A Self-adaptive Bit-level Color Image Encryption Algorithm Based on Generalized Arnold Map

Rui-Song YE[1,a], Hui-Qing HUANG[2], Yu-Cheng LI[1], Chang WANG[1] and Min-Yu LIAO[1]

[1]*Department of Mathematics, Shantou University, Shantou, Guangdong, 515063, China*
[2]*School of Mathematics, Jiaying University, Meizhou, Guangdong, 514015, China*

Abstract. A self-adaptive bit-level color image encryption algorithm based on generalized Arnold map is proposed. The red, green, blue components of the plain-image with height H and width W are decomposed into 8-bit planes and one three-dimensional bit matrix with size $H \times W \times 24$ is obtained. The generalized Arnold map is used to generate pseudo-random sequences to scramble the resulted three-dimensional bit matrix by sort-based approach. The scrambled 3D bit matrix is then rearranged to be one scrambled color image. Chaotic sequences produced by another generalized Arnold map are used to diffuse the resulted red, green, blue components in a cross way to get more encryption effects. Self-adaptive strategy is adopted in both the scrambling stage and diffusion stage, meaning that the key streams are all related to the content of the plain-image and therefore the encryption algorithm show strong robustness against known/chosen plaintext attacks. Some other performances are carried out, including key space, key sensitivity, histogram, correlation coefficients between adjacent pixels, information entropy and difference attack analysis, etc. All the experimental results show that the proposed image encryption algorithm is secure and effective for practical application.

1 Introduction

With the rapid development of computer network and multimedia processing technology, large amounts of information like audio, image and video transmitted by the network. More than 70% of the information transmitting in the network is related to image information. The security transmission and storage of image information attract more and more attention. Image encryption is usually adopted to handle the security issue. Image with large data volume, strong correlation and high redundancy, makes the traditional encryption algorithms like the International Data Encryption Algorithm (IDEA) and Advanced Encryption Standard (AES) reduce encryption efficiency, which cannot meet the requirements of real-time applications. Therefore, the traditional encryption algorithms cannot be suitable for encrypting digital images [1]. Thanks to the inherent natures of chaotic sequences generated by chaotic systems, such as pseudo-randomness, strong sensitivity to initial conditions and control parameters, unpredictability, etc., chaos-based image encryption technology has a broad application prospect. In the last two decades, many chaotic image encryption algorithms had been presented and investigated extensively [2-7].

[a] Corresponding author: rsye@stu.edu.cn

Color image contains the red (R) , green (G), blue (B) color components, among which provides more abundant information. Many of existed color image encryption algorithms used the same method to encrypt R, G, B components, which means that the image is encrypted three times independently, ignoring the correlation between R, G, B components, and those components are easily attacked. In this paper, we present a novel self-adaptive color image encryption algorithm with a sort-based scrambling strategy for the 3D bit matrix and color components' diffusion in a cross way. The R, G, B color components of the plain-image with size of $H \times W$ and 256 grayscale levels are decomposed into 8-bit planes and one three-dimensional (3D) bit matrix with size $H \times W \times 24$ is yielded. One generalized Arnold map is used to generate pseudo-random sequences to scramble the resulted 3D bit matrix by sort-based approach. The scrambled 3D bit matrix is then rearranged to be one scrambled color image. Such a kind of operation changes the pixels' positions as well as pixels color components' intensity values. It reduces the correlation between R, G, B components and therefore improves the security and performance significantly. Another generalized Arnold map is applied to diffuse the resulted R, G, B components in a cross way to get more encryption effects. To achieve desirable key sensitivity and plaintext sensitivity, both the scrambling stage and the diffusion stage adopt self-adaptive strategy, where the key streams are all related to the content of the plain-image and therefore the encryption algorithm show strong robustness against known/chosen plaintext attacks. Some other performances are carried out, including key space, key sensitivity, histogram, correlation coefficients between adjacent pixels, information entropy and difference attack analysis. All the experimental results show that the proposed image encryption algorithm is highly secure and demonstrates excellent performance.

2 The Proposed image encryption algorithm

A color plain-image P can be expressed by a 3D matrix of size $H \times W \times 3$ meaning that the image is of height H and width W . We assume that the three base color components are denoted by 2D matrices R, G, B respectively. The three base color components R, G, B are all composed of 8 bit planes. We rearrange the color plain-image in bit planes mode to get one 3D bit-plane matrix *P3D* sized $H \times W \times 24$. One generalized Arnold map is appied to generate chaotic sequences to scramble *P3D* at bit-level and another generalized Arnold map is used to diffuse the resulted R, G, B components in a cross way to achieve more encryption effect. The complete proposed image encryption algorithm is composed of one round of 3D bit matrix permutation and one diffusion process of the R, G, B components. It is outlined as follows.

Step 1. Generation of pseudo-random sequences x, y for 3D bit matrix *P3D*'s permutation process. With cipher keys x_0, y_0, a, b , we iterate the generalized Arnold map (1) for N times and reject the transient points $\{(x_k, y_k) : k=0, 1, \cdots, N-1\}$ to avoid the harmful effect. The values of (x_0, y_0) are reset to be (x_N, y_N) and the map (1) with new initial values (x_0, y_0) is applied to yield $\{(x_k, y_k) : k = 1, \cdots, L+12\}$ where $L = \max\{H, W\}$. We note that N is set to be related to the content of plain-image to enhance the encryption algorithm's robustness against known plaintext and chosen plaintext attacks. N is calculated by mod(*sum1*, 100)+30 where *sum1* is the number of one in the 3D bit-plane matrix *P3D* and function mod(e, f) resturns the remainder of integer e divided by integer f.

$$\begin{pmatrix} x_{n+1} \\ y_{n+1} \end{pmatrix} = \begin{pmatrix} 1 & a \\ b & 1+ab \end{pmatrix} \begin{pmatrix} x_n \\ y_n \end{pmatrix} \bmod 1. \tag{1}$$

Step 2. Sort three sequences

$\{x_k, k = 1, \cdots, H\}$, $\{y_k, k = 1, \cdots, W\}$, $\{v_k, k = 1, \cdots, 24\} = \{x_{H+1}, \cdots, x_{H+12}, y_{W+1}, \cdots, y_{W+12}\}$

in ascendant order to get the new sequences $\{x_k^{'}, k=1,\cdots,H\}$, $\{y_k^{'}, k=1,\cdots,W\}$, $\{v_1^{'},\cdots,v_{24}^{'}\}$ and the position indices $\{t_k, k=1,\cdots,H\}$, $\{s_k, k=1,\cdots,W\}$, $\{u_k, k=1,\cdots,24\}$ such that $x_k^{'}=x_{t_k}$, $y_k^{'}=y_{s_k}$, $v_k^{'}=v_{u_k}$ respectively. We get three P-boxes T,S,U: $T=\{t_1,\cdots,t_H\}, S=\{s_1,\cdots,s_W\}, U=\{u_1,\cdots,u_{24}\}$.

Step 3. Permute the 3D bit-plane matrix *P3D* by P-boxes T,S,U respectively to get one new 3D bit matrix *P3D1*: $P3D1(i,j,k)=P3D(t_i,s_j,u_k), i=1,...,H; j=1,...,W, k=1,...,24$. The resulted 3D bit matrix *P3D1* is converted to be one color image *PI* with height H and width W such that each color component consisting of 256 grayscale levels. The corresponding color components for *PI* are still denoted by R,G,B.

Step 4. Generation of pseudo-random sequences w,z for the diffusion process. With cipher keys w_0, z_0, a_1, b_1, N_1, we iterate the generalized Arnold map (1) for N_1 times and reject the transient points $\{(w_k,z_k): k=0,\ 1,\cdots,N_1-1\}$ to avoid the harmful effect. The values of (w_0,z_0) are reset to be (w_{N_1},z_{N_1}) and the chaotic map (1) with new initial values (w_0,z_0) is applied to yield $\{(w_k,z_k): k=1,\cdots,HW\}$. The resulted $\{(w_k,z_k): k=1,\cdots,HW\}$ is used to generate the pseudo-random gray value sequences by

$$x(n)=\mathrm{mod}(\mathrm{floor}(x_n\times 10^{14}),256), y(n)=\mathrm{mod}(\mathrm{floor}(y_n\times 10^{14}),256), n=1,\cdots,HW,$$

where function floor(x) returns the largest integer less than or equal to x. The vectors $x(n), y(n), n=1,2,\cdots,HW$ are then converted into two 2D matrices X,Y with size $H\times W$.

Step 5. We Calculate the sum of all the pixels' gray values of R,G and determine the value of $k2,k3$ by

$$k2=\mathrm{mod}(\sum_{i=1}^{H}\sum_{j=1}^{W}R(i,j),2),\ k3=\mathrm{mod}(\sum_{i=1}^{H}\sum_{j=1}^{W}G(i,j),2).$$

Then we substitute the pixels values of G,B components by

$$G1=\begin{cases}\mathrm{mod}(G+R+X,256), \text{if } k2=0,\\ \mathrm{mod}(G+R+Y,256), \text{if } k2=1,\end{cases}\quad B1=\begin{cases}\mathrm{mod}(B+G1+X,256), \text{if } k3=0,\\ \mathrm{mod}(B+G1+Y,256), \text{if } k3=1.\end{cases}$$

The sum of all the pixels' gray values of $B1$ is then used to determine the value of $k4$ and change the values of R by

$$k4=\mathrm{mod}(\sum_{i=1}^{H}\sum_{j=1}^{W}B1(i,j),2)\ ,\quad R1=\begin{cases}\mathrm{mod}(R+B1+X,256), \text{if } k4=0,\\ \mathrm{mod}(R+B1+Y,256), \text{if } k4=1.\end{cases}$$

The resulted color cipher-image of plain-image Lena (Figure 1(a)) consists of the three base color components $R1,G1,B1$, as shown in Figure 1(b).

(a) (b)

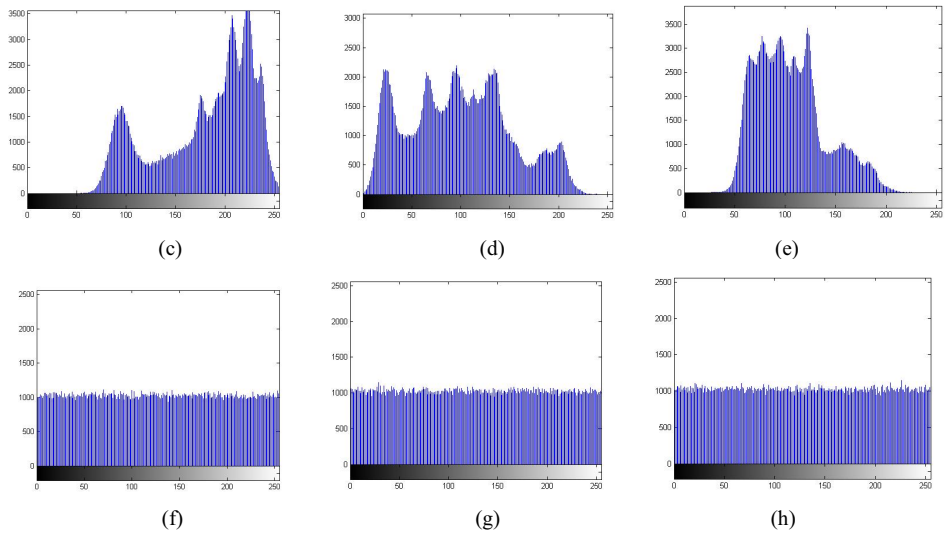

Figure 1. The encrypted results: (a) plain-image Lena; (b) cipher-image; (c)-(e) histograms for R, G, B components of Lena; (f)-(h) histograms for R, G, B components of cipher-image.

3 Performance analysis

An ideal cryptosystem requires high sensitivity to cipher keys, i.e., the cipher-text should have high correlation with cipher keys [1]. Furthermore, an ideal cryptosystem should have a large key space to make brute-force attack infeasible; it should also well resist various kinds of attacks like statistical attack, differential attack, etc. Some performance and security analysis will be carried out in this section, including the most important ones like key sensitivity, key space, statistical analysis (histogram analysis, information entropy analysis, correlation coefficients between adjacent pixels), and differential attack analysis. All the analysis shows that the proposed image encryption algorithm is secure enough thanks to its high sensitivity of the control parameters and initial conditions of the considered chaotic systems, large key space, and satisfactory self-adaptive approach.

3.1 Histogram analysis

Histogram analysis is applied to demonstrate the pixel distribution over the available intensity levels of an image. Regarding a 24-bit true color image, three histograms can be depicted for each 8-bit red, green and blue color components. We encrypt the color plain-image Lena one round with cipher keys $(x_0, y_0, a, b, w_0, z_0, a_1, b_1, N_1)$ to be (0.37, 0.59, 0.73, 0.81, 0.43, 0.59, 0.26, 0.41,100), and then plot the histograms of red, green, blue components of plain-image and cipher-image as shown in Figure 1(c)-(h). It follows from the histograms of the cipher-image that they are fairly uniform and significantly different from the corresponding histograms of the plain-image, and therefore the proposed image encryption algorithm is strongly robust against histogram analysis.

3.2 Correlation analysis between adjacent pixels

It is a common sense that each pixel is highly related to its adjacent pixels either at horizontal, vertical or diagonal direction for one nature image with definite visual content. An ideal cryptosystem should

generate cipher-images with less correlation between adjacent pixels. We calculate the correlation coefficient of the whole pairs by

$$Cr = \frac{cov(x,y)}{\sqrt{D(x)}\sqrt{D(y)}}, \quad cov(x,y) = \frac{1}{T}\sum_{i=1}^{T}(x_i - E(x))(y_i - E(y)) \quad E(x) = \frac{1}{T}\sum_{i=1}^{T}x_i, \quad D(x) = \frac{1}{T}\sum_{i=1}^{T}(x_i - E(x))^2 \quad (2)$$

where x_i, y_i form the *i*-th pair of horizontally, vertically or diagonally adjacent pixels and $T = H \times (W-1), (H-1) \times W, (H-1) \times (W-1)$ for the three cases respectively. The correlation coefficients of horizontally, vertically, diagonally adjacent pixels for plain-image Lena and its cipher-image are given in Table 1. One can observe from Table 1 that the proposed image encryption algorithm significantly reduces the correlation between the adjacent pixels of the plain-image.

We also calculate the correlation coefficients between the red, green, blue components of the plain-image and the cipher-image. The numerical results are given in Table 2, Table3. We can see from Table 2 that the coefficients Crg, Crb, Cgb between the R, G, B components of the plain-image is close to 1, while those of the cipher-image is close to zero. The correlation coefficients between the red, green, blue components of the plain-image and cipher-image are all close to zero. All these results indicate that the proposed image encryption algorithm reduces the correlations of the three base color components significantly.

Table 1. Correlation coefficients between adjacent pixels of plain-image and cipher-image.

		Correlation between adjacent pixels		
		Red	Green	Blue
Horizontal	Plain-image	0.9753	0.9871	0.9634
	Cipher-image	-0.0020	-0.0022	0.0005
Vertical	Plain-image	0.9748	0.9872	0.9630
	Cipher-image	-0.0007	-0.0039	-0.0021
Diagonal	Plain-image	0.9532	0.9741	0.9334
	Cipher-image	-0.0002	-0.0011	-0.0001

Table 2. Correlation coefficients between the R, G, B components within the plain-image and cipher-image.

	Crg	Crb	Cgb
Plain-image	0.8856	0.6995	0.9191
Cipher-image	-0.0003	-0.0007	0.0005

Table 3. Correlation coefficients between the R, G, B components of plain-image and cipher-image.

		Plain-image		
		Red	Green	Blue
Cipher-image	Red	0.0016	0.0038	0.0039
	Green	-0.0007	-0.0010	-0.0004
	Blue	-0.0008	-0.0014	-0.0014

3.3 Information entropy analysis

Information entropy is an index to measure the randomness of information sequence [8]. It can be applied to measure the uniformity of image histogram as well. The entropy $H(m)$ of a message source m can be calculated by $H(m) = -\sum_{i=0}^{K-1} p(m_i)\log(p(m_i))$ (bits), where K is the total number of symbols m, $p(m_i)$ represents the probability of occurrence of symbol m_i and log denotes the base 2 logarithm so that the entropy is expressed in bits. Considering a random source with 256 outcomes,

sharing equal probability, its entropy equals to 8. For a 24-bit color image, the information entropy for red, green, blue color components can be calculated respectively. We have calculated the information entropy for plain- image Lena and its cipher-image. The results are shown in Table 4. The values of information entropy for the cipher-image are very-very close to the expected value of truly random image, i.e., 8bits, implying that the proposed encryption algorithm is extremely robust against entropy attacks.

Table 4. Information entropy analysis.

	Red	Green	Blue
Plain-image	7.2634	7.5899	6.9854
Cipher-image	7.9994	7.9993	7.9993

3.4 Difference attack analysis

Differential cryptanalysis of a block cipher is the study of how differences in a plaintext can affect the resultant differences in the ciphertext with the same cipher key. It is usually done by implementing the chosen plaintext attack. Regarding image cryptosystems, attackers generally make a minor change (e.g., modify only one pixel) of the plain-image, and compare two cipher-images (obtained by applying the same cipher key on two plain-images having one pixel difference only) to find out some meaningful relationships between the plain-image and the cipher-image. If a meaningful relationship between plain-image and cipher-image can be found, it may further help the opponents to determine the cipher keys or equivalent key streams. If one slight change in the plain-image will cause significant, random and unpredictable changes in the cipher-image, then the encryption algorithm will resist differential attack efficiently.

Usually two most common measures NPCR (number of pixel change rate) and UACI (unified average changing intensity) are used to test the robustness of image cryptosystems against the differential cryptanalysis. If $C^{R/G/B}$ and $\bar{C}^{R/G/B}$ represent the R,G,B components for two cipher-images, then NPCR and UACI for each color component are calculated by

$$\mathrm{NPCR}^{R/G/B} = \frac{\sum_{i=1}^{H}\sum_{j=1}^{W} D_{i,j}^{R/G/B}}{W \times H} \times 100\%, \quad D_{i,j}^{R/G/B} = \begin{cases} 1, \text{ if } C_{i,j}^{R/G/B} \neq \bar{C}_{i,j}^{R/G/B}, \\ 0, \text{ if } C_{i,j}^{R/G/B} = \bar{C}_{i,j}^{R/G/B}, \end{cases}$$

$$\mathrm{UACI}^{R/G/B} = \frac{1}{W \times H} \sum_{i=1}^{H}\sum_{j=1}^{W} \frac{| C_{i,j}^{R/G/B} - \bar{C}_{i,j}^{R/G/B} |}{2^{L^{R/G/B}} - 1} \times 100\%.$$

We have performed the differential analysis by calculating NPCR and UACI on plain-image Lena. The analysis has been done by randomly choosing 10 pixels in plain-image, and changing one of the three color intensity values by one unit randomly at the selective pixel as well. The averages of 10 NPCR values and 10 UACI values thus obtained for all three color components are given in Table 5. It is clear that the NPCR and UACI values are very close to the expected values, thus the proposed image encryption technique shows good sensitivity to plaintext and hence invulnerable to differential attacks.

Table 5. Difference analysis of plain-image Lena.

Average NPCR (%)			Average UACI (%)		
Red	Green	Blue	Red	Green	Blue
99.5998	99.6110	99.6112	33.4212	33.4531	33.4738

3.5 Key sensitivity analysis

The key space of an encryption algorithm is composed of the total number of different cipher keys that can be used in the encryption procedure. The high sensitivity of the cipher-image to initial conditions and control parameters is inherent to any chaotic system. A good image encryption algorithm should contain sufficiently large key space. It should be sensitive to cipher keys as well, and thus can effectively prevent invaders decrypting original data even after they invest large amounts of time and resources. Strong key sensitivity is an essential feature of an effective cryptosystem. Key sensitivity of a cryptosystem can be observed and simulated from two aspects: (i) significantly different cipher-images should be generated even if keys with tiny difference are applied to encrypt the same plain-image; (ii) the cipher-image cannot be decrypted even if there is minor difference between encryption and decryption keys. We will use the following cipher keys to perform the simulations (one is master cipher key MKEY, the other keys are set by introducing a slight change to one of the parameters of master cipher key with all other parameters unchanged). Master cipher key is set to be $(x_0, y_0, a, b, w_0, z_0, a_1, b_1, N_1)$= (0.37, 0.59, 0.73, 0.81, 0.43, 0.59, 0.26, 0.41,100). Nine slightly different keys are

SKEY1 $(x_0 - 10^{-14}, y_0, a, b, w_0, z_0, a_1, b_1, N_1)$; SKEY2 $(x_0, y_0 - 10^{-14}, a, b, w_0, z_0, a_1, b_1, N_1)$;

SKEY3 $(x_0, y_0, a - 10^{-14}, b, w_0, z_0, a_1, b_1, N_1)$; SKEY4 $(x_0, y_0, a, b - 10^{-14}, w_0, z_0, a_1, b_1, N_1)$;

SKEY5 $(x_0, y_0, a, b, w_0 - 10^{-14}, z_0, a_1, b_1, N_2)$; SKEY6 $(x_0, y_0, a, b, w_0, z_0 - 10^{-14}, a_1, b_1, N_1)$;

SKEY7 $(x_0, y_0, a, b, w_0, z_0, a_1 - 10^{-14}, b_1, N_1)$; SKEY8 $(x_0, y_0, a, b, w_0, z_0, a_1, b_1 - 10^{-14}, N_1)$;

SKEY9 $(x_0, y_0, a, b, w_0, z_0, a_1, b_1, N_1 + 1)$.

(i) To estimate the key sensitivity for the first case, we encrypt plain-image Lena with MKEY and get the first cipher-image, then we encrypt Lena with SKEY1-SKEY9 and get nine cipher-images respectively. The correlation coefficients between the first cipher-image and the other nine cipher-images are calculated by Eq. (2). The experimental results are shown in Table 6. From the results, we can see that all the correlation coefficients are very small which indicate that even there is only slightly difference between the cipher keys, the cipher-images are greatly different. Hence the proposed encryption algorithm is extremely sensitive to the cipher keys.

(ii) Decryption using keys with minor difference is also performed in order to evaluate the key sensitivity of the second case. Firstly, we decrypt the cipher image using MKEY and we get the plain-image Lena. Secondly, nine decrypted images are produced as we decrypt the cipher-image using SKEY1-SKEY9 respectively. We have computed the correlation coefficients between Lena and nine decrypted images. The results are given in Table 7. It follows from Table 7 that there is even only a tiny difference between the decipher keys, the decrypted images have low correlation coefficients with the plain-image Lena. As for the second case, the proposed encryption algorithm also shows highly sensitivity to the cipher keys.

Since the permutation process is irrelevant to the diffusion process, the key space consists of the cipher keys in both the scrambling process and the diffusion process. We can conclude from the sensitivity analysis that the key space of the encryption algorithm is as large as $10^{14\times 8} \times 10^3 = 10^{75}$, if we set the seed N_1 to be one integer between 0 and 1000.

Conclusions

In this paper, we present a novel self-adaptive bit-level color image encryption algorithm using generalized Arnold maps. The 2D generalized Arnold maps are applied to generate chaotic sequences for both the sort-based scrambling stage and the color components' cross diffusion stage. The key streams are all related to the content of the plain-image and therefore the proposed image encryption algorithm show strong resistance against known-plaintext and chosen-plaintext attacks. All the experimental results show that the proposed image encryption algorithm is highly secure and demonstrates excellent performance.

Table 6. Key sensitivity analysis I.

	Correlation coefficients between the encrypted images obtained using MKEY and								
	SKEY1	SKEY2	SKEY3	SKEY4	SKEY5	SKEY6	SKEY7	SKEY8	SKEY9
Crr	0.0017	-0.0021	0.0007	0.0019	0.0004	0.0009	0.0009	-0.0015	0.0023
Crg	-0.0017	-0.0018	0.0033	-0.0003	0.0016	0.0021	0.0002	0.0010	-0.0001
Crb	-0.0017	0.0044	0.0008	0.0012	0.0031	-0.0011	-0.0012	0.0015	0.0014
Cgr	-0.0033	-0.0012	-0.0007	0.0012	0.0008	-0.0002	0.0015	-0.0039	0.0024
Cgg	-0.0002	-0.0013	0.0006	-0.0014	-0.0023	-0.0002	-0.0020	0.0032	-0.0024
Cgb	0.0017	0.0004	0.0003	0.0016	-0.0020	0.0016	0.0013	0.0009	-0.0004
Cbr	0.0005	-0.0041	0.0019	0.0016	0.0017	-0.0003	0.0003	-0.0026	0.0014
Cbg	-0.0001	-0.0010	-0.0012	0.0008	-0.0016	0.0012	0.0025	0.0016	0.0027
Cbb	-0.0022	0.0029	0.0001	0.0002	0.0013	0.0032	-0.0012	0.0001	0.0031

Table 7. Key sensitivity analysis II.

	Correlation coefficients between the decrypted images obtained using MKEY and								
	SKEY1	SKEY2	SKEY3	SKEY4	SKEY5	SKEY6	SKEY7	SKEY8	SKEY9
Crr	0.0031	-0.0065	0.0000	-0.0009	0.0009	-0.0028	0.0007	0.0010	-0.0003
Crg	0.0054	0.0001	-0.0020	0.0051	0.0001	0.0000	0.0016	0.0029	-0.0024
Crb	0.0039	0.0018	0.0090	-0.0068	0.0046	0.0038	-0.0001	-0.0037	0.0008
Cgr	0.0027	-0.0065	0.0005	-0.0015	0.0003	-0.0029	0.0001	0.0006	0.0004
Cgg	0.0062	0.0004	-0.0035	0.0052	0.0000	0.0002	0.0003	0.0022	-0.0021
Cgb	0.0051	0.0038	0.0073	-0.0040	0.0054	0.0036	0.0025	-0.0032	0.0008
Cbr	0.0000	-0.0037	-0.0021	-0.0031	-0.0001	-0.0027	-0.0003	0.0007	-0.0002
Cbg	0.0021	0.0009	-0.0014	0.0060	0.0002	0.0007	-0.0009	0.0011	-0.0018
Cbb	-0.0016	0.0032	0.0075	-0.0014	0.0049	0.0030	0.0041	-0.0017	0.0006

Acknowledgement

This research is partially supported by Entrepreneurship Training Program of Guangdong Colleges and partially by SRP of Science College of Shantou University.

References

1. D. R. Stinson, *Cryptography: Theory and Practice* (CRC Press, Boca Raton,1995)
2. J. Fridrich, Symmetric ciphers based on two-dimensional chaotic maps, Int. J Bifurcat. Chaos, **8**, 1259-1284 (1998)
3. R. Ye, A novel chaos-based image encryption scheme with an efficient permutation-diffusion mechanism, Opt. Commun., **284**, 5290-5298 (2011)
4. X. J. Tong, Design of an image encryption scheme based on a multiple chaotic map, Commun. Nonlinear Sci. Numer. Simulat., **18** , 1725-1733 (2013)
5. R. Ye, A novel image encryption scheme based on generalized multi-sawtooth maps, Fund. Inform., **133** , 87-104 (2014)
6. J. Chen, Z. Zhu, C. Fu, H. Yu, L. Zhang, An efficient image encryption scheme using gray code based permutation approach, Opt. Laser Eng., **67** , 191-204 (2015)
7. H. Liu, X. Wang, Color image encryption using spatial bit-level permutation and high-dimension chaotic system, Opt. Commun., **284**, 3895-3903(2011)
8. C. E. Shannon, Communication theory of secrecy system, Bell Syst. Tech. J, **28**, 656-715 (1949)