http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 13 (9): 1602-1611, 2014
ISSN 1812-5638 / DOI: 10.3923/itj.2014.1602.1611

An Efficient Medical Image Protection Scheme with Parallel Chaotic Key Stream Generation

Wei Zhou, Wen-Qi Liu, Dong-Liang Wang, Gui-Xiang Zhu, Ying-Jie Hu and Yong-Feng Zhan
General Hospital of Shenyang Military Command, Shenyang, 110016, China

Abstract: With the increasing utilization of telemedicine/tele health services, a vast number of medical images are now transmitted over the Internet and through wireless networks. Consequently, how to effectively protect the patients' privacy has become an important issue. Recently, chaos-based encryption approaches have been suggested and it is reported that they are more effective than traditional encryption algorithms when processing images. In this study, a novel permutation-diffusion architecture based medical image protection scheme is suggested. The new scheme introduces a parallel chaotic keystream generation mechanism so as to reduce the runtime cost on diffusion. Various intensive experiments are carried out to evaluate effectiveness and efficiency of the proposed cryptosystem. Computational results indicate that the cryptosystem can withstand all kinds of main attacks including brute-force attack, differential attack, known/chosen plain-text attack and various statistical attacks. We also show that the efficiency of the proposed cryptosystem is effectively improved.

Key words: Medical image cryptosystem, parallel chaotic keystream generation, chaotic logistic map, PACS

INTRODUCTION

With the advancement of multimedia and communication technologies, most medical images are now archived in digital format for easy storage, maintenance and retrieval. To manage those medical images and their accessories, a hospital usually employs a Picture Archiving and Communication Systems (PACS), which can provide convenient sharing of medical images and other data over the networks to promote high quality care for patient. Though PACS can manage medical image conveniently and makes telehealth applications feasible, most of existing systems do not combine with functions for data security over open networks, i.e., some of images have to be transferred outside a hospital for telehealth applications, such as the patient's home. Therefore, it is an important issue to protect patients' privacy when it is transferred over open networks (Cao *et al*., 2003; Li *et al*., 2005; Hu and Han, 2009; Lin *et al*., 2009; Fu *et al*., 2013). Usually, one can process ordinary data, such as text information, by traditional encryption algorithms, such as Tripe-DES, AES and IDEA. However, those conventional encryption algorithms are not suitable for practical medical image protection in PACS environment because of the size of image data and increasing demand for real-time telemedicine applications. In fact, some inherent features of medical images (e.g., high correlation among pixels and bulk data capacity) make traditional encryption algorithms hard to handle and thus medical image encryption is quite different from the ordinary data encryption. Recently, chaos-based encryption algorithms have been studied as a promising direction to settle the problem discussed above. Since 1990s, some researchers have concentrated on this direction using some inherent features of chaotic systems to design well-performance encryption algorithms. It is believed that, in chaotic dynamical systems, ergodicity, mixing property and sensitivity to initial conditions/system parameters are ideal cryptographic properties which can produce great efforts in confusion and diffusion operations of a cryptosystems. Based on this consideration, increasing number of studies on encryption algorithms based on chaotic systems has been carried out. In the followings, we provide a brief overview on some main recently proposed chaos-based image encryption schemes.

Fridrich (1998) suggested that there should be two stages in a secure image encryption algorithm with chaos-based approach: Chaotic confusion stage and pixel diffusion stage. Under this structure, the pixels of a plain image are firstly permutated, where the pixel positions of a plain image is rearranged by using an area-preserving chaotic map, whereas the diffusion stage alternates the grayscale values of all pixels one by one by and the modification made to a particular pixel usually depends on the accumulated effect of all the previous pixel values. Since that, this architecture has become the fundamental architecture for chaos-based image cipher, based on which researchers have suggested various image

Corresponding Author: Yong-Feng Zhan, General Hospital of Shenyang Military Command, Shenyang, 110016, China

cryptosystems. In the permutation stage, three two dimensional invertible chaotic maps, Arnold cat map, baker map and standard map are usually employed for generation of permutation table (Wong *et al*., 2008). In the diffusion stage, many discretized or continuous chaotic maps/systems such as logistic map, tent map, Lorenz system can be used as producing of pseudo-random keystream (Gao and Chen, 2008; Behnia *et al*., 2008). As an example Lian *et al.* (2005) presented a chaotic encryption method with two different chaotic maps, where a chaotic standard map is used in the permutation stage and a quantized logistic map is used in the diffusion stage and in each round the parameters of these chaotic maps have to be yielded by a keystream. As another example, Wang *et al.* (2009) suggested a chaos-based image encryption algorithm with variable control parameters. The control parameters used in the permutation stage and the keystream employed in the diffusion stage are generated from two chaotic maps related to the plain image. As a result, the algorithm can effectively resist all known attacks against permutation-diffusion architectures. Usually, chaotic maps used for permutation are based on 2D form in accordance with image presentation. To further enhance the effectiveness of permutation, 3D forms of the 2D chaotic cat map and baker map were developed (Chen *et al*., 2004; Mao *et al*., 2004; Tong and Cui, 2009). With these 3D maps several image encryption approaches were introduced. Besides, Xiang *et al.* (2007) presented a selective image encryption method based on chaotic keystream generated by a one-way coupled map lattice. To make permutation more effective, Belkhouche *et al.* (2005) proposed a novel permutation method that shuffles pixel positions by chaotic sequence sorting. This method well addresses the periodicity problems of discretized version 2D chaotic maps. To overcome the drawbacks of permutation-only type image cipher, Fu *et al.* (2011) introduced a significant substitution effect in permutation procedure using a two-stage bit-level shuffling technique. Compared with permutation-diffusion type image cipher, the new scheme has a comparable security level and a much lower computational complexity. As a successive work Fu *et al.* (2012) also presented a bidirectional diffusion strategy, which can accelerate the diffusion procedure. As a result, fewer overall encryption rounds are needed with the same level of security.

Considering PACS should have the ability to process thousands of images at the same time, the system requires more efficient encryption algorithms to achieve faster response for various online applications. To meet this requirement, we propose an efficient chaos-based medical image cryptosystem. In our cryptosystem, the key contribution is that a parallel keystream generation mechanism is proposed and embedded into the permutation-diffusion architecture, which significantly shortens the runtime of time-consuming diffusion procedure. When ciphering a certain pixel, one of these sequences is selected and the selection is plain image related, which ensures the cryptosystem secure against known/chosen plaintext attack. Moreover, we also introduce our permutation approach, where Arnold cat map is employed and the procedures of the complete medical image cryptosystem. Various experimental analyses, such as efficiency analysis, key space analysis, key sensitivity analysis and various statistical analyses, are intensively carried out. Through comparison of the experimental results, we conclude that our proposed cryptosystem is more efficient than other existing chaotic methods and is also secure enough to process medical images in PACS environment.

DIFFUSION STRATEGY WITH PARALLEL CHAOTIC KEYSTREAM GENERATION

In most cases, diffusion alternates pixels one by one with a chaotic sequence, where the image is processed as a one-dimensional sequence. This method can effectively confuse the relationship between cipher image and plain image. It is well-known that producing chaotic keystream is quite time-consuming procedure, even using the one-dimensional logistic map, which is one of the simplest chaotic maps/systems. This is because a large number of iterations of a chaotic map are required and the calculations are done over the field of real numbers. To shorten runtime of this step, as a key improvement, a parallel key stream generation scheme is proposed to achieve better efficiency. The detailed description of this algorithm is described as follows.

In the present study, the logistic map is employed to generate keystream. The logistic map is defined by:

$$x_{n+1} = \mu x_n (1-x_n),\ x(n) \in [0, 1],\ \mu \in [0, 4] \qquad (1)$$

where, μ is the control parameter and x_n is the state value. When $\mu \in [3.57, 4]$, the system is chaotic.

The diffusion approach has several threads: t (a predefined number of threads) threads to produce keystream and a main thread to encrypt images. The thread for keystream producing has the following steps:

- **Step 1:** Create t threads; each thread is assigned with a diffusion key K_{dt} produced by a key generator as initial value x_0 for the logistic map. The keystream produced by a certain thread is identified as Seq_i (i is the thread ID). The key scheming will be further discussed in Section 4

- **Step 2:** For each thread, iterate logistic map N_0 times to avoid the harmful effect of transitional procedure (N_0 = 200 in this study)
- **Step 3:** Iterate logistic map to yield an element for keystream Seq_i, the element is denoted as K:

$$K = \text{mod}\,[\text{floor}\,(x_n \times 10^{14}), L] \quad (2)$$

where, floor(x) is a function that returns the nearest integers less than or equal to x, mod(x, y) calculates the remainder after division and L is set to 256, as for the medical images in this study there are 256 possible gray levels

- **Step 4:** Buffer K to the end of Seq_i
- **Step 5:** Return to Step 3 until a stop message is received

It is noted that there are some 'weak' states when iterating the logistic map. For instance, it can not be used when μ is 4.0 and x_n is set to 0.5, as the dynamical system will trap into fixed point 0 and iterations will stop. Therefore, a slight perturbation mechanism should be employed to avoid this situation. Equation 3 defines the mechanism that makes the dynamic system jump to a new position from the current value of x_n:

$$x'_n = \begin{cases} x_n - 0.1 & x_n > 0.5, \\ x_n + 0.1 & x_n \le 0.5 \end{cases} \quad (3)$$

The main thread performs the following steps to complete diffusion:

- **Step 1:** Rearrange the plain image to a one-dimensional sequence in the order from left to right, top to bottom. Initialize n to 1
- **Step 2:** Let h = mod (p_{n-1}, t)+1, where p_{n-1} is the previously operated plain-pixel and p_0 is a pre-defined number ranging from [0, L] and it can be used as a part of secret key
- **Step 3:** Select the h-th keystream Seq_h and fetch the first element K_f if the queue is not empty. Otherwise, wait for the thread to produce an element
- **Step 4:** Calculate cipher-pixel value according to Eq. 4:

$$c_n = K_f \oplus \{[p_n + K_f] \bmod L\} \oplus c_{n-1} \quad (4)$$

where p_n, c_n and c_{n-1} are the currently operated pixel, output cipher-pixel and previous ciphered pixel and $\oplus$ represents a bit-wise exclusive OR operation

- **Step 5:** Remove K_f from Seq_h
- **Step 6:** n = n+1. Return to Step 2 until all the pixels are diffused
- **Step 7:** Send a stop message to all the keystream generation threads. Arrange the diffused pixels back to its original form and the cipher image is produced

The inverse diffusion operation for decryption is given by:

$$p_n = [k_n \oplus c_n \oplus c_{n-1} + L - k_n] \bmod L \quad (5)$$

As can be seen from above encryption procedures, the diffusion effort is enhanced by choosing different keystream according to previous plain pixels, which ensures the security against known/chosen plaintext attack.

IMAGE PERMUTATION BASED ON ARNOLD CAT MAP

In this section, we briefly describe our permutation method embedded in the cryptosystem. As mentioned above, three discretized 2D invertible chaotic maps, Arnold cat map, baker map and standard map, are frequently employed for permutation. These maps make a bijective mapping that changes each pixel in the image to some other place. This step is usually performed several times to achieve a satisfactory confusion effect. Though Arnold cat map and baker map have the drawback of short period after discretization compared with standard map, the two maps can complete permutation in a short time without complex computation on float numbers. Therefore, the cat map is employed for image permutation in the present study due to its lowest computational complexity.

The so called Arnold cat map is a two-dimensional invertible chaotic map introduced by Arnold and Avez (1968). The mathematical formula is presented as follows:

$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} \bmod 1 = A \begin{bmatrix} x_n \\ y_n \end{bmatrix} \bmod 1 \quad (6)$$

where, x mod 1 means the fractional part of x for any real number x. The map is area-preserving as det|A| = 1. The cat map can be easily described in geometric terms. As illustrated by Fig. 1, a unit square is firstly stretched by a linear transform matrix A and then folded back to the unit square by the module operation.

The cat map can be generalized by introducing two control parameters, p and q, as follows:

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

However, as x and y are real numbers, the generalized map is not suitable for image permutation that operates on a lattice of finitely many pixels. Therefore, the generalized map needs to be discretized so as to incorporate it into cryptosystem for performing pixel permutation. The discretized cat map is constructed by changing the range of (x, y) from the unit square I×I to the discrete lattice N×N, as described by:

$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} 1 & p \\ q & pq+1 \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} \bmod N = A \begin{bmatrix} x_n \\ y_n \end{bmatrix} \bmod N \quad (8)$$

where, N is the width or height of a square image. Obviously, the permutation key is composed of parameters p, q and the number of iterations (permutations) m.

KEY SCHEMING AND THE COMPLETE CRYPTOSYSTEM

The framework of the proposed medical image cryptosystem which is based on the permutation-diffusion architecture, is shown in Fig. 2.

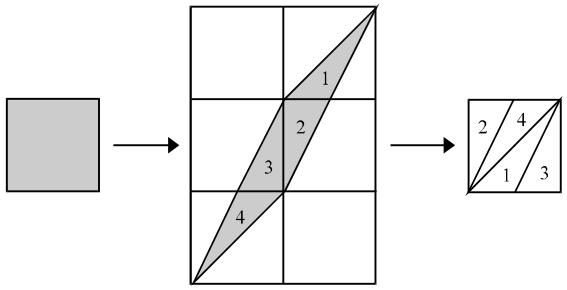

Fig. 1: Cat map

To make pixels sufficiently disordered, there should be 2 or more rounds for permutation, so we take m = 3 to achieve better performance. The whole permutation and diffusion operations repeat 2 rounds in the experiments carried out in the next section.

To satisfy the key sensitivity requirement, a key generator for producing diffusion keys for parallelly iterated logistic maps is introduced. The key generator is base on the well-known Lorenz system, which is a system of three ordinary differential equations now known as the Lorenz equations, as described by:

$$\begin{cases} \dfrac{dx}{dt} = \sigma(y-x), \\ \dfrac{dy}{dt} = x(\rho-z)-y, \\ \dfrac{dz}{dt} = xy-\beta z, \end{cases} \quad (9)$$

where, t is time and σ, ρ, β are the system parameters. When σ = 10, ρ = 8/3, β = 28, the system exhibits chaotic behavior. The initial state values x_0, y_0 and z_0 that control the dynamical behavior of Lorenz system serve as the secret key.

To generate proper diffusion keys, following procedures are carried out. Firstly, the Lorenz system is iterated for N_0 times with the same purpose discussed above. To solve the Eq. 4, fourth-order Runge-Kutta method is employed, as given by:

$$\begin{cases} x_{n+1} = x_n + (h/6)(K_1 + 2K_2 + 2K_3 + K_4), \\ y_{n+1} = y_n + (h/6)(L_1 + 2L_2 + 2L_3 + L_4), \\ z_{n+1} = z_n + (h/6)(M_1 + 2M_2 + 2M_3 + M_4) \end{cases} \quad (10)$$

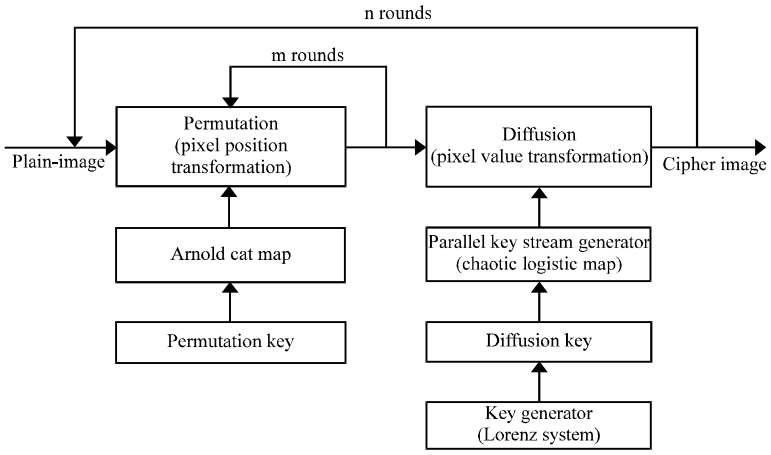

Fig. 2: Architecture of the proposed chaos-based medical image cryptosystem

Algorithm 1: Encryption procedures

Input: Plain image
Output: Cipher image

1. generate diffusion keys K_{d1}, K_{d2}, …, K_{dt} for parallely iterated logistic maps.
2. create one main thread deals with the permutation and diffusion operation.
3. create other t threads to produce diffusion keystreams Seq_1, Seq_2, …, Seq_t by using chaotic logistic map.
4. for 1 to n
5. for 1 to m
6. shuffle the plain image by using Arnold cat map (permutation keys are p, q and m).
7. end for
8. diffuse the shuffled image by masking plain-pixel with parallel chaotic sequences Seq_i ($1 \le i \le t$).
9. end for

Where:

$$\begin{cases} K_j = \sigma(y_n - x_n), \\ L_j = x_n(\rho - z) - y_n, \\ M_j = x_n y_n - \beta z_n, \\ (j = 1), \end{cases}$$

$$\begin{cases} K_j = \sigma[(y_n + hL_{j-1}/2) - (x_n + hK_{j-1}/2)], \\ L_j = (x_n + hK_{j-1}/2)(\rho - (z_n + hM_{j-1}/2)) - (y_n + hL_{j-1}/2), \\ M_j = (x_n + hK_{j-1}/2)(y_n + hL_{j-1}/2) - \beta(z_n + hM_{j-1}/2), \\ (j = 2, 3), \end{cases}$$

$$\begin{cases} K_j = \sigma[(y_n + hL_{j-1}) - (x_n + hK_{j-1})], \\ L_j = (x_n + hK_{j-1})(\rho - (z_n + hM_{j-1})) - (y_n + hL_{j-1}), \\ M_j = (x_n + hK_{j-1})(y_n + hL_{j-1}) - \beta(z_n + hM_{j-1}), \\ (j = 4) \end{cases}$$

and the step h is chosen as 0.0005. Then, the Lorenz system is iterated continuously and the iteration times depend on the number of threads. For example, if threads number t = 3, the Lorenz system will be iterated once and the three output states values x_n, y_n and z_n are used for diffusion keys generation. Finally, the diffusion keys D_{key} is generated by converting the output state values to the range of [0, 1] according to:

$$D_{key} = \phi_n\text{-floor}(\phi_n),\ \phi \in (x, y, z) \quad (11)$$

Algorithm 1 indicates the complete encryption procedure. One main thread deals with the permutation and diffusion operation. At the same time, diffusion keystreams are generated by other t parallel threads. In the diffusion stage, diffusion algorithm will fetch mask data from those sequences. The whole procedure will be repeated n times to achieve a satisfactory level of security. Corresponding decryption algorithm is the reversed procedures of encryption algorithm.

EFFICIENCY ANALYSIS

To demonstrate the efficiency of our proposed cryptosystem, the encryption time of medical images with different sizes is evaluated and compared with that of the classic DES and conventional chaos-based algorithms. The experiments are performed on a personal computer with Intel i7 2600 processor and 2GB RAM and the OS is Windows 7. We implement our cryptosystem with C++ and compile it under VC++ 2010 environment. To perform comparisons of running time, we only record the permutation and diffusion time (including the keystream generation), whereas time cost on reading/writing image files is not involved in the total time.

Table 1: Encryption speed of the proposed image cryptosystem and other encryption algorithms

Image size	Gray level	DES algorithm (msec)	Scheme of Fu C *et al.* (2012) (msec)	Proposed scheme (t = 3) (msec)	Proposed scheme (t = 7) (msec)
256×256	256	46	11	7	4
512×512	256	170	35	21	11
1024×1024	256	655	150	89	51

As can be seen from Table 1, the efficiency of the proposed cryptosystem is quite better than the DES algorithm and about 40% speed increment is achieved compared with conventional chaos-based algorithms with permutation-diffusion architecture if the number of keystream generation threads is up to 3. The result indicates that PACS embedded with the proposed image protection scheme can deal with real-time telemedicine applications with high efficiency, especially for the case of real-time secure medical image transmission over public networks.

SECURITY ANALYSIS

The premier evaluation of a medical image cryptosystem is ability of resisting attempts of unauthorized accesses which are probable to recover plain images. A good cryptosystems have to withstand all kinds of attacks known so far, such as brute-force attack, known/chosen plain-text attack, differential attack and various statistical attacks, to ensure security in practical applications. To demonstrate the robustness of our proposed medical image cryptosystem, thorough security analyses have been carried out in this section. In experiments, we randomly choose a group of arguments as keys for our cryptosystem.

Key space analysis: The key space is the total number of different keys that can be used in the encryption/decryption procedure. A good cryptosystem should have sufficiently large key space (usually more than 2^{100}) to make brute-force attack infeasible. The key space of our cryptosystem can be counted as follows. The system has two kinds of keys: K_c (keys for confusion stage) and K_d (keys for diffusion stage). As mentioned above, K_c consists of two control parameters (p, q) and

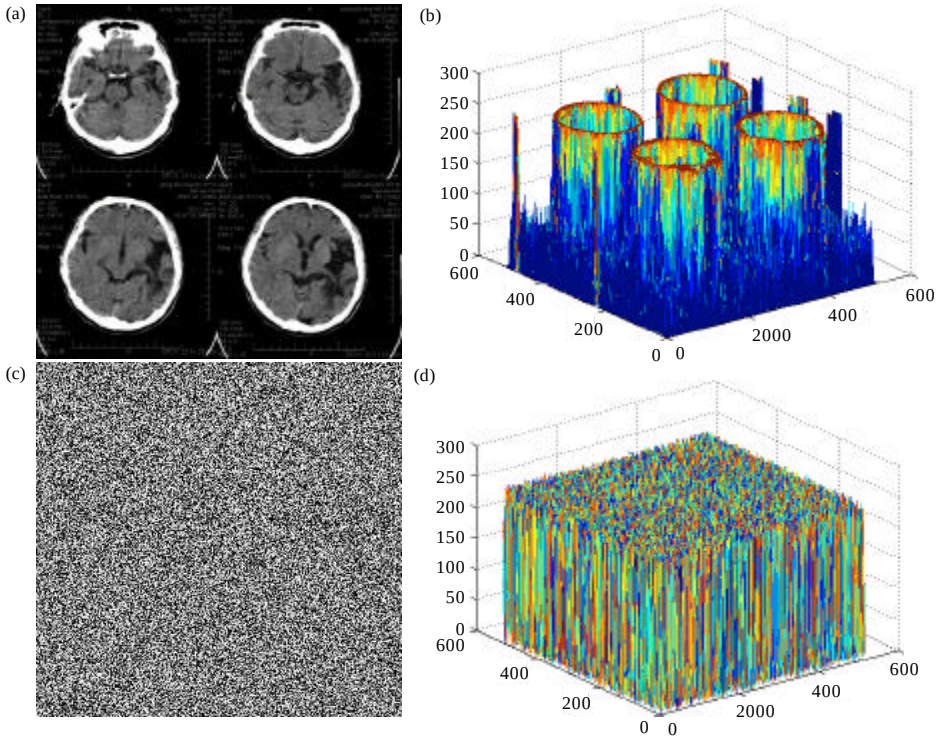

Fig. 3(a-d): Histograms, (a) Plain medical image 512×512 pixels with 256 gray levels, (b) Histogram of plain image, (c) Cipher image and (d) Histogram of cipher image

iteration times m of cat map, where, p, q∈[1, N] and N is the length or width of the plain image. Therefore, the size of K_c is $(N^2)^m$. K_d consists of three initial state values $(x_0, y_0, z_0)\in R$ of Lorenz system. According to the IEEE floating-point standard, the computational precision of those numbers is about 10^{-15}, so the total number of possible values of K_d is approximately 10^{45}. The two parts K_c and K_d are independent of each other and it is proposed to take m=3. If N = 512, the total size satisfies:

$$Key_{total} = K_c \times K_d \geq (512^2)^3 \times 10^{45} = 2^{203} \quad (12)$$

which is large enough to resist brute-force attack.

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 proposed cryptosystem, we have performed statistical analysis by calculating the histogram, the information entropy and the correlation of two adjacent pixels in the ciphered image.

Histogram is a visual representation of how pixels in an image are distributed. A well ciphered image should hide its redundancy and correlation between the cipher image and its corresponding plain image, so uniform distribution is preferred in histogram and hence one cannot observe any useful information from the cipher image. Figure 3b and d show the 3D histograms of a plain medical image (Fig. 3a) and its ciphered image (Fig. 3c) produced by the proposed scheme, respectively. It's clear from Fig. 3 that the histogram of the cipher image is quite uniform and therefore does not provide any clue to employ statistical analysis.

Next, we analyze the information entropy of the produced cipher image. Entropy, a significant feature of disorder in information theory, is extensively used in measure the unpredictability of a cryptosystem. To calculate the entropy H (s) of a source s, we have:

$$H(s) = -\sum_{i=0}^{2^N-1} P(s_i)\log_2 P(s_i) \quad (13)$$

where, N is the number of bits representing a symbol $s_i\in s$ and P (s_i) is the probability of symbol s_i. Therefore, for a random image with 256 gray levels, the entropy should ideally be H (s) = 8.

To compare the information entropy of the plain image and cipher image, the number of occurrence of each grey level and the probability of occurrence are recorded.

With those recorded data, we calculate entropy of the plain image is H (s) = 5.7887, whereas the entropy of the corresponding cipher image is H (s) = 7.9998. The entropy of the output ciphered image is very close to the theoretical value of 8, which indicates that the disorderliness of the ciphered image is satisfactory and there is no information leakage.

Finally, the correlations between adjacent pixels of the ciphered image are analyzed. For an ordinary image with meaningful content, most of pixels are highly correlated with their adjacent pixels. Take Fig. 3a as an example, most pairs of adjacent pixels in horizontal, vertical and diagonal directions are with same grayscale value. However, with the fact that low correlation of the adjacent pixels makes statistical analysis impossible, the correlations of a well-ciphered image should ideally be 0.

To compute and compare the correlations of adjacent pixels in the plain and cipher image, the following procedure is carried out. Firstly, 3,000 pairs of adjacent pixels are randomly chosen from horizontal, vertical and diagonal directions. Then, the correlation variances are calculated by using Eq. 14-17:

$$r_{xy} = \frac{\text{cov}(x, y)}{\sqrt{D(x)}\sqrt{D(y)}} \tag{14}$$

$$\text{cov}(x, y) = E\,(x-E(x))(y-E(y)) \tag{15}$$

$$E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{16}$$

$$D(x) = \frac{1}{N}\sum_{i=1}^{N} (x_i - E(x))^2 \tag{17}$$

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

The results of correlation coefficients for horizontal, vertical and diagonal adjacent pixels in the plain image and its ciphered image are given in Table 2. It can be seen from Table 2 that the correlations between adjacent pixels of the plain image is very strong (>0.9), whereas the values of the cipher image are significantly decreased to the ideal value 0.

Moreover, the visual testing of the correlation of adjacent pixels is also carried out by plotting the distribution of the adjacent pixels in the plain image and its corresponding cipher image. Figure 4a and b illustrate the correlation distribution of two horizontally adjacent pixels of the plain image and its corresponding ciphered image, respectively. Similar results can be obtained for vertical and diagonal directions. It's clear from Table 2 and Fig. 4 that the strong correlations between adjacent pixels in plain image are effectively eliminated by using the proposed cryptosystem.

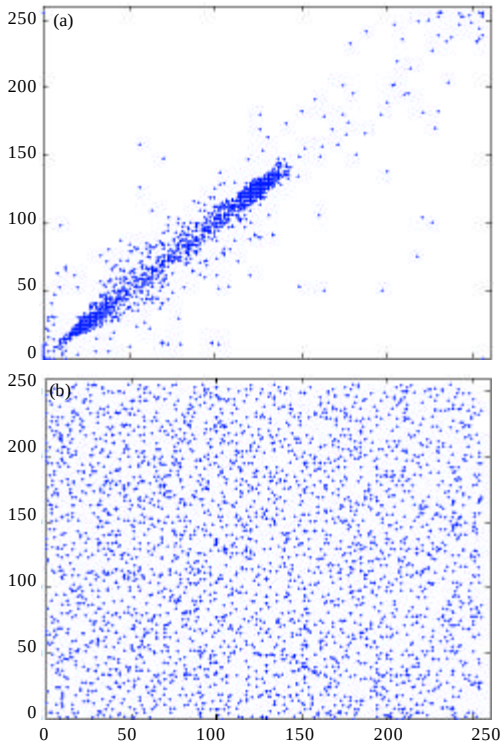

Fig. 4(a-b): Correlations of two horizontally adjacent pixels of the plain image and its ciphered image, (a) Plain image and (b) Cipher image

Table 2: Correlation coefficients of two adjacent pixels in two images

Direction	Plain image	Cipher image
Horizontal	0.9528	-0.0159
Vertical	0.9301	-0.0054
Diagonal	0.9287	0.0113

Differential analysis: In order to implement known plaintext attack, chosen plaintext attack and more advanced adaptive chosen plaintext attack, an opponent may make a slight change (maybe one bit) in a plain image and then compare the two produced cipher images to find out which parts are different. With the help of other analysis methods the secret key may be obtained. However, this differential analysis will not be available if a small change in the plain image can be diffused to the entire cipher image. To test the diffusion effect when changing one-bit in a plain image, two common measures NPCR and UACI are employed.

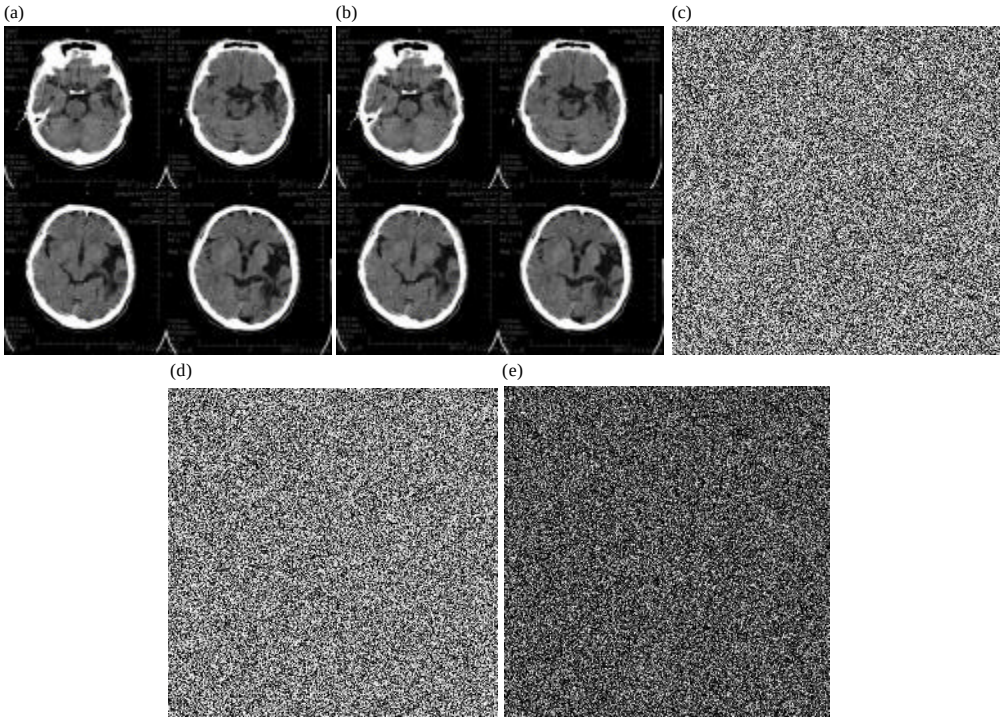

Fig. 5(a-d): NPCR and UACI tests, (a, b) Two plain medical images with only one bit difference at the lower right corner, (c) Cipher image of (a), (d) Cipher image of (b), (e) Differential image between (c) and (d)

Table 3: Decryption keys used for key sensitivity test

Figure	Decryption key
6 (a)	(x_0 = 6.3, y_0 = -3.5, z_0 = 9.9)
6 (b)	(x_0 = 6.30000000000001, y_0 = -3.6, z_0 = 9.9)
6 (c)	(x_0 = 4.3, y_0 = -2.69999999999999, z_0 = 6.8)
6 (d)	(x_0 = 4.3, y_0 = -2.7, z_0 = 6.80000000000001)

The NPCR is used to measure the percentage of different pixels between two images. Let P_1 (i, j) and P_2 (i, j) be the pixels on the position (i, j) of two images P_1 and P_2, respectively, NPCR is defined as:

$$NPCR = \frac{\sum_{i=1}^{W}\sum_{j=1}^{H} D(i,j)}{W \times H} \times 100\% \quad (18)$$

where, W and H are the width and height of P_1 or P_2, respectively. D (i, j) is set to 0 if P_1 (i, j) = P_2 (i, j) and 1 otherwise.

The second measurement, UACI is used to measure the average intensity of differences between the two images. It is defined as:

$$UACI = \frac{1}{W \times H}\left[\sum_{i=1}^{W}\sum_{j=1}^{H} \frac{|P_1(i,j) - P_2(i,j)|}{L-1}\right] \times 100\% \quad (19)$$

The NPCR and UACI values for two truly random images with 256 grey levels, namely the ideal values for the two measurements, are 99.609 and 33.464%, respectively.

To evaluate the NPCR and UACI of the proposed medical image cryptosystem, the following procedures are carried out. Firstly, we select the pixel locatedat the lower-right corner in the plain image and change one bit on it, as shown in Fig. 5a and b. Then, the two images before and after the change are encrypted with the same key and two cipher images are produced, as shown in Fig. 5c, d. Figure 5e shows the differential image between the twocipher images. Finally, we calculate the NPCR and UACIby means mentioned above and obtain NPCR = 99.59% and UACI = 33.45%. The results indicate that a slight change in the original image willresult in a significant change in the ciphered image, so the proposed scheme is robust against differential attack.

Key sensitivity analysis: Another essential property requiredby a good cryptosystem is key sensitivity, which ensures that no data can be recovered from ciphertext even though there is only a slight difference between the

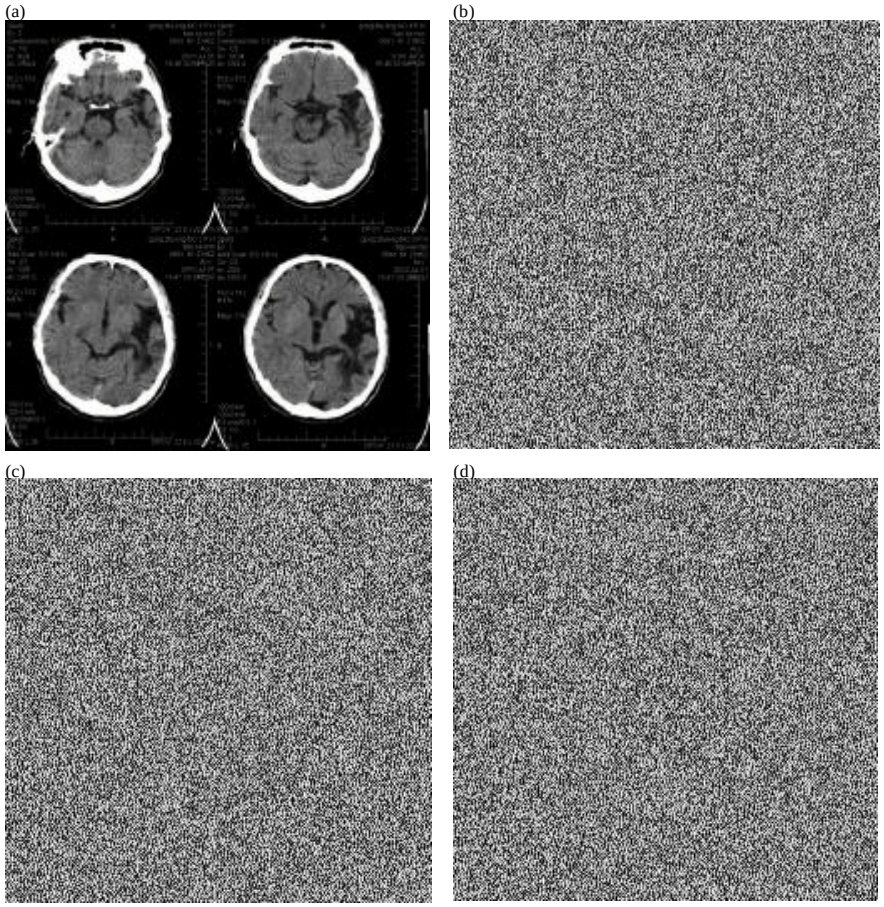

Fig. 6 (a-d): Deciphered image using slightyly different keys, (a) Deciphered image with correct keys and (b, c, d) Image decrypted with 3 groups of wrong keys from Table 3, respectively

encryption and decryption keys. To evaluate the key sensitivity of the proposed cryptosystem,the test image (Fig. 3a) is firstly encrypted using the test key (x_0 = 6.3, y_0 = -3.5, z_0 = 9.9). Then, the ciphered image is tried to be decrypted using four decryption keys, as listed in Table 3. The resultant deciphered images are shown in Fig. 6a-d, respectively, from which we can see that even an almost perfect guess of the key does not reveal any information about the plain image. Similar results are obtained with a slight change in permutation key. Therefore, it can be concluded that the proposed scheme well satisfies the key sensitivity requirement.

CONCLUSION

In this study, we have presented a novel medical image encryption scheme with a parallel diffusion approach. The scheme is based on permutation-substitution architecture using chaotic cat map and logistic map. To improve the efficiency of the time-consuming diffusion procedure, the proposed scheme produces the chaotic keystream in a parallel manner. Moreover, when ciphering a certain pixel, one of these sequences is selected and the selection is plain image related, which ensures the cryptosystem secure against known/chosen plaintext attack. Computer simulations are performed to validate the security of our cryptosystem. Security analyses including key space analysis and various statistical analyses have been discussed, which demonstrated that the proposed algorithm can achieve high security. Furthermore, the encryption efficiency is improved greatly as parallel approaches are involved in the algorithm. Therefore, we conclude that our cryptosystem provides a good candidate for secure medical image transmission over open networks.

REFERENCES

Arnold, V.I. and A. Avez, 1968. Ergodic Problems of Classical Mechanics. 1st Edn., W.A. Benjamin Publishing, New York, USA., Pages: 286.

Behnia, S., A. Akhshani, H. Mahmodi and A. Akhavan, 2008. A novel algorithm for image encryption based on mixture of chaotic maps. Chaos Solitons Fractals, 35: 408-419.

Belkhouche, F., I. Gokcen and U. Qidwai, 2005. Chaotic gray-level image transformation. J. Electron. Imaging, Vol. 14. 10.1117/1.2135792

Cao, F., H.K. Huang and X.Q. Zhou, 2003. Medical image security in a HIPAA mandated PACS environment. Comput. Med. Imaging Graph., 27: 185-196.

Chen, G., Y. Mao and C.K. Chui, 2004. A symmetric image encryption scheme based on 3D chaotic cat maps. Chaos Solitons Fractals, 21: 749-761.

Fridrich, J., 1998. Symmetric ciphers based on two-dimensional chaotic maps. Int. J. Bifurcat. Chaos, 8: 1259-1284.

Fu, C., B.B. Lin, Y.S. Miao, X. Liu and J.J. Chen, 2011. A novel chaos-based bit-level permutation scheme for digital image encryption. Optics Commun., 284: 5415-5423.

Fu, C., J.J. Chen, H. Zou, W.H. Meng, Y.F. Zhan and Y.W. Yu, 2012. A chaos-based digital image encryption scheme with an improved diffusion strategy. Optics Express, 20: 2363-2378.

Fu, C., W.H. Meng, Y.F. Zhan, Z.L. Zhu, F.C.M. Lau, C.K. Tse and H.F. Ma, 2013. An efficient and secure medical image protection scheme based on chaotic maps. Comput. Biol. Med., 43: 1000-1010.

Gao, T.G. and Z. Chen, 2008. A new image encryption algorithm based on hyper-chaos. Phys. Lett. A, 372: 394-400.

Hu, J. and F. Han, 2009. A pixel-based scrambling scheme for digital medical images protection. J. Network Comput. Appl., 32: 788-794.

Li, M., R. Poovendran and S. Narayanan, 2005. Protecting patient privacy against unauthorized release of medical images in a group communication environment. Comput. Med. Imaging Graph., 29: 367-383.

Lian, S., J. Sun and Z. Wang, 2005. A block cipher based on a suitable use of the chaotic standard map. Chaos Solitons Fractals, 26: 117-129.

Lin, C.F., C.H. Chung and J.H. Lin, 2009. A chaos-based visual encryption mechanism for clinical EEG signals. Med. Biol. Eng. Comput., 47: 757-762.

Mao, Y., G. Chen and S. Lian, 2004. A novel fast image encryption scheme based on 3D chaotic baker maps. Int. J. Bifurcat. Chaos, 14: 3613-3624.

Tong, X. and M. Cui, 2009. Image encryption scheme based on 3D baker with dynamical compound chaotic sequence cipher generator. Signal Process., 89: 480-491.

Wang, Y., K.W. Wong, X. Liao, T. Xiang and G. Chen, 2009. A chaos-based image encryption algorithm with variable control parameters. Chaos Solitons Fractals, 41: 1773-1783.

Wong, K.W., B.S.H. Kwok and W.S. Law, 2008. A fast image encryption scheme based on chaotic standard map. Phys. Lett. A, 372: 2645-2652.

Xiang, T., K.W. Wong and X. Liao, 2007. Selective image encryption using a spatiotemporal chaotic system. Chaos, Vol. 17. 10.1063/1.2728112.