SCIENCE CHINA
Information Sciences

• **Supplementary File** •

Spatial image encryption algorithm based on chaotic map and pixel frequency

Guodong YE[1,2*] & Xiaoling HUANG[1]

[1]*Faculty of Mathematics and Computer Science, Guangdong Ocean University, Zhanjiang* 524088*, China;*
[2]*College of Information Science and Electronic Engineering, Zhejiang University, Hangzhou* 310027*, China*

Appendix A Chaotic behavior for 3D cat map

3D cat map will be chaotic if the keys are set as x_0, y_0, $z_0 \in (0, 1)$. Figure A1 shows the chaotic behavior of this map with initial conditions $x_0 = 0.3124$, $y_0 = 0.5567$, and $z_0 = 0.8321$ (here, the first 100 iterated values are dropped to avoid transit effects [1]).

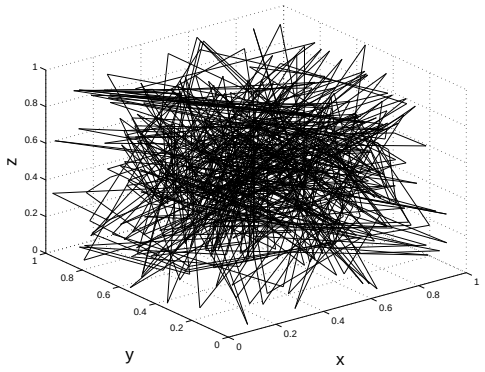

Figure A1 Chaotic behavior of the 3D cat map

Appendix B Decryption

The decryption process to the Algorithms 1 and 2 can be implemented in following Algorithms B1 and B2 respectively.

Algorithm B1 (Decryption to algorithm 1)

Input: image E, keys x_0, y_0, z_0.
Output: image P.
Let $P = zeros(m,\ n)$, $D = P$;
for i=m: -1: 1
Compute s using equation (C4) for D;
$s = floor(s \times 10^{14}) + i \bmod m$;
$H = T((i-1)n + 1 + s : in + s)$;
$P(i,\ :) = E(i,\ :) - H - E(i-1,\ :) \bmod 256$;
$D(i,\ :) = E(i,\ )$;
end

* Corresponding author (email: guodongye@hotmail.com, guodongye@gmail.com)

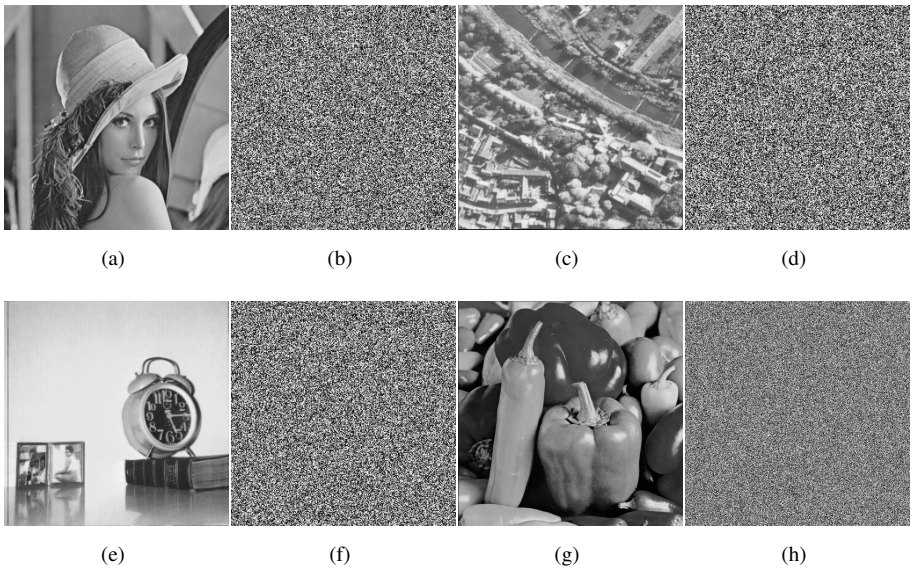

Figure C1 Tests: (a) plain-image of Lena, (b) cipher-image of (a), (c) plain-image of Aerial, (d) cipher-image of (c), (e) plain-image of Clock, (f) cipher-image of (e), (g) plain-image of Peppers, (h) cipher-image of (g).

Algorithm B2 (Decryption to algorithm 2)

Input: cipher-image C, keys x_0, y_0, z_0.
Output: image E.
Let $E = zeros(m,\ n)$, $D = E$;
for j=n: -1: 1
Compute s using equation (C4) for D;
$s = floor(s \times 10^{14}) + j \bmod n$;
$L = T((j-1)m + 1 + s :\ jm + s)$;
$E(:,\ j) = C(:,\ j) - L - C(:,\ j-1) \bmod 256$;
$D(:,\ j) = E(:,\ j)$;
end

Appendix C Simulation and analysis

In this section, simulation is performed using the proposed algorithm. Some common plain-images, shown in Figures C1(a), (c), (e), and (g), are randomly chosen for testing. With keys $x_0 = 0.5801$, $y_0 = 0.2378$, and $z_0 = 0.8414$, the corresponding cipher-images are displayed in Figure C1(b), (d), (f), and (h), respectively. The work was implemented by Matlab 2011b on a platform of Windows 7. To keep high security and avoid linearity, at least two rounds of confusion-diffusion should be implemented in encryption.

(1) Keyspace analysis

The 3D cat map is used to generate the chaotic sequences in both stages of permutation and diffusion. The key combinations for x_0, y_0, and z_0 reach 10^{42}, with the precision set to 10^{-14}. The proposed algorithm is secure enough to frustrate cryptanalysis in any brute-force attack.

(2) Statistical analysis

1) Correlation analysis

Correlation coefficient is usually used to test the randomness before and after encryption by the following equation:

$$r_{xy} = \frac{U((x - U(x))(y - U(y)))}{\sqrt{V(x)}\sqrt{V(y)}} \tag{C1}$$

where $U(x) = \frac{1}{N}\sum_{i=1}^{N} x_i$, $V(x) = \frac{1}{N}\sum_{i=1}^{N}(x_i - U(x))^2$, x and y represent the gray values of two adjacent pixels in the image, and N is the total number of the samples. The results for different images are given in Table C1 (PLena: Plain-image of Lena, CLena: Cipher-image of Lena, PBoat: Plain-image of Boat, CBoat: Cipher-image of Boat.). Clearly, the correlation coefficients are nearly zeros in the cipher-images, which can frustrate the confidence of attackers.

2) Histogram analysis

To see the pixel distribution of an image, histogram can supply a clear picture. A good algorithm should have an uniform histogram. Figure C2 shows the histogram tests for the Lena image of size 256×256. To evaluate further the quality of

Table C1 Correlation coefficients analysis

Directions	PLena	CLena	PBoat	CBoat
Horizontally	0.9419	0.0633	0.9801	0.0288
Vertically	0.9514	0.0478	0.9695	-0.0060
Diagonally	0.9817	0.0621	0.9588	0.0309

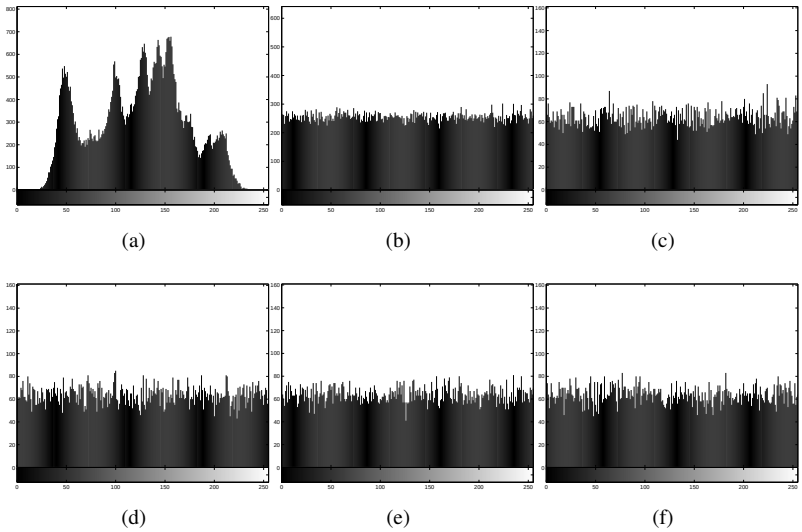

Figure C2 Histogram tests for gray Lena: (a) plain-image, (b) cipher-image; Blocks for cipher-image: (c) top-left, (d) top-right, (e) bottom-left, (f) bottom-right.

the cipher-image, block histogram is also employed to demonstrate the performance, as shown by Figure C2. All histogram test results show uniform distributions for pixels in the cipher-image.

3) Information entropy

Another rule of randomness test for images (messages) is the information entropy, computed by

$$I(\varphi) = \sum_{i=1}^{2^{Len}-1} p(\varphi_i) log_2 \frac{1}{p(\varphi_i)} \tag{C2}$$

where Len denotes the length of the pixel value in bits, φ is the message received, and $p(\varphi_i)$ is the probability of the symbol φ_i. Table C2 lists the results for different plain-images and their corresponding cipher-images. Clearly, the values for cipher-images are close to the theoretical value 8. Furthermore, Table C3 shows the results for color images.

(3) Sensitivity analysis

1) Key sensitivity

Any change in the key should lead to a wrong decrypted image with high sensitivity. Again, the gray Lena is taken to test the sensitivity performance. Figure C3 shows the wrong decryption results when a tiny change is added into x_0, y_0, and z_0. Similar test results on the gray Peppers are shown in Figure C3. Therefore, high key sensitivity can be achieved by the proposed algorithm.

2) Plain-image sensitivity

Commonly, UACI (unified averaged changed intensity) and NPCR (number of changing pixel rate) [4] are used to measure the sensitivity of plain-images. Test results are given in Table C4. Here, case-one denotes one bit shifting in a randomly

Table C2 Information entropy for different images

Images	Lena	Barb	Boat	Peppers
Plain-image	7.453238	7.466426	7.123758	7.571478
Cipher-image	7.997618	7.999282	7.999249	7.999263

Table C3 Information entropy for color images

Channels	R	G	B
Lena	7.9974	7.9970	7.9971
Baboon	7.9969	7.9972	7.9971
House	7.9972	7.9971	7.9972

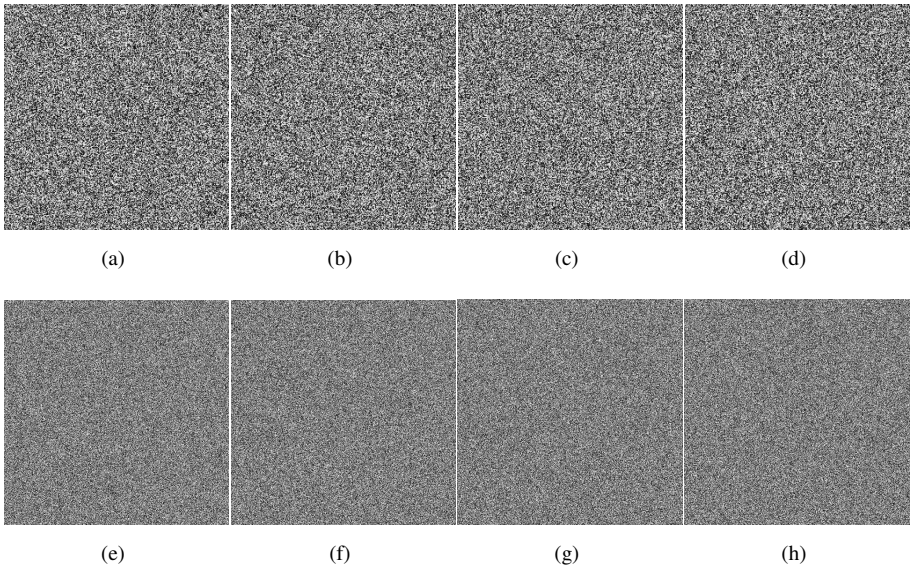

Figure C3 Key sensitivity tests for: Gray Lena: (a) cipher-image, (b) wrong decryption with $x_0 + 10^{-14}$, (c) wrong decryption with $y_0 + 10^{-14}$, (d) wrong decryption with $z_0 + 10^{-14}$; Gray Peppers: (e) cipher-image, (f) wrong decryption with $x_0 + 10^{-14}$, (g) wrong decryption with $y_0 + 10^{-14}$, (h) wrong decryption with $z_0 + 10^{-14}$.

Table C4 UACI and NPCR for Lena

Cases	Case-one	Case-two	Case-three	Case-four
UACI	33.5318	33.3524	33.6664	33.5637
NPCR	99.5987	99.6017	99.5926	99.6567

Table C5 Results for ISSD and BISSD

Images	Lena	Boat	Aerial	Baboon
ISSD	0.0024	0.0017	0.0012	0.0022
BISSD	0.0038	-0.0016	0.0006	0.0002

pixel at (58,108); case-two represents one bit shifting in two pixels at (116,67) and (213,170) synchronously but with the same summation; case-three means two bits shifting; case-four gives the results for changing three pixels simultaneously at (116,67), (213,170), and (44,142). The values are near the ideal values (UACI≈33.4635%, NPCR≈99.6094%) [5, 6].

(4) Image structure similarity degree

To evaluate image quality, image structure similarity degree (ISSD, or, SSIM) [7] can be applied as an objective method to quantify the visibility of differences between a distorted image and a reference image. It is defined as follows:

$$\mathrm{ISSD}(x,y)=\frac{(2\mu_x\mu_y+\varepsilon_1)(2\sigma_{xy}+\varepsilon_2)}{(\mu_x^2+\mu_y^2+\varepsilon_1)(\sigma_x^2+\sigma_y^2+\varepsilon_2)} \tag{C3}$$

where μ_x denotes the mean value of vector x; μ_y computes the mean of y; σ_x and σ_y return the standard deviation values for x and y respectively; $\varepsilon_1=(K_1L)^2$ and $\varepsilon_2=(K_2L)^2$ are positive numbers with small constant numbers $K_1\ll 1$ and $K_2\ll 1$, and L is the dynamic range of the pixel values ($L=255$ for 8-bit grayscale image). Here, ε_1 and ε_2 are used to avoid unstable results when either $\mu_x^2+\mu_y^2$ or $\sigma_x^2+\sigma_y^2$ is very close to zero [7]. Table C5 gives the test results for ISSD and BISSD (block ISSD for local evaluation) using the proposed algorithm, which shows that all values are near zero, implying that the cipher-image has a huge distortion to the plain-image.

(5) Know-plaintext and chosen-plaintext attacks

It is well known that to resist the known-plaintext and chosen-plaintext attacks, the keystreams generated by any algorithm should depend on the plain-image [8, 9]. In the confusion operation, equation (C4) is taken to calculate the properties for the plain-image, obtaining new initial keys x_0', y_0', and z_0'. As a result, the keystream can be generated with dependence on the plain-image. Furthermore, in Algorithms 1 and 2 for the diffusion operation, the choices of H and L are also related to the s computed by the confused image, respectively. Thus, both the production of the keystream in the confusion stage and the selection of the keystream in the diffusion stage are dependent on the plain-image. Consequently, the know-plaintext and chosen-plaintext attacks are infeasible to the proposed algorithm.

$$s=\frac{\sum_{i=1}^{256} r_i^2 v_i+1}{255\sum_{i=1}^{256} r_i^2+2} \tag{C4}$$

(6) Extra transmission analysis

To have the symmetric cipher structure, there should not be any extra transmission when communicating with the cipher-image, except for the secret keys shared by the sender and the receiver. In the proposed algorithm, after receiving the cipher-image, shared keys x_0, y_0, z_0 are used to iterate the 3D cat map to get T for the diffusion operation in encryption. Starting from the last column to the first column, the recovery of columns can be implemented as shown in Algorithm 3 by setting D be a zero matrix in the beginning. Then, matrix E can be obtained from the cipher-image C in a column-by-column manner. Similarly, the inverse diffusion operation for rows can also be carried out to E, to obtain a confused image P using Algorithm 4. After that, equation (2) is applied to P and s is formed again, which is employed to update the keys x_0, y_0, z_0 in the inverse confusion stage. Therefore, new keys x_0', y_0', z_0' are obtained, respectively, by equation (3). Then, iterate the 3D cat map to get random vectors p and q. As a result, the plain-image A can be recovered by inverse circular confusion (permutation) from P. Thus, all the intermediate parameters s, H, and L can be obtained in a self-adaptive way. That is, extra transmission is not needed in the new algorithm, except for secret keys.

(7) Randomness test using NIST

In this section, NIST 800-22 test [10] is employed to evaluate the true-random and pseudorandom numbers for the cipher-images obtained by the proposed algorithm. Images Peppers and Barb are randomly chosen for testing, with results listed in Table C6. It is clear that all P values are larger than 0.01. So, the proposed encryption algorithm can pass the randomness test with promising high security.

(8) Chi-square analysis

Table C6 Randomness test by NIST

Test items	P-value for Peppers	P-value for Barb	Results
Frequency	0.580656	0.735091	Pass
Block frequency	0.949503	0.576420	Pass
Cumulative sums	0.974349	0.657322	Pass
Runs	0.595737	0.401813	Pass
Longest run	0.355117	0.126734	Pass
Rank	0.923590	0.939736	Pass
FFT	0.638220	0.436477	Pass
Non-Overlapping Template	0.652599	0.434148	Pass
Overlapping Template	0.869997	0.098530	Pass
Universal	0.511340	0.757272	Pass
Approximate Entropy	0.612397	0.278902	Pass
Random Excursions	0.717785	0.300164	Pass
Random Excursions Variant	0.798159	0.215095	Pass
Serial	0.841614	0.064342	Pass
Linear Complexity	0.433072	0.931300	Pass

Table C7 Chi-square tests

Images	Lena	Peppers	Barb
$\chi^2_{256,0.05}$	294	294	294
χ^2_{test}	217	269	261
Pass or not	Pass	Pass	Pass

The uniform distribution for a message can also be proved by Chi-square [11], which is defined by (C5):

$$\chi^2_{test} = \sum_{i=1}^{k} \frac{(o_i - e_i)^2}{e_i} \tag{C5}$$

where, $k = 256$ (for a 8-bit gray image), o_i represents the observed occurrence frequency of each gray value, e_i denotes the expected occurrence frequency. By testing the proposed algorithm, Table C7 lists the results. It is noted that $\chi^2_{test} < \chi^2_{256,0.05}$ implies that the distribution is uniform [11]. Therefore, our method can pass the Chi-square test.

(9) Comparisons

In this section, information entropy, UACI, and NPCR are compared to show the encryption effect [3, 12, 13]. By taking color Lena image of size 256×256, with one round in encryption, the comparison results are shown in Table C8 for information entropy, in which the results obtained by the proposed algorithm are all larger than or equal to 7.9970. Table C9 displays the UACI and NPCR values with one-bit change in the gray Lena, while Table C10 lists the UACI and NPCR comparisons under the same pixel summation but with one bit change each at random positions (100, 101) and (200, 33) for different images of the same size 256×256. As to the running speed, Table C11 gives a comparison with two new methods reported in recent years. Less time is needed when using the proposed algorithm. Therefore, the proposed algorithm shows better performance by comparisons, especially for the invariance of pixel summation, for example [2].

References

1 Chai X L, Gan Z H, Lu Y, et al. A novel color image encryption algorithm based on genetic recombination and the four-dimensional memristive hyperchaotic system. Chinese Phys B, 2016, 25: 100503.

2 Wang X Y, Liu C M, Xu D H, et al. Image encryption scheme using chaos and simulated annealing algorithm. Nonlinear Dynam, 2016, 84: 1417-1429

Table C8 Comparisons for information entropy with color Lena

Channels	Ref. [2]	Ref. [5]	Ref. [14]	Ours
R	7.9884	7.9971	7.9968	7.9974
G	7.9894	7.9966	7.9975	7.9970
B	7.9894	7.9972	7.9983	7.9971
All $\geqslant$ 7.9970	No	No	No	Yes

Table C9 Comparisons with one bit change on gray Lena

Positions	(1, 1)	(180, 190)	(256, 256)
UACI of Ref. [2]	33.537316	33.537771	33.532308
UACI of Ours	33.393890	33.498775	33.543737
NPCR of Ref. [2]	99.624634	99.586487	99.620056
NPCR of Ours	99.639893	99.635315	99.555969

Table C10 Comparisons for UACI and NPCR with two bit changes

Images	Lena	Aerial	Clock
UACI of Ref. [2]	0.620021	0.423422	0.874658
UACI of Ours	33.534983	33.483534	33.402632
NPCR of Ref. [2]	78.820801	83.168030	27.757263
NPCR of Ours	99.618530	99.617004	99.603271

Table C11 Comparisons of speeds (unit: second)

Methods	[2]	[15]	Ours
256×256	0.934446	0.531963	0.240241
512×512	3.641063	2.088853	1.492930

3 Belazi A, El-Latif A A A, Belghith S. A novel image encryption scheme based on substitution-permutation network and chaos. Signal Process, 2016, 128: 155-170.
4 Guesmi R, Farah M A B, Kachouri A, et al. Hash key-based image encryption using crossover operator and chaos. Multimed Tools Appl, 2016, 75: 4753-4769.
5 Wang X Y, Teng L, Qin X. A novel colour image encryption algorithm based on chaos. Signal Process, 2012, 92: 1101-1108.
6 Hua Z Y, Zhou Y C. Image encryption using 2D Logistic-adjusted-Sine map. Inform Sciences, 2016, 339: 237-253.
7 Wang Z, Bovik A C, Sheikh H R, et al. Image quality assessment: from error visibility to structural similarity. IEEE T Image Prpcess, 2004, 13: 600-612.
8 Alvarez G, Li S J. Some basic cryptographic requirements for chaos-based cryptosystems. Int J of Bifurcat Chaos, 2006, 16: 2129-2151.
9 Xiao D, Wang L, Xiang T, et al. Multi-focus image fusion and robust encryption algorithm based on compressive sensing. Opt Laser Technol, 2017, 91: 212-225.
10 Cicek I, Pusane A E, Dundar G. An integrated dual entropy core true random number generator. IEEE T Circuits-II, 2017, 64: 329-333.
11 Chen J X, Zhu Z L, Fu C, et al. Reusing the permutation matrix dynamically for efficient image cryptographic algorithm. Signal Process, 2015, 111: 294-307.
12 Li Y P, Wang C H, Chen H. A hyper-chaos-based image encryption algorithm using pixel-level permutation and bit-level permutation. Opti Lasers in Eng, 2017, 90: 238-246. (Online)
13 Tong X J, Cui M G. Feedback image encryption algorithm with compound chaotic stream cipher based on perturbation. Sci China Inform Sci, 2010, 53: 191-202.
14 Hua Z Y, Zhou Y C, Pun C M, et al. 2D Sine Logistic modulation map for image encryption. Inform Sciences, 2015, 297: 80-94.
15 Eslami Z, Bakhshandeh A, An improvement over an image encryption method based on total shuffling. Opt Commun, 2013, 286: 51-55.