Reversible Data Hiding for Encrypted Image Based on Arnold Transformation

Dan Wu

Department of Mathematics,China Jilang University, Hangzhou, 310018, China

Abstract. A reversible data hiding scheme for encrypted image was proposed based on Arnold transformation. In this scheme, the original image was divided into four sub-images by sampling, the sub-images were scrambled by Arnold transformation using two secret keys, then the scrambled sub-images were reconstituted an encrypted image. Subsequently, additional data was embedded into the encrypted image by modifying the difference between two adjacent pixels. With an encrypted image containing additional data, the receiver can obtain a decrypt image using the decryption key. Meanwhile, with the aid of the decryption key and information hiding key, the receiver can pick the hiding information and recover the original image without any error. Experiment result shows that the proposed scheme can obtain a higher payload with good image quality.

1 Introduction

Reversible data hiding is a technology that embeds additional information into the images, such as military or medical image, which not only extracts the hiding data, but also perfectly reconstructs the original host signal from the embedded work. With the rapid development of cloud computing, users are beginning to move their data to the cloud servers in order to avoid troublesome data management at local machines and enjoy convenient service. In cloud computing, users need to pass private data onto the cloud, but do not trust the cloud, so the data will be encrypted and passed to the cloud. Reversible data hiding allows the service provider to embed additional messages, e.g. the image labels, the notations or the authentication information, into the encrypted images, and has the reversibility feature of extracting the additional message and recovering the original image. The technology has a good application prospect. For example, the patient's medical image is encrypted and uploaded to the hospital's server or cloud (Zhang, 2011). The administrator can embed relevant information of the image, such as patient information, shooting time, and shooting location into the encrypted images. The theories of reversible data hiding algorithm are derived from the research results of the plaintext domain (Tian, 2003, Thodi, 2007, Ni, 2006, Cleik, 2005). Recently, many researchers have paid more attention to reversible data hiding algorithms in encrypted images since the encrypted images are widely generated and stored in cyberspace. A popular reversible data hiding algorithm is contributed by Zhang (Zhang, 2011). In this work, Zhang (Zhang, 2011) divided encrypted image into blocks, embedded one bit into block by flipping three LSBs of a half of the block pixels, and extracted secret data by defining a fluctuation function in terms of spatial correlation in natural images. Although some useful RDH algorithms for encrypted image have been reported (Hong, 2012. Zhang, 2012. Qian, 2013.), they cannot correctly extract all secret bits from stego image (Tang, 2018).

This paper proposed a method of reversible data hiding for encrypted images based on Arnold transformation. In this scheme, the original image was divided into four sub-images by sampling, the sub-images were scrambled by Arnold transformation using two secret keys, then the scrambled sub-images were reconstituted an encrypted image. Then the information are embedded into the encrypted image by modifying the difference between two adjacent pixels. If the receiver has the data-hiding key, it can extract the hiding data. If the receiver has the encryption key, it can roughly recover the image. If the receiver has both the data-hiding key and the encryption key, it can extract the additional data and recover the original content without any error. Experiment result shows that the proposed scheme can obtain a higher payload with good image quality.

The rest of the paper is organized as follows. Section 2 presents the principle of image encryption by using Arnold transformation and details the proposed reversible data hiding for encrypted image. Conclusion are shown in Section 3.

* Corresponding author: wudan@cjlu.edu.cn

2 PROPOSED SYSTEM

2.1 Image Encryption

Supposing that the size of a grayscale image A is 512× 512, we divide it into 2×2 pieces (Fig.1), and get 256× 256 blocks. For every piece we choose the upper left pixel, the upper right pixel, the lower left pixel, and the lower right pixel to form four sub-images (I, II, III, and IV as shown in Fig.2).

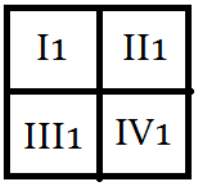
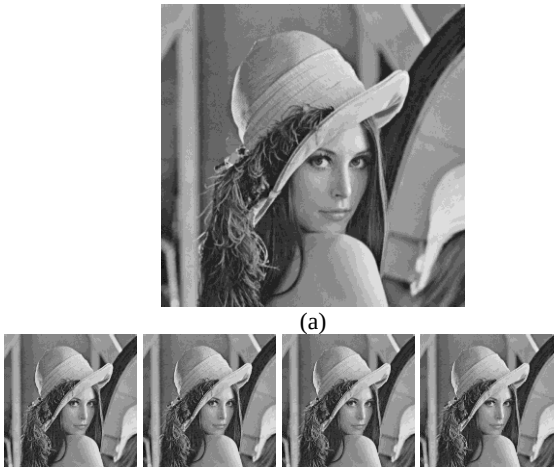
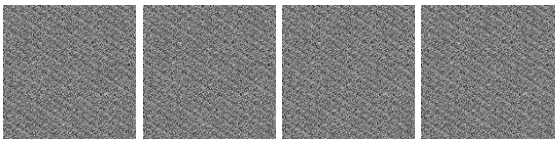


Figure 1. 2×2 piece



(a) (b) (c) (d) (e)

Figure 2. (a) Lena (512×512) (b) the sub-image I (256×256) (c) the sub-image II(256×256) (d) the sub-image III (256×256)(e) the sub-image IV (256×256)



(a) (b) (c) (d)

Figure 3. (a)the sub-image IW (n1=128) (b) the sub-image IIW (n1=128) (c) the sub-image IIIW (n2=120) (d) the sub-image IVW (n2=120)

We transform every sample using Arnold transformation.

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \bmod 256 \quad (1)$$

where (x,y) are the coordinates of a pixel in a image, and (x',y') are the coordinates after scrambling. We choose I and II to iterate the above formula for a certain number of times n1 (as a secret key K1), disturb III and IV n2 times (another key K2), and get the scrambled images IW, IIW, IIIW, and IVW (Fig.3). Rearranging the IW, IIW, IIIW, and IVW as shown in Fig.1 we get the encrypted image WA(Fig.4).

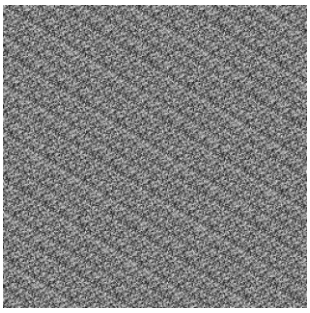


Figure 4. the encrypted Lena WA

2.2 Data Embedding

The data-hider scans the encrypted image WA horizontally, get the sequence of encrypted pixel $WA = \{wa_1, wa_2, \cdots, wa_{512\times512-1}, wa_{512\times512}\}$, and divide WA into 512×512/2 blocks $\overline{WA} = \{(wa_1, wa_2), \cdots, (wa_{512\times512-1}, wa_{512\times512})\}$.

If wa_1 and wa_2 are the gray values of a pixel-pair, the integer-mean wm and the difference d are defined as

$$\begin{cases} wm = floor((wa_1 + wa_2)/2) \\ d = wa_1 - wa_2 \end{cases} \quad (2)$$

Using the difference d , the bit b can be hided via the following equations

$$\begin{cases} d' = 2d + b & -K \le d \le K-1 \\ d' = d + K & d \ge K \\ d' = d - K & d < -K \end{cases} \quad (3)$$

where K is the threshold controlled by the hider.

$$\begin{cases} wa'_1 = wm + floor((d'+1)/2) \\ wa'_2 = wm - floor(d'/2) \end{cases} \quad (4)$$

The embedded pixel wa'_1 and wa'_2 can be calculated by d' and wm via (4). And we can get the encrypted image WA' containing embedded data. When $d \ge K$ or $d < -K$, we shift the difference d further away from the zero point, which is called the difference shifting, and leave $[K, 2K-1]$ and $[-2K-1, -K-1]$ empty for difference expansion. When $-K \le d \le K-1$, we expand the difference d to embed the watermark bit b . Table 1 shows the relationship between the threshold and payload.

Table 1. Relationship between the threshold and payload

threshold	payload
K=1	38359
K=2	58595
K=3	75295
K=4	87690
K=5	96857

2.3 Data Extracting and Image Recovery

With an encrypted image EA' containing embedded data, if the receiver has only the data-hiding key, he may scan the image horizontally, get the sequence of pixels denoted as $\overline{WA}' = \{(wa'_1, wa'_2), \cdots, (wa'_{512\times512-1}, wa'_{512\times512})\}$, and calculate the difference d' and the integer-mean M of the pair (wa'_{2i-1}, wa'_{2i}).

When $-2K \le d' \le 2K-1$, the hiding data can be obtained via (5).

$$b = d' \bmod 2 \tag{5}$$

If the user has the encryption key but do not know the data-hiding key, he cannot extract the hiding information and recover the original image losslessly. However, the original image can be roughly recovered. Denoting the encrypted image containing embedded data as EA', the receiver can decrypt the received image. The image can be divided into 2×2 pieces (Fig.5), and get 256×256 blocks. For every block we select the upper-left pixel, the upper-right pixel, the lower-left pixel, and the lower-right pixel to form four sub-images (EI', EII', EIII', and EIV' as shown in Fig.2). We disturb EI' and EII' m1 times (m1=196-n1), EIII' and EIV' m2 times (m2=196-n2), and get the decrypted images I', II', III', and IV', then rearrange the images and get the decrypted image of high PSNR.Table 2 shows the relationship between payload and PSNR of decrypt image without the hiding key.

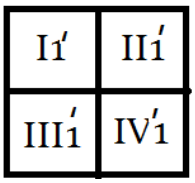


Figure 5. 2×2 piece

Table 2. Relationship between the payload and PSNR (decrypt without the data hiding key)

PSNR	payload
49.9997	38359
47.0584	58595
48.0893	75295
43.7050	87690
42.6788	96857

If the user has the encryption key and the data-hiding key, he can extract the hiding data, recover the encrypted image using equation (6) and (7), and recover the original image losslessly using the same method as above.

$$\begin{cases} d = \left\lfloor \dfrac{d'}{2} \right\rfloor & -2K \le d \le 2K-1 \\ d = d' - K & d \ge 2K \\ d = d' + K & d < -2K \end{cases} \tag{6}$$

$$\begin{cases} wa_1 = wm + floor((d+1)/2) \\ wa_2 = wm - floor(d/2) \end{cases} \tag{7}$$

The test image Lena sized 512×512 shown in Fig. 6(a) was used as the original image in the experiment. After Arnold transformation the encrypted image are generated as shown in Fig. 6 (b). Then, we let $K = 4$ to embed 87690 additional bits into the encrypted image. The encrypted image containing the additional information is shown in Fig. 6(c), and the embedding rate is 0.3345 bit per pixel (bpp). With an encrypted image containing embedded data, we could extract the additional data using the data-hiding key. If we directly decrypt the encrypted image containing embedded data using the encryption key, the value of PSNR in the decrypted image is 43.7050 dB. The directly decrypted image is given as Fig. 6(d). By using both the data-hiding key and the decryption key, the embedded data could be successfully extracted and the original image could be perfectly recovered from the encrypted image containing embedded data.

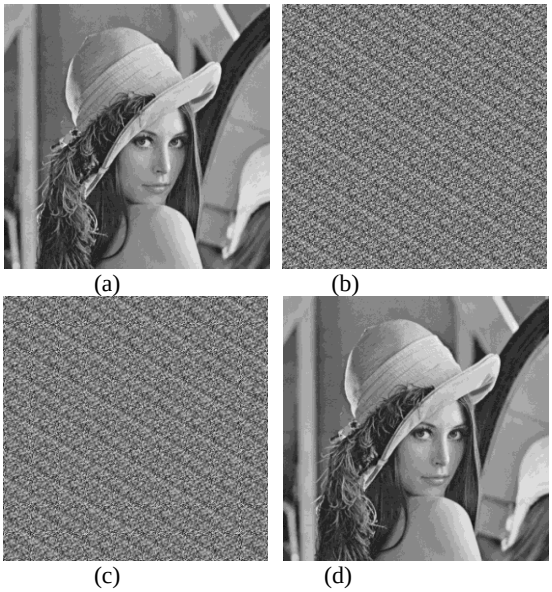


Figure 6. (a) Original Lena, (b) its encrypted version, (c) encrypted image containing embedded data with embedding rate 0.3345 bpp, and (d) directly decrypt

3 Conclusions

In this paper we proposed a method of reversible data hiding for encrypted images based on Arnold transformation.The original image was encrypted by Arnold transformation, and the information was embedded into the encrypted image using difference expansion and shifting. If the receiver has the data-hiding key, he can extract the hiding data. If the receiver has both the data-hiding key and the decryption key, he can extract the additional data and recover the original content losslessly. Experiment result shows that the proposed scheme can obtain a higher payload with good image quality.

References

1. Zhang, X., 2011. Reversible data hiding in encrypted image. *IEEE on Signal Processing Letters*, 18(4): 255-258
2. Tian, J., 2003, Reversible data embedding using a difference expansion. *IEEE Transactions on Circuits System and Video Technology*, 13(8): 890-896.
3. Thodi, D. M., Rodriguez, J. J., 2007. Expansion embedding techniques for reversible watermarking. *IEEE Transactions on Image Processing*, 16(3): 721-730.
4. Ni, Z., Shi, Y. Q., Ansari, N., Su, W., 2006. Reversible data hiding. *IEEE Transactions on Circuits and Systems for Video Technology*, 16(3): 354-362.
5. Celik, M. U., Sharma, G., Tekalp, A. M., Saber, E., 2005. Lossless generalized-LSB data embedding. *IEEE Transactions on Image Processing*, 14(2): 253-266.
6. Hong, W., Chen, T. S., Wu, H. Y., 2012. An improved reversible data hiding in encrypted images using side match. *IEEE on Signal Processing Letters*, 19(4): 199-202.
7. Zhang, X., 2012. Separable reversible data hiding in encrypted image. *IEEE Transactions on Information Forensics and Security*, 7(2): 826-832.
8. Qian, Z., Han, X., Zhang, X.. 2013. Separable reversible data hiding in encrypted images by n-nary histogram modification. *The Third International Conference on Multimedia Technology*, Atlantis Press, Paris. 869-876.
9. Tang, Z., Lu, Q., Lao, H., Yu, C., Zhang, X. 2018. Error-free reversible data hiding with high capacity in encrypted image. *Optik* ,157: 750-760.