Optics Communications 284 (2011) 5415–5423

Contents lists available at SciVerse ScienceDirect

Optics Communications

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

A novel chaos-based bit-level permutation scheme for digital image encryption

Chong Fu [a,*], Bin-bin Lin [b], Yu-sheng Miao [b], Xiao Liu [b], Jun-jie Chen [b]

[a] School of Information Science and Engineering, Northeastern University, Shenyang 110004, China
[b] Software College, Northeastern University, Shenyang 110004, China

ARTICLE INFO

Article history:
Received 23 June 2011
Received in revised form 5 August 2011
Accepted 5 August 2011
Available online 23 August 2011

Keywords:
Image encryption
Chaos
Bit-level permutation
Chaotic sequence sorting
Arnold Cat map

ABSTRACT

Confidentiality is an important issue when digital images are transmitted over public networks, and encryption is the most useful technique employed for this purpose. Image encryption is somehow different from text encryption due to some inherent features of image such as bulk data capacity and high correlation among pixels, which are generally difficult to handle by conventional algorithms. Recently, chaos-based encryption has suggested a new and efficient way to deal with the intractable problems of fast and highly secure image encryption. This paper proposes a novel chaos-based bit-level permutation scheme for secure and efficient image cipher. To overcome the drawbacks of conventional permutation-only type image cipher, the proposed scheme introduced a significant diffusion effect in permutation procedure through a two-stage bit-level shuffling algorithm. The two-stage permutation operations are realized by chaotic sequence sorting algorithm and Arnold Cat map, respectively. Results of various types of analysis are interesting and indicate that the security level of the new scheme is competitive with that of permutation–diffusion type image cipher, while the computational complexity is much lower. Therefore the new scheme is a good candidate for real-time secure image communication applications.

1. Introduction

With the advancements of multimedia and network technologies, a vast number of digital images are now transmitted over the Internet and through wireless networks for convenient accessing and sharing. As a result, protection of digital images against illegal copying and distribution has become an important issue. Conventional encryption algorithms such as DES, AES, IDEA and, RSA are not suitable for practical image cipher in the aspect of efficiency, especially when the image is large. Chaos-based algorithm has been widely investigated for a decade or so, to meet the demand for real-time secure image communication over open networks. Since 1990s, many researchers have noticed that there exists a close relationship between chaos and cryptography. The fundamental features of chaotic dynamical systems such as ergodicity, mixing property, sensitivity to initial conditions/system parameters, etc. can be considered analogous to some ideal cryptographic properties such as confusion, diffusion, balance, avalanche properties, etc.

Ever since Fridrich proposed the chaotic image encryption scheme in 1998 [1], there have been increasing researches on chaos-based image cryptosystem. The major core of these systems consists of one or several chaotic maps serving the purpose of either just encrypting the image or shuffling the image and subsequently encrypting the resulting shuffled image [1–34]. In [2], Scharinger proposed a chaotic Kolmogorov-flow-based image encryption algorithm. In his scheme, the plain-image is firstly permuted through a key-controlled chaotic system based on the Kolmogorov flow and then substituted by using shift-registered pseudo-random number generator, which alters the statistical property of the cipher-image. In [4,5], the 2D chaotic Cat map and Baker map are generalized to 3D for designing a real-time secure symmetric encryption scheme. The two approaches employ the 3D map to shuffle the positions of image pixels and use another chaotic map to confuse the relationship between the cipher-image and plain-image. In [9], Lian et al. employed a chaotic standard map in the substitution stage and a quantized logistic map in the diffusion stage. The parameters of these two chaotic maps are determined by a key stream generated in each round. In [14], Xiang et al. proposed a selective gray-level image encryption scheme. In this scheme, only a portion of significant bits of each pixel is encrypted by the key-stream generated from a one-way coupled map lattice. In [16], a typical coupled map was mixed with a one-dimensional chaotic map and used for high degree security image encryption. In [20], Wong et al. introduced certain diffusion effect in the confusion stage with the purpose of reducing the workload of the time-consuming diffusion procedure, and thus the efficiency of the cryptosystem is prompted. In [21], Sun et al. proposed a novel image encryption scheme based on spatial chaos map. The basic idea is to encrypt the image in space with spatial chaos map pixel by pixel, and then the pixels are confused in multiple directions of space. In [26], Rhouma et al. proposed an OCML-based color image encryption scheme with a stream cipher structure.

* Corresponding author. Tel.: +86 24 23388825.
E-mail address: fuchong@ise.neu.edu.cn (C. Fu).

0030-4018/$ – see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.optcom.2011.08.013

In this scheme, a 192-bit-long external key is used to generate the initial conditions and the parameters of the OCML by making some algebraic transformations to the secret keys. In [29], Wong et al. proposed an efficient diffusion mechanism using simple table lookup and swapping techniques as a light-weight replacement of the 1D chaotic map iteration. In [32], Elashry et al. proposed a new homomorphic image cryptosystem. The idea of this system is based on encrypting the reflectance component after the homomorphic transform and embedding the illumination component as a least significant bit water mark into the encrypted reflectance component.

This work proposes a novel chaos-based bit-level permutation scheme for secure and efficient image cipher. Unlike conventional permutation-only schemes which operate on pixel-plane, the new scheme performs a two-stage shuffling operations on bit-plane and hence a significant diffusion effect is introduced in permutation procedure. Results of various analyses have indicated that the security level of the proposed scheme is competitive with that of permutation–diffusion type image cipher, while the computational complexity is much lower. The rest of this paper is organized as follows. Section 2 presents the architecture of the proposed scheme. The detailed two-stage bit-level permutation algorithms are discussed in Sections 3 and 4, respectively. In Section 5, we analyze the security of the proposed image cipher and evaluate its performance through various statistical analyses, key space analyses, key sensitivity analyses, speed analyses, etc. Finally, Section 6 concludes the paper.

2. Architecture of the proposed permutation scheme

Chaos-based image cryptosystems can be classified into three categories according to their architecture. These categories are: permutation-only, diffusion-only and the combination form. Among them, the permutation-only type image cipher is superior in the aspect of efficiency due to its lowest computational complexity. It only shuffles the position of each pixel in a secret order while it does not alter its value. Three types of two-dimensional investable chaotic maps named Arnold Cat map, Baker map, and Standard map are commonly employed to realize pixel permutation. However, the security of this kind of image cipher has been proved to be relatively weak due to the following two reasons.

(1) The histogram of the shuffled image is completely unchanged since permutation-only type encryption scheme only shuffles the pixel positions without changing their values. This weakness has caused the development of various kinds of attacks, in which statistical attack is the most widely used to break such a cryptosystem.
(2) The invertible chaotic map must be discretized when applied to image permutation. Unfortunately, not all useful features of chaos can be inherited by discretization. The most serious problem is that an aperiodic chaotic map may become periodic after discretization, which will downgrade the security of the cryptosystem, because the possible intruders may apply the same chaotic map to the cipher-image to recover the plain-image through simple iteration operations.

As a remedy, the permutation–diffusion type image cipher introduces a substitution module, which alters the pixel values sequentially and the modification made to a pixel usually depends on the accumulated effect of all the previous pixel values, so that a slight change in one pixel could be spread out to almost all the subsequent pixels. Chaotic map is used as generation of key stream for substitution and the substitution could be one of simple operations such as XOR, XNOR, shift, add, and subtract or a combination of these simple operations.

The introduction of substitution module significantly enhanced the security of the cryptosystem. However, it brings another problem. The substitution is a time-consuming process since a considerable amount of computation load is devoted to the real number arithmetic operation and the subsequent quantization required by the key stream generation. While for security purpose, the computing precision cannot be too low. Such computation complexity greatly downgrades its advantage in practical large image encryption.

To overcome the drawbacks of conventional permutation-only type image cipher, a novel bit-level permutation scheme is proposed, as shown in Fig. 1.

There are two iterative permutation stages in proposed scheme. Firstly, the plain-image is extended to bit-plane. In first permutation stage, each bit in the bit-plane is shuffled by using chaotic sequence sorting algorithm. Then, the permutated bit-plane is divided into squares with equal size. In second permutation stage, each square is shuffled independently by using two-dimensional area-preserving chaotic map. Finally, all the permutated squares are concatenated together and recovered to pixel-plane to generate the cipher image. To sufficiently disturb the correlation among adjacent pixels and confuse the relationship between cipher image and plain image, there are m rounds iteration in second permutation stage with $m \geq 1$ and the overall permutation operation is performed for n rounds. The initial parameters and conditions of the chaotic maps that govern the permutation operation serve as the secret keys.

Since the two-stage permutation operations are performed on bit-plane, both the shuffling on pixel position and the modification of pixel value are carried out simultaneously. As a result, a significant diffusion effect is introduced in the new permutation scheme and hence the first security weakness of the conventional permutation-only scheme is well solved. Moreover, the second security weakness can also be well solved by using the chaotic sequence sorting based permutation algorithm. The two-stage permutation operations are realized by chaotic Chebyshev map and Arnold Cat map, respectively, which will be discussed next.

3. Bit-level permutation using chaotic sequence sorting

Before stage-1 permutation, the plain image is firstly extended to bit-plane. For a grayscale image, the brightness from black to white is

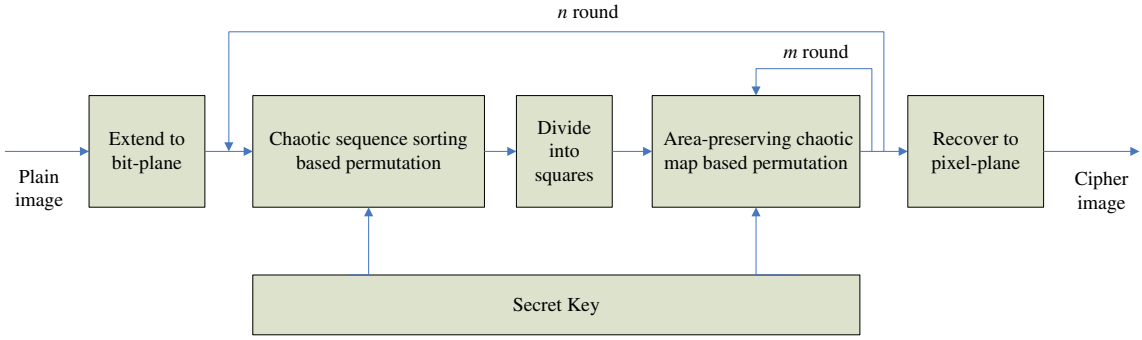

Fig. 1. Architecture of proposed permutation scheme.

Fig. 2. Plain Lenna image.

quantized into a number of levels. The division of brightness depends on the representation precision in digitalization, and the most commonly used one is 256. In digital computer system, these gray levels are represented by an 8-bit format, given by

$$G(x,y) = b(7)b(6)b(5)\cdots b(0), \tag{1}$$

where $G(x, y)$ is the value of the pixel at coordinate (x, y) and the number in parentheses indicates the bit index from highest bit 7 to the lowest bit 0. Therefore, an M^*N size image with 256 gray levels can be extended to a bit-plane with size $M^*(N^*8)$, which is a binary image because only two possible values (0 and 1) exist for each pixel. We take Lenna image as example, the plain image and its extended bit-plane are shown in Figs. 2, 3, respectively.

To realize stage-1 permutation, a novel chaotic sequence sorting based shuffling algorithm is proposed. The chaotic sequence is generated by Chebyshev map, as described by

$$x_{n+1} = T_k(x_n) = \cos\left(k \cdot \cos^{-1} x_n\right), \quad x_n \in [-1, 1] \tag{2}$$

where k and x_n are parameter and state value, respectively. If one chooses $k \in [2, \infty)$, the system is chaotic as illustrated by Fig. 4. The initial value x_0 and parameter k are used as the key.

The detailed permutation procedure is described as follows:

Step 1 Iterate Eq. (2) for N_0 times to avoid the harmful effect of transitional procedure, where N_0 is a constant.

Step 2 We continue to iterate Chebyshev map m times and obtain a chaotic sequence $X = \{x_1, x_2, \ldots, x_m\}$ from the state value. Here, notice that the value of -1 is a 'bad' point, trapping the iterations to the fixed point 0. If this case is encountered, a tiny perturbation should apply.

Step 3 The chaotic sequence X is sorted in ascending order, and we get a new set $Y = \mathrm{sort}(X) = \{y_1, y_2, \ldots, y_m\}$.

Step 4 Let $P = \{p_1, p_2, \ldots, p_m\}$ denote the permutation vector of X, such that $Y = X(P)$. Then rearrange each row of the bit-plane according to P, that is, move the first row to p_1th row, the second row to p_2th row, …, the last row to p_mth row, as illustrated by Fig. 5.

Step 5 Each column of the bit-plane is shuffled in the same way using a chaotic sequence of length N^*8.

Fig. 6 shows the results of applying above permutation algorithm on plain bit-plane (Fig. 3). The keys are $k = 4.0$, $x_0 = 0.30000000$.

Each pixel in shuffled image can be represented as

$$G'(x,y) = b'(7)b'(6)b'(5)\cdots b'(0), \tag{3}$$

where $b'(n)$ $(n \in [0\text{–}7])$ are the bits moved from other positions on the bit-plane. Obviously, the probability of $G(x, y) \equiv G'(x, y)$ is $(1/2)^8 = (1/256)$, thus a significant diffusion effect is introduced.

In decryption process, same permutation vector is generated using the same keys. Then the plain image can be obtained by moving back the rows and columns according to the permutation vector.

4. Bit-level permutation using Arnold Cat map

In the second permutation stage, the shuffled bit-plane (Fig. 6) is firstly divided into eight bit-squares from left to right, as shown in Fig. 7(a)–(h), respectively.

Then each bit-square is shuffled independently with different control parameters by using Arnold Cat map. The well-known Cat map is a two-dimensional invertible chaotic map introduced by Arnold and Avez [35]. The mathematical formula is:

$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} \bmod 1 = \boldsymbol{A} \begin{bmatrix} x_n \\ y_n \end{bmatrix} \bmod 1 \tag{4}$$

where x mod 1 means the fractional part of x for any real number x. The map is area-preserving since $\det|\boldsymbol{A}| = 1$. The Cat map is most easily described in geometric terms. As shown in Fig. 8, a unit square is first stretched by the linear transform matrix $\boldsymbol{A}$ and then folded back to the unit square by the modulo operation.

The above 2D Cat map can be generalized by introducing two control parameters, p and q, as follows:

$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} 1 & p \\ q & pq+1 \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} \bmod 1 = \boldsymbol{A} \begin{bmatrix} x_n \\ y_n \end{bmatrix} \bmod 1. \tag{5}$$

In order to incorporate generalized chaotic Cat map into image encryption that operated on a finite set, it has to be discretized, while reserving some of its useful features such as the mixing property and the sensitivity to initial conditions and parameters. The discretized version Cat map can be obtained simply by changing the range of (x, y) from the unit square $I \times I$ to the discrete lattice $N \times N$.

$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} 1 & p \\ q & pq+1 \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} \bmod N = \boldsymbol{A} \begin{bmatrix} x_n \\ y_n \end{bmatrix} \bmod N \tag{6}$$

where N is the width or length of a square image. Thus the parameters p, q and the number of iterations M all can be used as the secret keys.

Since there only exist a linear transformation and mod function, it is very efficient to shuffle the bit-squares by using the Cat map. The results of applying the discretized Cat map to the eight bit-squares are shown in Fig. 9(a)–(h), respectively, the iteration times are set to three. As can be seen from Fig. 9, the bits in each bit-square are well-distributed, thus the diffusion effect is further enhanced.

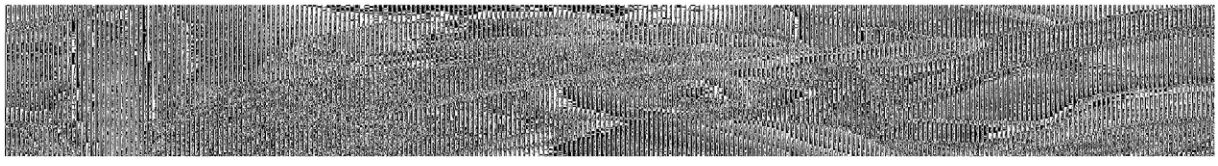

Fig. 3. Bit-plane of plain Lenna image.

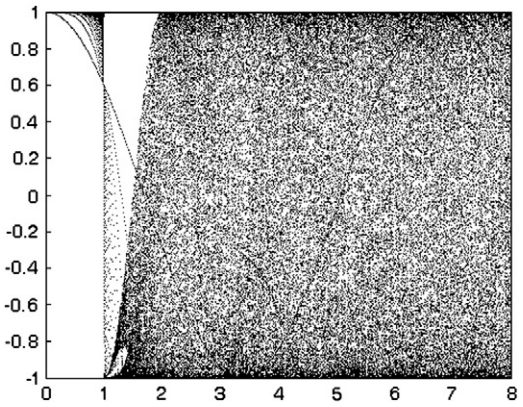

Fig. 4. The Chebyshev map.

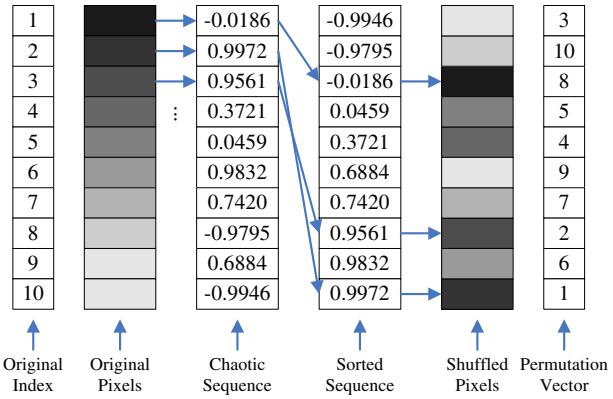

Fig. 5. Bit-level permutation based on chaotic sequence sorting.

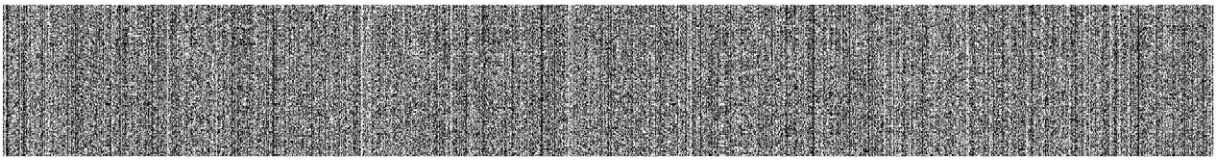

Fig. 6. The shuffled bit-plane.

The inverse transform of the Cat map for deciphering is given by

$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} pq+1 & -p \\ -q & 1 \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} \bmod N. \quad (7)$$

Finally, all the 8 bit-squares are concatenated from left to right and recovered to pixel-plane to generate the cipher image.

5. Security analysis

The crucial measure for the quality of a cryptosystem is its capability to withstand the attempts of an unauthorized participant, or an opponent, to gain knowledge about the unencrypted information. A good cryptosystem should resist all kinds of known attacks, such as known/chosen plain-text attack, cipher-text only attack, statistical attack, differential attack, and various brute-force attacks. Some security analysis has been performed on the proposed scheme, including the most important ones like key space analysis, statistical analysis (including histogram, information entropy, and correlation of adjacent pixels), and key sensitivity analysis, which has demonstrated the satisfactory security of the new scheme, as discussed in the following.

5.1. Key space analysis

The key space is the total number of different keys that can be used in the encryption/decryption procedure. For an effective cryptosystem, the key space should be large enough to make brute-force attack

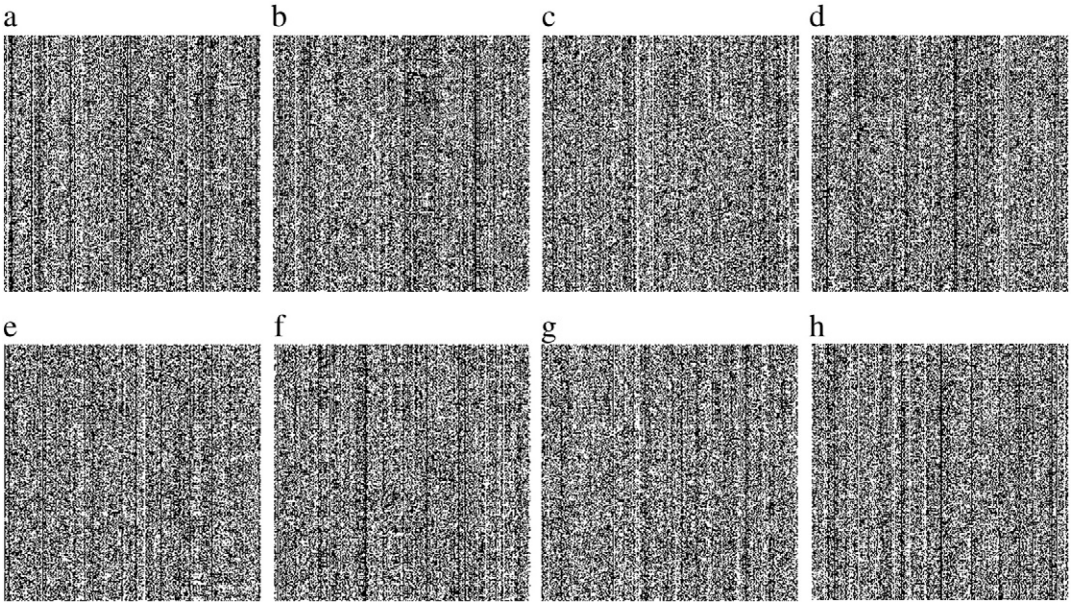

Fig. 7. Eight bit-squares of the shuffled bit-plane. (a)–(h) is the bit-square from left to right, respectively.

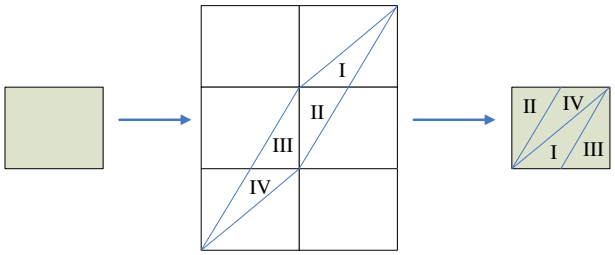

Fig. 8. The Cat map.

infeasible. As mentioned above, the key of the proposed cryptosystem is composed of two parts: ① the initial parameter and condition (k, x_0) of chaotic Chebyshev map, where $x_0 \in [-1, 1]$ and k can have any real value greater than 2.0; ② the control parameters (p, q) and iteration times m of Cat map, where $p, q, m \in N^+$.

According to the IEEE floating-point standard [36], the computational precision of the 64-bit double-precision number is about 10^{-15}. Therefore, the total number of possible values of x_0 that can be used as a part of the key is approximately 2×10^{15}. As in the proposed image encryption scheme, k can have any real value greater than 2.0 hence it has infinite number of possible values that can be used as a part of the key. However, the range of k should be restricted to a particular interval of 2π to prevent Chebyshev map from producing periodic orbits, then for k there will be approximately $2\pi\times10^{15}$ different values possible.

For discretized Cat map, the values of p, q should be restricted to the set N since the four-tuple $[1, (p+k_1N), (q+k_2N), (p+k_1N)(q+k_2N)+1]$ generates the same cipher as the four-tuple $[1, p, q, (pq+1)]$ for any $k_1, k_2, k_3, k_4 \in Z$. Thus the total number of ciphering keys for the Cat map is $(N^2)^m$, where N is the bit-square size.

The two parts of the key are independent of each other. Therefore, the complete key space of the proposed image encryption scheme is

$$H(x_0, k, p, q, m) \approx 12.57 \times 10^{30} \times \left(N^2\right)^m. \tag{8}$$

Here, it is proposed to take $m = 3$. If $N \geq 256$, the total size satisfies

$$H(x_0, k, p, q, m)\ N\ 3.54 \times 10^{45} \approx 2^{153} \tag{9}$$

which is large enough to resist brute-force attack.

5.2. Statistical analysis

It is well known that many ciphers have been successfully analyzed with the help of statistical analysis and several statistical attacks have been devised on them. Therefore, an effective cipher should be robust against any statistical attack. To prove the robustness of the new permutation scheme, we have performed statistical analysis by calculating the histogram, the information entropy and the correlation of two adjacent pixels in the ciphered image.

5.2.1. Histogram

An image histogram illustrates that how pixels in an image are distributed by plotting the number of pixels at each grayscale level. The distribution of cipher-text is of much importance. More specifically, it should hide the redundancy of plain-text and should not leak any information about the plain-text or the relationship between plain-text and cipher-text.

The histograms of plain-image and its ciphered image generated by the proposed scheme are shown in Fig. 10(a), (b), respectively. It's clear from Fig. 10(b) that the histograms of the cipher-image are fairly uniform and significantly different from that of the plain image and hence do not provide any clue to employ statistical attack. Fig. 10(c), (d) shows the histograms of ciphered images produced by conventional permutation-only scheme and permutation–diffusion scheme, respectively. As mentioned above, the conventional permutation-only image cipher only shuffles the pixel position without changing its value and hence the histogram of the shuffled image is the same as that of the plain-image. In [37], Zhu et al. proposed a chaos-based symmetric image encryption scheme using a bit-level permutation. In this scheme, a grayscale image is decomposed into 8 bit-planes and each bit-plane is shuffled separately by using Arnold Cat map, thus a certain diffusion effect is introduced in permutation stage. Fig. 10(e)

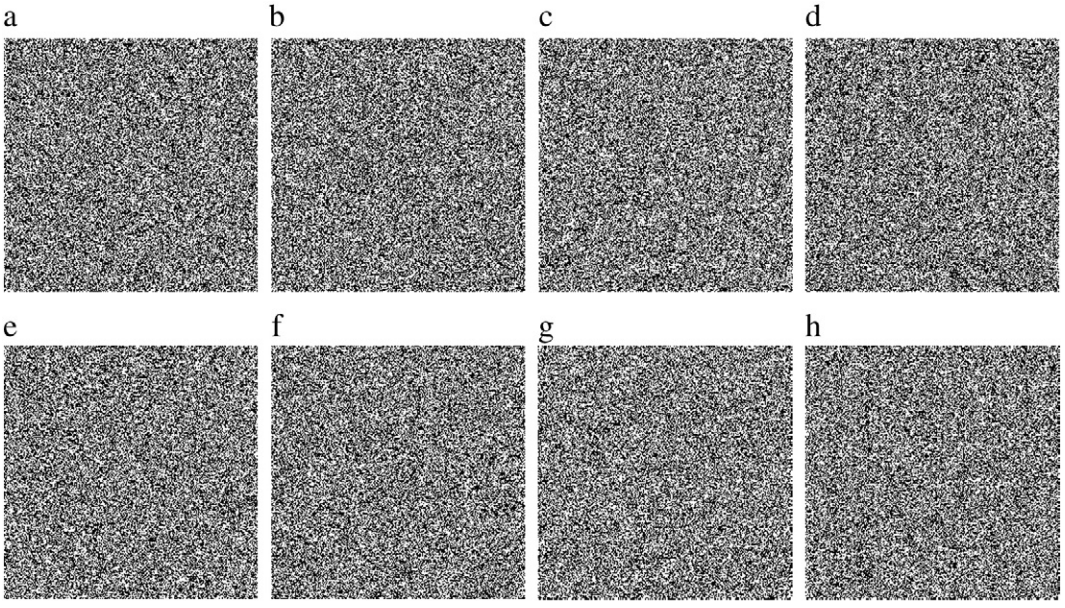

Fig. 9. The eight shuffled bit-squares with various control parameters. (a) $p=40$, $q=9$. (b) $p=35$, $q=8$. (c) $p=30$, $q=7$. (d) $p=25$, $q=6$. (e) $p=20$, $q=5$. (f) $p=15$, $q=4$. (g) $p=10$, $q=3$. (h) $p=5$, $q=2$.

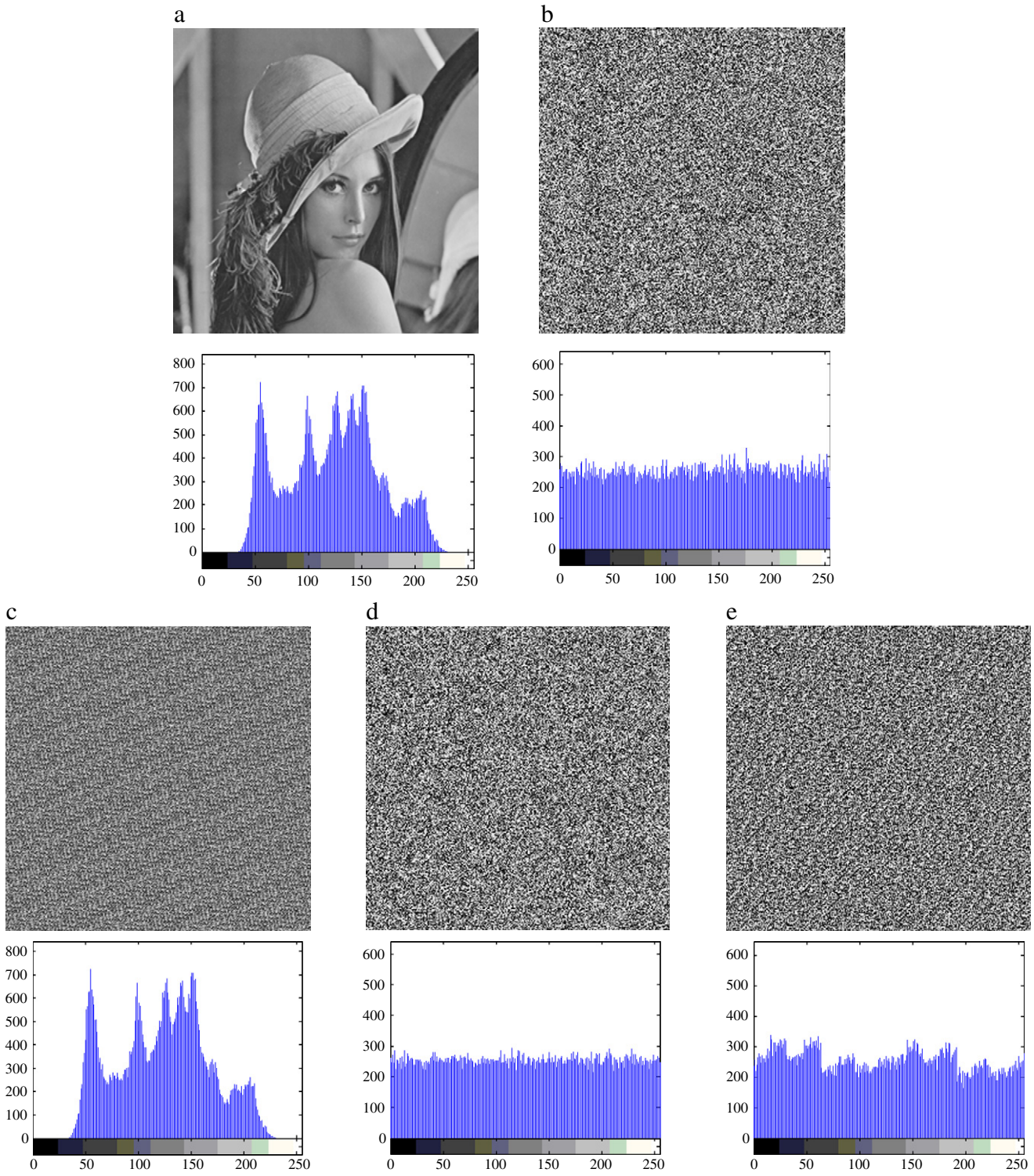

Fig. 10. The histogram. (a) Plain-image. (b) Proposed scheme. (c) Conventional permutation-only scheme. (d) Permutation–diffusion scheme. (e) Zhu et al.'s scheme.

Table 1
Information entropies for plain-image and ciphered-images.

Schemes	Plain-image	Proposed scheme	Conventional permutation-only scheme	Permutation–diffusion scheme
Information entropy	7.3507	7.9880	7.3507	7.9901

shows the histogram of shuffled image produced by Zhu et al.'s scheme. By comparing Fig. 10(b) with Fig. 10(d) and (e), it can be concluded that the histogram performance of the new permutation scheme is superior to that of Zhu et al.'s scheme and comparable with that of permutation–diffusion scheme owing to the significant diffusion effect introduced in the shuffling process.

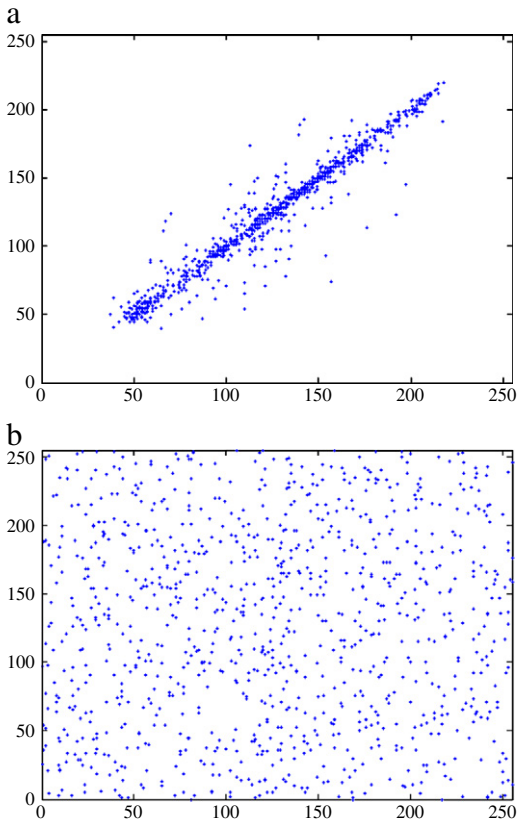

Fig. 11. Correlation of horizontal adjacent two pixels. (a) Plain image. (b) Ciphered image.

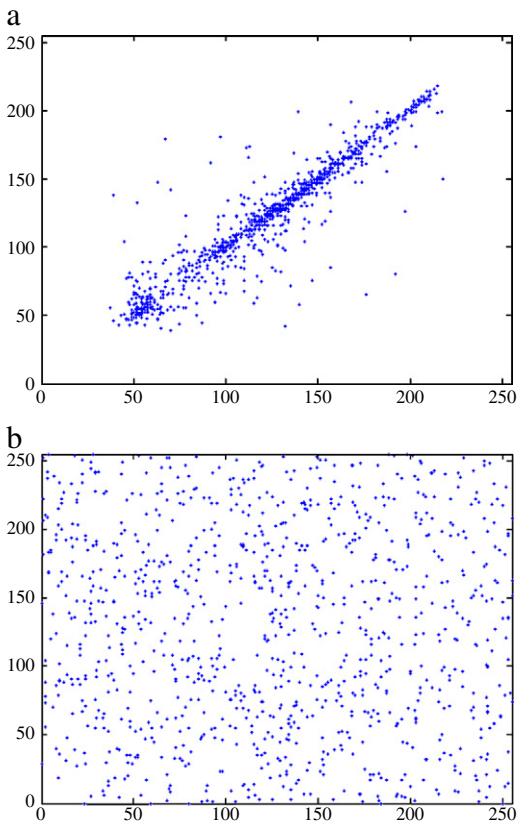

Fig. 12. Correlation of vertical adjacent two pixels. (a) Plain image. (b) Ciphered image.

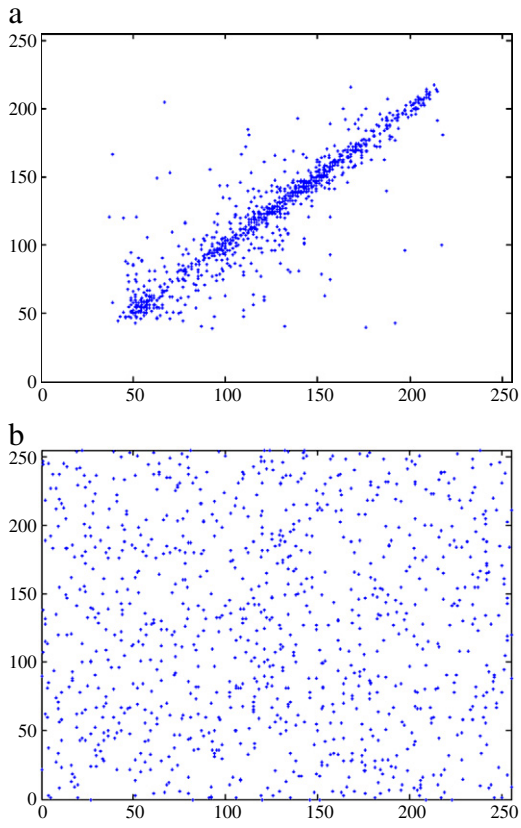

Fig. 13. Correlation of diagonal adjacent two pixels. (a) Plain image. (b) Ciphered image.

5.2.2. Information entropy

In information theory, entropy is the most significant feature of disorder, or more precisely unpredictability. To calculate the entropy $H(s)$ of a source s, we have:

$$H(s) = -\sum_{i=0}^{2^N-1} P(s_i)\ log_2 P(s_i) \tag{10}$$

where N is the number of bits to represent a symbol $s_i \in s$ and $P(s_i)$ represents the probability of symbol s_i so that the entropy is expressed in bits.

For a truly random source emitting 2^N symbols, the entropy is $H(s) = N$. therefore, for a ciphered image with 256 gray levels, the entropy should ideally be $H(s) = 8$. If the output of a cipher emits symbols with entropy less than 8, there exists certain degree of predictability, which threatens its security.

The entropies for plain image and ciphered images using various schemes are calculated and listed in Table 1. As can be seen from Table 1, the entropy of the cipher image produced by the new permutation scheme is very close to the theoretical value of 8 and is competitive with that of permutation–diffusion scheme. This means that information leakage in the encryption process is negligible and the cryptosystem is secure against entropy attack. However, the

Table 2
Correlation coefficients of two adjacent pixels in plain-image and ciphered-images.

Direction	Plain image	Proposed scheme	Permutation-only scheme	Permutation–diffusion scheme
Horizontal	0.9710	0.0368	0.0012	−0.0254
Vertical	0.9349	−0.0392	−0.0379	0.0119
Diagonal	0.9077	0.0068	−0.0439	0.0341

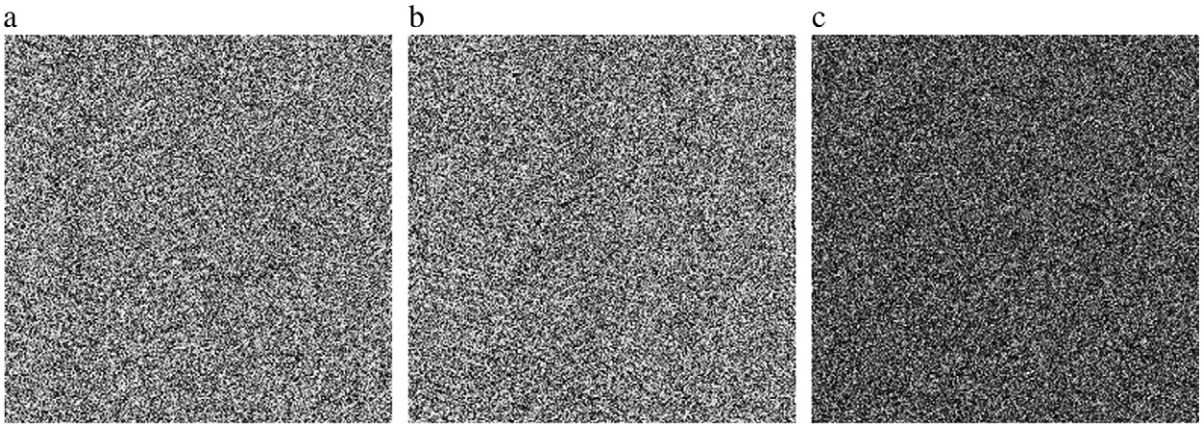

Fig. 14. Key sensitivity test: result 1. (a) Ciphered image using key ($k=4.0$, $x_0=0.70000000$). (b) Ciphered image using key ($k=4.0$, $x_0=0.70000001$). (c) Differential image between (a) and (b).

entropy of the ciphered-image produced by conventional permutation-only scheme is the same as that of plain image for the same reason discussed in Section 5.2.1.

5.2.3. Correlation of adjacent pixels

For an ordinary image having definite visual content, each pixel is highly correlated with its adjacent pixels either in horizontal, vertical or diagonal direction. However, an efficient image cryptosystem should produce the cipher image with sufficiently low correlation in the adjacent pixels.

The visual testing of the correlation of adjacent pixels can be done by plotting the distribution of the adjacent pixels in the plain image and its corresponding cipher image. The correlation distribution of two horizontally adjacent pixels, two vertically adjacent pixels and two diagonally adjacent pixels of the plain image and the cipher image produced by the proposed scheme is shown in Figs. 11–13, respectively.

It's clear from Figs. 11–13 that the strong correlation between adjacent pixels in plain image is greatly reduced in the cipher image produced by the proposed scheme.

To quantify and compare the correlations of adjacent pixels in the plain and cipher image, the following procedure is carried out. First, randomly select 1000 pairs of adjacent pixels in each direction from the plain image and its ciphered image. Then, calculate the correlation coefficient $r_{x,y}$ of each pair by using the following four formulas:

$$r_{xy} = \frac{cov(x,y)}{\sqrt{D(x)}\sqrt{D(y)}} \tag{11}$$

$$cov(x,y) = E[(x-E(x))(y-E(y))] \tag{12}$$

$$E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{13}$$

$$D(x) = \frac{1}{N}\sum_{i=1}^{N} (x_i-E(x))^2 \tag{14}$$

where x and y are grayscale values of two adjacent pixels in the image, and N denotes the total number of samples.

The results of the correlation coefficients for horizontal, vertical and diagonal adjacent pixels for the plain image and its ciphered images produced by various schemes are given in Table 2, from which we can see that all the three schemes provide a satisfactory correlation performance.

5.3. Key sensitivity

A good cryptosystem should be sensitive to the key which guarantees the security of the cryptosystem against the brute-force attack to some extent. The key sensitivity of a cryptosystem can be observed in two ways: (i) completely different cipher images should be produced when slightly different keys are used to encrypt the same plain image; (ii) the cipher image cannot be correctly decrypted even though there is only a slight difference between the encryption and decryption keys.

To evaluate the key sensitivity of the first case, the plain Lenna image is firstly encrypted using the test key ($k=4.0$, $x_0=0.70000000$) and the resultant cipher image is shown in Fig. 14(a). Then the test key is slightly changed to ($k=4.0, x_0=0.70000001$), which is used to encrypt the same plain image and the result is shown in Fig. 14(b). The difference between two ciphered images is 99.64% and the differential image is drawn in Fig. 14(c). Similar results are obtained with a slight change in other components of the key.

To evaluate the key sensitivity of the second case, the plain Lenna image is firstly encrypted using the test key ($k=3.99999999$, $x_0=0.7$) and the resultant cipher image is shown in Fig. 15(a). Then the ciphered image is tried to be decrypted using the keys ($k=3.99999999$, $x_0=0.7$) and ($k=4.0$, $x_0=0.7$). The resultant decrypted images are shown in Fig. 15(b), (c), respectively. The difference between wrong deciphered image to plain image is 99.61%. So it can be concluded that the proposed scheme is highly sensitive to the key. Instead any attempt to decrypt with a wrong key is in fact another encryption operation. Therefore differential attack would become very inefficient and practically useless.

5.4. Speed performance

Apart from the security consideration, efficiency is also an important aspect for a good image cryptosystem, particular for real-time Internet applications. The encryption speed of images with different sizes by using the proposed scheme and the permutation–diffusion scheme as well as the well-known DES algorithm is listed in Table 3. The computer used in this test is 2.4 GHz Intel Core2 Duo with 2 G memory. From Table 3 we can see that the speed of the proposed cryptosystem is superior to that of permutation–diffusion scheme and much faster than the classic DES algorithm. With such a speed, this image cryptosystem is appropriate to be used for real-time secure image transmission over broadband network, where the encryption time should be short relative to the transmission time.

Fig. 15. Key sensitivity test: result 2. (a) Ciphered image using key ($k=3.99999999$, $x_0=0.7$). (b) Deciphered image using key ($k=3.99999999$, $x_0=0.7$). (c) Deciphered image using key ($k=4.0$, $x_0=0.7$).

Table 3
Efficiency comparison of three encryption schemes.

Image size	Gray level	Encryption speed of proposed scheme (ms)	Encryption speed of permutation–diffusion scheme (ms)	Encryption speed of DES algorithm (ms)
256×256	256	23	31	46
512×512	256	74	93	170
1024×1024	256	278	343	655

6. Conclusions

In this paper, a novel chaos-based bit-level permutation scheme has been proposed for secure and efficient image cipher. To overcome the drawbacks of conventional permutation-only type image cipher, the new scheme introduced a significant diffusion effect in permutation procedure through a two-stage bit-level shuffling algorithm. Compared with permutation–diffusion type image cipher, the new scheme has a comparable security level and a much lower computational complexity. Extensive security analysis has been carried out on the proposed image encryption technique using various statistical analyses, key space analyses, key sensitivity analyses, etc. Based on the results of our analysis, we conclude that the proposed image encryption technique is perfectly suitable for the real time secure image transmission over public networks.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (No. 60872040, 61071124) and the Fundamental Research Funds for the Central Universities (No. N100404016).

References

[1] J. Fridrich, International Journal of Bifurcation and Chaos 8 (6) (1998) 1259.
[2] J. Scharinger, Journal of Electronic Imaging 7 (2) (1998) 318.
[3] J.C. Yen, J.I. Guo, IEEE Proceedings-Vision Image and Signal Processing 147 (2) (2000) 167.
[4] G.R. Chen, Y.B. Mao, C.K. Chui, Chaos Solitons & Fractals 21 (3) (2004) 749.
[5] Y.B. Mao, G.R. Chen, S.G. Lian, International Journal of Bifurcation and Chaos 14 (10) (2004) 3613.
[6] Z.H. Guan, F.J. Huang, W.J. Guan, Physics Letters A 346 (1–3) (2005) 153.
[7] J.B. Shen, X.G. Jin, C. Zhou, Lecture Notes in Computer Science 3768 (2005) 270.
[8] F. Belkhouche, I. Gokcen, U. Qidwai, Journal of Electronic Imaging 14 (4) (2005) 043001.
[9] S.G. Lian, J.S. Sun, Z.Q. Wang, Chaos Solitons & Fractals 26 (1) (2005) 117.
[10] H.J. Gao, Y.S. Zhang, S.Y. Liang, et al., Chaos Solitons & Fractals 29 (2) (2006) 393.
[11] N.K. Pareek, V. Patidar, K.K. Sud, Image and Vision Computing 24 (9) (2006) 926.
[12] A.N. Pisarchik, N.J. Flores-Carmona, M. Carpio-Valadez, Chaos 16 (3) (2006) 033118.
[13] H.S. Kwok, W.K.S. Tang, Chaos Solitons & Fractals 32 (4) (2007) 1518.
[14] T. Xiang, K.W. Wong, X.F. Liao, Chaos 17 (2) (2007) 023115.
[15] S. Behnia, A. Akhshani, S. Ahadpour, et al., Physics Letters A 366 (4–5) (2007) 391.
[16] S. Behnia, A. Akhshani, H. Mahmodi, et al., Chaos Solitons & Fractals 35 (2) (2008) 408.
[17] F. Huang, Y. Feng, X.H. Yu, International Journal Of Innovative Computing, Information And Control 3 (6B) (2007) 1593.
[18] T.G. Gao, Z.Q. Chen, Physics Letters A 372 (4) (2008) 394.
[19] T.G. Gao, Z.Q. Chen, Chaos Solitons & Fractals 38 (1) (2008) 213.
[20] K.W. Wong, B.S.H. Kwok, W.S. Law, Physics Letters A 372 (15) (2008) 2645.
[21] F.Y. Sun, S.T. Liu, Z.Q. Li, et al., Chaos Solitons & Fractals 38 (3) (2008) 631.
[22] S.J. Xu, J.Z. Wang, S.X. Yang, Chinese Physics B 17 (11) (2008) 4027.
[23] X.J. Tong, M.G. Cui, Signal Processing 89 (4) (2009) 480.
[24] V. Patidar, N.K. Pareek, K.K. Sud, Communications in Nonlinear Science and Numerical Simulation 14 (7) (2009) 3056.
[25] C.K. Huang, H.H. Nien, Optics Communications 282 (11) (2009) 2123.
[26] R. Rhouma, S. Meherzi, S. Belghith, Chaos Solitons & Fractals 40 (1) (2009) 309.
[27] D. Xiao, X.F. Liao, P.C. Wei, Chaos Solitons & Fractals 40 (5) (2009) 2191.
[28] Y. Wang, K.W. Wong, X.F. Liao, et al., Chaos Solitons & Fractals 41 (4) (2009) 1773.
[29] K.W. Wong, B.S.H. Kwok, C.H. Yuen, Chaos Solitons & Fractals 41 (5) (2009) 2652.
[30] S. Mazloom, A.M. Eftekhari-Moghadam, Chaos Solitons & Fractals 42 (3) (2009) 1745.
[31] G.D. Ye, Imaging Science Journal 57 (5) (2009) 266.
[32] I.F. Elashry, O.S.F. Allah, A.M. Abbas, et al., Journal of Electronic Imaging 18 (3) (2009) 033002.
[33] S.E. Borujeni, M. Eshghi, Mathematical Problems in Engineering 762652 (2009).
[34] Y.L. Xiao, L.M. Xia, Communications in Theoretical Physics 52 (5) (2009) 876.
[35] E.A. Arnold, A. Avez, Ergodic Problems of Classical Mechanics, Chap. 1, Benjamin, W. A, New Jersey, 1968, p. 6.
[36] IEEE Computer Society, IEEE Standard for Binary Floating-Point Arithmetic, ANSI/IEEE std, August 1985, p. 754.
[37] Z.L. Zhu, W. Zhang, K.W. Wong, et al., Information Sciences 181 (6) (2011) 1171.