2012 IEEE International Conference on Systems, Man, and Cybernetics
October 14-17, 2012, COEX, Seoul, Korea

A New Image Encryption Algorithm Using Truncated P-Fibonacci Bit-planes

Weijia Cao, Student Member, IEEE
Department of Computer and Information Science, University of Macau
Macau, China

Yicong Zhou, Member, IEEE
Department of computer and Information Science, University of Macau
Macau, China
yicongzhou@umac.mo

C.L. Philip Chen, Fellow, IEEE
Department of Computer and Information Science, University of Macau
Macau, China

Abstract—Image encryption is an effective approach to protect privacy and security of images. This paper introduces a novel image encryption algorithm using the Truncated P-Fibonacci Bit-planes as security key images to encrypt images. Simulation results and security analysis are provided to show the encryption performance of the proposed algorithm.

Keywords-image encryption; truncated P-Fibonacci bit-plane

I. INTRODUCTION

Multimedia communication plays an increasingly important role of information exchange between people. Digital information technologies and networks bring us the convenience, as well as, however, the hidden hazards: sensitive information within images may be easily stolen, falsified, or illegally copied and disseminated. Providing security of images becomes growingly significant in the digital age [1]. Image encryption is an effective approach to protect privacy and security of images.

In recent years, many encryption algorithms based on image bit-plane decomposition have been developed. Some of them encrypt only the selective bit-planes of images [2, 3]. Another algorithm uses exclusive-OR operations to encrypt images [4]. However, these algorithms have a low level of security due to the limited key space and the fact that the decomposition results of traditional bit-plane decomposition results are predictable. The number of bit-planes and their content for a specific image are fixed. To achieve a higher level of security, our previous work has applied edge maps to image encryption [5]. It generates edge maps from the original image. However, this is not convenient for image decryption because the edge maps have to be sent to the authorized users for decryption. To further extend the concept of using edge map for image encryption, binary key images have been used for image encryption [6]. These binary images could be other existing binary images, or edge maps, or bit-planes generated from another image.

To achieve a higher security level, in this paper, we introduce a novel algorithm to encrypt images. Instead of using the traditional bit-plane decomposition method to generate the binary key images, this algorithm uses the Truncated P-Fibonacci Bit-planes [7] for image encryption. It takes advantage of the parametric character of the truncated P-Fibonacci Sequence and Truncated P-Fibonacci Bit-planes. Different p values result in different bit-planes which increases the security key space of the proposed encryption algorithm.

The rest of this paper is organized as follows: Section II reviews the Truncated P-Fibonacci Bit-plane, which will be used for the new image encryption algorithm introduced in Section III. The experimental results and security analysis are presented in Section IV and Section V, respectively. Section VI reaches a conclusion.

II. BACKGROUND

This section briefly reviews the Truncated P-Fibonacci Sequence (TPFS) and the Truncated P-Fibonacci Bit-plane (TPFB).

A. Truncated P-Fibonacci Sequence (TPFS)

In mathematics, a recursive sequence called the Truncated P-Fibonacci Sequence is defined by [7]:

$$T_p(i)=\begin{cases}0 & i<0\\ 1 & i=0\\ F_p(i+p) & i>0\end{cases} \quad (1)$$

where i is the index number of the sequence; the non-negative integer p is a distance parameter; and $F_p(i+p)$ is the P-Fibonacci Sequence defined by [8]

$$F_p(n)=\begin{cases}0 & n<0\\ 1 & n=0\\ F_p(n-1)+F_p(n-p-1) & n>0\end{cases} \quad (2)$$

This work was partially supported by Research Committee of the University of Macau under grant SRG007-FST12-ZYC, MYRG113(Y1-L3)-FST12-ZYC, and conference grant.

978-1-4673-1714-6/12/$31.00 ©2012 IEEE

Table I provides several example sequences when the *p* value changes.

TABLE I. SAMPLE SEQUENCES OF TPFS WITH DIFFERENT P VALUES

P	i								
	0	***1***	***2***	***3***	***4***	***5***	***6***	***7***	***...***
0	1	2	4	8	16	32	64	128	...
1	1	2	3	5	8	13	21	34	...
2	1	2	3	4	6	9	13	19	...
3	1	2	3	4	5	7	10	14	...
4	1	2	3	4	5	6	8	11	...
5	1	2	3	4	5	6	7	9	...
6	1	2	3	4	5	6	7	8	...
...	...	...	...	...	...	...	...	...	...

As shown in Table I, the sequence is a power of 2 series when *p* value is equal to 0. When *p* is equal to 1 the sequence is the traditional Fibonacci sequence.

According to Equation (2) the number of duplicate "1" in the beginning of the P-Fibonacci Sequence is growing when the *p* value increases. The TPFS in Equation (1) was introduced to minimize the redundancy of "1" in the sequence.

B. Truncated P-Fibonacci Bit-plane (TPFB)

A decimal number *N* can be represented by the Truncated P-Fibonacci codes because the binary sequence is a special instance of the TPFS when the *p* value is 0 [7].

$$N = \sum_{i=0}^{n-1} t_i * T_p(i) = t_0 * T_p(0) + t_1 * T_p(1) + ... + t_{n-1} * T_p(n-1) \quad (3)$$

where $T_p(i)$ is the TPFS defined in Equation (1), t_i is the weight coefficient.

Therefore, a decimal number can be represented by the Truncated P-Fibonacci code $(t_{n-1},...,t_2,t_1,t_0)$. However, such representation is not unique. For example, if *N* is equal to 25 and *p* is equal to 2, there are several Truncated P-Fibonacci codes of 25 as shown in TABLE II.

TABLE II. DIFFERENT TRUNCATED P-FIBONACCI CODES OF 25 WITH P = 2

N=25, P=2	$T_p(i)$							
	19	***13***	***9***	***6***	***4***	***3***	***2***	***1***
25	1	0	0	1	0	0	0	0
25	1	0	0	0	1	0	1	0
25	1	0	0	0	0	1	1	1
25	...	...	...	...	...	...	...	...

Thus, we need a rule to ensure that every non-negative decimal number is represented by a unique Truncated P-Fibonacci code. In this paper, we choose a rule presented in [9].

$$N = T_p(i) + q \quad (4)$$

where $T_p(i)$ is the i^{th} element of TPFS with a specific *p* value, $0 \le i < n$; and *q* is a non-negative decimal number which is a reminder, $0 \le q < T_p(i-p)$.

After applying the rule in Equation (4), every decimal number will have only one representation of the Truncated P-Fibonacci code. For example, the Truncated P-Fibonacci code of decimal number 25 will be (1,0,0,1,0,0,0,0) when *p* value is 2. Namely, $25=(1,0,0,1,0,0,0,0)_2$.

According to the TPFB definition in Equation (3), a grayscale image can be decomposed into several binary bit-planes using the Truncated P-Fibonacci codes, which is called the Truncated P-Fibonacci Bit-plane decomposition. An example is shown in Figure 1. The decomposition results and the number of TPFBs change with parameter *p* values. We will use these TPFBs as security key images for image encryption in the following sections.

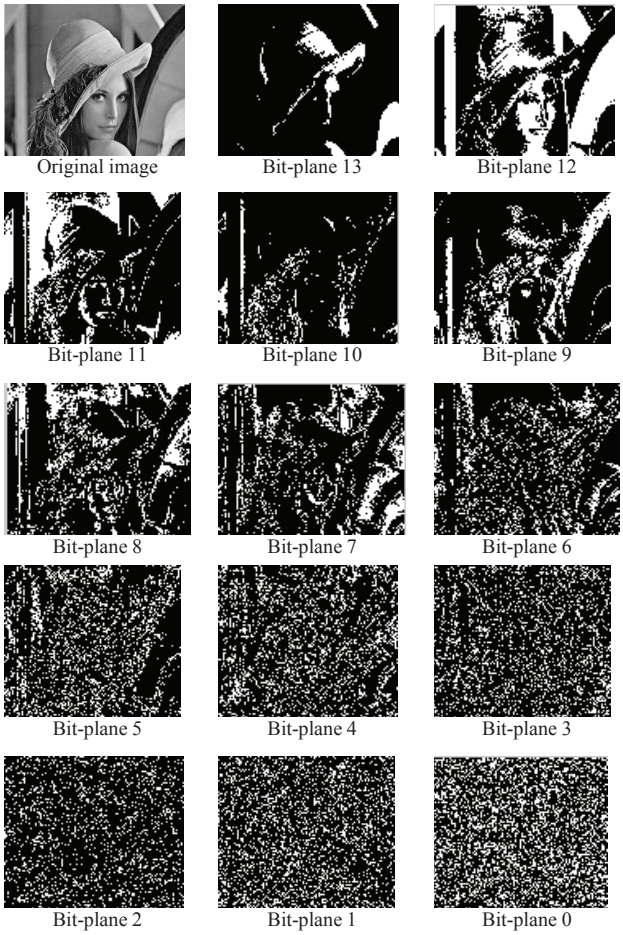

Figure 1. The Truncated P-Fibonacci bitplane decomposition of a grayscale image, p = 2

III. NEW IMAGE ENCRYPTION ALGORITHM

In this section, a new image encryption algorithm is introduced. It uses one of the TPFBs to encrypt the original image.

The new image encryption algorithm is shown in Figure 2. The r^{th} TPFB of a source image is selected as a security key image to encrypt the original image. The source image is another new or existing image with the same size as the original image. The original image is decomposed into binary bit-planes. An XOR operation is performed between the security key image (the r^{th} TPFB of a source image) and each bit-plane of the original image. The XORed bit-planes are then combined. A scrambling algorithm is used to change the pixel locations of the combined image, obtaining the final encrypted image.

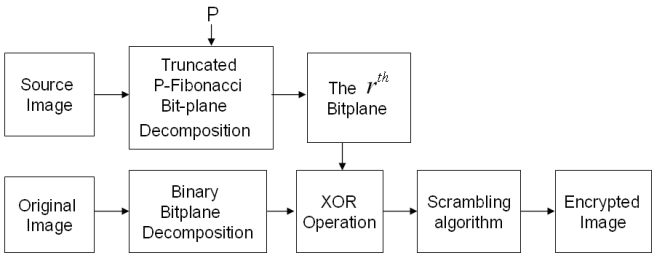

Figure 2. The new image encryption algorithm

The users have the flexibility to select: (1) any new or existing image as the source image; (2) any specific p value to decompose the source image into the TPFBs; (3) any one of the TPFBs as the security key image; and (4) any image scrambling algorithm to change image pixel locations. The security keys of the new algorithm consist of the source image (or the location of the source image); the parameter p value, the TPFB index, as well as the scrambling algorithm and its security keys.

IV. Simulation Results

The proposed new algorithm can be used to encrypt different types of images. In this section, several experimental results will be provided to show its encryption performance.

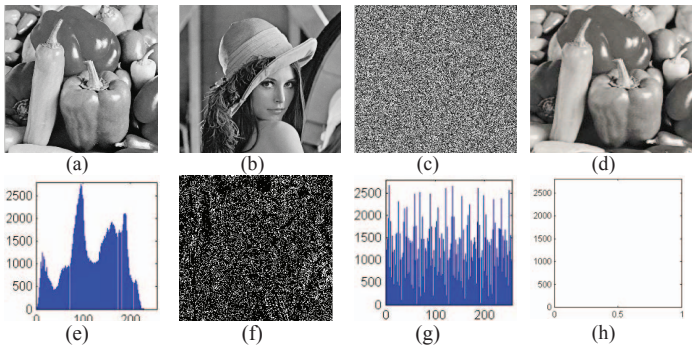

Figure 3. Grayscale image encryption. (a) The original grayscale image with the size of 512×512; (b) The source image with the size of 512×512; (c) The encrypted image; (d) The recovered image; (e) Histogram of the original image in (a); (f) the 4^{th} TPFB of the source image in (b), p=2; (g) Histogram of the encrypted image in (c); (h) Histogram of the difference between (a) and (d).

As shown in the Figure 3, the original grayscale image in Figure 3(a) can be encrypted by the proposed encryption algorithm and perfectly recovered in the decryption process.

The encrypted image in Figure 3(c) is completely different with the original image in Figure 3(a). The histogram of the encrypted image also shows that its pixel distribution is almost uniform. This ensures that the attackers are difficult to break the encrypted images using statistic methods. The recovered image in Figure 3(d) is visually the same as the original image. The histogram in Figure 3(h) plots the difference between the original and recovered images pixel by pixel. The results are zeros. This means that the recovered and original images are identical.

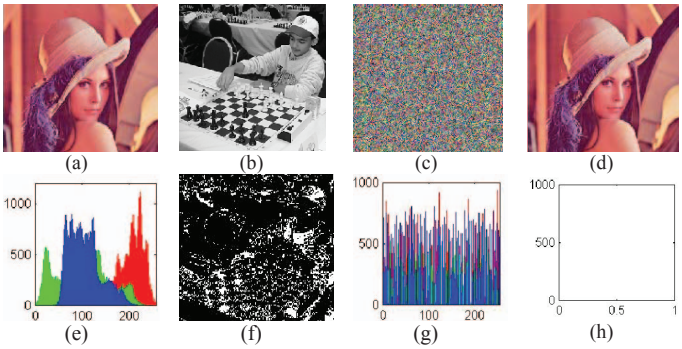

Figure 4. Color image encryption. (a) The original color image with the size of 256x256; (b) The source image with the size of 256x256; (c) The encrypted color image; (d) The recovered image; (e) Histogram of the original image in (a); (f) the 4^{th} TPFB of the source image in (b), p=2; (g) Histogram of the encrypted image in (c); (h) Histogram of the difference between (a) and (d).

The proposed algorithm can be also used for color image encryption. Similar results can be obtained when color images are encrypted by the proposed algorithm as shown in Figure 4. These experimental results further prove that our proposed algorithm is a lossless encryption method.

V. Security Analysis

This section addresses the security issues of the proposed image encryption algorithm.

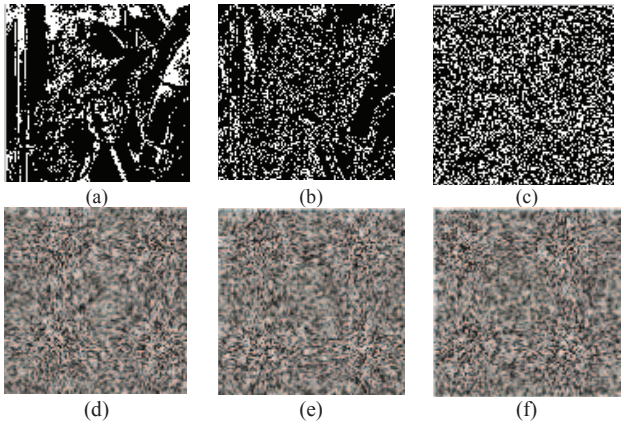

Figure 5. Image encrypted using different TPFBs of a source image in Figure 3(b) with p=3. (a) The 8^{th} TPFB; (b) The 5^{th} TPFB; (c) The 0^{th} TPFB; (d)The encrypted image using the bit-plane in (a); (e) The encrypted image using the bit-plane in (b); (f) The encrypted image using the bit-plane in (c).

The proposed encryption algorithm contains the following security keys: (1) the source image (or the location of the

source image); (2) the parameter *p* value; (3) the TPFB index; and (4) the scrambling algorithm and its security keys. The possible choices of these security keys form the size of the key space of the proposed algorithm. There are a huge number of choices for the source images. For a specific source image, the TPFB generating method is also parametric. Moreover, other portion of the security keys has large number of choices. Therefore, the proposed image encryption algorithm has a sufficiently large key space. This ensures that its encrypted images are difficult to be decoded by unauthorized users, achieving a high level of security.

We provide several encryption examples to show the encryption performance when the TPFB changes. We use the image in Figure 3(a) as the original images and the image in Figure 3(b) as the source image for our tests. Figure 5 provides the encryption results using different TPFBs obtained from a same image with a specific p value. Figure 6 uses the TPFBs generated by different p values. As shown in Figures 5(d)-(f) and 6(d)-(f), different TPFBs result in different encrypted images which are visually unrecognizable and completely different from the original image in Figure 3(a). This means that changing the security key images (the TPFBs) will not change the encryption quality. This is another advantage of the proposed encryption algorithm.

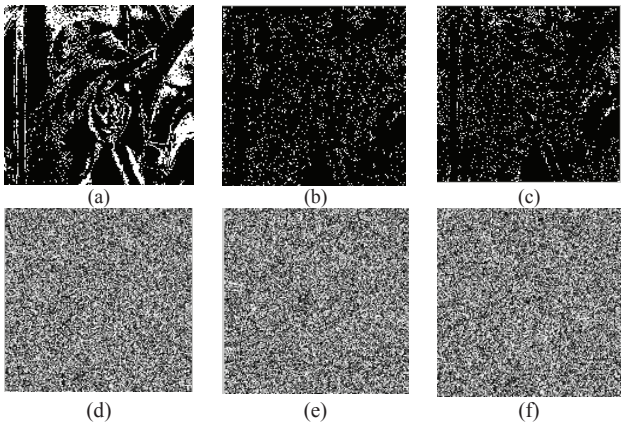

Figure 6. The encrypted results using the TPFBs with different p values (a) The 9^{th} TPFB, p=3; (b) The 5^{th} TPFB, p=5; (c) The 8^{th} TPFB, p=7; (d) The encrypted image using (a); (e) The encrypted image using (b); (f) The encrypted image using (c).

VI. CONCLUSION

This paper has introduced a new image encryption algorithm, which use a TPFB of another source image as the security key image. The user has the flexibility to use any new or existing image as a source image to generate the TPFBs and to include any scrambling algorithm into our proposed encryption algorithm. The experimental results have shown that the proposed algorithm has excellent performance for image encryption. It has a sufficiently large key space, ensuring a high security level. It has potential applications in privacy and copyright protection.

REFERENCES

[1] Y. Zhou, K. Panetta, and S. Agaian, "A Lossless Encryption Method for Medical Images Using Edge Maps," in 2009 the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Minneapolis, MN, 2009, pp. 3707-3710.

[2] M. Podesser, H. Schmidt, and A. Uhl, "Selective bitplane encryption for secure transmission of image data in mobile environments," in The 5th Nordic Signal Processing Symposium -NORSIG-2002, on board Hurtigruten, Norway, 2002, p. 1037.

[3] D. Moon, Y. Chung, S. B. Pan, K. Moon, and K. Chung, "An Efficient Selective Encryption of Fingerprint Images for Embedded Processors," ETRI Journal, vol. 28, pp. 444-452, Aug 2006.

[4] J.-W. Han, C.-S. Park, D.-H. Ryu, and E.-S. Kim, "Optical image encryption based on XOR operations," Optical Engineering, vol. 38, pp. 47-54, 1999.

[5] Y. Zhou, K. Panetta, and S. Agaian, "Image Encryption Based on Edge Information," in IS&T / SPIE Electronic Imaging 2009: Multimedia on Mobile Devices 2009, San Jose, CA, 2009, pp. 725603-11.

[6] Y. Zhou, K. Panetta, and S. Agaian, "Image Encryption Using Binary Key-Images," in 2009 IEEE International Conference on Systems, Man and Cybernetics, San Antonio, TX, 2009, pp. 4569-4574.

[7] Y. Zhou, K. Panetta, R. Cherukuri, and S. Agaian, "Selective Object Encryption for Privacy Protection," in SPIE Defense, Security, and Sensing 2009: Mobile Multimedia/Image Processing, Security, and Applications 2009, Orlando, FL, 2009, pp. 73510F-10.

[8] D. Z. Gevorkian, K. O. Egiazarian, S. S. Agaian, J. T. Astola, and O. Vainio, "Parallel algorithms and VLSI architectures for stack filtering using Fibonacci p-codes," Signal Processing, IEEE Transactions on, vol. 43, pp. 286-295, 1995.

[9] Y. Zhou, K. Panetta, S. Agaian, and C. L. P. Chen, "Image encryption using P-Fibonacci transform and decomposition," Optics Communications, vol. 285, pp. 594-608, 2012.