A Novel Image Encryption Scheme Using Josephus Permutation and Image Filtering

Binxuan Xu, Zhongyun Hua, and Hejiao Huang(✉)

School of Computer Science and Technology, Harbin Institute of Technology Shenzhen Graduate School, Shenzhen 518055, China
huanghejiao@hit.edu.cn

Abstract. To efficiently protect digital images, this paper proposes a novel image encryption scheme using the concepts of Josephus permutation and image filtering. It adopts the well-known architecture of confusion and diffusion. The Josephus permutation is designed to achieve the confusion property by fast separating adjacent pixels into different rows and columns, while the image filtering is to obtain the diffusion property by spreading tiny change in plain-image to the whole cipher-image. Simulation results demonstrate that the proposed image encryption scheme can encrypt different kinds of images into noise-like cipher-images. Security analysis shows that it has high security level and can outperform several other image encryption schemes.

Keywords: Image encryption · Image filtering · Josephus permutation · Security analysis

1 Introduction

Nowadays, more and more digital data are produced by various digital devices and transmitted through all kinds of networks. A large portion of these digital data is represented by image format, because image can display information in a visualized way. Thus, image security attracts increasing attentions [1–3]. Among all kinds of image security technologies, image encryption is an efficient way that transforms digital images into unrecognized formats. Only with the correct key, can one recover the original image information [4–6].

One strategy of encrypting images is to treat digital images as data sequences, and then encrypt them using traditional data encryption technologies, such as the Advanced Encryption Standard (AES) [7]. However, every pixel of digital images may be represented by 8 or more bits. Treating image as data sequence can't sufficiently utilize the property of image pixel, and thus may result in low encryption efficiency. To address this problem, many image encryption algorithms considering image properties have been developed [4,8]. Among all these image encryption technologies, the chaos-based encryption is one of the most popular technologies, because chaotic systems have many inner properties that are similar with encryption. In [9], Zhou *et al*. developed a novel image encryption using the combination of chaotic systems. In [10], an image encryption

S. Wen et al. (Eds.): CSS 2017, LNCS 10581, pp. 307–319, 2017.
https://doi.org/10.1007/978-3-319-69471-9_23

algorithm using a two-dimensional chaotic system was designed. For these chaos-based image encryption algorithms, their security performance are highly dependent on the chaos performance of the used chaotic systems. Because chaotic systems have unstable properties when implemented in digital platforms, their developed image encryption algorithms can't obtain stable high security [11,12]. Thus, it is desirable to develop image encryption schemes with high security levels using other technologies.

This paper proposes a new image encryption scheme using Josephus permutation and image filtering. It strictly follows the structure of confusion and diffusion. The Josephus permutation, derived from the theory of Josephus problem [13], can achieve the confusion property by first selecting a set of pixels in image, and then shuffling these pixels into different rows and columns. With a random mask generated from secure key, the image filtering can randomly change a pixel value using its adjacent pixels, and thus can obtain confusion property. After two rounds of Josephus permutation and image filtering, a natural image can be encrypted into a random-like one. Simulation results prove the efficiency of the proposed encryption scheme in encrypting different images. Security analysis demonstrates that it can protect digital images with high security levels.

The rest of this paper is organized as follows. Section 2 presents the proposed image encryption scheme. Section 3 simulates it using different kinds of images and discusses its properties. Section 4 analyzes its security from different aspects and Sect. 5 concludes this paper.

2 Proposed Encryption Scheme

This section introduces the proposed image encryption scheme. Its structure is shown in Fig. 1. As can be seen, the secure key is to generate parameters for the Josephus permutation and image filtering. The Josephus permutation, derived from the well-known Josephus problem [13], is to randomly separate image pixels to different rows and columns, while the image filtering is used to randomly change image pixel values. After two rounds of Josephus permutation and image filtering, a meaningful image can be encrypted as a random-like cipher-image. The rest of this section will present each of these operations in detail.

2.1 Parameter Generation

The secure key $\mathbf{K}$ is of length 256 bits and it consists of four parts, $\mathbf{K} = \{\mathbf{f}_1, \mathbf{f}_2, \mathbf{s}_1, \mathbf{s}_2\}$. Among the four parts, $\mathbf{f}_1$ and $\mathbf{f}_2$ are original initial components, while $\mathbf{s}_1$ and $\mathbf{s}_2$ are distribution parameters to enlarge the key space. Each of them is of length 64 bits. The two sub-keys, $\mathbf{k}_1$ and $\mathbf{k}_2$ for the two encryption rounds can be generated by the equations

$$\mathbf{k}_i = \mathbf{f}_i \oplus \mathbf{s}_i, \tag{1}$$

where $i \in \{1, 2\}$. The sub-key also has the length of 64 bits and it generates parameters for the Josephus permutation and image filtering.

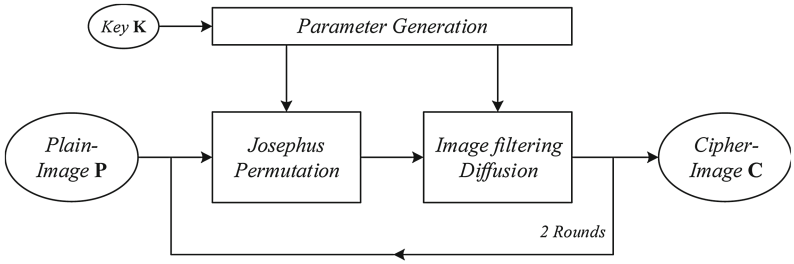

Fig. 1. Structure of the proposed image encryption scheme.

2.2 Josephus Permutation

As high adjacent correlations may exist in natural images, an image encryption scheme should have the ability to decorrelate these correlations. To achieve this property, We designed a Josephus permutation in our encryption scheme according to the Josephus problem [13]. It can randomly select pixels from an image and permutate them to different rows and columns.

Suppose the image to be encrypted is of size $M \times N$, a set of parameters is first generated using the 64-bit sub-key $\mathbf{k}$,

$$\begin{aligned} MP &= (\text{bi2de}(\mathbf{k}(1:8)) \mod M) + 1 \\ NP &= (\text{bi2de}(\mathbf{k}(9:16)) \mod N) + 1 \\ CP &= (\text{bi2de}(\mathbf{k}(25:32)) \mod N) + 1 \\ MStep &= \text{bi2de}(\mathbf{k}(17:20) + 1 \\ NStep &= \text{bi2de}(\mathbf{k}(21:24) + 1 \\ CStep &= \text{bi2de}(\mathbf{k}(33:36) + 1 \end{aligned} \tag{2}$$

where $[MP, NP]$ is the initial pixel position, $MStep$ is the row skipped step and $NStep$ is the column skipped step, CP denotes the column position that the first selected pixel will move to, and $CStep$ is the skipped step of CP. The whole procedure of the Josephus permutation can be described as follows,

Step 1: Initialize an M-dimensional vector $\mathbf{A}$ and two N-dimensional vectors $\mathbf{B}$ and $\mathbf{C}$, and assign $\mathbf{A}(i) = i$, $\mathbf{B}(j) = \mathbf{C}(j) = j$;

Step 2: A set of N pixels is selected as follows: (1) set MP as their row positions, remove MP from $\mathbf{A}$, cyclic skip $MStep$ in $\mathbf{A}$, and assign the next number to MP; (2) set the column position of first pixel as NP, and remove NP from $\mathbf{B}$; (3) starting from the previously removed number, cyclic skip $NStep$ cells in $\mathbf{B}$ and assign the next number to the column position of second pixel; (4) repeat the operation (3) until all the column positions of N pixels are determined; (5) assign the last selected column position to NP;

Step 3: Repeat *Step 2* until M sets of N pixels are recorded;

Step 4: Initialize an N-dimensional vector $\mathbf{CI}$ and assign it as follows: (1) set $\mathbf{CI}(1) = CP$ and remove CP from $\mathbf{C}$; (2) starting from the previously

removed number, cyclic skip $CStep$ cells in $\mathbf{C}$ and assign the next number to $\mathbf{CI}(2)$; (3) repeat the operation (2) until all the N numbers in $\mathbf{C}$ have been assigned to $\mathbf{CI}$;

Step 5: For the N pixels in i-th set, permute them into the positions $\{(i, \mathbf{CI}(1)), (i+1, \mathbf{CI}(2)), \cdots, (i+N, \mathbf{CI}(N))\}$. Note that if $i+j$ is bigger than M, we use its modulus result to M instead.

Algorithm 1 shows the detailed procedure of the Josephus permutation.

Algorithm 1. Josephus permutation

Input: The plain-image $\mathbf{P}$ with size of $M \times N$, parameters MP, NP, CP, $MStep$, $NStep$, and $CStep$ calculated by Eq. (2).
Output: The permutation result $\mathbf{I}$.

$\mathbf{A}$=1:M; $\mathbf{B}$=1:N; $\mathbf{C}$=1:N;
Set $\mathbf{CI} \in \mathbb{N}^{1\times N}$, $NP\_new = NP$.
for $k = 1$ to N **do**
  $\mathbf{CI}(k) = \mathbf{C}(CP)$;
  $\mathbf{C}(CP) = [\,]$;
  $CP = ((CP - 2 + CStep) \mod (N-k)) + 1$;
  $CStep = CStep + 1$;
end for
for $i = 1$ to M **do**
  $row = \mathbf{A}(MP)$;
  $\mathbf{A}(MP) = [\,]$;
  $NP = NP\_new$;
  $\mathbf{B} = 1 : N$;
  for $j = 1$ to N **do**
    $column = \mathbf{B}(NP)$;
    $\mathbf{I}(((i+j-2) \mod M) + 1, CI(j)) =$
    $\mathbf{P}(row, column)$;
    $NP\_new = \mathbf{B}(NP)$;
    $\mathbf{B}(NP) = [\,]$;
    $NP = (NP - 2 + NStep) \mod (N-j) + 1$;
    $NStep = NStep + 1$;
  end for
  $MP = (MP - 2 + MStep) \mod (M-i) + 1$;
  $MStep = MStep + 1$;
end for

2.3 Image Filtering

An image encryption algorithm should have the diffusion property, which means that slight change in plain-image can cause the change of all the pixels in cipher-image. To obtain this property, we use the concept of image filtering to randomly change pixel values in our image encryption algorithm. The image filtering is widely used in image processing, such as image denoising and edge detection. It does convolution operation using a mask with adjacent pixels. By this way, the current pixel can be affected by its adjacent pixels.

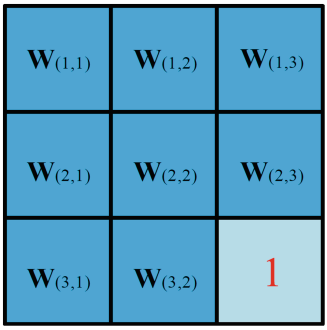

Fig. 2. The mask of image filtering.

Figure 2 shows an example of the mask $\mathbf{W}$ used in our experiment. As can be seen, it is of size 3×3 and its lower right position $\mathbf{W}(3,3)$ corresponds to the current pixel. To recover the current pixel value, the weight coefficient $\mathbf{W}(3,3)$ is set as one and other weight coefficients are integers determined by the sub-key $\mathbf{k}$, which is defined as

$$\begin{aligned} WV(i) &= \text{bi2de}(\mathbf{k}_i((8i-7):8i)) \\ [WS, IX] &= \text{sort}(WV) \\ w_i &= WV(i) + \text{find}(IX == i) \end{aligned}$$

Using the mask presented in Fig. 2, we can perform the image filtering to the permutation result. First, initialize the result $\mathbf{C}$ using the permutation result. Then, update the pixel value of $\mathbf{C}$ using the following equation,

$$\mathbf{C}(x,y) = \sum_{i,j \in \{1,2,3\}} \mathbf{W}(i,j)\mathbf{C}(x-i+1, y-j+1) \mod F, \tag{3}$$

where F is number of allowed pixel value in the original image, e.g. $F = 256$ if every pixel is represented by 8 bits.

Using the same mask $\mathbf{W}$, the inverse filtering operation in the decryption process is defined as

$$\begin{aligned} \mathbf{C}(x,y) = (\mathbf{C}(x,y) - \\ \sum_{i,j \in \{1,2,3\} \cap i,j \neq (1,1)} \mathbf{W}(i,j)\mathbf{C}(x-i+1, y-j+1)) \mod F \end{aligned} \tag{4}$$

Note that the operation order in the decryption process is opposite to that in the encryption process.

For these border pixels in the left two columns and top two rows, they don't have or have insufficient left and top adjacent pixels. To address this problem, when processing these border pixels, we use the right and bottom pixels to extend the images, which is demonstrated as Fig. 3. It is noticed that this won't enlarge the size of encrypted image, because we don't need to store these extended pixels.

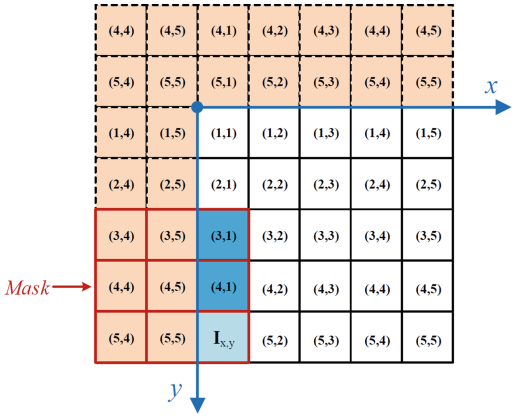

Fig. 3. Demonstration of extending image.

After two rounds of Josephus permutation and image filtering, a plain-image can be encrypted into cipher-image with high security level.

3 Simulations Results

This section simulates the proposed image encryption scheme in software environment. The test images are selected from USC-SIPI[1] and CVonline[2] image databases.

3.1 Simulation Results

An universal image encryption algorithm should have the ability to process different kinds of images. Figure 4 shows the simulation results of the proposed image encryption scheme for binary, grayscale and color images. As can be seen that the proposed encryption scheme can encrypt these digital images into noise-like images with uniform distribution. One can't obtain any information about the original images from the visual effects or from their pixel distributions. Using the corresponding key, the decryption process can completely recover the original images.

3.2 Discussion

As the architecture of confusion and diffusion is followed, the used Josephus permutation has high efficiency of separating pixels and the image filtering can randomly change pixel values, the proposed image encryption scheme can achieve

[1] http://sipi.usc.edu/database/.
[2] http://homepages.inf.ed.ac.uk/rbf/CVonline/Imagedbase.htm.

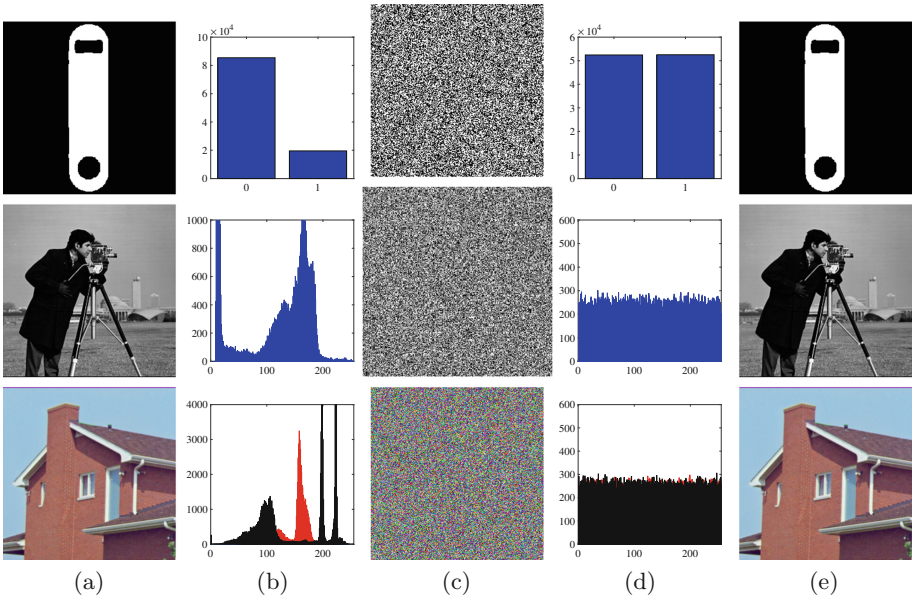

Fig. 4. Simulation results of the proposed image encryption scheme. (a) The plain-images of the binary, grayscale and color images; (b) histograms of (a); (c) encrypted results of (a); (d) histograms of (c); (e) decrypted results of (c).

the following advantages. (1) It can achieve the confusion and diffusion properties. (2) It can resist many commonly used security attacks, such as the brute-force attack, differential attack; This will be verified by experimental results in Sect. 4; (3) It has high encryption speed, because only two encryption rounds are performed and the Josephus permutation has a linear computation complexity.

4 Security Analysis

A high efficient image encryption algorithm should have high security level. That is to say, its encrypted cipher-images are expected to resist the commonly used security attacks. The proposed image encryption scheme has strong ability of resisting these attacks. This section analyzes the security performance of the proposed scheme and compares it with schemes in [14] and [15].

4.1 Key Sensitivity

First of all, an encryption algorithm should have secure key. The key security includes two parts. First, the length of key should be large enough to resist brute-force attack. Our encryption scheme has the secure key with length of 256 bits, which has a proper length of resisting brute-force attack. Besides, the secure key should be extremely sensitive, which means that only the correct key

can recover the original image. Another key with slight change can't recover any useful information about the original image.

Figure 5 shows the key sensitivity analysis in both the encryption and decryption processes. The encryption and decryption processes are represented as $\mathbf{C} = \mathrm{En}(\mathbf{P}, \mathbf{K})$ and $\mathbf{C} = \mathrm{En}(\mathbf{P}, \mathbf{K})$, respectively. The secure keys $\mathbf{K}_1$, $\mathbf{K}_2$ and $\mathbf{K}_3$ are three different keys that have only one bit difference with each other. As can be seen, when encrypting an identical image using two secure keys with only one bit difference, the two obtained cipher-images are totally different (see Fig. 5(d)). On the other hand, only the correct key can completely recover the original image (see Fig. 5(e)). Using secure keys with slight difference to decrypt a cipher-image, the two obtained decrypted results are random-like, and also completely different (see Figs. 5(f)–(h)). Thus, the key of the proposed encryption scheme are quite sensitive in both encryption and decryption processes.

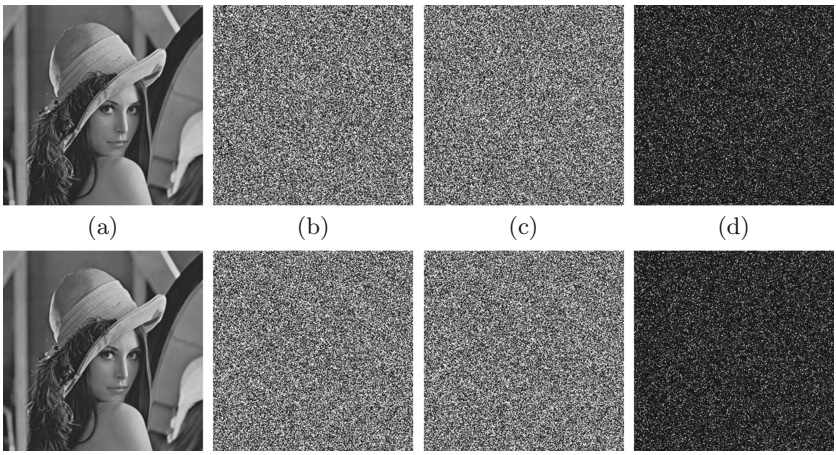

Fig. 5. Key sensitivity analysis. (a) Plain-image $\mathbf{P}$; (b) cipher-image $\mathbf{C}_1 = \mathrm{En}(\mathbf{P}, \mathbf{K}_1)$; (c) cipher-image $\mathbf{C}_2 = \mathrm{En}(\mathbf{P}, \mathbf{K}_2)$; (d) the difference between $\mathbf{C}_1$ and $\mathbf{C}_2$, $|\mathbf{C}_1 - \mathbf{C}_2|$; (e) decrypted image $\mathbf{D}_1 = \mathrm{De}(\mathbf{C}_1, \mathbf{K}_1)$; (f) decrypted image $\mathbf{D}_2 = \mathrm{De}(\mathbf{C}_1, \mathbf{K}_2)$; (g) decrypted image $\mathbf{D}_3 = \mathrm{De}(\mathbf{C}_1, \mathbf{K}_3)$; (f) the difference between $\mathbf{D}_2$ and $\mathbf{D}_3$, $|\mathbf{D}_2 - \mathbf{D}_3|$.

4.2 Local Shannon Entropy

To resist statistic attack, an ideal cipher-image is expected to have uniform distribution. The local Shannon entropy is designed to test the randomness of a block data from local view and it can provide a quantitative description to the randomness of cipher-image [16]. Mathematically, the calculation procedure of local Shannon entropy is defined as

$$\overline{H_{k,T_B}} = \sum_{i=1}^{k} \frac{H(S_i)}{k}, \tag{5}$$

where k is the number of image block, T_B denotes the number of pixels in each image block, $S_1 \sim S_k$ indicates the randomly selected image blocks and $H(S_i)$ is S_i's information entropy, which is defined as

$$H(S_i) = -\sum_{j=1}^{L} Pr(s_j) \log_2 Pr(s_j), \tag{6}$$

where L is the number of allowed values and s_j represents the j-th possible pixel value in image block S_i, and $Pr(s_j)$ is the probability of s_j.

Table 1. Local Shannon entropy values of several image encryption schemes. $\alpha = 0.05$, $k = 30$, $T_B = 1936$.

File name	Ref. [14]	Ref. [15]	Proposed
6.1.01	7.90153079	7.90625727	7.90212377
6.1.02	7.90302019	7.89802040	7.90253124
6.1.03	7.90157342	7.90364200	7.90264621
6.1.04	7.90393573	7.90203689	7.90220704
motion01.512	7.90397726	7.89959521	7.90219051
motion02.512	7.90304833	7.90257791	7.90270603
motion03.512	7.90241218	7.90466502	7.90243374
motion04.512	7.90339060	7.89972857	7.90230303
5.3.01	7.90085003	7.89893143	7.90232586
5.3.02	7.90465776	7.89797578	7.90279932
Pass rate	2/10	2/10	10/10

According to the recommendation in [16], our experiments set the significance level $\alpha = 0.5$, $k = 30$, and $T_B = 1936$. Then the local Shannon entropy is expected to fall into the interval $(h^*_{left}, h^*_{right}) = (7.901901305, 7.903037329)$ to pass the test for 8-bit grayscale image. Table 1 lists the local Shannon result of cipher-images generated by different image encryption schemes. As one can see that all the ten cipher-images generated by our proposed image encryption scheme can pass the local Shannon entropy. On the other hand, not all the cipher-images generated by [14] and [15] can pass the test. As a consequence, our proposed image encryption algorithm can generate cipher-images with high randomness.

4.3 Differential Attack

The differential attack is a widely used and efficient security attack. By observing how the difference in plaintext affects the change in ciphertexts, it aims to build the relationship between plaintext and ciphertext, and uses the built relationship

to recover the original information without secure key. The success of differential attack can even totally recover the original data. Worse yet, it can recover the secure key or equivalent key. An encryption algorithm with diffusion property can efficiently resist the differential attack. The diffusion property means that little change in plaintext can spread to the whole ciphertext. Figure 6 demonstrates the visualized effect of diffusion property of our image encryption algorithm. When using the same secure key to encrypt two plain-images with only one bit difference, the two obtained cipher-images are totally different and their difference can be seen in Fig. 6(f). This means that the proposed encryption scheme has good diffusion property.

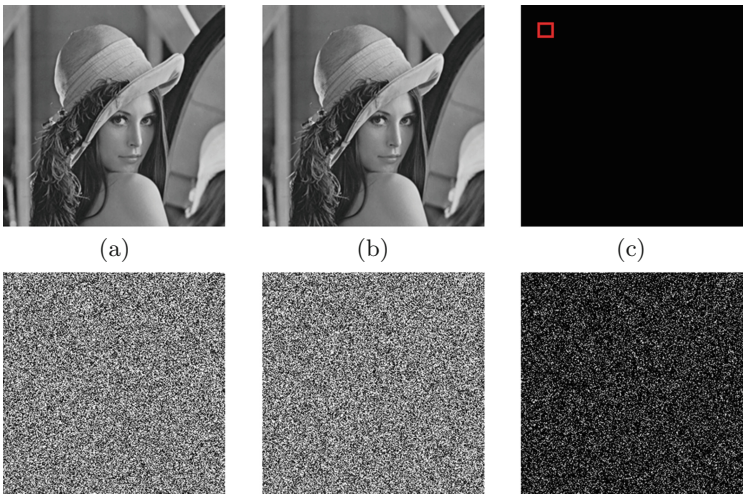

Fig. 6. Visualized effect of diffusion property. (a) original image $\mathbf{P}_1$; (b) original image $\mathbf{P}_2$, which has one bit difference with $\mathbf{P}_1$; (c) difference between $\mathbf{P}_1$ and $\mathbf{P}_2$, $|\mathbf{P}_1 - \mathbf{P}_2|$; (d) encrypted image $\mathbf{C}_1 = \mathrm{En}(\mathbf{P}_1, \mathbf{K}_1)$; (e) encrypted image $\mathbf{C}_2 = \mathrm{En}(\mathbf{P}_2, \mathbf{K}_1)$; (f) difference between $\mathbf{C}_1$ and $\mathbf{C}_2$, $|\mathbf{C}_1 - \mathbf{C}_2|$.

The number of pixel change rate (NPCR) and unified averaged changed intensity (UACI) can provide a quantitative description about the ability of resisting differential attack. Suppose $\mathbf{C}_1$ and $\mathbf{C}_2$ are two cipher-images encrypted from two plain-images with only one bit difference, the NPCR between $\mathbf{C}_1$ and $\mathbf{C}_2$ can be defined as

$$NPCR(\mathbf{C}_1, \mathbf{C}_2) = \sum_{i=1}^{M} \sum_{j=1}^{N} \frac{\mathbf{D}(i,j)}{G} \times 100\%, \tag{7}$$

where M, N are the height and width of the images, respectively, G denotes the total number pixels in an image, and $\mathbf{D}$ represents the difference between $\mathbf{C}_1$ and $\mathbf{C}_2$ and can be obtained by

$$\mathbf{D}(i,j) = \begin{cases} 0, & \text{if } \mathbf{C}_1(i,j) = \mathbf{C}_2(i,j), \\ 1, & \text{if } \mathbf{C}_1(i,j) \neq \mathbf{C}_2(i,j). \end{cases}$$

The UACI is mathematically defined as

$$UACI(\mathbf{C}_1, \mathbf{C}_2) = \sum_{i=1}^{M} \sum_{j=1}^{N} \frac{|\mathbf{C}_1(i,j) - \mathbf{C}_2(i,j)|}{T \times G} \times 100\%,$$

where T is the largest allowed pixel value.

Recently, strict critical values were developed in [17]. Firstly, an interval $(\mathcal{U}_\alpha^{*-}, \mathcal{U}_\alpha^{*+})$ for NPCR and a threshold $\mathcal{N}_\alpha^*$ for UACI were calculated, where α is the significance level. The obtained NPCR value falls into $(\mathcal{U}_\alpha^{*-}, \mathcal{U}_\alpha^{*+})$ and the UACI value is bigger than $\mathcal{N}_\alpha^*$ are considered to pass the corresponding test. According to the setting in [17], our experiments set $\alpha = 0.05$. Then for size of 256×256, $\mathcal{N}_\alpha^* = 99.5693\%$ and $(\mathcal{U}_\alpha^{*-}, \mathcal{U}_\alpha^{*+}) = (33.2824\%, 33.6447\%)$. For the size of 512×512, $\mathcal{N}_\alpha^* = 99.5893\%$ and $(\mathcal{U}_\alpha^{*-}, \mathcal{U}_\alpha^{*+}) = (33.3730\%, 33.5541\%)$. For the size of 1024×1024, $\mathcal{N}_\alpha^* = 99.5994\%$ and $(\mathcal{U}_\alpha^{*-}, \mathcal{U}_\alpha^{*+}) = (33.4183\%, 33.5088\%)$.

Table 2. NPCR results of several image encryption schemes with different images.

File name	Ref. [14]	Ref. [15]	Proposed
6.1.01	97.6165%	99.6051%	99.6322%
6.1.02	98.6066%	99.6087%	99.6017%
6.1.03	93.6384%	99.6065%	99.6002%
6.1.04	93.6402%	99.6057%	99.6200%
motion01.512	95.6271%	99.6091%	99.6181%
motion02.512	95.6280%	99.6085%	99.5922%
motion03.512	85.6641%	99.6114%	99.6017%
motion04.512	91.6419%	99.6099%	99.6192%
5.3.01	91.6390%	99.6083%	99.6109%
5.3.02	95.6293%	99.5809%	99.6171%
Pass rate	0/10	10/10	10/10

Tables 2 and 3 list the NPCR and UACI results of several image encryption schemes with different sizes of images. One can see that the image encryption in [14] can't pass most of the test images for both the NPCR and UACI tests, and the scheme in [15] fails one test in the UACI test. On the other hand, our proposed scheme can pass all the tests in both NPCR and UACI. As a result, our proposed image encryption scheme can achieve a strong ability of resisting differential attack.

Table 3. UACI results of several image encryption schemes with different images.

File name	Ref. [14]	Ref. [15]	Proposed
6.1.01	32.8546%	33.4473%	33.4799%
6.1.02	33.4571%	33.4697%	33.5830%
6.1.03	31.4938%	33.4356%	33.5983%
6.1.04	31.5274%	33.4731%	33.4297%
motion01.512	32.1218%	33.4901%	33.4531%
motion02.512	32.1020%	33.4684%	33.4457%
motion03.512	28.7945%	33.4603%	33.5269%
motion04.512	30.7904%	33.4740%	33.4532%
5.3.01	30.7987%	33.4460%	33.4941%
5.3.02	32.1332%	33.4104%	33.4768%
Pass rate	1/10	9/10	10/10

5 Conclusion

This paper introduced a new image encryption scheme according to Josephus permutation and image filtering. The Josephus permutation can break the high correlations between adjacent pixels by fast shuffling image pixels into different rows and columns. The image filtering using a random mask can efficiently spread the slight change of plain-image to the whole pixels of cipher-images. We provided the simulation results of the proposed image encryption scheme using different images. The security analysis was performed from the security key analysis, local Shannon entropy and differential attack analysis. The results show that, compared with several other image encryption schemes, the proposed scheme can encrypt images into cipher-images with higher security levels.

Acknowledgement. This work was financially supported by National Natural Science Foundation of China with Grant No. 61701137, No. 11371004 and No. 61672195, National Key Research and Development Program of China with Grant No. 2016YFB0800804 and No. 2017YFB0803002, and Shenzhen Science and Technology Plan with Grant No. JCYJ20170307150704051, No. JCYJ20160318094336513, No. JCYJ20160318094101317 and No. KQCX20150326141251370.

References

1. Jain, Y., Bansal, R., Sharma, G., Kumar, B., Gupta, S.: Image encryption schemes: a complete survey. Int. J. Sig. Process. Image Process. Patt. Recogn. **9**(7), 157–192 (2016)
2. Zhang, L.Y., Liu, Y., Pareschi, F., Zhang, Y., Wong, K.W., Rovatti, R., Setti, G.: On the security of a class of diffusion mechanisms for image encryption. IEEE Trans. Cybern. **PP**, 1–13 (2017)

3. Hua, Z., Wang, Y., Zhou, Y.: Image cipher using a new interactive two-dimensional chaotic map. In: 2015 IEEE International Conference on Systems, Man, and Cybernetics (SMC), pp. 1804–1808. IEEE (2015)
4. Chai, X., Chen, Y., Broyde, L.: A novel chaos-based image encryption algorithm using DNA sequence operations. Opt. Lasers Eng. **88**, 197–213 (2017)
5. Hua, Z., Zhou, Y.: Design of image cipher using block-based scrambling and image filtering. Inf. Sci. **396**, 97–113 (2017)
6. Chen, J., Zhu, Z.L., Zhang, L.B., Zhang, Y., Yang, B.Q.: Exploiting self-adaptive permutation-diffusion and dna random encoding for secure and efficient image encryption. Sig. Process. **142**, 340–353 (2018)
7. FIPS PUB 197: Advanced encryption standard (AES) (2001)
8. Ye, G., Huang, X.: An efficient symmetric image encryption algorithm based on an intertwining logistic map. Neurocomputing **251**, 45–53 (2017)
9. Zhou, Y., Bao, L., Chen, C.P.: A new 1D chaotic system for image encryption. Sig. Process. **97**, 172–182 (2014)
10. Hua, Z., Zhou, Y.: Image encryption using 2D Logistic-adjusted-Sine map. Inf. Sci. **339**, 237–253 (2016)
11. Arroyo, D., Rhouma, R., Alvarez, G., Li, S., Fernandez, V.: On the security of a new image encryption scheme based on chaotic map lattices. Chaos Interdisc. J. Nonlinear Sci. **18**(3), 033112 (2008)
12. Ping, P., Xu, F., Wang, Z.J.: Image encryption based on non-affine and balanced cellular automata. Sig. Process. **105**, 419–429 (2014)
13. Halbeisen, L., Hungerbühler, N.: The josephus problem. J. de théorie des nombres de Bordeaux **9**(2), 303–318 (1997)
14. Xu, L., Li, Z., Li, J., Hua, W.: A novel bit-level image encryption algorithm based on chaotic maps. Opt. Lasers Eng. **78**, 17–25 (2016)
15. Wang, X., Wang, Q., Zhang, Y.: A fast image algorithm based on rows and columns switch. Nonlinear Dyn. **79**(2), 1141–1149 (2015)
16. Wu, Y., Zhou, Y., Saveriades, G., Agaian, S., Noonan, J.P., Natarajan, P.: Local shannon entropy measure with statistical tests for image randomness. Inf. Sci. **222**, 323–342 (2013)
17. Wu, Y., Noonan, J.P., Agaian, S.: NPCR and UACI randomness tests for image encryption. Cyber J. Multi. J. Sci. Technol. J. Sel. Areas Telecommun. (JSAT), 31–38 (2011)