Nonlinear Dyn (2014) 75:567–576
DOI 10.1007/s11071-013-1086-2

ORIGINAL PAPER

A novel image encryption algorithm based on dynamic S-boxes constructed by chaos

Xingyuan Wang · Qian Wang

Received: 17 May 2013 / Accepted: 17 September 2013 / Published online: 10 October 2013

Abstract In this manuscript, an image encryption based on dynamic S-boxes is presented, in which the S-boxes are constructed by chaotic systems. An external 256-bit key and the last pixel of plain image are used to generate the parameters and initial states of the chaotic systems for the first S-box. The plain image is divided into groups in which the pixels are substituted by S-boxes and in order to smash the correlation of adjacent pixels the image is grouped in four directions. After encrypting previous group, the initial states of chaotic systems are altered by encrypted image pixels and the S-box for the next group is generated. This algorithm scheme can make it resist differential attacks and chosen plain-text attacks. Moreover, because in the all process we only need to construct less than 50 S-boxes, the progress time is reduced. Superiority in speed and security is analyzed by applying the algorithm on 256-grey images.

Keywords Image encryption · S-box · Logistic map · Kent map

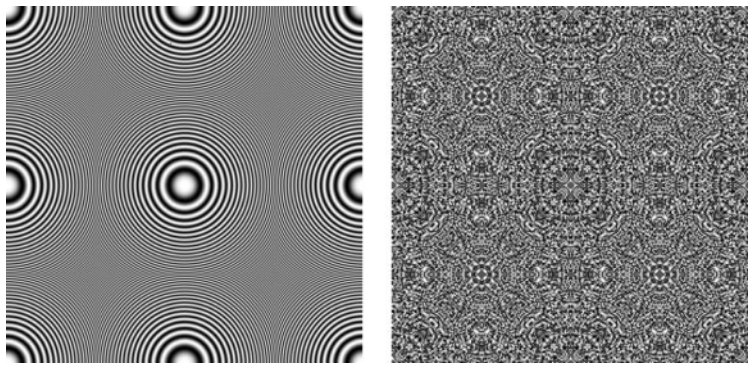
X. Wang · Q. Wang (✉)
Faculty of Electronic Information and Electrical Engineering, Dalian University of Technology, Dalian 116024, China
e-mail: wq605053@163.com

X. Wang
e-mail: wangxy@dlut.edu.cn

1 Introduction

Coupled with the development of multimedia technology and Internet, encryption of images has become an emergency research in recent years. Images are classified by its high correlation between adjacent pixels, so traditional encryption by S-box of DES (Data Encryption Standard) or AES (Advanced Encryption Standard) is not enough. Figure 1 shows the result of encryption by a single S-box, which is the core component in DES and AES. From the figure, we can see that the outline of the image is clear, so we must find a more efficient method for image encryption.

Chaotic systems have good features of sensitive dependence on initial conditions, pseudo-randomness, periodicity, and reproduction. Lots of image encryption algorithms based on chaos are proposed [1–22]. The authors presented an image encryption algorithm based on reversible cellular automata combining chaos in [1]; in [2, 15, 17] coupled chaotic system was used; the authors proposed an image encryption algorithm based on linear hyperbolic chaotic system of partial differential equations in [3]; the hyperchaotic system was used in the algorithm proposed in [19]; the authors in [6, 14] used piecewise linear chaotic map in algorithms; delayed fractional-order chaotic logistic system was employed by the authors of [7]; the authors of [8] used classical chaotic masking technique; the authors of [11–13] employed chaotic maps and S-box in their algorithms; in [20], the authors presented a double optical image encryption, which used discrete



(a) Zone.bmp (b) Cipher image encrypted by S-box

Fig. 1 Encryption result by a single S-box

Chirikov standard map and chaos-based fractional random transform; the authors presented a method using self-synchronizing to improve security of image encryption algorithms based on multichaos in [21]; in [22], the authors analyzed a chaos-base image encryption and improved the algorithm. In recent years, not only in image encryption chaotic systems are also used to construct S-boxes. S-box is one of the core components in block cipher and has been widely used in cryptographic standards such as DES and AES. Recent researches show that it is a novel and promising direction to utilize the nonlinear property of chaos to design S-boxes. Lots of S-box construction algorithms [23–26] based on chaotic systems have been proposed in recent years. Furthermore, S-box was also used in image encryption [11–13] and watermarking [27]. In [28], the authors proved that the algorithms cannot resist chosen plain-text attacks in which the S-boxes are generated before encryption [11–13].

All above image encryption algorithms were complexly designed, and some of them cost more time while the others cannot resist certain attacks. In this paper, we proposed an image encryption algorithm, which uses dynamic S-boxes, that is, the S-boxes are generated according to the plain image when the encryption is proposed. In order to obtain a fast and secure image encryption method, we observed that if the correlation between adjacent pixels could be smashed, then we could use S-box in image encryption. To smash the correlation, a method is to shuffle the image pixels first and another method is to substitute adjacent pixels by different S-boxes. In the algorithm, the latter is employed and chaotic systems are used to construct the S-boxes. Because for only one image pixel constructing an S-box is time-consuming and unrealistic, the image pixels are divided to several groups according to rows and adjacent rows are in different groups. For each group, a new S-box is generated and used. By this way, the corrections between vertical adjacent pixels are smashed. And then we use the same method on columns in order to smash its correction between horizontal adjacent pixels. In order to get higher security, we do the encryption from four directions. An external 256-bit secret key is employed in order to get large key space. The last pixel of plain image and encrypted pixel values are going to influence the construction of S-boxes in order to resist differential attacks and chosen plain-text attacks. It is proved that the algorithm has better character in encryption speed and security.

The rest of the paper includes: In Sect. 2, chaotic systems and the construction method of a S-box are introduced, which is employed in the proposed algorithm; the encryption and decryption algorithm will be presented in Sect. 3; in Sect. 4, the speed and security analysis of the algorithm and comparison with other algorithms are illustrated and the conclusion is given in Sect. 5. Finally, the references are listed.

2 Chaotic systems and construction of S-box

The logistic map and the Kent map are two of the frequently-used chaotic maps. They can be presented as follows:

$$x_{n+1} = \mu x_n(1 - x_n), \tag{1}$$

$$x_{n+1} = \begin{cases} \frac{x_n}{b}, & 0 \le x \le b, \\ \frac{1-x_n}{1-b}, & b < x \le 1, \end{cases} \tag{2}$$

where $x_n \in (0, 1)$, $\mu \in (0, 4]$ and $b \in (0, 1)$. When $\mu = 4.0$, the data range of x_n in Eq. (1) is $[0, 1]$. In our algorithm, we use $\mu = 4.0$.

The construction algorithm of S-boxes would use the chaotic maps presented in Eqs. (1)–(2) [25]. A brief introduction of the construction method would be stated in following, and the feathers of the S-boxes constructed have been proved in [25].

Step 1. Set a sequence $Y = [0, 1, 2, \ldots, n-1]$ and an empty sequence $Z = [\,]$ where $n = N \times N$ if we want to construct a $N \times N$ S-box.

Step 2. Divide $(0, 1)$ into n minizones and label them as T_i $(i = 0, 1, \ldots, n-1)$.

Step 3. Iterate Eqs. (1)–(2) several times alternately, that is, after iterating Eq. (1) use its state as initial state of Eq. (2), iterate once and then use its state as initial state of Eq. (1). A state is obtained and if it belongs to the minizone T_i and i is not in sequence Z, add i to the end of sequence Z.

Step 4. Repeat *Step* 3 until there are n elements in sequence Z.

Step 5. Translate sequence Z to a $N \times N$ table, then the S-box is gotten.

In proposed image encryption algorithm, 16×16 S-boxes are used, so $n = 256$.

3 Encryption and decryption algorithm

In order to resist chosen plain-text attacks, the encryption S-boxes must be different due to different images, so we also use the last pixel value of the plain image as the secret key which is presented by pk. Coupled with pk an external 256-bit secret key is needed for calculating the initial state and parameters of chaotic systems in the proposed algorithm. The image pixels are divided into groups according to rows or columns and different groups use different S-boxes constructed by chaotic maps. Before processing the next group, we alter the initial state of the chaotic system according to the encrypted group pixel values in order to make the algorithm robust to resisting differential attacks and chosen plain-text attacks. The encryption algorithm will be stated in detail as follows:

Step 1. Present the external 256-bit secret key, the plain image and pk as

$$K = [k_1, k_2, \ldots, k_{32}], \tag{3}$$

$$I = [v_{i,j}]\ (i, j = 1, 2, \ldots, 256), \tag{4}$$

$$pk = v_{256,256}, \tag{5}$$

and calculate the initial state and parameters of chaotic system according to Eqs. (3)–(17).

$$xsum = pk \oplus k_1 \oplus k_2 \oplus \cdots \oplus k_{32}, \tag{6}$$

$$\mu = 4.0, \tag{7}$$

$$b = \left\{(k_{32} + k_1 + xsum)/2^8 + 0.01\right\}, \tag{8}$$

$$d_1 = (k_{26} + k_7 + xsum) \bmod 5 + 5, \tag{9}$$

$$d_2 = (k_{27} + k_6 + xsum) \bmod 5 + 5, \tag{10}$$

$$d = [d_1, d_2], \tag{11}$$

$$x_1 = \left\{(k_{23} + k_{10} + xsum)/2^8 + 0.01\right\}, \tag{12}$$

$$x_2 = \left\{(k_{22} + k_{11} + xsum)/2^8 + 0.01\right\}, \tag{13}$$

$$x_3 = \left\{(k_{21} + k_{12} + xsum)/2^8 + 0.01\right\}, \tag{14}$$

$$x_4 = \left\{(k_{20} + k_{13} + xsum)/2^8 + 0.01\right\}, \tag{15}$$

$$xr = [x_1, x_2], \tag{16}$$

$$xc = [x_3, x_4], \tag{17}$$

$$round = 1, \tag{18}$$

where μ and b are parameters of logistic map and the Kent map, respectively; $\{x\}$ can get decimal part of x; d is a vector by which the image pixels are divided into groups; xr, xc are the initial states of chaotic system while encrypting groups in rows and columns, respectively, by S-boxes.

Step 2. Do

$$pace = d(round),$$

$$x_n = xr(round),$$

$$xsum = 0.$$

And make

$$group_1 = [v_{1,j}]$$

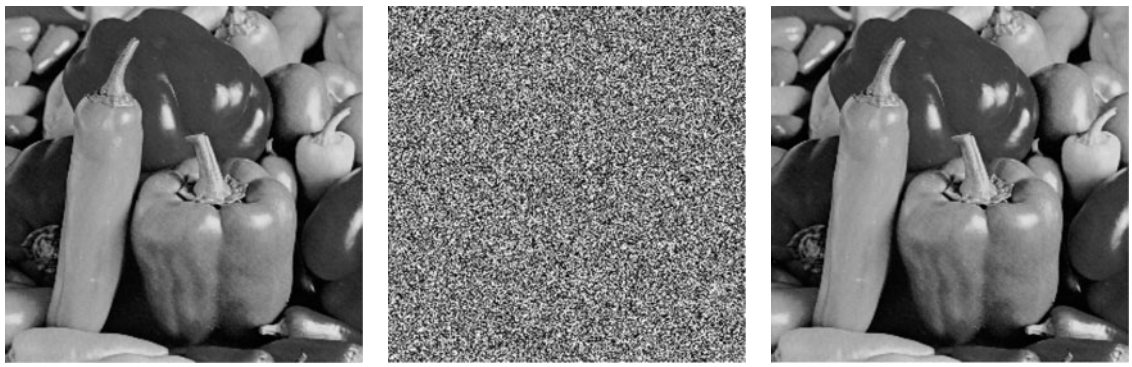
and

$$group_l = [v_{l,j}, v_{l+pace,j}, v_{l+2pace,j}, \ldots]$$

$$(l = 2, 3, \ldots, pace + 1).$$

Step 3. Iterate the chaotic maps alternately and construct an S-box.

Step 4. Substitute the pixels of $group_1$ by the S-box constructed in *Step* 3 and after each substitution xor the encrypted pixel to $xsum$.



(a) Peppers (b) Cipher image (c) Decoding image

Fig. 2 Experimental results of the algorithm proposed

Step 5. Alter the x_n by

$$x_n = \{x_n + xsum/2^8\}.$$

Initialize

$$xsum = 0.$$

Step 6. Repeat the *Steps* 3–5 for $group_l$.
Step 7. Do

$$x_n = xc(round),$$

and

$$xsum = 0;$$

make

$$group_1 = [v_{i,1}]$$

and

$$group_l = [v_{i,l}, v_{i,l+pace}, v_{i,l+2pace}, \ldots].$$

Step 8. Repeat *Steps* 3–6. Then reverse the image pixels and do $round = 2$.
Step 9. Repeat *Steps* 2–8.

Now, the cipher image is gotten. The decryption steps are almost the same with the encryption steps except that the reverse of image pixels must be done first and S-boxes must be changed into their inverse S-boxes. The experimental result is shown in Fig. 2. Figure 2(a) shows the plain-text image of peppers and its cipher-text image and decoding image are shown in Fig. 2(b) and Fig. 2(c), respectively.

4 Security analysis and speed analysis

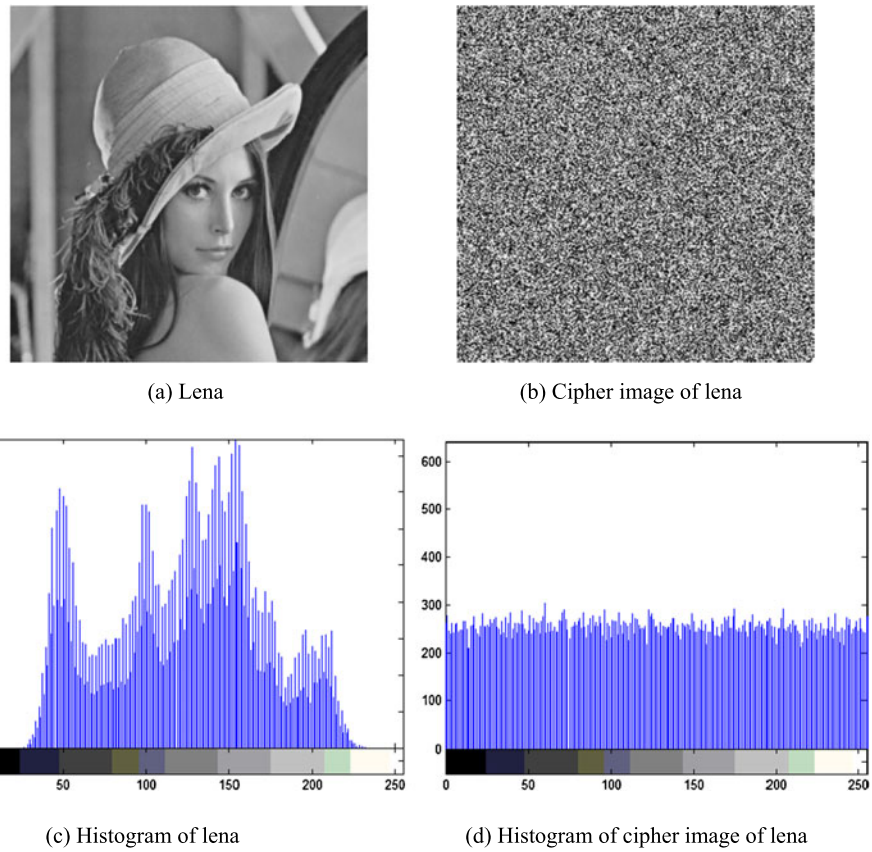
4.1 Statistical analysis

It is well known that the statistical property of a cipher image is enormously vital and an ideal image algorithm should be robust against any statistic attacks. Histogram and correlation of two adjacent pixels are two important indicators of statistical analysis. To describe the statistical property of the proposed algorithm, the authors applied the encryption algorithm to 256×256 256-grey images.

Histogram: Histograms of plain image and cipher image are plotted, through which we can intuitively see the number of pixels of each value. A good image algorithm should make the histogram of cipher image as much as possible flat. The histograms of lena and its cipher image are shown in Fig. 3. Figures 3(a) and (c) are lena image and its histogram; Figs. 3(b) and (d) are the cipher image of lena and its histogram.

Correlation of two adjacent pixels: Generally speaking, the two adjacent pixels of a plain image would come near to each other and a good image encryption algorithm could smash this relation between them. 10000 pairs of adjacent pixels from plain image of a bird and its cipher image are selected randomly in horizontal direction, vertical direction, and diagonal direction, respectively, and the correlation of them is plotted out. The results are showed in Fig. 4. Moreover, the correlation coefficients r_{xy} of each pair are calculated using the following equations [17]:

$$\operatorname{cov}(x, y) = E\{(x - E(x))(y - E(y))\}, \tag{19}$$



(a) Lena (b) Cipher image of lena

(c) Histogram of lena (d) Histogram of cipher image of lena

Fig. 3 Histograms of plain image and cipher images

Table 1 Correlation coefficients of two adjacent pixels in the plain and cipher images

Name	Plain image			Cipher image		
	Horizontal	Diagonal	Vertical	Horizontal	Diagonal	Vertical
lena	0.9395	0.9286	0.9789	0.0097	0.0178	0.0136
moon	0.9001	0.9001	0.9360	0.0205	0.0105	0.0406
boat	0.9130	0.8862	0.9362	0.0143	0.0072	0.0014
brain	0.9489	0.9293	0.9682	0.0231	0.0045	0.0714
finger	0.9021	0.8379	0.9500	0.0180	0.0036	0.0040

$$r_{xy} = \frac{\operatorname{cov}(x, y)}{\sqrt{D(x)}\sqrt{D(y)}}, \tag{20}$$

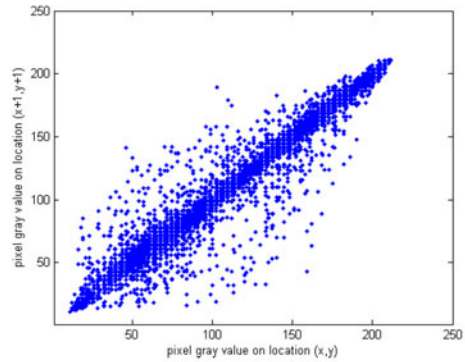
where x and y are values of the two adjacent pixels in the image,

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

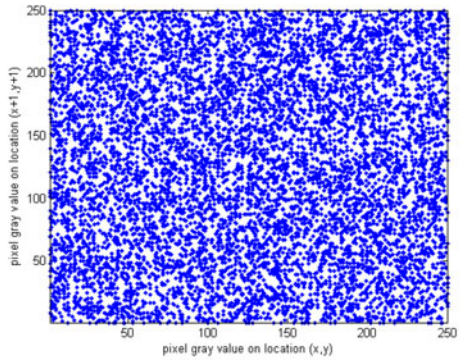
and

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

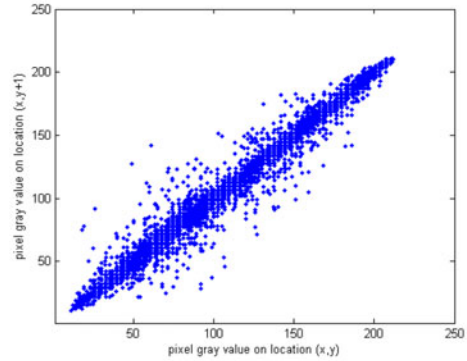
The correlation coefficients of the plain image and cipher images are shown in Table 1. From Fig. 4 and Table 1, it could be known that there are no detectable



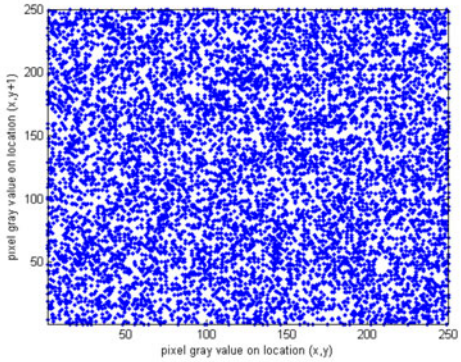
(a) Diagonal direction in plain image



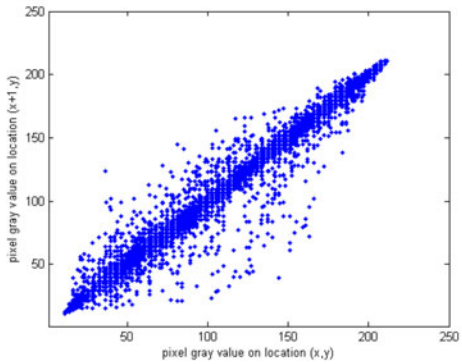
(b) Diagonal direction in cipher image



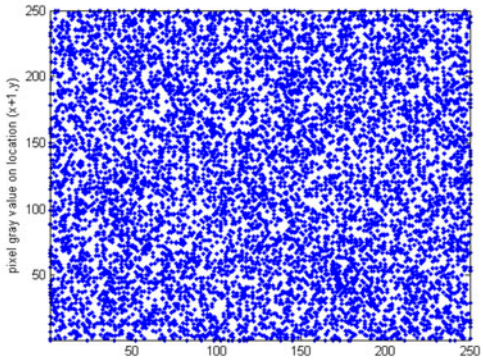
(c) Horizontal direction in plain image



(d) Horizontal direction in cipher image



(e) Vertical direction in plain image



(f) Vertical direction in cipher image

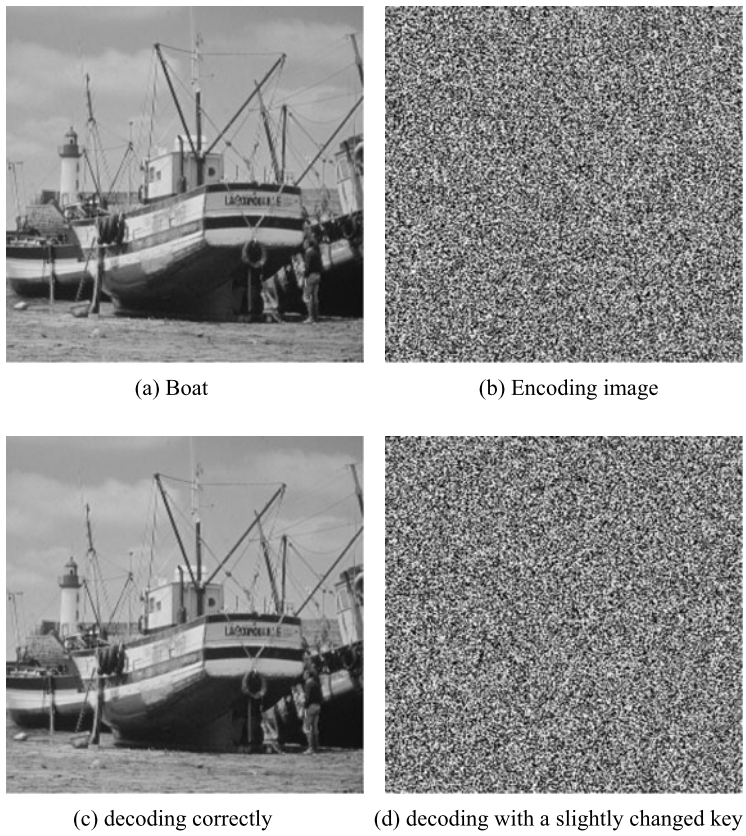
Fig. 4 Correlations of two adjacent pixels

correlations exist between the plain image and its corresponding cipher images.

4.2 Key space analysis

It is widely known that a good encryption algorithm should have big key space and be sensitive to its key. In this research, an external 256-bit secret key is designed and all initial conditions, parameters, and group basis are generated by the secret key and the last pixel value of plain image.

Key space: In the proposed algorithm, a 256-bit external key is needed, so that the key space is 2^{256} and it is big enough to stand up brute force attacks.



(a) Boat (b) Encoding image

(c) decoding correctly (d) decoding with a slightly changed key

Fig. 5 The plain image, encoding image, decoding correctly image and decoding by a slightly changed key

Sensitivity to secret key: the external key "116, 3, 163, 12, 213, 82, 4, 130, 56, 112, 27, 101, 127, 90, 110, 19, 216, 19, 1, 157, 149, 24, 223, 68, 1, 30, 41, 65, 23, 64, 19, 16" in which each 8-bit is presented by decimalism is used to encrypt the plain image "boat.bmp." A slightly changed key "116, 3, 163, 12, 213, 82, 4, 130, 56, 112, 27, 101, 127, 90, 110, 19, 216, 19, 1, 157, 149, 24, 223, 68, 1, 30, 41, 65, 23, 64, 19, 17" and the correct key are used to decrypt the cipher image, respectively. The results are shown Fig. 5. The different rates of cipher images, which are encrypted by the correct key and the slightly changed key, respectively, are shown in Table 2.

Table 2 Different rate of the cipher images encrypted by correct key and the slightly changed key

Name	Different rate of the cipher images
house	0.9965
zone	0.9962
plane	0.9960
bird	0.9963
finger	0.9959

4.3 Information entropy

The information entropy can be calculated by

$$H(m) = \sum_{0}^{M-1} p(m_i) \log \frac{1}{p(m_i)}, \tag{21}$$

in which m is a set of symbols; M is the total number of symbols $m_i \in m$; $p(m_i)$ is the probability of m_i. In the experiment, 256 gray level images are used, so the theoretical $H(m)$ should be 8. The information entropies of cipher images using our algorithm are shown in Table 3. From Table 3, it is known that the information entropies are close to eight, so the algorithm proposed has good property of information entropy.

4.4 Resisting differential attack analysis

In order to resist differential attacks, a petty change in plain image must cause amount of differences of pixels in cipher images. Two quantitative measures, which

are number of pixels change rate (NPCR) and unified average changing intensity (UACI), are used to measure the influences on cipher images of a one-pixel change in plain image. A cipher image and its plain image are presented as C_1, I_1 and another cipher image and its plain image are presented as C_2, I_2 where there is only one pixel value different between I_1 and I_2. A matrix D is created, where if $C_1(i, j) = C_2(i, j)$, $D(i, j) = 0$; otherwise $D(i, j) = 1$. NPCR and UACI are calculated by [18]:

$$NPCR = \frac{\sum_{i,j} D(i, j)}{W \times H} \times 100\ \%, \tag{22}$$

$$UACI = \frac{1}{W \times H}\left(\sum_{i,j} \frac{|C_1(i, j) - C_2(i, j)|}{255}\right) \times 100\ \%, \tag{23}$$

where W, H are the width and height of the image, respectively.

A pixel from plain image is changed by adding 1 to it and a new cipher image is got which is compared with the old cipher image to calculate NPCR and UACI. In the experiment, for each sample image the last pixel and 999 randomly selected pixels are taken to change and 1000 pairs of cipher images are compared, then the average, the minimum, and the maximum of NPCRs and UACIs are calculated, respectively. The experiment results are shown in Table 4 and a conclusion can be inferred that the proposed algorithm has good property in resisting differential attacks. By contrast, the algorithm proposed in [19] cannot resist differential attacks, because the authors did not do any diffusion in the encryption progress, the cipher pixels only relating with the hyperchaos and plain image's current pixels.

Table 3 Information entropies of cipher image

Name	Entropies of cipher image
house	7.9970
zone	7.9970
lena	7.9971
bird	7.9976
finger	7.9972

4.5 Resisting known-plaintext and chosen-plaintext attacks analysis

In [28], the authors proposed a cryptanalysis algorithm of S-boxes-only cipher images against chosen attacks. This algorithm can break image encryption algorithms in which the S-boxes are constructed in advance.

But according to our algorithm, the construction of S-boxes are doing in the encryption procedure and the initial conditions of the chaotic system for constructing S-box are influenced by the last pixel of plain image and encrypted pixel values. So, with different images, the S-boxes for groups are different. Even if the attackers get a certain plain image and its cipher image, they could not use it in other cipher images. Therefore, the proposed algorithm can well resist the known- and chosen-plaintext attacks (as described in [29–31]). And as described above in Sect. 4.4, because of that the algorithm in [19] also cannot resist known- and chosen-plaintext attacks.

4.6 Speed analysis

In the proposed algorithm, we desert the traditional method of doing shuffling and diffusion stage and directly substitute the image pixels value by S-boxes for four times. We do not need to construct S-box for each row or column of image; we divide the image pixels into groups and use different S-boxes for different

Table 4 NPCRs and UCAIs between different ciphers while plain images only have only one different pixel

Name	Average		Minimum		Maximum	
	NPCR	UCAI	NPCR	UCAI	NPCR	UCAI
plane	0.9961	0.3345	0.9955	0.3321	0.9969	0.3368
lena	0.9961	0.3342	0.9956	0.3319	0.9967	0.3366
bird	0.9961	0.3348	0.9957	0.3323	0.9966	0.3368
house	0.9961	0.3347	0.9955	0.3321	0.9967	0.3365
boat	0.9961	0.3346	0.9956	0.3324	0.9969	0.3369

Table 5 Comparison the proposed algorithm with other algorithms on speed

Name	Our algorithm	Ref. [14] (1 round)	Ref. [16] (3 rounds)	Ref. [17] (2 rounds)	Ref. [19] (1 round)
Boat	1.0460	2.1400	2.4530	35.2650	1.4690
Lena	1.0740	2.1410	2.4540	34.8280	1.4840
Bird	1.0150	2.1410	2.4840	34.7190	1.5160
Finger	1.0520	2.1090	2.4690	34.8750	1.4530
Moon	1.0470	2.1250	2.4690	34.9850	1.5000
Plane	1.0940	2.1090	2.4690	34.8280	1.4840
Brain	1.0350	2.1710	2.4530	34.7190	1.4680
Peppers	1.0820	2.1100	2.4530	34.8280	1.5310
Zone	1.0710	2.1090	2.4070	34.9530	1.4840
House	1.0850	2.1560	2.4530	34.8130	1.4680

groups instead. So we reduce the time in constructing S-boxes. Comparisons with [14, 16, 17] and [19] are made and the results are shown in Table 5. Our experimental environments are MATLAB R2012a and a PC computer with Intel Core 2 Duo CPU E4500@ 2.20 GHz, 2.19 GHz, 1.98 G RAM, and Window XP OS. From experimental results shown in Table 5, it can be proved that our algorithm has better property in speed. In order to meet the highest security, we run the algorithms for different number of rounds accordingly.

In [17], they combine the shuffling and diffusion stage and reduce the iteration of the chaotic system, but they do too much logic operation on 64 numbers, which are used to encrypt a block. This makes the program scanning the 64 numbers repeatedly in encryption of each block, so its encryption speed is slow.

5 Conclusion

In this paper, an image encryption algorithm based on dynamic S-boxes is proposed and the security of the algorithm is analyzed. The cipher image is divided into groups and each group uses an S-box. In the experiment, we divide the image into less than ten groups, so although we scan the image for four times, we only need to construct less than 50 S-boxes. Before the construction of a new S-box the initial state would be altered by the prior encrypted group. And comparisons with the encryption algorithms proposed in [14, 16, 17], and [19] are made for proving that the proposed algorithm has better property in resisting differential attacks and speed. It is also illustrated that this algorithm has big key space and can stand up to information entropy analysis, known plaintext attacks, and chosen plaintext attacks.

Although the proposed algorithm has higher speed and good secure feature, there is still a lot of work to do. When an algorithm is complicated, the speed must be expensed and absolutely a high security will be obtained. By contrast, when an algorithm is simple, less time is consumed and the security will be low. We must find an image encryption algorithm, which has high security and consume as less time as possible. Also, the precision of the decimal in different computers or different OS could affect the decryption of image and it is important because the proposed algorithm is used in communication in Internet. So, the next work will be solving the accordance between different environments.

Acknowledgements This research is supported by the National Natural Science Foundation of China (Nos. 61370145, 61173183, and 60973152), the Doctoral Program Foundation of Institution of Higher Education of China (No. 20070141014), Program for Liaoning Excellent Talents in University (No. LR2012003), the National Natural Science Foundation of Liaoning province (No. 20082165) and the Fundamental Research Funds for the Central Universities (No. DUT12JB06).

References

1. Wang, X.Y., Luan, D.P.: A novel image encryption algorithm using chaos and reversible cellular automata. Commun. Nonlinear Sci. Numer. Simul. **18**(11), 3075–3085 (2013)
2. Rhouma, R., Meherzi, S., Belghith, S.: OCML-based colour image encryption. Chaos Solitons Fractals **40**(1), 309–318 (2009)

3. Zhang, Y.S., Xiao, D., Shu, Y.L., Li, J.: A novel image encryption scheme based on a linear hyperbolic chaotic system of partial differential equations. Signal Process. Image Commun. **28**(3), 292–300 (2013)
4. Liu, H.J., Wang, X.Y.: Color image encryption using spatial bit-level permutation and high-dimension chaotic system. Opt. Commun. **284**(16–17), 3895–3903 (2011)
5. Patidar, V., Pareek, N.K., Sud, K.K.: A new substitution-diffusion based image cipher using chaotic standard and logistic maps. Commun. Nonlinear Sci. Numer. Simul. **14**(7), 3056–3075 (2009)
6. Wang, X.Y., Jin, C.Q.: Image encryption using game of life permutation and PWLCM chaotic system. Opt. Commun. **285**(4), 412–417 (2011)
7. Wang, Z., Huang, X., Li, N., Song, X.N.: Image encryption based on a delayed fractional-order chaotic logistic system. Chin. Phys. B **21**(5), 050506 (2012)
8. Xiang, T., Liao, X.F., Tang, G.P., Chen, Y., Wong, K.: A novel block cryptosystem based on iterating a chaotic map. Phys. Lett. A **349**(1–4), 109–115 (2006)
9. Ye, R.S.: A novel chaos-based image encryption scheme with an efficient permutation-diffusion mechanism. Opt. Commun. **284**(22), 5290–5298 (2011)
10. Huang, C.K., Nien, H.H.: Multi-chaotic systems based pixel shuffle for image encryption. Opt. Commun. **282**(11), 2123–2127 (2009)
11. Hussain, I., Shah, T., Gondal, M.A.: An efficient image encryption algorithm based on S-8 S-box transformation and NCA map. Opt. Commun. **285**(24), 4887–4890 (2012)
12. Hussain, I., Shah, T., Gondal, M.A., Mahmood, H.: A novel image encryption algorithm based on chaotic maps and GF(2(8)) exponent transformation. Nonlinear Dyn. **72**(1–2), 399–406 (2013)
13. Hussain, I., Shah, T., Gondal, M.A.: Image encryption algorithm based on PGL(2,GF(2(8))) S-boxes and TD-ERCS chaotic sequence. Nonlinear Dyn. **70**(1), 181–187 (2012)
14. Behnis, S., Akhshani, A., Ahadpour, S., Mahnodi, H., Akhavan, A.: A fast chaotic encryption scheme based on piecewise nonlinear chaotic maps. Phys. Lett. A **366**(4–5), 391–396 (2007)
15. Seyedzadeh, S.M., Mirzakuchaki, S.: A fast color image encryption algorithm based on coupled two-dimensional piecewise chaotic map. Signal Process. **92**(5), 1202–1215 (2012)
16. Wang, X.Y., Zhao, J.F., Liu, H.J.: A new image encryption algorithm based on chaos. Opt. Commun. **285**(5), 562–566 (2012)
17. Wang, Y., Wong, K.W., Liao, X.F., Chen, G.R.: A new chaos-based fast image encryption algorithm. Appl. Soft Comput. **11**(1), 514–522 (2011)
18. Chen, G.R., Mao, Y., Chui, C.K.: A symmetric image encryption scheme based on 3D chaotic cat maps. Chaos Solitons Fractals **21**(3), 749–761 (2004)
19. Gao, T.G., Chen, Z.Q.: A new image encryption algorithm based on hyper-chaos. Phys. Lett. A **372**(4), 394–400 (2008)
20. Zhang, Y.S., Xiao, D.: Double optical image encryption using discrete Chirikov standard map and chaos-based fractional random transform. Opt. Lasers Eng. **51**(4), 472–480 (2013)
21. Xiao, D., Shih, F.Y.: Using the self-synchronizing method to improve security of the multi chaotic systems-based image encryption. Opt. Commun. **283**(15), 3030–3036 (2010)
22. Xiao, D., Liao, X.F., Wei, P.C.: Analysis and improvement of a chaos-based image encryption algorithm. Chaos Solitons Fractals **40**(15), 2191–2199 (2009)
23. Wang, Y., Wong, K.W., Li, C.B., Li, Y.: A novel method to design S-box based on chaotic map and genetic algorithm. Phys. Lett. A **376**(6–7), 827–833 (2012)
24. Peng, J., Jin, S.Z., Lei, L., Liao, X.F.: Construction and analysis of dynamic S-boxes based on spatiotemporal chaos. In: 11th IEEE International Conference on Cognitive Informatics & Cognitive Computing, Kyoto, Japan, 22–24 August 2012, pp. 274–278 (2012)
25. He, B., Luo, L.Y., Xiao, D.: A method for generating S-box based on iterating chaotic maps. J. Chongqing Univ. Posts Telecommun. (Nat. Sci.) **22**(1), 89–93 (2010)
26. Hussain, I., Shah, T., Gondal, M.A., Mahmood, H.: An efficient approach for the construction of LFT S-boxes using chaotic logistic map. Nonlinear Dyn. **71**(1–2), 133–140 (2013)
27. Hussain, I., Shah, T., Gondal, M.A., Mahmood, H.: Construction of new S-boxes over finite field and their application to watermarking. Z. Naturforsch. A, J. Phys. Sci. **67**(12), 705–710 (2012)
28. Zhang, Y.S., Xiao, D.: Cryptanalysis of S-box-only chaotic image ciphers against chosen plaintext attack. Nonlinear Dyn. **72**(4), 751–756 (2013)
29. Wang, Y., Liao, X., Xiang, T., Wong, K.W., Yang, D.G.: Cryptanalysis and improvement on a block cryptosystem based on iteration a chaotic map. Phys. Lett. A **363**(4), 277–281 (2007)
30. Wei, J., Liao, X.F., Wong, K.W., Zhou, T.: Cryptanalysis of a cryptosystem using multiple one-dimensional chaotic maps. Commun. Nonlinear Sci. Numer. Simul. **12**(5), 814–822 (2007)
31. Pei, W.J., Wang, K., Zou, L.H., Song, A.G., He, Z.Y.: On the security of 3D cat map based symmetric image encryption scheme. Phys. Lett. A **343**(6), 432–439 (2005)