Multimedia Tools and Applications (2019) 78:20511–20531
https://doi.org/10.1007/s11042-018-6824-5

A novel fast image encryption algorithm for embedded systems

Zhihao Lin[1] · Jizhao Liu[1] · Jing Lian[1] · Yide Ma[1] · Xinguo Zhang[1]

Received: 3 May 2018 / Revised: 22 October 2018 / Accepted: 24 October 2018 /
Published online: 2 March 2019

Abstract
Nowadays, embedded systems can be found everywhere in daily life. In the development of embedded systems, data security is one of the critical factors. Encryption is an effective way to protect data from threats. Among encryption algorithms, chaos-based methods have strong cryptographic properties since chaotic systems are sensitive to initial conditions and parameters. However, most of these algorithms cannot be applied in practice because their encryption speed is not fast enough. In this paper, a fast image encryption algorithm is proposed. Compared with traditional chaos-based image encryption algorithms, the proposed method utilizes mixed-sequence and decorrelation operation to enhance the randomness of chaotic sequence. Moreover, it used minimum length of the sequence which is determined by experiments. Therefore, the proposed scheme spends much less computation time, which is an important advantage for being applied in practice. Testing results have shown that this algorithm has good performance in resisting known attacks, such as known-plaintext attacks, chosen ciphertext attacks, statistical attacks, differential attacks, and various brute-force attacks.

Keywords Fast image encryption · Chaos · Embedded system · Security · Decorrelation

1 Introduction

Internet networks especially mobile networks are rapid developed in last decade. People take and upload pictures with smart phones to share with their families and friends on social networking platform. Moreover, image and video transmission are extensively used in industrial scene supervision, E-government, military cooperative operations, telemedicine and distance learning. These sensitive information resources include personal privacy, trade secrets and even state secrets which shall not be disclosed to anyone not involved. Owing to frequent

✉ Yide Ma
yidema@gmail.com

[1] School of Information Science and Engineering, Lanzhou University, No. 222, TianShui Road (south), Lanzhou, China

flow of digital image and video across the world over the transmission media, it has become essential to secure them from leakages [18].

Data encryption is an effective way to protect data from threats. Due to bulky data capacity and high correlation among pixels in image files, traditional techniques are not suitable for image encryption [27]. In last few years, a great number of image encryption schemes have been proposed. Among encryption algorithms, chaos-based methods have strong cryptographic properties, with their high sensitivity to their initial conditions and parameters, their uncorrelated, their random-like nature and their unpredictability, can be very helpful in improving the security of transmitting communications [13]. Although chaos-based encryption schemes have very high security and reliability, high computation complexity is a serious obstacle to be widely applied in industrial production and daily life.

With development of the Internet, a large number of devices based on embedded systems are applied in different fields, from smart phones, intelligent recognition devices [10–12], industrial control systems, to military application like unmanned aerial vehicle (UAV) [3] and cooperative operations. Take smart phones as an example, data transmission of these devices is not safe enough, particularly when linking unsafe network such as public Wi-Fi. A network engineer can easily intercept data such as encrypted password and chat record from a network. Therefore, encryption of transmitted data plays a significant role in the defense of information disclosure. If use chaos-based encryption schemes with high security on devices based on embedded systems to encrypt transmitted information, the risk of information disclosure will be reduced drastically. Usually, embedded systems have small size and high stability, which is suitable to be used for many specific occasions. But compared with computer platform, embedded systems normally present much less power of computation and very limited memory size [15]. Since most of traditional chaos-based image encryption schemes have high computational complexity, low computing power of embedded systems limits time efficiency. Accordingly, more high-performance encryption schemes for embedded systems are needed.

In this paper, a fast image encryption algorithm which could be implemented to embedded systems is proposed. A continuous third-order hyperbolic sine chaotic system is chosen to generate chaotic sequence. The proposed method utilizes mixed-sequence and decorrelation operation to enhance the randomness of chaotic sequence. Moreover, it used minimum length of sequence determined by experiments. Therefore, the proposed scheme could significantly reduce computational complexity and processing time. This efficient algorithm is able to be applied in practice, like images transmission from UAV to ground control station [17, 22] and industrial control systems [6]. As for us, we are trying to utilize this scheme to a program of UAV with high performance of information security.

The rest of this paper is organized as follows. In section 2, entire encryption process will be detailed introduced. Section 3 presents the operations to enhance randomness of sequence. Section 4 is the experiments to determine the minimum length of chaotic sequence. And section 5 is the implementation in embedded systems. Section 6 displays the experimental results and performance analysis of encryption. Last section is conclusion.

2 Encryption algorithm

As is shown in Fig. 1. This paper uses a general encryption structure proposed by Jiri Fridrich [4].

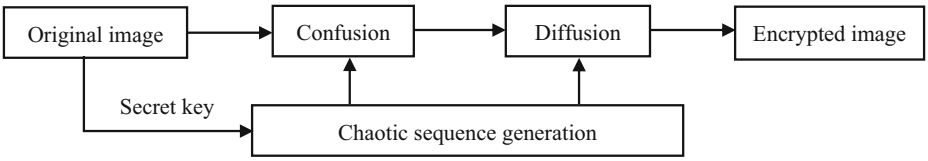


Fig. 1 Encryption process of algorithm

In the chaotic sequence generation step, a continuous third-order hyperbolic sine chaotic system is used, which is given by Ref [14]:

$$x + 0.75\ddot{x} + x + 1.2 \times 10^{-6} sinh\left(\frac{\dot{x}}{0.026}\right) = 0 \tag{1}$$

Lyapunov exponents characterize the rate of separation of infinitesimally close trajectories in state space as time tends to infinity [13]. Lyapunov exponents of this system was calculated by experiment under initial conditions of $(\dot{x}, \ddot{x}, x) = (0.1, 0.1, 0.1)$, results are as follow:

$$(\lambda_1, \lambda_2, \lambda_3) = \left(0.154851, 1.91018 \times 10^{-9}, -0.904851\right) \tag{2}$$

Where λ_1, λ_2, λ_3 are three Lyapunov exponents of this system. $\lambda_1 > 0$, $\lambda_2 \approx 0$, $\lambda_3 < 0$, which proves this system has strange attractor thus it is chaotic. Therefore, proposed chaos-based algorithm is convergent.

Encryption process is divided into two parts: confusion and diffusion. Confusion is a procedure of changing pixels' position of a digital image in particular order. We proposed to change the position of pixels by rows and columns instead of all pixels to the consideration of simplifying computation complexity. To have a better understanding of this procedure, a pseudo-code is listed in Appendix A. Diffusion aim to change the value of pixels. We proposed to use XOR operation as most papers do [16, 21, 23]. To have a better understanding of this procedure, a pseudo-code is listed in Appendix B.

3 Operations to enhance randomness of sequence

In this section, mixed-sequence and decorrelation operation is discussed to enhance the randomness of chaotic sequence. We used NIST to test chaotic sequence and choose the best result, which could reconcile low computational complexity and high security level.

3.1 Mixed-sequence

Pseudo-random is a significant property of chaotic system. However, sequence generated from $\dot{x}$ of Eq. (1) could not pass the pseudo-random test, which means this sequence was not random enough to be used in encryption. Therefore, encryption scheme utilizes this sequence may be easily cracked. In proposed algorithm, in order to obtain high uncorrelated sequence, we chose

elements from $\dot{x}$ and $\ddot{x}$ of Eq. (1) respectively to generate a mixed-sequence. The process is as follows:

$$\begin{cases} s(2n) = \dot{x}(2n) \\ s(2n+1) = \ddot{x}(2n+1) \end{cases} \tag{3}$$

Where $s(n)$ is the generated chaotic sequence.

3.2 Decorrelation operation

Decorrelation operation is an effective method to lower sequence correlation and then improve security. We decorrelated the chaotic sequence by following equations:

$$c(n) = s(n) \times 10^k - floor\left(s(n) \times 10^k\right) \tag{4}$$

As for the value of k, we carried out a series of experiments. We used NIST SP 800–22 test to test generated chaotic sequence. NIST test is used to detect the random characteristic of a pseudorandom sequence. If $p-value$ is greater than α, the tested sequence can pass the test. Usually, α is set to 0.01. We respectively tested $k = 0$~9. According to the experiments results, we found that when $k = 6$, sequence passed all the tests. Same results occurred when $k = 7$, 8, 9. In C language environment, long double, a data type with the longest significant digits, usually guarantees the accuracy of eighteen decimal places. If k is too big, the computational accuracy of different devices may hardly be guaranteed. In this case, the decrypted image may be different from plain image. Based on the principle of minimum data destruction, we chose $k = 6$ in this paper. Testing results have shown that the decorrelation operation has no effect on encryption scheme. Some of the experimental results are shown in Table 1 ($k = 0$), Table 2 ($k = 4$),Table 3 ($k = 6$) and Table 4 ($k = 8$).

Table 1 NIST SP 800–22 test ($k = 0$)

Test	*P* value	Result
Frequency	0.026525	Success
BlockFrequency	0.000028	Failure
Runs	0.513263	Success
LongestRun	0.107384	Success
Rank	0.004293	Failure
FFT	0.000092	Failure
NonOverlappingTemplate	0.000009	Failure
OverlappingTemplate	0.091876	Success
Universal	0.000005	Failure
LinearComplexity	0.473331	Success
Serial	0.715640	Success
ApproximateEntropy	0.071583	Success
CumulativeSums	0.331282	Success
RandomExcursions	0.134407	Success
RandomExcursionsVariant	0.228340	Success

Table 2 NIST SP 800–22 test ($k = 4$)

Test	P value	Result
Frequency	0.003496	Failure
BlockFrequency	0.058927	Success
Runs	0.513263	Success
LongestRun	0.341018	Success
Rank	0.356565	Success
FFT	0.839031	Success
NonOverlappingTemplate	0.113426	Success
OverlappingTemplate	0.114002	Success
Universal	0.005759	Failure
LinearComplexity	0.583945	Success
Serial	0.367623	Success
ApproximateEntropy	0.652735	Success
CumulativeSums	0.007888	Failure
RandomExcursions	0.413413	Success
RandomExcursionsVariant	0.388996	Success

4 Operations to reduce computational complexity

For most traditional chaos-based image encryption algorithms, one pixel of a digital image corresponds to one element of chaotic sequences, which guarantees the randomness and uniformity of distribution of encrypted image. A $m \times n$ length of sequence is needed for encrypting a digital image of size $m \times n$. However, for most images, length of sequence $m \times n$ usually reaches millions. Generation of such a long sequence would cost a lot of hardware resources and computation time. Therefore, these methods could hardly be applied in embedded systems or practice.

In this section, we proposed a simpler correspondence of pixels and sequence, an entire row or column of image correspond to one element of sequence. Details are shown in Appendix A and Appendix B. Furthermore, we consider chaotic sequence as a cyclic array, one element of sequence may be used more than once. Therefore, the length of sequence is significantly

Table 3 NIST SP 800–22 test ($k = 6$)

Test	P value	Result
Frequency	0.072340	Success
BlockFrequency	0.378525	Success
Runs	0.507790	Success
LongestRun	0.158383	Success
Rank	0.416271	Success
FFT	0.066240	Success
NonOverlappingTemplate	0.690672	Success
OverlappingTemplate	0.050298	Success
Universal	0.205584	Success
LinearComplexity	0.859249	Success
Serial	0.799248	Success
ApproximateEntropy	0.031193	Success
CumulativeSums	0.593591	Success
RandomExcursions	0.540358	Success
RandomExcursionsVariant	0.256494	Success

Table 4 NIST SP 800–22 test ($k = 8$)

Test	P value	Result
Frequency	0.670461	Success
BlockFrequency	0.396687	Success
Runs	0.351384	Success
LongestRun	0.091835	Success
Rank	0.821438	Success
FFT	0.048462	Success
NonOverlappingTemplate	0.801468	Success
OverlappingTemplate	0.838185	Success
Universal	0.733547	Success
LinearComplexity	0.077362	Success
Serial	0.412651	Success
ApproximateEntropy	0.455348	Success
CumulativeSums	0.136011	Success
RandomExcursions	0.560307	Success
RandomExcursionsVariant	0.328002	Success

reduced, as well as computation time. Because of decorrelation operation, the security of algorithm is still holding at a high level.

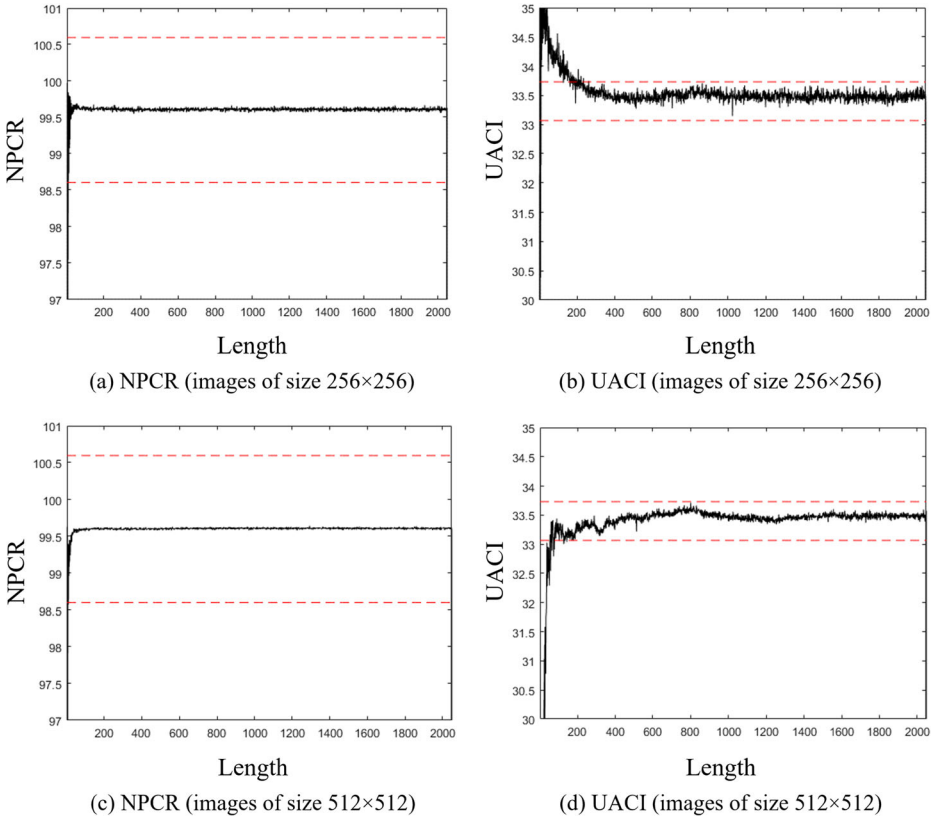
The shorter length of chaotic sequence the better efficiency of encryption performance. We did several experiments to determine the minimum length of sequence by calculating the effect on NPCR and UACI score, due to NPCR and UACI are two important factors to evaluate the performance of an encryption scheme. Experiments are formulated to test a specific hypothesis under different length of sequences. Theoretical value of NPCR is 99.6 and UACI is 33.4. We chose a critical value of 1%. The hypothesis says if fractional error of NPCR and UACI exceed the critical value, hypothesis of high-level security scheme is rejected. Otherwise, hypothesis is accepted. We calculated average value of NPCR and UACI under different length of sequences. Experimental results are shown in Fig. 2. Testing images are from USC-SIPI. Interval between two red dotted lines is 1% of the theoretical value of NPCR and UACI.

From Fig. 2, when length of sequence exceeds a certain range, values of NPCR and UACI will fluctuate up and down close to the theoretical. For image of size 256×256, when length exceeds 200, values of NPCR and UACI are in the interval. Same as image of size 512×512 when length of sequence exceeds 300. However, we chose longer length of 400 in case of some extreme situation.

In traditional encryption schemes, the length of sequence will be 65,536 and 262,144 for 256×256 images and 512×512 images respectively. Compared with these schemes, only 400 lengths of sequence are used in proposed scheme to achieve the same security level, which contribute much to low computational complexity and high time efficiency.

5 Implemented in embedded system

In this section, we implemented proposed algorithm in embedded system development board based on QT interface. We used NanoPC-T3 development board which contains a Samsung S5P6818 processor with a 7.0-in. touchable LCD screen. Fig. 3 shows the software architecture diagram of embedded system and Fig. 4 shows the Hardware architecture diagram.



(a) NPCR (images of size 256×256) (b) UACI (images of size 256×256)

(c) NPCR (images of size 512×512) (d) UACI (images of size 512×512)

Fig. 2 Effect of length on NPCR and UACI

5.1 Performance of development board

Detailed performance of embedded system development board is shown in Table 5. We not only ported Linux system but Qt/Embedded to the development board [9, 19]. Qt is a powerful cross platform GUI program development framework for C++, which provides modularization programming method and comprehensive graphical interfaces. Today, most handheld devices are developed based on Qt. Therefore, this algorithm has feature of powerful transplantable capability.

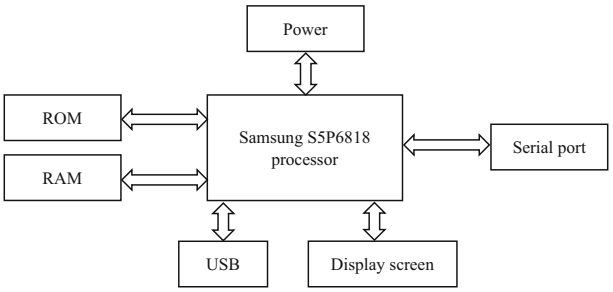


Fig. 3 Software architecture diagram

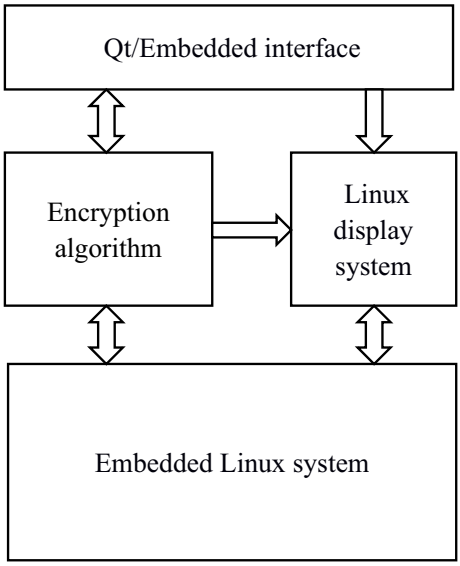


Fig. 4 Hardware architecture diagram

5.2 Image processing for development board

It is known to all that C language has no libraries for image processing. In order to process image, we complied Opencv 2.4.9 in Linux and ported libraries to the development board [26]. Furthermore, we connected a 7.0-in. touchable LCD screen to development board to achieve user interface interaction.

5.3 Encryption results

Encryption results can be seen in Fig. 5. Further encryption performance analyses are shown in next section.

6 Performance analyses

Experimental results and performance analyses of proposed encryption algorithm are provided in this section.

6.1 Encryption results

We used image Lena of size 512 × 512 and Cameraman of size 256 × 256 both with 256 Gy levels. Encryption results are shown in Fig. 6.

Table 5 Performance of embedded system development board

Clock Speed	RAM	ROM	System	Qt Version
400 MHz −1.4GHz	1GB DDR3	8GB eMMC	Debian 8.1	Qt/Embedded 2.2

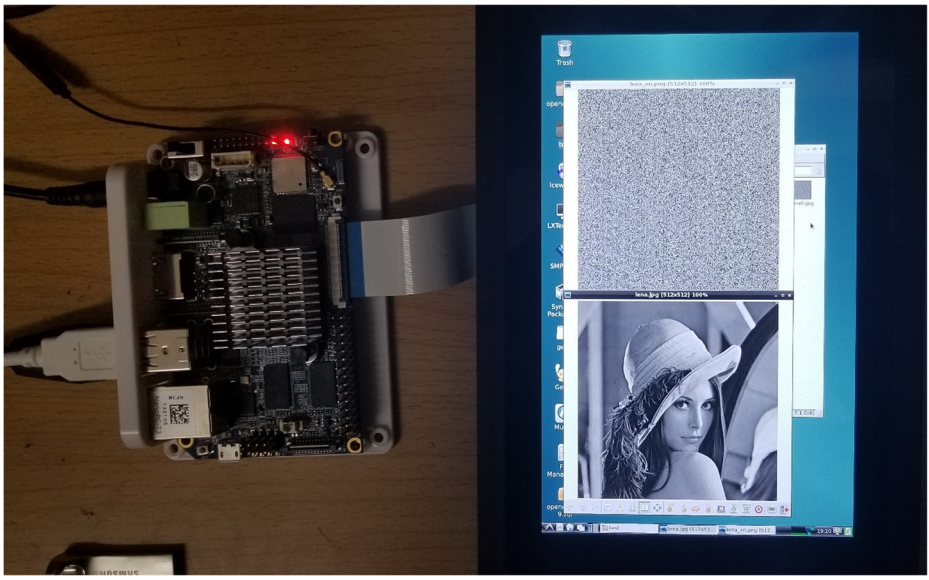


Fig. 5 Encryption results for embedded system

6.2 Key space analysis

According to Ref [1], a cryptosystem must have more than 2^{100} possible secret keys to defeat a brute-force attack. The proposed encryption algorithm is based on a continuous third-dimensional chaotic system. Consider limited precision in computer, the algorithm has at least $10^{48} \approx 2^{159}$ secret keys (account to 16 decimal places). Therefore, key space of scheme is huge enough to defeat a brute-force attack.

6.3 Gray-level histogram

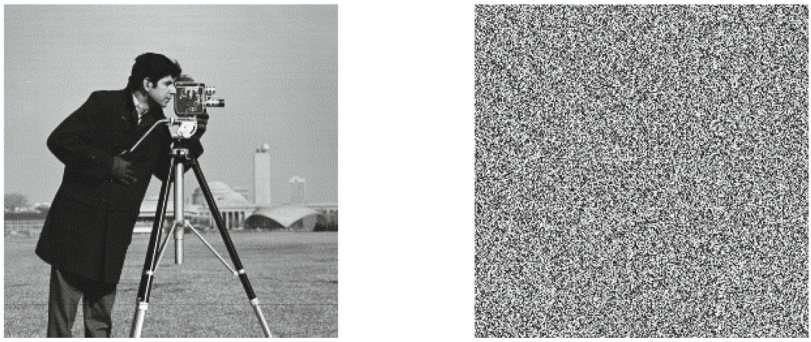
Gray-level histogram is a function showing the number of pixels in the image for each gray level [24]. It is a method of showing the random value of pixels directly. Fig. 7 shows gray-level histogram of original image and encrypted image. As we can see, number of pixels of each gray-level is close and pixels are uniform distribution.

6.4 Key sensitivity analysis

Since chaotic systems are sensitive to its initials, we changed its initials slightly and encrypt same image to find out the difference of changing initials [5]. For sensitivity analysis of initial conditions of $(\dot{x}, \ddot{x}, x) = (0.1, 0.1, 0.1)$, we set a slight change of $\Delta x = 10^{-15}$. The modified keys are $(\dot{x} + \Delta x, \ddot{x}, x) = \left(0.1 + 10^{-15}, 0.1, 0.1\right)$, $(\dot{x}, \ddot{x} + \Delta x, x) = \left(0.1, 0.1 + 10^{-15}, \ 0.1\right)$ and $(\dot{x}, \ddot{x}, x + \Delta x) = \left(0.1, 0.1, 0.1 + 10^{-15}\right)$. We used these four keys as input initials to encrypt same image of Fig. 6(b). It was shown that 99.6% of the pixels are different. Four ciphered images are shown in Fig. 8.

Therefore, difference between images encrypted by different keys is large enough to maintain high security against known-plaintext attack.

(a) Lena (b) Encrypted Lena



(c) Cameraman (d) Encrypted Cameraman

Fig. 6 Encryption performance of two examples

6.5 Information entropy

Information entropy is the expected value of the information contained in an image. A meaningless image often has big entropy [20]. For an image with 256 Gy levels, the maximum value of entropy is 8. Simulation experiment shows the entropy of Fig. 6(b) is 7.99899138 and (d) is 7.99538375. It proves the encrypted image is highly stochastic.

6.6 Correlation analysis

Generally, for a meaningful image, its adjacent pixels have intense correlation [7]. Its horizontal correlation, vertical correlation, and diagonal correlation are very close to 1. Thus, a reasonable encryption algorithm should destroy its pixels' correlation.

Correlation of two adjacent pixels in the original image and encrypted Lena image is shown in Fig. 9. Calculation results of horizontal correlation, vertical correlation, and diagonal correlation are shown in Table 6.

As we can see, proposed encryption algorithm effectively destroys the correlation of adjacent pixels.

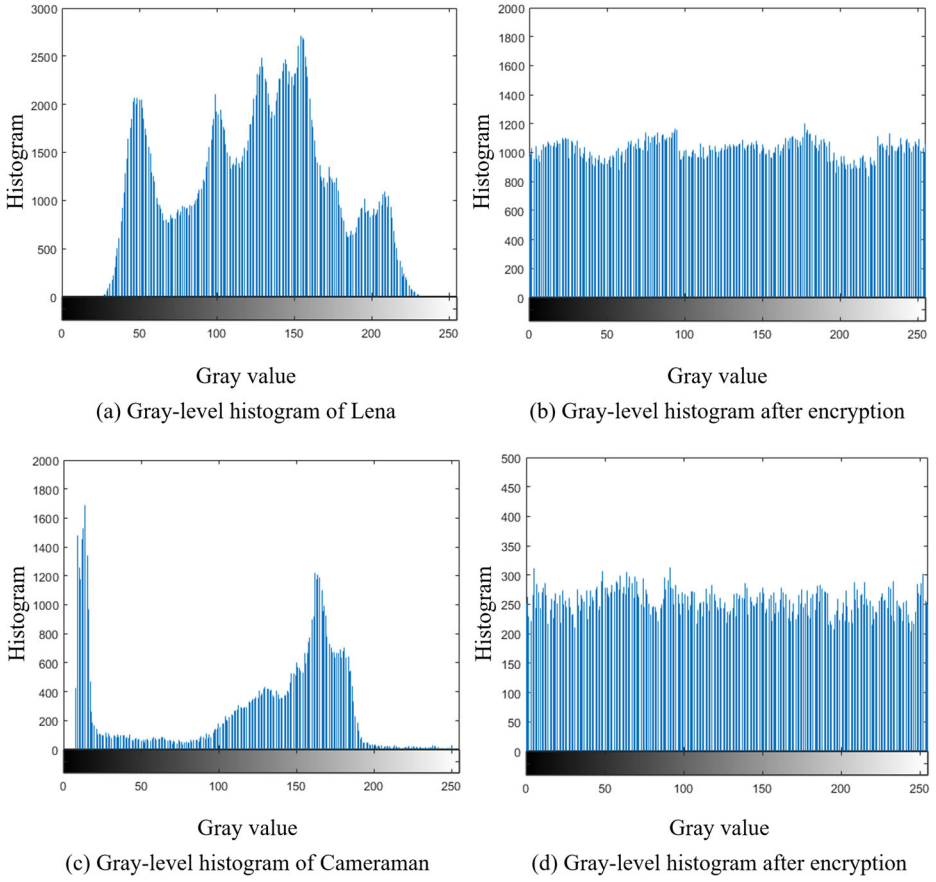


Fig. 7 Gray-level histogram of original image and encrypted image

6.7 NPCR and UACI

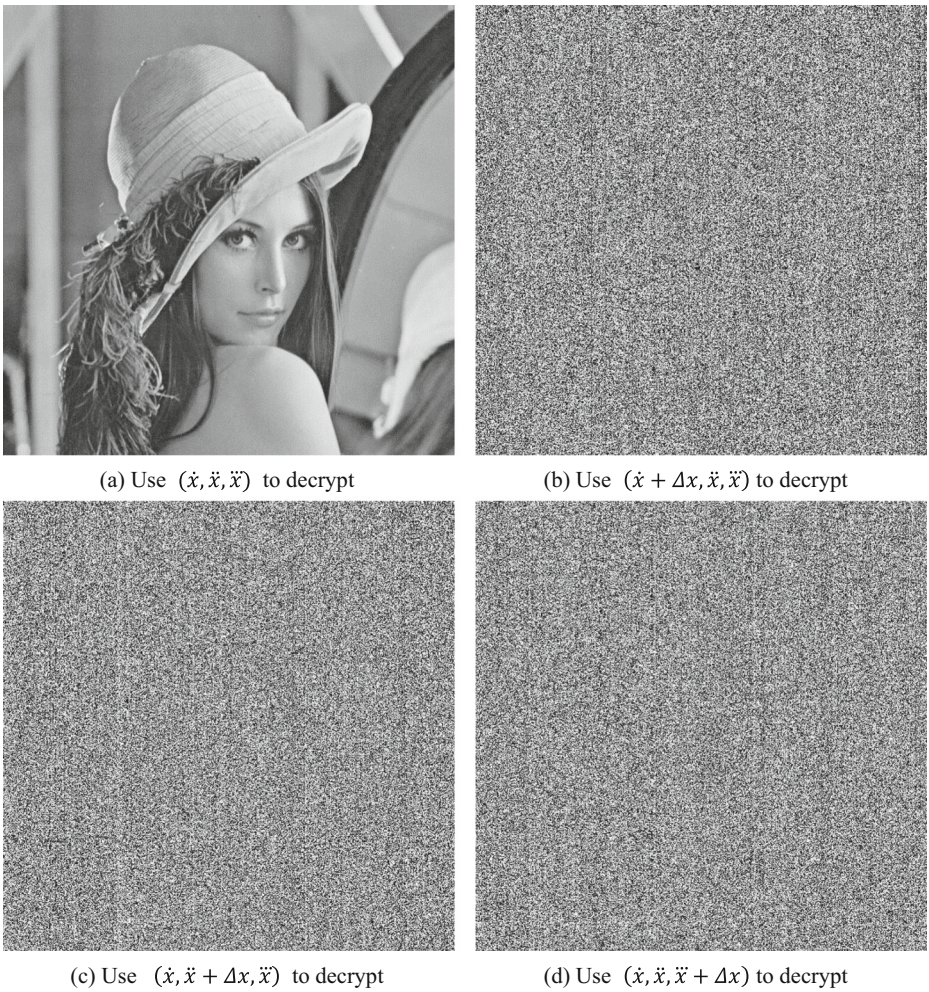
Number of pixels change rate (NPCR) and unified average changing intensity (UACI) are two important indices of resisting differential attacks [2, 25]. We changed the value of one pixel of original images randomly and encrypted them. Calculation results of NPCR and UACI are shown in Table 7. Testing images are from USC-SIPI and Caltech image database (492 images). Besides, the values of NPCR and UACI of images from database is the average of all images.

Apparently, proposed encryption algorithm is capable of resisting differential attacks.

6.8 Algorithm complexity analysis

According to Fig. 1, proposed algorithm contains four computational procedure, secret key computation, sequence generation, confusion and diffusion. The sum of frequency of all statements in algorithm can be obtained by adding computation of each part:

$$T(m,n) = m \times n + 400 \times 15 + 400 + 4 \times m \times n + m \times n \tag{5}$$



(a) Use $(\dot{x}, \ddot{x}, \dddot{x})$ to decrypt

(b) Use $(\dot{x} + \Delta x, \ddot{x}, \dddot{x})$ to decrypt

(c) Use $(\dot{x}, \ddot{x} + \Delta x, \dddot{x})$ to decrypt

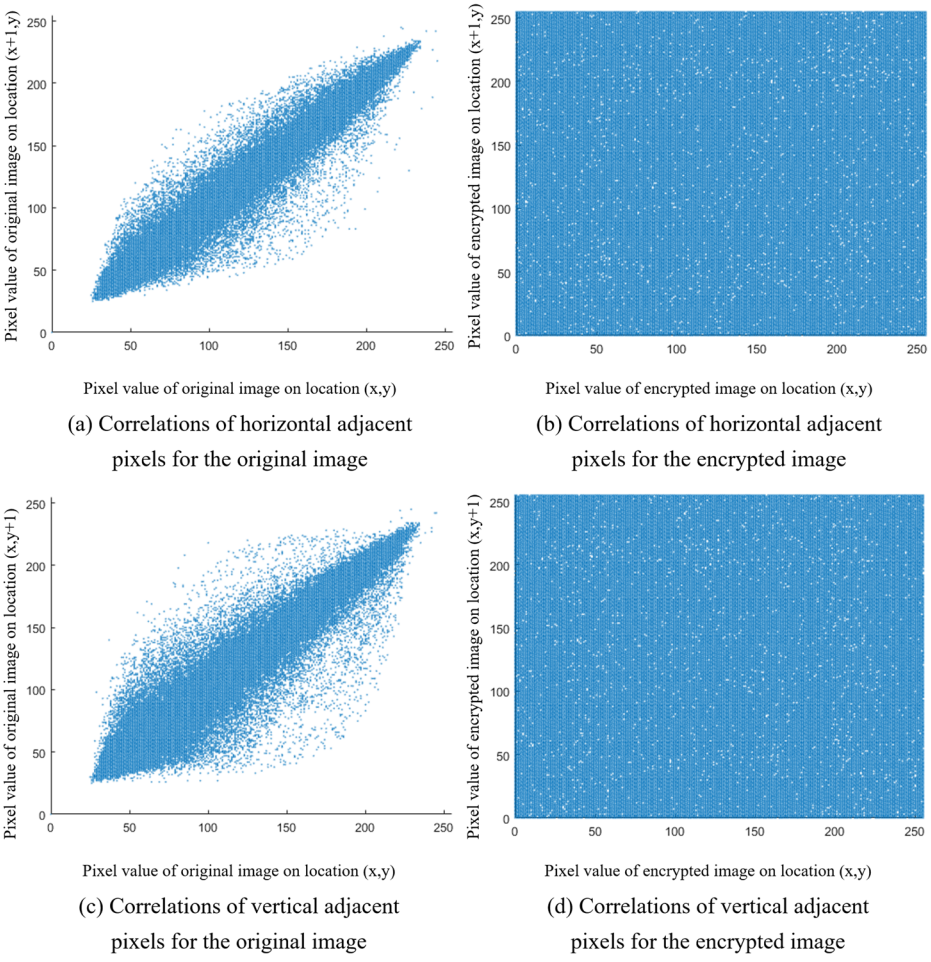
(d) Use $(\dot{x}, \ddot{x}, \dddot{x} + \Delta x)$ to decrypt

Fig. 8 Difference between two encrypted images

Where m, n are the size of original image. Hence the complexity of proposed algorithm is $O(6 \times m \times n)$.

6.9 Running time on PC

Encryption time, especially for chaos-based encryption algorithm, is a decisive factor of whether it can be used in practice. Running time of each encryption step are shown in Table 8. The computer configuration used in this test is a 2.4GHz Core i5-6300 U processor with 4GB memory. And the program is running in Visual Studio 2017 with Opencv 3.2.0. Testing images



(a) Correlations of horizontal adjacent pixels for the original image

(b) Correlations of horizontal adjacent pixels for the encrypted image

(c) Correlations of vertical adjacent pixels for the original image

(d) Correlations of vertical adjacent pixels for the encrypted image

Fig. 9 Correlation analysis for original and encrypted Lena image

are from USC-SIPI and Caltech image database (492 images). Besides, running time of images from database is the average of all images.

Therefore, proposed encryption algorithm is significantly efficient. An encryption of a 512 × 512 Gy image uses less than 40 ms (except the time of reading and saving image), which is an extraordinary short encryption time for a chaos-based encryption scheme.

Table 6 Results of horizontal correlation, vertical correlation, and diagonal correlation

Image	Horizontal correlation	Vertical correlation	Diagonal correlation
Lena	0.004639	0.006763	0.010818
Barbara	0.001906	0.004093	0.015255
Cameraman	0.008107	0.014632	0.037752
Couple	0.002843	0.006146	0.003127

Table 7 Results of NPCR and UACI

Image	NPCR	UACI
Lena	99.61624145	33.31693313
Barbara	99.59793091	33.39792588
Cameraman	99.59749512	33.60269583
Couple	99.63073730	33.49121094
SIPI database	99.60835480 (average)	33.38765596 (average)

6.10 Running time on embedded system

After a series of optimization, proposed algorithm can run on embedded system development board fast. The running time of each encryption step for embedded system is shown in Table 9. The embedded system development board is NanoPC-T3 which contains a Cortex -A53 Samsung S5P6818 processor, with a 7.0-in. touchable LCD screen. Testing images are from USC-SIPI and Caltech image database (492 images). Besides, running time of images from database is the average of all images.

6.11 Comparison results

We compared the performance of our scheme to several typical image encryption algorithms [8, 13, 28]. We evaluate the performance of our method by conducting several tests based on image quality and other evaluation metrics. To compare the correlation values, we use the following equation:

$$C = \frac{C_h + C_v + C_d}{3} \tag{6}$$

Ref [13] introduced a new simple chaotic system and its application in medical image encryption. Ref [28] described an image encryption method using a chaotic system of the mixed linear–nonlinear coupled map lattices for diffusion in the image encryption. Ref [8] proposed a novel image encryption scheme based on the Zaslavsky chaotic map.

The comparison of algorithm performance is shown in Table 10. Obviously, running time of proposed scheme is much shorter than others, which conforms to our main ideas. Such encryption method with high quality and efficient could be appropriately applied in practice. As for us, we are trying to utilize this scheme to a program of UAV with high performance of information security.

Table 8 Running time of each encryption step for PC

Image	Secret key computation	Sequence generation	Confusion	Diffusion
Lena	2 ms	2 ms	22 ms	2 ms
Barbara	2 ms	2 ms	21 ms	2 ms
Cameraman	2 ms	2 ms	14 ms	<1 ms
Couple	2 ms	2 ms	14 ms	<1 ms
SIPI database (512 × 512)	2.023 ms	2.025 ms	21.368 ms	2.036 ms
SIPI database (256 × 256)	1.883 ms	2.013 ms	13.699 ms	<1 ms

Table 9 Running time of each encryption step for embedded system

Image	Secret key computation	Sequence generation	Confusion	Diffusion
Lena	8 ms	4 ms	28 ms	8 ms
Barbara	8 ms	4 ms	29 ms	8 ms
Cameraman	2 ms	4 ms	16 ms	3 ms
Couple	2 ms	4 ms	15 ms	2 ms
SIPI database (512 × 512)	7.883 ms	4.323 ms	29.525 ms	8.354 ms
SIPI database (256 × 256)	2.178 ms	4.424 ms	15.882 ms	2.658 ms

Table 10 Comparison of the proposed image encryption scheme with recent state-of-the-art encryption algorithms in same environment (test image is image Lena of size 512 × 512 with 256 Gy levels)

Method name	Key space	Entropy	Correlation coefficient	NPCR	UACI	Running time
Ideal value	$>2^{100}$	≈8	≈0	≈99.6	≈33.4	N/A
Proposed scheme	$>2^{159}$	7.9989	0.0040	99.6162	33.3979	30ms
Ref [13]	$>2^{300}$	7.9969	0.0025	99.6140	33.4805	15.268s
Ref [1]	$>2^{400}$	N/A	0.0006	99.7826	33.4964	26.452s
Ref [24]	2^{711}	7.9978	0.0031	99.6100	33.5000	34.480s

7 Conclusion

A novel fast image encryption algorithm based on chaos for embedded systems is proposed in the paper. The proposed scheme can encrypt digital images with high time efficiency. Mixed-sequence and decorrelation operation are also utilized to enhance the randomness of chaotic sequence. Moreover, the algorithm used minimum length of the sequence which is determined by experiments. Therefore, it spends much less computation time and can be implemented in embedded systems. Performance analysis including key space analysis, gray-level histogram, different initials analysis, information entropy, correlation analysis, NPCR, UACI, complexity analysis and running time. All the experimental results show that proposed encryption scheme can resist known attacks, such as known-plaintext attacks, chosen ciphertext attacks, statistical attacks, differential attacks, and various brute-force attacks. Besides, it performs well in embedded systems thereby can be applied in programs.

All these properties make the proposed scheme an efficient encryption algorithm with high security. However, Cortex -A53 processor we used in embedded system is a relatively powerful one among embedded processors. Performance of proposed algorithm in some systems with low efficiency is not as good as we expect. In the future, we aim to find more approaches to keep on reducing computational complexity without effecting the security. Additional future work could explore to video encryption for more application scenarios.

Acknowledgements Thanks for the useful suggestions provided by Yide Ma and Jizhao Liu. This study was supported by the Fundamental Research Funds for the Central Universities (No.lzujbky-2016-238). National Natural Science Foundation of China (No.61175012). 2017 s batch of innovation base and innovative talents (Small and medium enterprises innovation fund 17CX2JA018).

Appendix A Process of confusion

Input: Original image Org_Img, Length of Chaotic sequence k, Chaotic sequence c(k).
Output: Image En_Con_r after confusion operation.

```
[m,n] ← size(Org_Img)
u ← 0                                   % u is a count variable
for j=1:1:m
   u ← u + 1
   if u > k
      u ← u - k
   end
   cm(j) ← c(u)*10^6                    % Decorrelation of chaotic sequence
   cm(j) ← cm(j)-floor(cm(j))           % cm is the sequence used in column transformation
end
En_Con_c ← zeros(m,n,'uint8')           % Create a new blank image to save the data after
                                        % column transformation
[x1,temp_col] ← sort(cm)                % Sorting cm by ascending order, x1 is the results
                                        % Another sequence temp_col is the same size as cm
                                        % and describes the arrangement of the elements of cm
                                        % into x1 along the sorted dimension.
for i=1:1:m
   for j=1:1:n
      En_Con_c(i,j) ← Org_Img(temp_col(i),j)        % Column transformation
   end
end
for j=1:1:m
   u ← u + 1
   if u > k
      u ← u - k
   end
   cn(j) ← c(u)*10^6                    % Decorrelation of chaotic sequence
   cn(j) ← cn(j)-floor(cn(j))           % cn is the sequence used in row transformation
end
En_Con_r ← zeros(m,n,'uint8')           % Create a new blank image to save the data after
                                        % row transformation
[x2,temp_row] ← sort(cn)                % Sorting cn by ascending order, x2 is the results
                                        % Another sequence temp_row is the same size as cn
                                        % and describes the arrangement of the elements of cn
                                        % into x2 along the sorted dimension.
for j=1:1:n
   for i=1:1:m
      En_Con_r(i,j) ← En_Con_c(i,temp_row(j))       % Row transformation
   end
end
```

Appendix B process of diffusion

Input: Image En_Con_r after confusion operation, Length of Chaotic sequence k, Chaotic sequence c(k).
Output: Encryption image En_Img.

```
[m,n] ← size(En_Con_r)

En_Img ← zeros(m,n,'uint8')          % Create a new blank image to save the data after diffusion

for i=1:1:k

   cd(i) ← c(i)*10^8                 % cd is the sequence used in diffusion operation

   cd(i) ← floor(cd(i)) & 255        % This step ensure cd(i) is between 1 to 256

end

u ← floor(k/2)                       % u is a count variable

for i_row=1:1:m
   for i_col=1:1:n

      u ← u + 1

      if u > k

         u ← 1

      end

      En_Img (i_row,i_col) ← bitxor(c(u),En_Con_r(i_row,i_col))      % Diffusion operation

   end
end
```

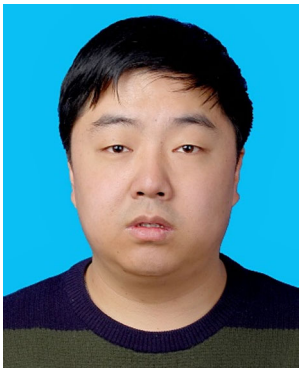

References

1. Alvarez G, Li S (2006) Some basic cryptographic requirements for chaos-based cryp- tosystems. Int J Bifurcation Chaos 16:2129–2151
2. Chen G, Mao YB, Chui CK (2004) A symmetric image encryption scheme based on 3D chaotic cat maps. Chaos, Solitons Fractals 21(3):749–761
3. chuan Y, Xijun WANG, GAO Yongke LI, Qingmin TIAN (2015) A new scheme for UAV TT&C and information transmission system. Int Conf Adv Mech Eng Indust Inform (AMEII) 2015(2015):1018–1022
4. Fridrich J (1998) Symmetric ciphers based on two-dimensional chaotic maps. Int J Bifurcation Chaos 8(6): 1259–1284

5. Galatolo S (2003) Complexity, initial condition sensitivity, dimension and weak chaos in dynamical systems. Nonlinearity 16(4):1219–1238
6. Reinhard Gerndt, Sören Michalik, Stefan Krupop (2011) Embedded vision system for robotics and industrial automation, 2011 9th IEEE International Conference on Industrial Informatics. 895–899
7. Guo W, Zhao J, Ye R (2014) A Chaos-based pseudorandom permutation and bilateral diffusion scheme for image encryption, international journal of image. Graph Sign Proc 6(11):50–61
8. Hamza R, Titouna F (2016) A novel sensitive image encryption algorithm based on the Zaslavsky chaotic map. Inform Sec J A Glob Perspect 25(4–6):162–179
9. Li Y, Wang YL (2015) Customizing dynamic libraries of Qt based on the embedded Linux. Appl Mech Mater 740:782–785
10. Liu Y, Nie L, Han L, Zhang L, David S (2015) Rosenblum, Action2Activity: recognizing complex activities from sensor data. Proc Twenty-Fourth Int Joint Conf Artificial Intel (IJCAI) 2015:1617–1623
11. Liu Y, Zhang L, Nie L, Yan Y, Rosenblum DS (2016) Fortune teller: predicting your career path. Proc Thirtieth AAAI Conf Artific Intel (AAAI-16):201–207
12. Liu Y, Nie L, Liu L, Rosenblum DS (2016) From action to activity: sensor-based activity recognition. Neurocomputing 181:108–115
13. Liu J, Ma Y, Li S, Lian J, Xinguo Zhang A (2018) New simple chaotic system and its application in medical image encryption. Multimed Tools Appl 4:1–22
14. Liu J, Sprott JC, Wang S, Ma Y (2018) Simplest chaotic system with a hyperbolic sine and its applications in DCSK scheme. IET Commun
15. Malinowski A, Yu H (2011) Comparison of embedded system Design for Industrial Applications. IEEE Trans Indust Inform 7(2):244–254
16. Murillo-Escobar MA, Cruz-Hernández C, Abundiz-Pérez F, López-Gutiérrez RM (2016) Implementation of an improved chaotic encryption algorithm for real-time embedded systems by using a 32-bit microcontroller. Microprocess Microsyst 45:297–309
17. Mykhatsky OYu , KuzmenkoNS , Savchenko OV (2013) Experimental UAV flight data structuring transmission and visualization by narrowband telemetry transmission, 2013 IEEE 2nd International Conference Actual Problems of Unmanned Air Vehicles Developments Proceedings (APUAVD). 118–121
18. Pareek NK, Vinod Patidar KKS (2006) Image encryption using chaotic logistic map. Image Vis Comput 24: 926–934
19. Song R, Lang WC, Pan SW (2009) Development of embedded system GUI based on Qt/embedded. Appl Mech Mater 109:586–590
20. Talarposhti KM, Jamei MK (2016) A secure image encryption method based on dynamic harmony search (DHS) combined with chaotic map. Opt Lasers Eng 81:21–34
21. Wong KW, Kwok SH, Law WS (2008) A fast image encryption scheme based on chaotic standard map. Phys Lett A 372(15):2645–2652
22. Xu Z, Wu L, Gerke M, Wang R, Yang H (2016) Skeletal camera network embedded structure-from-motion for 3D scene reconstruction from UAV images. ISPRS J Photogramm Remote Sens 121:113–127
23. Xu H, Tong X, Meng X (2016) An efficient chaos pseudo-random number generator applied to video encryption. Optik - Int J Light Electron Optics 127(20):9305–9319
24. Yao Y, Zhang W, Yu N (2017) Inter-frame distortion drift analysis for reversible data hiding in encrypted H.264/AVC video bitstreams. Signal Process 132:19–28
25. YAOBIN MAO, GUANRONG CHEN, SHIGUO LIAN (2004) A novel fast image encryption scheme based on the 3D chaotic baker map. Int J Bifurcat Chaos 14(10):3613–3624
26. Yin SJ, Xia DC, Ma J (2014) Application of machine vision in ARM based on OpenCV and Qt. Adv Mater Res 1049-1050(1):1938–1941
27. Zhang G, Liu Q (2011) A novel image encryption method based on total shuffling scheme. Opt Commun 284(12):2775–2780
28. Zhang YQ, Wang XY (2014) A symmetric image encryption algorithm based on mixed linear–nonlinear coupled map lattice. Inf Sci 273(8):329–351

Zhihao Lin received the B.S. degree from South China University of Technology, Guangzhou, China, in 2016. He is currently pursuing the M.S. degree at Lanzhou University, Lanzhou, China. His research activities and interests include secure communications, image encryption and embedded systems.

Jizhao Liu received the B.S. and M.S. degree from Lanzhou University, Lanzhou, China, in 2012 and 2015, respectively. He is currently pursuing the Ph.D. degree at Lanzhou University. His research activities and interests are currently focused on chaotic system, secure communications and image encryption,



Jing Lian received the B.S. and M.S. degrees in communication and information engineering from Lanzhou Jiaotong University, Gansu, China, in 2005 and 2010, respectively. He is currently pursuing the Ph.D. degree in radio physics at Lanzhou University. His current research interests include artificial neural networks, image processing, object recognition.

Yide Ma received the B.S. and M.S. degrees in radio technology from the University of Electronics Science and Technology, Chengdu, China, in 1984 and 1988, respectively, and the Ph.D. degree from the Department of Life Science, Lanzhou University, Lanzhou, China, in 2001. He is currently a Professor with the School of Information Science and Engineering, Lanzhou University. He has authored over 150 papers in major journals and international conferences and several textbooks, including Applications of Pulse-Coupled Neural Networks (Springer\& Higher Education Press (Beijing, China), 2010, in English) and Principle and Application of Microcomputer (Beijing, China: Higher Education Press, 2011, in Chinese). His current research interests include artificial neural networks, digital image processing, pattern recognition, digital signal processing, and computer vision.

Xinguo Zhang received the B.S. degree from Lanzhou University, Lanzhou, China. He is retired from Lanzhou University, before that, he is a senior engineer in School of Information Science and Engineering, Lanzhou University, Lanzhou. His research interests include the design and analysis of nonlinear circuits and nonlinear signal processing.