Security and performance evaluation

All images used in the following experiments are binary, greyscale and colour images with different sizes. For colour images, all colour channels are transformed into 1D arrays for encryption then reconstructed to obtain the corresponding cipherimage. Fig. 9(a), (d) and (g) are the plainimages of the Cameraman, Lena and a small black box image (image with odd width and height) respectively, whereas Fig. 9(b), (e) and (h) illustrate their corresponding cipherimage. The proposed algorithm successfully decrypts the cipherimages as shown in Fig. 9(c), (f) and (i) respectively. At a glance, the cipherimages have low correlation with the plainimages as they resemble white noise. The decryption process can successfully recover the original images without any loss of information. The following subsection will analyse the security of the proposed cipher in further detail in terms of statistical characteristics and resistance against cryptanalysis. Whenever applicable, the proposed cipher will be compared against other existing chaos-based image ciphers that utilize the same evaluation metrics.

4.1. Security analysis

4.1.1. Histogram analysis

A histogram is plotted to depict the distribution of the pixels in an image. The histogram for four plainimages and their corresponding cipherimages are illustrated in Fig. 10. It can be seen that the histograms for the cipherimages are evenly distributed as compared to their corresponding plainimages. Thus, no information with regards to the plainimage can be inferred based on the distribution of pixels of the cipherimage to be used in a statistical attack.

In addition to visual inspection, we further analyse the uniformity of the cipherimages' histogram in terms of maximum deviation, irregular deviation and deviation from uniform histogram (DUH). These metrics are then used for comparison with other schemes [7,10]. Firstly, maximum deviation is a statistical analysis that is used to scale the quality of an encryption scheme by calculating the change in pixel values of the plainimage and cipherimage. A high maximum deviation value implies that the pixel values of the encrypted image have been modified to a great extent, which in turn implies that the encryption scheme is secure. Maximum deviation can be calculated as

$$M_D = \frac{D_0 + D_{N-1}}{2} + \sum_{i=1}^{N-1} D_i \tag{8}$$

where $D_i = |D_{C_i} - D_{P_i}|$ is the absolute difference for each grey value in the cipherimage and plainimage, N is the number of grey values with the range of [0,256]. If M_D has a high value, there is a large difference between the pixel values of the plainimage and cipherimage.

Secondly, irregular deviation measures the extent of deviation from a uniform statistical distribution. Irregular deviation can be calculated as

$$I_D = \sum_{i=0}^{N-1} |h_i - A_H| \tag{9}$$

where h_i is the absolute difference of the histogram between plainimage and cipherimage at index i whereas A_H is the mean of the histogram values. A low value of I_D indicates that the pixels in the cipherimage are uniformly distributed, which is a desirable characteristic for an image cipher.

Thirdly, DUH examines the deviation of the cipherimage's histogram from an ideal one. To achieve minimum deviation, pixels in an encrypted image should have a high probability to be quite equal to pixel values from an uniform histogram. The pixel values in a uniform histogram is dependent on the size of plainimage and

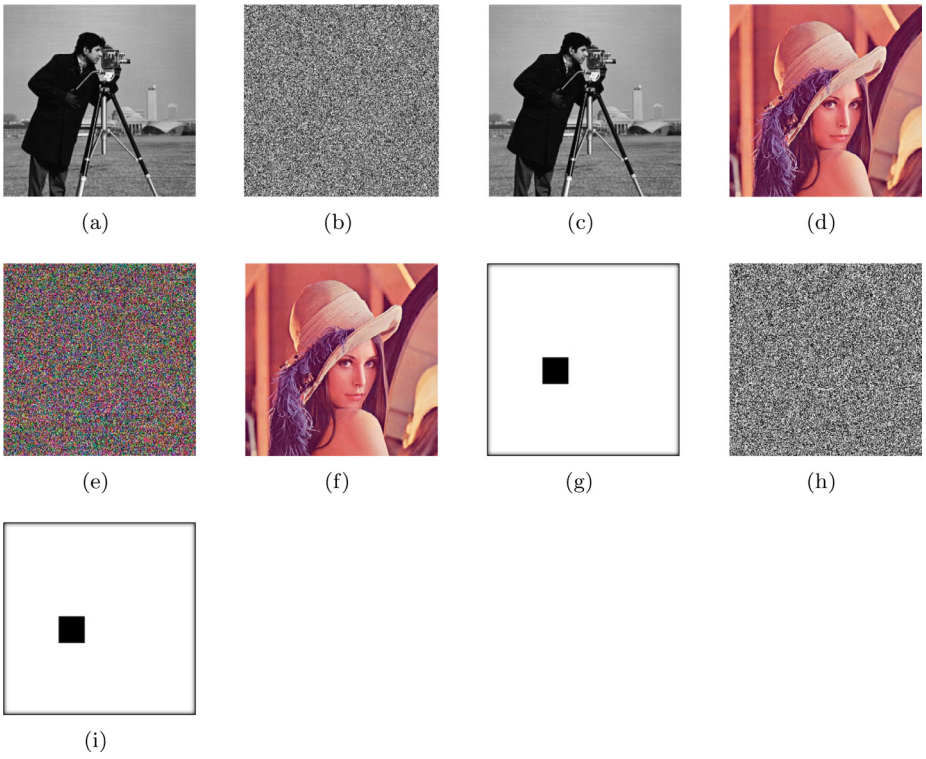

Fig. 9. (a) Cameraman plainimage (size 256×256), (b) Cameraman cipherimage, (c) Cameraman recovered plainimage, (d) Lena plainimage (size 512×512), (e) Lena cipherimage, (f) Lena recovered plainimage, (g) small black box plainimage (size 211×207), (h) small black box cipherimage, (i) small black box recovered plainimage.

it can be calculated as

$$H_C = \begin{cases} \frac{M \times N}{256}, & 0 \leqslant i \leqslant 255 \\ 0, & \text{elsewhere,} \end{cases} \tag{10}$$

where M and N are the width and height of an image respectively. DUH can then be calculated as

$$D_H = \frac{\sum_{C_i=0}^{255} |H_{C_i} - H_C|}{M \times N} \tag{11}$$

where H_{C_i} is the grey value of the cipherimage's histogram at index i. A low value of D_H indicates that a low deviation from the ideal histogram, which implies a uniform distribution.

The results of these histogram analyses are tabled in Table 1. Two different plainimages (Lena and peppers) were used for comparison with other image encryption schemes which utilized the same evaluation metrics. It is clearly shown that the proposed encryption algorithm outperforms the others for all three evaluation metrics.

4.1.2. Correlation coefficient analysis

Correlation coefficients are calculated to analyse the relationship strength between the plainimages and cipherimages for vertically, horizontally, diagonally adjacent pixels. The correlation coefficient is calculated as

$$C_r = \frac{\mathrm{COV}(X, Y)}{\sqrt{D(X)D(Y)}}, \tag{12}$$

where $\mathrm{COV}(x, y) = E([x - E(x)][y - E(y)])$, $E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i$ and $D(x) = \frac{1}{N}\sum_{i=1}^{N}[x_i - E(x)]^2$. x and y are the values of adjacent pixels in the plainimage or cipherimage.

For $N = 4096$ randomly selected pairs of vertically adjacent pixels from the Lena image and its corresponding cipherimage, the correlation coefficient is calculated. The same experiment is repeated for horizontally and diagonally adjacent pixels. As shown in Fig. 11, there is no correlation between adjacent pixel values of the cipherimage in all directions. The experiment is repeated for multiple plainimages. The correlation coefficient between the entire plainimage and cipherimage (CC_{whole}) is also calculated. All results are tabulated in Table 2. In all cases, correlation coefficients are near-zero. This implies that the cipher highly resistant to statistical-based attacks.

4.1.3. Information entropy

To assess randomness, the global Shannon entropy (GSE) and local Shannon entropy (LSE) of an image can be calculated as

$$H(m) = -\sum_{i=0}^{L-1} p(m_i)\log_2 p(m_i), \tag{13}$$

where $p(m_i)$ is the discrete probability density function (PDF), m_i is the pixel value from 0 to 255, and L is grey level ($L = 256$ for an 8-bit grey image). GSE is calculated for the entire image whereas LSE is calculated based on selecting randomly non-overlapping blocks of an image [41].

The ideal value of GSE for an 8-bit grey level image is $H(m) = 8$, indicating that the information contained in the image is dense and it cannot be further compressed. It also reflects the balanced distribution of the pixel values. The ideal entropy value of LSE is related to the level of significance α, the number of non-overlapping image blocks K, and the number of pixels T_B. If the average of LSE values fall between the critical values $[h^*_{\text{left}}, h^*_{\text{right}}]$, the image sample under scrutiny has successfully passed the test. The GSE of the cipherimages generated by the proposed cipher are calculated to be near-ideal for GSE and whereas most of the cipherimages pass the LSE test as shown in Table 2. For comparison with several existing image ciphers, we select all images from the USC-SIPI 'Miscellaneous' image dataset and calculate both LSE and GSE as shown in Table 3. Despite being slightly outperformed by

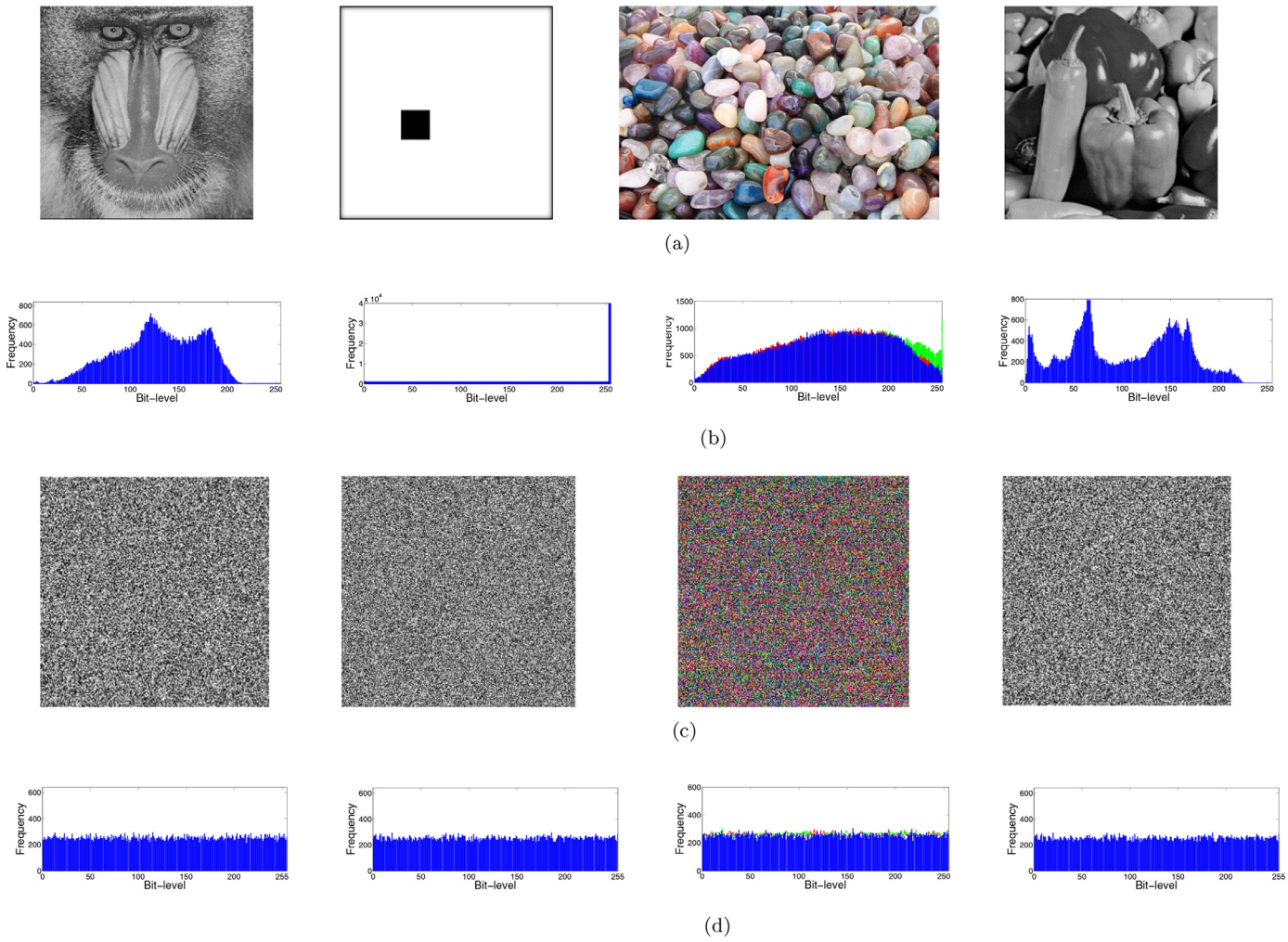

Fig. 10. (a) Plainimages, (b) histogram of the plainimages, (c) cipherimages, (d) histogram of the cipherimages.

Table 1
Histogram analysis of the proposed scheme compared to other schemes.

Scheme (image)	Maximum deviation	Irregular deviation	DUH
Proposed scheme (Lena)	37,980	20,053	0.0471
Ref. [40] (Lena)	21,786	40,904	0.0974
Ref. [5] (Lena)	21,339	40,480	0.0994
Proposed scheme (Peppers)	37,968	23,863	0.0522
Ref. [40] (Peppers)	24,254	35,242	0.0938
Ref. [5] (Peppers)	22,935	35,088	0.0917

Hua et al. [36], the proposed scheme depicts a high level of randomness for a variety of different input images.

4.1.4. Differential attack

To evaluate an image cipher's resistance against differential attacks [43], the number of pixels change rate (NPCR), unified average changing intensity (UACI) and avalanche criterion can be computed. The NPCR and UACI between two encrypted images $C1$ and $C2$ are described as follows:

$$\mathrm{NPCR} = \frac{\sum_{i,j}^{M,N} D(i,j)}{M \times N} \times 100\% \tag{14}$$

$$\mathrm{UACI} = \frac{1}{M \times N} \sum_{i,j}^{M,N} \frac{|C_1(i,j) - C_2(i,j)|}{255} \times 100\% \tag{15}$$

$$D(i,j) = \begin{cases} 1, & \text{if } C_1(i,j) \neq C_2(i,j) \\ 0, & \text{if } C_1(i,j) = C_2(i,j), \end{cases} \tag{16}$$

where M and N are the width and height size of the plainimage respectively, $C1$ and $C2$ are the encrypted images values before and after changing one bit of the plainimage. The avalanche criterion is calculate as the average number of bits that differ between $C1$ and $C2$.

The following experiment is performed: a pixel value is randomly selected from the plainimage. The pixel value is changed by 1 bit to obtain a new plainimage. These plainimages are encrypted and the NPCR, UACI and avalanche criterion are computed. This experiment is repeated 100 times for various images. The minimum, maximum and mean values of the NPCR, UACI and avalanche criterion for each image are listed in Table 4. Additionally, Fig. 12 shows the values of NPCR, UACI and avalanche criterion of for 100

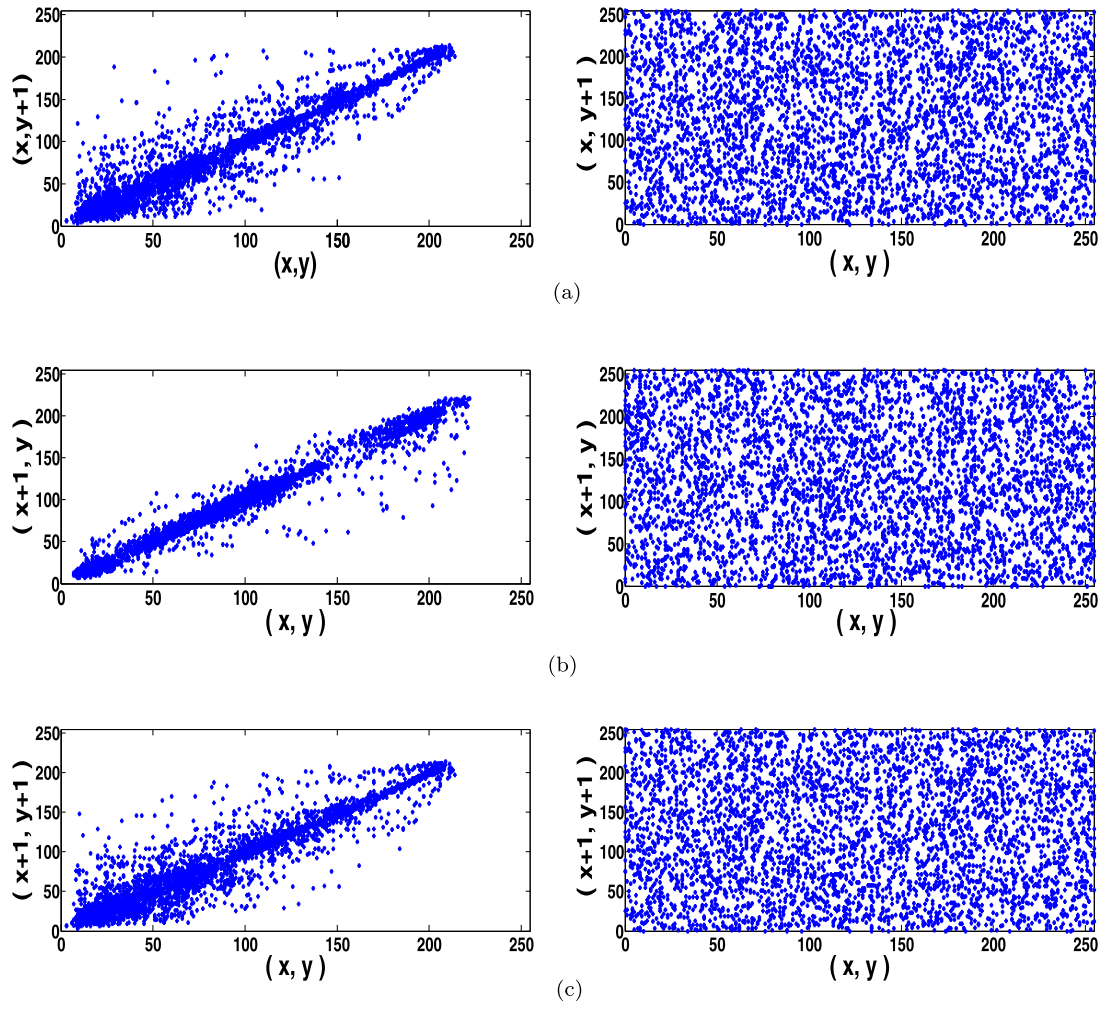

Fig. 11. Correlation coefficient analysis of the Lena image for its plainimage and cipherimage: (a) vertical correlation coefficient, (b) horizontal correlation coefficient, (c) diagonal correlation coefficient.

repetitions of the experiment on the Lena image. The critical values for NPCR and UACI have been calculated under $\alpha = 0.05$ as (NPCR $\geq$ 99.5693) and UACI $\in$ [33.22, 33.70] [44]. Avalanche criterion also has ideal value of $\approx$50. Experimental results show that the proposed cipher can pass NPCR and UACI tests and has near-ideal results for avalanche criterion. These results show that encryption scheme has high resistance to differential cryptanalysis and has good diffusion.

4.1.5. Secret key analysis

In order to resist the brute force attacks, a cipher must have a sufficiently large keyspace. The minimum key size for symmetric ciphers are recommended to be at least 80 bits and 128 bits for lightweight and regular ciphers respectively [45]. In addition, the cipher should be highly sensitive to slight changes in the key. The secret key of the proposed cipher has a length of 312 bits which is distributed into six keys states $r1$, $r2$, x_1, y_1, x_0, y_0. This is more

Table 2
Correlation coefficient and entropy analysis for multiple images.

Image	Plainimage					Cipherimage					CC_{whole}
	Correlation coefficient					Correlation coefficient					
	Vertical	Horizontal	Diagonal	GSE	LSE	Vertical	Horizontal	Diagonal	GSE	LSE	
Baboon	0.550	0.459	0.575	7.3649	7.1049	−0.0015	−0.0026	0.0014	7.9971	7.9021	0.0062
Barbara	0.932	0.977	0.898	7.3924	6.9556	−0.0004	0.0061	0.0008	7.9973	7.9014	0.004
Boat	0.863	0.972	0.914	7.2024	6.7425	0.0009	0.0208	0.0021	7.9971	7.9029	−0.003
Couple	0.935	0.967	0.708	7.1845	6.7497	−0.0004	0.0090	0.0016	7.9970	7.8987	−0.0065
Elaine	0.972	0.942	0.910	7.4717	6.9702	0.0032	−0.0047	−0.0003	7.9971	7.9020	0.0008
Fingerprint	0.661	0.838	0.476	7.5948	7.4278	−0.0006	−0.0206	−0.0003	7.9971	7.9029	−0.0005
Gold Hill	0.938	0.792	0.852	7.4450	7.0245	−0.003	0.0143	0.0021	7.9975	7.9020	0.0005
Lena	0.955	0.940	0.931	7.5764	6.8134	−0.0017	−0.0084	−0.0019	7.9975	7.9051	0.0036
Peppers	0.970	0.952	0.958	7.5701	6.9377	0.0024	−0.0131	0.0002	7.9970	7.9043	−0.002
Average	0.864	0.871	0.802	7.4224	6.9696	0.0013	0.0111	0.0012	7.9972	7.9024	0.003

LSE of cipherimages $\in$ [7.9015, 7.9034] a pass ($K = 30$, $T_B = 1936$, $\alpha = 0.001$).

Table 3
Values of LSE and GSE of Miscellaneous image dataset. (The bold results have passed LSE test.)

File name	LSE				GSE
	Ref. [42]	Ref. [22]	Ref. [36]	Proposed scheme	
5.1.09	7.904191	**7.903354**	**7.902127**	**7.902758**	7.9965
5.1.10	**7.902371**	**7.902443**	**7.903402**	**7.902215**	7.9972
5.1.11	7.900799	**7.902756**	**7.902687**	**7.901642**	7.9973
5.1.12	**7.903374**	**7.901526**	**7.901906**	**7.902335**	7.9954
5.1.13	7.904566	7.904563	**7.902825**	7.903598	7.9963
5.1.14	**7.903111**	**7.902954**	**7.902340**	7.900207	7.9973
5.2.08	**7.901762**	**7.902356**	**7.903327**	**7.902487**	7.9992
5.2.09	7.905854	7.899853	**7.901765**	**7.902681**	7.9990
5.2.10	**7.902768**	**7.902654**	**7.902748**	**7.902150**	7.9987
5.3.01	7.901040	**7.902647**	**7.901772**	**7.902531**	7.9993
5.3.02	7.900981	7.910474	**7.903328**	**7.902483**	7.9992
7.1.01	**7.902145**	**7.902634**	7.901305	**7.902725**	7.9980
7.1.02	**7.902157**	**7.901634**	**7.901578**	7.893536	7.9949
7.1.03	7.900645	7.905423	**7.903099**	7.900743	7.9983
7.1.04	7.904141	**7.902125**	**7.902607**	**7.902163**	7.9985
7.1.05	7.900027	7.883653	7.905305	**7.902208**	7.9988
7.1.06	**7.901736**	**7.902356**	**7.902695**	**7.903055**	7.9990
7.1.07	7.900802	**7.902364**	**7.902896**	**7.902832**	7.9987
7.1.08	7.900944	7.904456	**7.901632**	**7.902407**	7.9988
7.1.09	7.905658	**7.903012**	**7.903173**	**7.902674**	7.9985
7.1.10	7.893848	**7.901598**	**7.901524**	**7.902699**	7.9988
7.2.01	7.904525	**7.901989**	**7.902454**	**7.901923**	7.9988
boat.512	7.900712	**7.901879**	**7.903088**	**7.902488**	7.9993
elaine.512	**7.902030**	**7.902989**	**7.901720**	**7.903052**	7.9993
gray21.512	**7.902149**	7.905107	**7.902688**	7.887108	7.9903
numbers.512	7.903579	7.892351	**7.901657**	**7.902447**	7.9992
ruler.512	7.901428	**7.903001**	**7.903052**	7.898703	7.9978
testpat.1k	**7.903343**	**7.901681**	**7.902752**	**7.902771**	7.9847
Mean	7.902167	7.901923	**7.902488**	7.901379	7.9974
Pass rate	11/28	20/28	26/28	22/28	–

LSE $\in$ [7.901515698, 7.903422936] that means pass ($K = 30, T_B = 1936, \alpha = 0.001$).

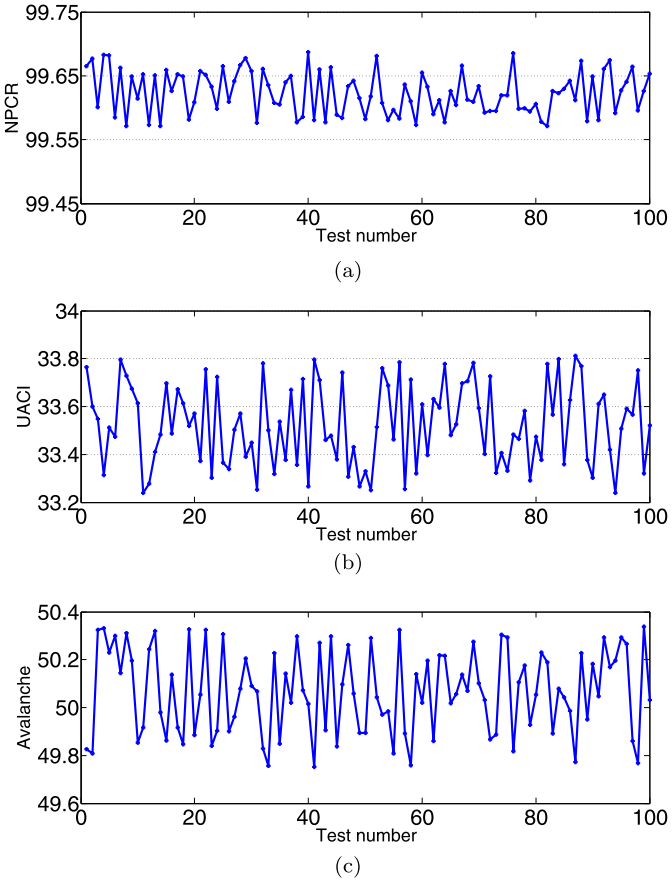

Fig. 12. Differential attack analysis of Lena image: (a) NPCR (NPCR*0.05) ($\geq$99.5693) [44], (b) UACI (UACI*0.05) [33.22,33.70] [44], (c) avalanche criterion $\approx$50.

than sufficient to withstand brute force attacks with the current computational power available today. Moreover, any changes to the secret key will be diffused throughout the key states based on the key generation algorithm in Algorithm 1. This slight change will then be further amplified by the sensitivity of the hybrid chaotic maps.

The key sensitivity of the proposed cipher is analysed based on the following experiment: A plainimage is first encrypted with a secret key, $K_0 = 7_1, 7_2, \ldots, 7_{78}$. 312 different secret keys K_1 to K_{312} are selected by sequentially toggling a single bit within K_0 and are used to encrypt the same plainimage. To quantitatively measure the sensitivity of the key, the correlation coefficients (CC), NPCR, UACI, and avalanche criterion are calculated for the encrypted images using valid and invalid keys as shown in Fig. 13. The CC is close to zero for all cases, indicating a lack of relationship between the encrypted images and the original encrypted image via K_0. We can also see that the NPCR, UACI and avalanche values are close to critical values [44], which confirms that any slight change to the key will affect the encryption of the entire image. Moreover, these key sensitivity has same effect on decryption. As a result, none of the modified keys can successfully decrypt the cipherimage. This depicts the sensitivity of the proposed cipher to slight changes to its encryption key.

Table 4
Values of NPCR, UACI and avalanche criterion. The bold values are the average NPCR/UACI/Avalanche values that are compared against the average values achieved by other proposed image ciphers in ref [5,38,40].

Image	NPCR			UACI			Avalanche		
	min	max	mean	min	max	mean	min	max	mean
Baboon	99.531	99.641	99.601	33.242	33.712	33.424	49.942	50.142	49.997
Barbara	99.547	99.693	99.615	33.295	33.723	33.527	49.715	50.151	50.014
Boat	99.561	99.684	99.618	33.288	33.701	33.521	49.745	50.211	49.997
Couple	99.524	99.681	99.619	33.312	33.664	33.497	49.985	50.161	50.145
Elaine	99.522	99.674	99.624	33.212	33.680	33.559	49.950	50.134	49.958
Fingerprint	99.521	99.681	99.613	33.181	33.671	33.518	49.784	50.117	49.943
Gold Hill	99.511	99.684	99.650	33.245	33.631	33.523	49.817	50.132	50.101
Lena	99.533	99.671	99.620	33.231	33.812	33.505	49.714	50.312	50.010
Peppers	99.546	99.663	99.617	33.315	33.712	33.391	49.814	50.144	50.022
Average	99.532	99.674	**99.618**	33.257	33.700	**33.496**	49.829	50.167	**49.999**
Ref. [38]	99.542	99.707	**99.612**	33.245	33.648	**33.457**	–	–	**49.999**
Ref. [40]	99.514	99.642	**99.618**	33.225	33.709	**33.606**	–	–	**49.999**
Ref. [5]	99.521	99.661	**99.621**	33.221	33.701	**33.666**	–	–	**49.999**

(NPCR*0.05) ($\geq$99.5693), (UACI*0.05) [33.22,33.70] and Avalanche criterion $\approx$50.

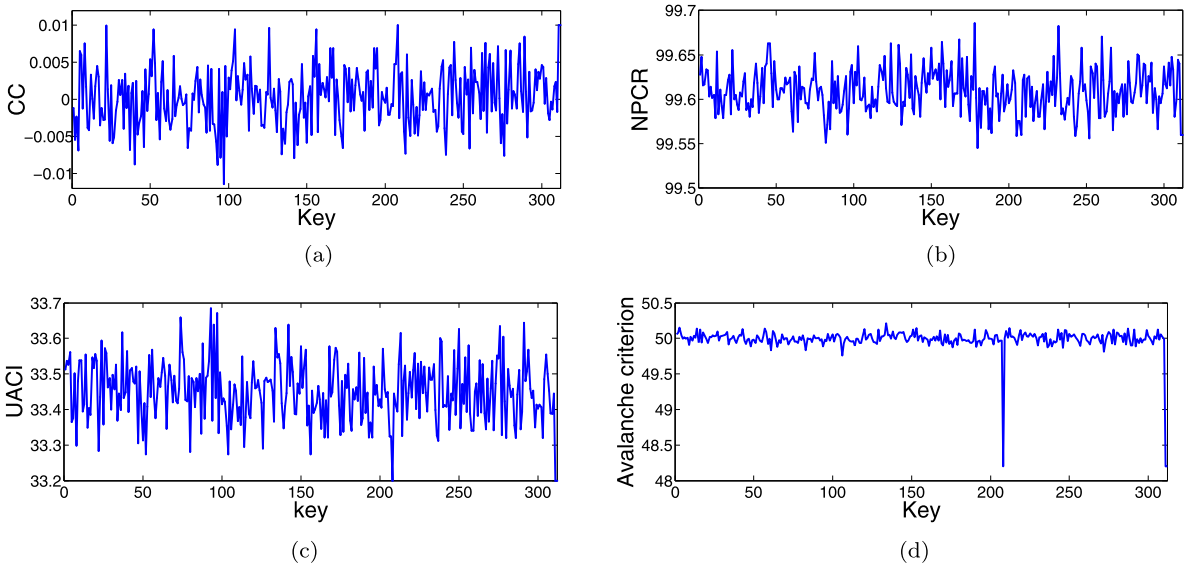

Fig. 13. Encrypted results for a black image for (a) CC of different keys, (b) NPCR of different keys (NPCR*0.05) ($\geq$99.5693), (c) UACI of different keys (UACI*0.05) [33.22,33.70], (d) avalanche criterion of different keys $\cong$50.

Table 5
Chosen plaintext attack/known plaintext attack analysis.

4×4 gray level matrix of plainimage	1	2	3	4	1	1	1	1	255	255	255	255
	1	2	3	4	1	1	1	1	255	255	255	255
	1	2	3	4	1	1	1	1	255	255	255	255
	1	2	3	4	1	1	1	1	255	255	255	255
4×4 gray level matrix of cipherimage	124	187	51	219	142	120	20	85	91	201	10	180
	254	150	29	40	201	180	106	14	77	21	24	132
	165	120	125	133	251	151	100	7	12	185	119	108
	241	142	255	217	181	150	21	20	124	132	41	4

4.1.6. Resisting known-plaintext and chosen-plaintext attacks

Adversaries often use specific type of images to cryptanalyse encryption algorithms such as all black or white images. These types of images allow the adversary to locate observable patterns that can be used to extract key-related information from the resulting cipherimage. To depict the capability of the proposed cipher in eliminating all traces of chosen patterns in a plainimage, three different 4×4 images were encrypted. The resulting pixel values after before and after encryption are shown in Table 5. It can be seen that any specific patterns, white or black images would be fully randomized after encryption with no observable patterns.

The capability of the proposed algorithm is also compared against image ciphers proposed by Wang et al. [46] and Liu et al. [47]. An all black image was encrypted by using all three ciphers but running only one round of the encryption process for each. Even without performing the two full rounds of encryption, the proposed algorithm is able to successfully encrypt the all-black plainimage as shown in Fig. 14(a). However, there still exists observable patterns in Wang and Liu's algorithms as shown in Fig. 14(b) and (c). The attacker is unable to obtain any useful information from the encrypted known-plainimage or chosen-plainimage for the proposed algorithm. These results show that the proposed image encryption algorithm is resistant against the attacks discussed in [48,49].

4.2. Performance analysis

All experiments were conducted by using Matlab R2012b on a computer with a 2.60 GHz processor, 2 GB RAM memory and Windows 7. Lena image with a size of 256×256 was encrypted and decrypted 50 times with an average of speed of 0.6212 s and 0.6121 s respectively. Table 6 provides a speed comparison between the proposed cipher with other image encryption algorithms for an image size of 256×256 pixels. The efficiency of the proposed algorithm is comparable with existing chaos-based image ciphers.

In terms of computational complexity, the proposed encryption scheme was calculated to require $O(MN\log(MN))$ iterations encrypt an image with a width and height of M and N respectively. Table 6 compares performance data of the proposed algorithm with other encryption schemes.

4.3. Comparisons with existing work

The experiments conducted in this work has used the same parameters and input data as other existing ciphers to ensure that a fair comparison can be made in terms of its security characteristics. Table 7 shows a comparison of statistical results where the 256×256 Lena image was used as the plainimage. The entropy of the proposed cipher is higher as compared to those in [3,10,52,53]. The correlation coefficients of the proposed system is also closer to 0 compared to [3,10,47].

The proposed encryption algorithm has the flexibility to support multiple key sizes of up to 312 bits which is large enough to resist against brute force attacks. The keyspace is larger than ciphers proposed in [3,47,52,53], and less than the methods in [10,38]. However, a larger key size does not necessarily guarantee higher security, especially against statistical-based cryptanalytic attacks.

The proposed system also demonstrates strong a resistance against differential attacks where the NPCR and UACI values close

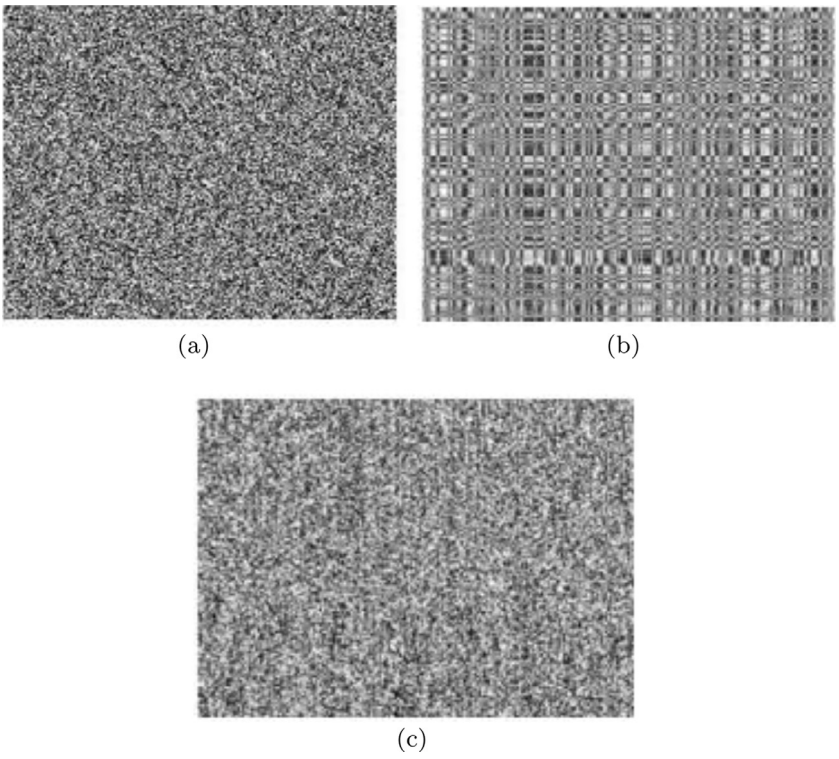

Fig. 14. Encrypted results for a black image for (a) one round of the proposed system, (b) one round of Wang's algorithm [46], (c) one round of Liu's algorithm [47].

Table 6
Encryption speed for an image size of 256×256 pixels and computational complexity.

Algorithm	Encryption time(s)	Number of operations	
		Confusion	Diffusion
Proposed system	0.6212	$6MN + 2MN\log(\frac{MN}{2}))$	$12MN$
Ref. [50]	2.40172	$8MN\log(2MN)$	$4MN$
Ref. [51]	3.80120	$12MN$	$24MN + \log(4MN)$

Table 7
Comparison of the proposed cipher against other image ciphers.

Algorithms	GSE	Correlation coefficient			NPCR	UACI	Key space
		Horizontal	Vertical	Diagonal			
Proposed system	7.9975	−0.0017	−0.0084	−0.0019	99.62	33.50	2^{312}
Ref. [47]	7.9976	−0.003	−0.0024	−0.0034	99.61	33.33	2^{256}
Ref. [52]	7.9972	0.0022	0.0001	−0.0017	99.65	33.44	2^{128}
Ref. [3]	7.9971	−0.0016	−0.0033	0.013	99.62	33.45	2^{128}
Ref. [10]	7.9963	−0.0048	−0.0112	−0.0045	99.62	33.7	2^{624}
Ref. [38]	7.9979	0.0008	0.0008	0.0005	99.61	33.45	2^{512}
Ref. [53]	7.9971	–	–	–	99.62	33.65	2^{100}

to the ideal values of 99.61% and 33.46%. In this aspect, the proposed cipher slightly outperforms all the benchmark ciphers. Overall, the proposed algorithm has shown to be flexible, efficient and highly secure, making it a viable choice when image encryption is required.

5. Conclusion

This paper has proposed a new image encryption based on a hybrid chaotic system and chaotic perturbation based on plainimage pixels. The hybrid chaotic maps show better chaotic complexity and performance as compared to other existing chaotification methods. The proposed chaotification method can be used to enhance the dynamical properties of any 1D chaotic map for cryptographic applications because it does not affect the phase space of the original map. Thus, no changes need to be made to the cryptosystem that utilizes the aforementioned map. Chaotic state and system parameter perturbation based on plainimage pixels are used in the permutation and diffusion phases of the image cipher, respectively. Due to the novel perturbation algorithm, the proposed scheme is highly sensitive to any changes in the plainimage or cipherimage. Security and performance evaluation show that the proposed cipher has various desirable characteristics such as efficiency, flexibility, and resistance against cryptanalytic attacks. Thus, the proposed cipher is suitable for practical applications. This in turn also demonstrates the capability of the proposed hybrid

chaotic system in designing secure cryptosystems in the future. However, the high sensitivity of the proposed algorithm to slight changes in the plaintext or ciphertext makes it susceptible to noise and cropping attacks. Improving the robustness against these attacks will be addressed in future work.

Conflict of interest

The authors declare that they have no conflicts of interest.

Acknowledgements

This work has been partially supported by Universiti Sains Malaysia under grant no. 304/PKOMP/6316280 and the National Natural Science Foundation of China under grant no. 61702212.

References

[1] M. Kanafchian, B. Fathi-Vajargah, A novel image encryption scheme based on clifford attractor and noisy logistic map for secure transferring images in navy, Int. J. e-Navig. Marit. Econ. 6 (2017) 53–63, doi:10.1016/j.enavi.2017.05.007.
[2] A.Y. Niyat, M.H. Moattar, M.N. Torshiz, Color image encryption based on hybrid hyper-chaotic system and cellular automata, Opt. Lasers Eng. 90 (2017) 225–237.
[3] X. Chai, Z. Gan, K. Yang, Y. Chen, X. Liu, An image encryption algorithm based on the memristive hyperchaotic system, cellular automata and DNA sequence operations, Signal Process. Image Commun. 52 (2017) 6–19, doi:10.1016/j.image.2016.12.007.
[4] S. Tedmori, N. Al-Najdawi, Image cryptographic algorithm based on the haar wavelet transform, Inf. Sci. 269 (2014) 21–34.
[5] A. Belazi, A.A.A. El-Latif, S. Belghith, A novel image encryption scheme based on substitution-permutation network and chaos, Signal Process. 128 (2016) 155–170.
[6] X. Chai, Z. Gan, Y. Chen, Y. Zhang, A visually secure image encryption scheme based on compressive sensing, Signal Process. 134 (2016) 35–51.
[7] J.S. Khan, J. Ahmad, Chaos based efficient selective image encryption, Multidimension. Syst. Signal Process. (2018), doi:10.1007/s11045-018-0589-x.
[8] X. Chai, Y. Chen, L. Broyde, A novel chaos-based image encryption algorithm using DNA sequence operations, Opt. Laser Eng. 88 (2017) 197–213.
[9] D. Ravichandran, P. Praveenkumar, J.B.B. Rayappan, R. Amirtharajan, DNA chaos blend to secure medical privacy, IEEE Trans. Nanobiosci. 16 (8) (2017) 850–858.
[10] A. Belazi, A.A. El-Latif, S. Belghith, A novel image encryption scheme based on substitution-permutation network and chaos, Signal Process. 128 (2016) 155–170, doi:10.1016/j.sigpro.2016.03.021.
[11] U. Çavuşoğlu, S. Kaçar, I. Pehlivan, A. Zengin, Secure image encryption algorithm design using a novel chaos based S-box, Chaos Solitons Fract. 95 (2017) 92–101, doi:10.1016/j.chaos.2016.12.018.
[12] A. Akhavan, A. Samsudin, A. Akhshani, A symmetric image encryption scheme based on combination of nonlinear chaotic maps, J. Franklin Inst. 348 (8) (2011) 1797–1813, doi:10.1016/j.jfranklin.2011.05.001.
[13] Z. Hua, F. Jin, B. Xu, H. Huang, 2D logistic-sine-coupling map for image encryption, Signal Process. 149 (2018) 148–161.
[14] Y.Z.Z. Hua, S. Yi, Medical image encryption using high-speed scrambling and pixel adaptive diffusion, Signal Process. 149 (2018) 148–161.
[15] A.G. Radwan, S.H. AbdElHaleem, S.K. Abd-El-Hafiz, Symmetric encryption algorithms using chaotic and non-chaotic generators: a review, J. Adv. Res. (2016) 193–208.
[16] J.S. Teh, K. Tan, M. Alawida, A chaos-based keyed hash function based on fixed point representation, Cluster Comput. (2018) 1–12, doi:10.1007/s10586-018-2870-z.
[17] B. Yang, X. Liao, Period analysis of the logistic map for the finite field, Sci. China Inf. Sci. 60 (2017) 022302.
[18] B. Yang, X. Liao, Some properties of the logistic map over the finite field and its application, Signal Process. 153 (2018) 231–242.
[19] Y. Zhou, Z. Hua, C. Pun, C.L.P. Chen, Cascade chaotic system with applications, IEEE Trans. Cybern. 45 (9) (2015) 2001–2012.
[20] R. Lan, J. He, S. Wang, T. Gu, X. Luo, Integrated chaotic systems for image encryption, Signal Process. 147 (2018) 133–145.
[21] Y. Liu, Y. Luo, S. Song, L. Cao, J. Liu, J. Harkin, Counteracting dynamical degradation of digital chaotic chebyshev map via perturbation, Int. J. Bifurcation Chaos. 27 (03) (2017) 1750033, doi:10.1142/S021812741750033X.
[22] Y. Zhou, L. Bao, C. Chen, Image encryption using a new parametric switching chaotic system, Signal Process. 93 (2013) 3039–3052.
[23] W. Yue, Z. Yicong, L. Bao, Discrete wheel-switching chaotic system and applications, IEEE Trans. Circuits Syst. I Regul. Pap. 12 (2014) 3469–3477.
[24] D. Ravichandran, P. Praveenkumar, J.B.B. Rayappan, R. Amirtharaja, Chaos based crossover and mutation for securing dicom image, Comput. Biol. Med. 72 (2016) 170–184.
[25] Y. Zhou, L. Bao, C.P. Chen, A new 1D chaotic system for image encryption, Signal Process. 97 (2014) 172–182.
[26] C. Pak, L. Huang, A new color image encryption using combination of the 1d chaotic map, Signal Process. 138 (2017) 129–137.
[27] R. Parvaz, M. Zarebnia, A combination chaotic system and application in color image encryption, Opt. Laser Technol. 101 (2018) 30–41.
[28] Y. Deng, H. Hu, N. Xiong, W. Xiong, L. Liu, A general hybrid model for chaos robust synchronization and degradation reduction, Inf. Sci. 305 (2015) 146–164.
[29] C. Shannon, Communication theory of secrecy systems, Bell Syst. Tech. J. 28 (4) (1949) 656–715.
[30] V. Patidar, N. Pareek, G. Purohit, K.K. Sud, Modified substitution-diffusion image cipher using chaotic standard and logistic maps, Commun. Nonlinear Sci. Numer. Simul. 15 (10) (2010) 2755–2765, doi:10.1016/j.cnsns.2009.11.010.
[31] Y. Wang, K.W. Wong, X. Liao, G. Chen, A new chaos-based fast image encryption algorithm, Appl. Soft Comput. 11 (1) (2011) 514–522, doi:10.1016/j.asoc.2009.12.011.
[32] W. Yue, Z. Yicong, B. Long, Image encryption based on three-dimensional bit-matrix permutation, Signal Process. 118 (2016) 36–50.
[33] Z.I. Zhu, W. Zhang, K.W. Wong, H. Yu, A chaos-based symmetric image encryption scheme using a bit-level permutation, Inf. Sci. 181 (6) (2011) 1171–1186.
[34] P. Ping, F. Xu, Y. Mao, Z. Wang, Designing permutation-substitution image encryption networks with Henon map, Neurocomputing 283 (2018) 53–63.
[35] C. Cao, K. Sun, W. Liu, A novel bit-level image encryption algorithm based on 2D-LICM hyperchaotic map, Signal Process. 143 (2018) 122–133.
[36] Z. Hua, Y. Zhou, C.M. Pun, C.L.P. Chen, 2D sine logistic modulation map for image encryption, Inf. Sci. 297 (2015) 80–94.
[37] M.A. Murillo-Escobar, C. Cruz-Hernández, F. Abundiz-Pérez, R.M. López-Gutiérrez, O.D. Campo, A RGB image encryption algorithm based on total plain image characteristics and chaos, Inf. Sci. 109 (6) (2015) 119–131.
[38] B. Norouzi, S.M. Seyedzadeh, S. Mirzakuchaki, R. Mosavi, A novel image encryption based on hash function with only two-round diffusion process, Multimedia Syst. 20 (1) (2014) 45–64.
[39] D. Arroyo, J.M. Amigo, S. Li, G. Alvarez, On the inadequacy of unimodal maps for cryptographic applications, in: XI RECSI 2010, 2010, pp. 37–42. arXiv:0805.4355.
[40] X. Wang, L. Teng, X. Qin, A novel colour image encryption algorithm based on chaos, Signal Process. 92 (4) (2012) 1101–1108.
[41] Y. Wu, Y. Zhou, G. Saveriades, S. Agaian, J. Noonan, P. Natarajan, Local Shannon entropy measure with statistical tests for image randomness, Inf. Sci. 222 (2013) 323–342.
[42] X. Liao, S. Lai, Q. Zhou, A novel image encryption algorithm based on self-adaptive wave transmission, Signal Process. 90 (2010) 2714–2722.
[43] E. Biham, A. Shamir, Differential cryptanalysis of deslike cryptosystems, J. Cryptol. 4 (1) (1991) 3–72.
[44] Y. Wu, J.P. Noonan, S. Agaian, NPCR and UACI randomness tests for image encryption, J. Sel. Areas Telecommun. 4 (1) (2011) 31–38.
[45] Algorithms, key size and protocols report 2018. Technical Report, ECRYPT-CSA. http://www.ecrypt.eu.org/csa/documents/D5.4-FinalAlgKeySizeProt.pdf.
[46] X. Wang, Q. Wang, Y. Zhang, A fast image algorithm based on rows and columns switch, Nonlinear Dyn. 79 (2) (2015) 1141–1149.
[47] W. Liu, K. Sun, C. Zhu, A fast image encryption algorithm based on chaotic map, Opt. Lasers Eng. 84 (2016) 26–36, doi:10.1016/j.optlaseng.2016.03.019.
[48] X. Ge, F. Liu, B. Lu, C. Yang, Improvement of Rhouma's attacks on gao algorithm, Phys. Lett. A 374 (11) (2010) 1362–1367, doi:10.1016/j.physleta.2010.01.024.
[49] G. Alvarez, V. Fernandez, D. Arroyo, A basic framework for the cryptanalysis of digital chaos-based cryptography, in: 6th International Multi-Conference on Systems, Signals and Devices SSD'09, 2009, IEEE, 2009, pp. 1–6.
[50] L. Teng, X. Wang, A bit-level image encryption algorithm based on spatiotemporal chaotic system and selfadaptive, Opt. Commun. 285 (20) (2012) 4048–4054.
[51] L. Xu, Z. Li, J. Li, W. Hua, A novel bit-level image encryption algorithm based on chaotic maps, Opt. Laser Eng. 78 (21) (2016) 17–25.
[52] A.Y. Niyat, M.H. Moattar, M.N. Torshiz, Color image encryption based on hybrid hyper-chaotic system and cellular automata, Opt. Lasers Eng. 90 (2017) 225–237, doi:10.1016/j.optlaseng.2016.10.019.
[53] G. Ye, X. Huang, An efficient symmetric image encryption algorithm based on an intertwining logistic map, Neurocomputing 251 (2017) 45–53, doi:10.1016/j.neucom.2017.04.016.