Encrypted Image-Based Reversible Data Hiding with Public Key Cryptography from Interpolation-Error Expansion

Fuqiang Di, Junyi Duan, Minqing Zhang(✉), Yingnan Zhang, and Jia Liu

Engineering University of the People's Armed Police, Xi'an, China
18710752607@163.com

Abstract. This paper proposes an improved version of Shiu's encrypted image-based reversible data hiding with public key cryptography (EIRDH-P). The original work vacates embedding room by difference expansion technique and embeds one bit into each pair of adjacent encrypted pixels. The data extraction and image recovery can be achieved by comparing all pairs of decrypted pixels. Shius' work did not fully exploit the correlation inherent in the neighborhood of a pixel and required side information to record the location map. These two issues could reduce the amount of differences and in turn lessen the potential embedding capacity. This letter adopts a better scheme for vacating room before public key encryption using prediction-error expansion method, in which the pixel predictor is utilized by interpolation technique. The experimental results reveal that the proposed method offers better performance over Shiu's work and existing EIRDH-P schemes. For example, when the peak signal-to-noise ratio of the decrypted Lena image method is 35, the payload of proposed method is 0.74 bpp, which is significantly higher than 0.5 bpp of Shius's work.

Keywords: Reversible data hiding · Interpolation-error expansion · Encrypted image · Public key cryptography

1 Introduction

Reversible data hiding (RDH) [1–3] aims to embed some additional information into a carrier image, while the original image can be recovered by one hundred percent after data extraction. In many scenarios, since cryptography is used to convert normal image data to cipher form for secure communication, encrypted image-based reversible data hiding (EIRDH) has attracted much attention in recent years and has a lot of important applications in medical, military and other fields [4–6]. For example, medical images of patients which have been uploaded to the hospital servers or the cloud are encrypted so as to protect privacy of patients. On the one hand, managers need to embed relevant information such as the owner information and recording time into the corresponding cipher text; On the other hand, the original medical images must be recovered without error.

© Springer International Publishing AG 2018
L. Barolli et al. (eds.), *Advances in Internetworking, Data & Web Technologies*,
Lecture Notes on Data Engineering and Communications Technologies 6,
DOI 10.1007/978-3-319-59463-7_14

Some classic reversible data hiding algorithms including difference expansion [7], histogram shifting [8] and redundancy compression [9] are not so suitable for encrypted covers as unencrypted covers. Zhang [10] proposed the first EIRDH algorithm with flipping pixel values. He embedded additional data in the image encrypted by stream cipher, and recovered the original content using the correlation between pixels. An improved version of Zhang's method was proposed by Hong et al. [11], but the algorithm does not work when the block size is small. Lots of EIRDH algorithms have been present [12–17] to improve embedding payload and image quality.

However, symmetric cryptosystem based EIRDH algorithms have the drawbacks such as difficulty of key management and unsuitable for multi-party computation problems, while encrypted image-based reversible data hiding with public key cryptography (EIRDH-P) is a natural issue. The first EIRDH-P algorithms is proposed by Chen et al. [18], which encrypts image using the public key and decrypts embedded image by the secret key of receiver. Each pixel is divided into an even integer and a bit, and both of them are encrypted by homomorphic encryption. In the embedding phase, the second parts of two adjacent pixels are modified to embed a bit. Due to the shared key, Chen's scheme no longer depends on a secure channel among the image provider, the data-hider and the receiver, but it has the inherent overflow since the summation of two adjacent pixel values may be overflow. Some improved EIRDH-P algorithms [19, 20] based on additive homomorphic encryption are proposed, but these schemes provides low embedding capacity, and the directly decrypted images is distorted significantly.

To overcome the weakness of the above schemes, Shiu et al. [21] constructs an efficient EIRDH-P scheme from difference expansion (DE). In this scheme, a preprocessing is needed so as to vacate room for data embedding procedure. Then, by side information and pixel difference expansion, the additional bits are hidden. Concerning the additive homomorphic encryption, Shiu et al. embed additional data in decrypted domain by vacating room for hiding data before encryption. However, the scheme does not fully exploits the correlation inherent in the neighborhood of a pixel and the side information required to record the location map can considerably lower embedding capacity.

In this paper, a new EIRDH-P scheme is introduced, in which one quarter of the total pixels are used to predict other pixels based on interpolation technique and the interpolation-error expansion is adopted to embed additional data. Then, the embedding data can be extracted perfectly and the original images can be losslessly recovered. In general, the proposed scheme obtains excellent performance compared with the existing algorithms.

The rest of this paper is organized as follows. In Sect. 2, some preliminaries are introduced. The proposed EIRDH-P scheme is shown in Sect. 3. In Sect. 4, the experimental results are provided. Finally, conclusions of our work are given in Sect. 5.

2 Preliminaries

2.1 Prediction Error Expansion

Prediction error expansion (PEE) [22–24] is a new approach firstly proposed by Thodi et al. [22] to improve the difference expansion (DE). Here we review this method. For a pixel $I(i,j)$, let $I^*(i,j)$ be the predicted pixel value derived from a prediction algorithm, then the prediction error is defined as follows:

$$e = I(i,j) - I^*(i,j) \tag{1}$$

If the additional bit to be embedded is $w \in \{0, 1\}$, the expansion and embedding process is described by

$$e^* = 2e + w \tag{2}$$

where e^* is the new prediction error. Then, the new pixel value after embedding is

$$I_e = I^*(i,j) + e^* \tag{3}$$

It is easy to show that

$$I_e = 2I(i,j) - I^*(i,j) + w \tag{4}$$

After receiving I_e, the receiver compute the predicted value $I^*(i,j)$ using the same prediction algorithm, and compute

$$I_e^* = I_e - I^*(i,j) = 2I(i,j) - 2I^*(i,j) + w \tag{5}$$

Since $I(i,j)$ and $I^*(i,j)$ are both integers, I_e^* and w have the same parity. The extraction of data can be cast as

$$w = \begin{cases} 0, & \text{if } I_e^* \bmod 2 = 0 \\ 1, & \text{if } I_e^* \bmod 2 = 1 \end{cases} \tag{6}$$

Then the image recovery process can be described by

$$I(i,j) = \frac{I_e^* - w}{2} + I^*(i,j) \tag{7}$$

2.2 Paillier Encryption

Homomorphic encryption [25, 26] is a very useful tool that allows computations to be carried out on ciphertext. However, the decrypted results matches the results of operations performed on the plaintext. Paillier encryption [27] is a classical homomorphic encryption with additive homomorphic property. The algorithm can be described as follows.

Select two large primes a and b, and computes

$$p = a \cdot b \tag{8}$$

$$\lambda = lcm(a - 1, b - 1) \tag{9}$$

where $lcm(x, y)$ means the least common multiple of x and y. The private key is λ, and the public key is composed of p and a randomly selected integer g. If the plaintext is m, then it can be encrypted by

$$c = E[m, r] = g^m \cdot r^p \bmod p^2 \tag{10}$$

where r represents a randomly selected small integer. The decryption process can be described as

$$D(c) = \frac{L(c^\lambda \bmod p^2)}{L(g^\lambda \bmod p^2)} \bmod p \tag{11}$$

where $L(\bullet)$ is defined as

$$L(u) = \frac{u - 1}{p} \tag{12}$$

The additive homomorphic property of Paillier encryption can be shown that

$$D[E[m_1, r_1] \bullet E[m_2, r_2] \bmod p^2] = (m_1 + m_2) \bmod p^2 \tag{13}$$

3 The Proposed Algorithm

In this section, the details of the proposed reversible data hiding algorithm in encrypted images using interpolation-error expansion and homomorphic encryption are illustrated, which is made up of image provider, data-hide and receiver. First of all, a preprocessing is employed to vacate room for data embedding procedure and the image is encrypted with public key based on homomorphic cryptosystem. Then, the data-hider embeds some additional data into the carrier image. The receiver can perfectly extract embedding data and obtain the recover image at last. Figure 1 shows the sketch of the proposed EIRDH-P.

3.1 Preprocessing

We assume that the original image I is a 8 bit grayscale image of size $N \times M$, and all pixel values belong to the range $[0, 255]$. Represent each pixel value with $X(i,j)$, $1 \leq i \leq N$, $1 \leq j \leq M$. In our work, an interpolation technique in [28] is adopted for pixel prediction. We classify all pixels in the original image into two sets: sample

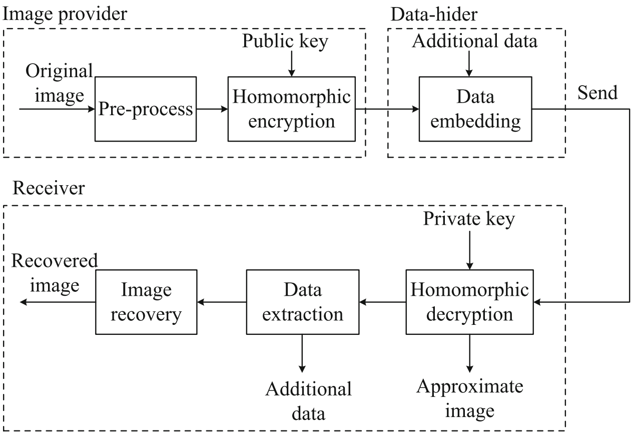


Fig. 1. Sketch of the proposed EIRDH-P scheme

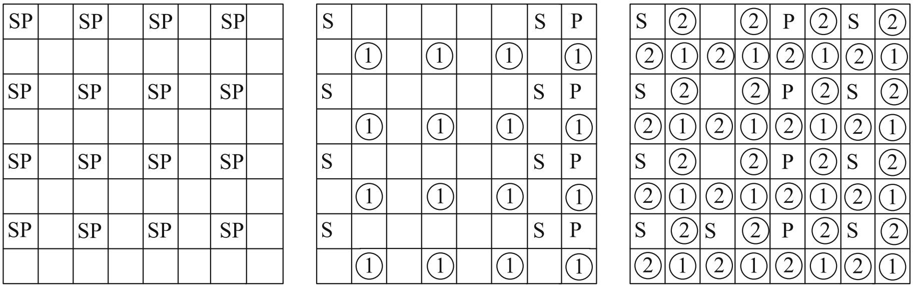
pixels (SP) and non-sample pixels (NSP). Then, the pixels in the set of SP consists of $X(2n-1, 2m-1)$ with $n = 1, 2, \ldots, N/2$, $m = 1, 2, \ldots, M/2$, as depicted in Fig. 2(a). The sample pixels are used to predict non-sample pixels. The prediction process consists two rounds. In the first round, the pixels $X(2n, 2m)$ marked as '①' in Fig. 2(b) can be estimated by the four nearest sample pixels $X(2n-1, 2m-1)$, $X(2n-1, 2m+1)$, $X(2n+1, 2m-1)$, $X(2n+1, 2m+1)$. We compute two prediction value $X_{45}(2n, 2m)$ and $X_{135}(2n, 2m)$ along two orthogonal directions: 45° diagonal and 135° diagonal by

$$X_{45}(2n, 2m) = (X(2n-1, 2m+1) + X(2n+1, 2m-1))/2 \tag{14}$$

$$X_{135}(2n, 2m) = (X(2n-1, 2m-1) + X(2n+1, 2m+1))/2 \tag{15}$$

Select an optimal pair of weights w_{45} and w_{135} to give a good estimate value $X^{*}(2n, 2m)$ with

$$X^{*}(2n, 2m) = w_{45} \cdot X_{45}(2n, 2m) + w_{135} \cdot X_{135}(2n, 2m) \tag{16}$$



SP		SP		SP		SP	
SP		SP		SP		SP	
SP		SP		SP		SP	
SP		SP		SP		SP	

(a) Sample pixels

S						S	P
	①		①		①		①
S						S	P
	①		①		①		①
S						S	P
	①		①		①		①
S						S	P
	①		①		①		①

(b) The first round of prediction

S	②		②	P	②	S	②
②	①	②	①	②	①	②	①
S	②		②	P	②	S	②
②	①	②	①	②	①	②	①
S	②		②	P	②	S	②
②	①	②	①	②	①	②	①
S	②	S	②	P	②	S	②
②	①	②	①	②	①	②	①

(c) The second round of prediction

Fig. 2. Illustration of image interpolation

According to [28],

$$w_{45} = \frac{\sigma_{135}}{\sigma_{135} + \sigma_{45}},\ w_{135} = 1 - w_{45} \tag{17}$$

where

$$\begin{cases} \sigma_{45} = \frac{1}{3}\sum\limits_{k=1}^{3} \left(S_{45}(k) - u\right)^2 \\ \sigma_{135} = \frac{1}{3}\sum\limits_{k=1}^{3} \left(S_{135}(k) - u\right)^2 \end{cases} \tag{18}$$

and

$$\begin{cases} S_{45} = \{\mathrm{X}(2n-1, 2m+1),\ X_{45}(2n, 2m),\ X(2n+1, 2m-1)\} \\ u = \frac{1}{4}(\mathrm{X}(2n-1, 2m+1) + X(2n+1, 2m-1) + \mathrm{X}(2n-1, 2m-1) + X(2n+1, 2m+1)) \\ S_{135} = \{\mathrm{X}(2n-1, 2m-1), X_{135}(2n, 2m), X(2n+1, 2m+1)\} \end{cases} \tag{19}$$

In the second round, the non-sample pixels $X(2n-1, 2m)$ and $X(2n, 2m-1)$ marked as '②' in Fig. 2(c) can be estimated by the four nearest pixels along two orthogonal directions: 0° diagonal and 90° diagonal by the same method. After two prediction rounds, the predicted values of all the non-sample pixels can be obtained. Assume the original pixels value and the predicted pixels value are respectively $X(i,j)$ and $X^*(i,j)$, then the interpolation-error is

$$e(i,j) = X(i,j) - X^*(i,j) \tag{20}$$

Since overflow or under flow will happen when the interpolation-error is high, we set a parameter θ to overcome this problem. Then, the preprocess of non-sample pixels can be described as

$$X_p^*(i,j) = \begin{cases} 2X(i,j) - X^*(i,j), & |e(i,j)| \le \theta \\ X(i,j), & |e(i,j)| > \theta \end{cases} \tag{21}$$

where $X_p^*(i,j)$ is the new non-sample pixel after preprocessing. However, all the sample pixels remain unchanged after preprocessing.

3.2 Encryption and Embedding

Since the operation of data embedding based on interpolation-error mainly includes the addition, our algorithm adopts Paillier encryption. Let $X_p^*(i,j)$ be pixel after preprocessing, and the encrypted pixel is calculated by

$$c(i,j) = E[X_p^*(i,j), r(i,j)] = g^{X_p^*(i,j)} \cdot (r(i,j))^p \bmod p^2 \tag{22}$$

where $r(i,j)$ is a randomly selected small integer, and $c(i,j)$ is the encrypted pixel. To embed data by $c(i,j) + m(i,j)$, the corresponding operation in encrypted domain is

$$c_e(i,j) = \begin{cases} c(i,j) \cdot g^{m(i,j)} \cdot (r_e(i,j))^p \text{ mod } p^2, & (i,j) \in SP \text{ and } |e(i,j)| \le \theta \\ c(i,j), & (i,j) \in NSP \end{cases} \tag{23}$$

where $m(i,j) \in \{0,1\}$ is the additional message, $r_e(i,j)$ is a randomly selected small integer, and $c_e(i,j)$ is the pixel value after data embedding.

3.3 Data Extraction and Image Recovery

After receiving the encrypted image, the receiver need decrypt the image using The private key λ firstly. Assume the pixels before and after decryption is $c_e(i,j)$ and $m^*(i,j)$ respectively, then the decryption process is

$$m^*(i,j) = \frac{L([c_e(i,j)]^\lambda \bmod p^2)}{L(g^\lambda \text{ mod } p^2)} \bmod p \tag{24}$$

where $L(\bullet)$ is defined as

$$L(u) = \frac{u-1}{p} \tag{25}$$

As the embedding distortion is very little, the image directly after decrypting can be used as an approximate image in some special scenarios. According to Eqs. (21)–(24), the relationship between $m^*(i,j)$ and $X(i,j)$ is

$$m^*(i,j) = \begin{cases} 2X(i,j) - X_p(i,j) + m(i,j) & (i,j) \in NSP \\ X(i,j) & (i,j) \in SP \end{cases} \tag{26}$$

The receiver can obtain the same predicted non-sample pixel values using the same method, since the sample pixels are not embedded and remain unchanged after decryption. For the non-sample pixel $m^*(i,j)$, $(i,j) \in NSP$, the new interpolation-error $m_t^*(i,j)$ can be compute as follows

$$m_t^*(i,j) = m^*(i,j) - X_p(i,j) = 2X(i,j) - 2X_p(i,j) + m(i,j) \tag{27}$$

Then the data extraction process can be described as

$$m(i,j) = \begin{cases} 0, & m_t^*(i,j) \bmod 2 = 0 \\ 1, & m_t^*(i,j) \bmod 2 = 1 \end{cases} \tag{28}$$

and the original value of the non-sample pixel can be recovered by

$$X(i,j) = \begin{cases} \frac{m_t^*(i,j) - m(i,j)}{2} + X_p(i,j), & (i,j) \in NSP \\ m_t^*(i,j), & (i,j) \in SP \end{cases} \tag{29}$$

4 Experimental Results

The proposed method is verified in this section with the experimental environment of MATLAB R2012b under windows 7. The test is conducted on a 2.6 GHz, Intel(R) Core(TM) i7-6700 HQ system with 8 GB RAM running. We measure the embedding capacity and image quality respectively by *Payload* (bpp) and *PSNR* (Peak signal-to-noise ratio, dB). The *Payload* is the proportion of the total number of embedded bits to the total number of pixels in original image. Moreover, we consider *PSNR* and *MSE* as

(a) (b) (c)

(d) (e) (f)

Fig. 3. Comparisons of payload and PSNR among original Lena, directly decrypted Lena with different values of θ, and recovered Lena. (a) Original image. (b) $\theta = 0$, 0.09 bpp, 61.45 dB. (c) $\theta = 4$, 0.56 bpp, 43.50 dB. (d) $\theta = 7$, 0.66 bpp, 40.13 dB. (e) $\theta = 31$, 0.75 bpp, 34.22 dB. (f) Recovered image, PSNR = +∞

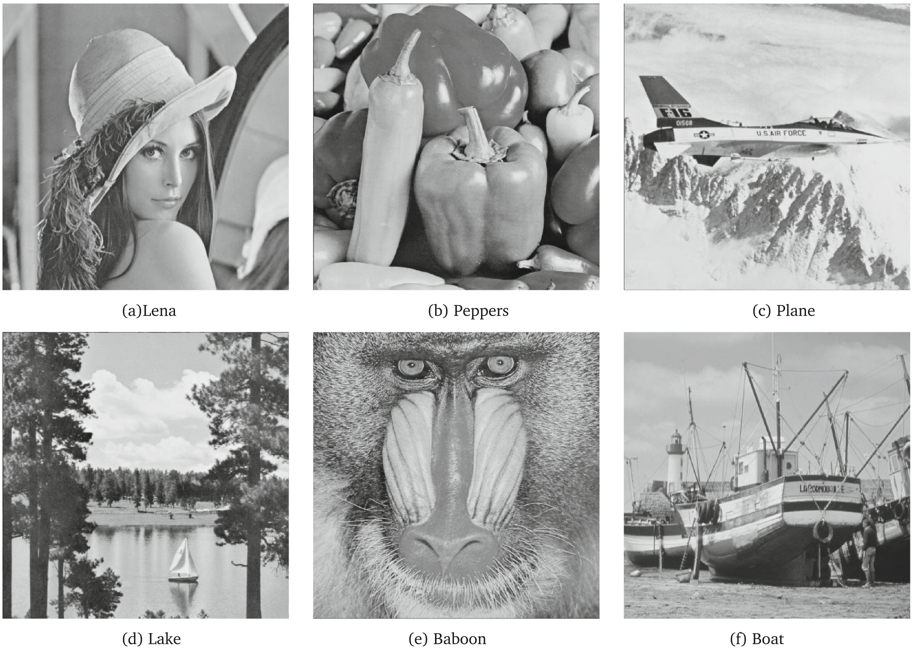
Table 1. Experimental results with different values of θ

Value of θ		0	1	2	3	4	5	7	10	20	30	40	50
Lena	*payload*	0.093	0.261	0.394	0.491	0.558	0.603	0.657	0.695	0.736	0.746	0.749	0.750
	PSNR	61.44	53.39	48.62	45.56	43.49	42.03	40.13	38.46	35.61	34.32	33.67	33.39
Peppers	*payload*	0.067	0.195	0.312	0.410	0.489	0.55	0.631	0.689	0.734	0.743	0.746	0.748
	PSNR	62.92	54.6	49.38	45.87	43.4	41.56	39.12	37.14	34.82	33.86	33.26	32.79
Plane	*payload*	0.134	0.338	0.459	0.533	0.578	0.609	0.648	0.680	0.725	0.739	0.744	0.745
	PSNR	59.87	52.33	48.39	45.95	44.33	43.12	41.28	39.45	35.83	34.02	33.13	32.66
Lake	*payload*	0.060	0.170	0.260	0.333	0.394	0.445	0.525	0.604	0.704	0.732	0.743	0.747
	PSNR	63.37	55.25	50.35	47.00	44.51	42.57	39.7	36.9	32.75	30.99	30.03	29.52
Baboon	*payload*	0.028	0.082	0.135	0.185	0.230	0.271	0.341	0.422	0.573	0.649	0.693	0.719
	PSNR	66.70	58.30	52.92	49.09	46.22	43.99	40.66	37.31	31.43	28.37	26.37	25.07
Boat	*payload*	0.089	0.248	0.368	0.448	0.501	0.538	0.590	0.641	0.716	0.738	0.745	0.748
	PSNR	61.68	53.62	48.99	46.13	44.21	42.79	40.64	38.26	34.01	32.19	31.28	30.78

$$PSNR = 10\log_{10}\left(\frac{255^2}{MSE}\right) \tag{30}$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(p_i - p_i^*)^2 \tag{31}$$

where n is the total number of pixels in original image, p_i and p_i^* are respectively the original pixel value and embedded pixel value. We firstly choose the 8-bit grayscale



(a)Lena

(b) Peppers

(c) Plane

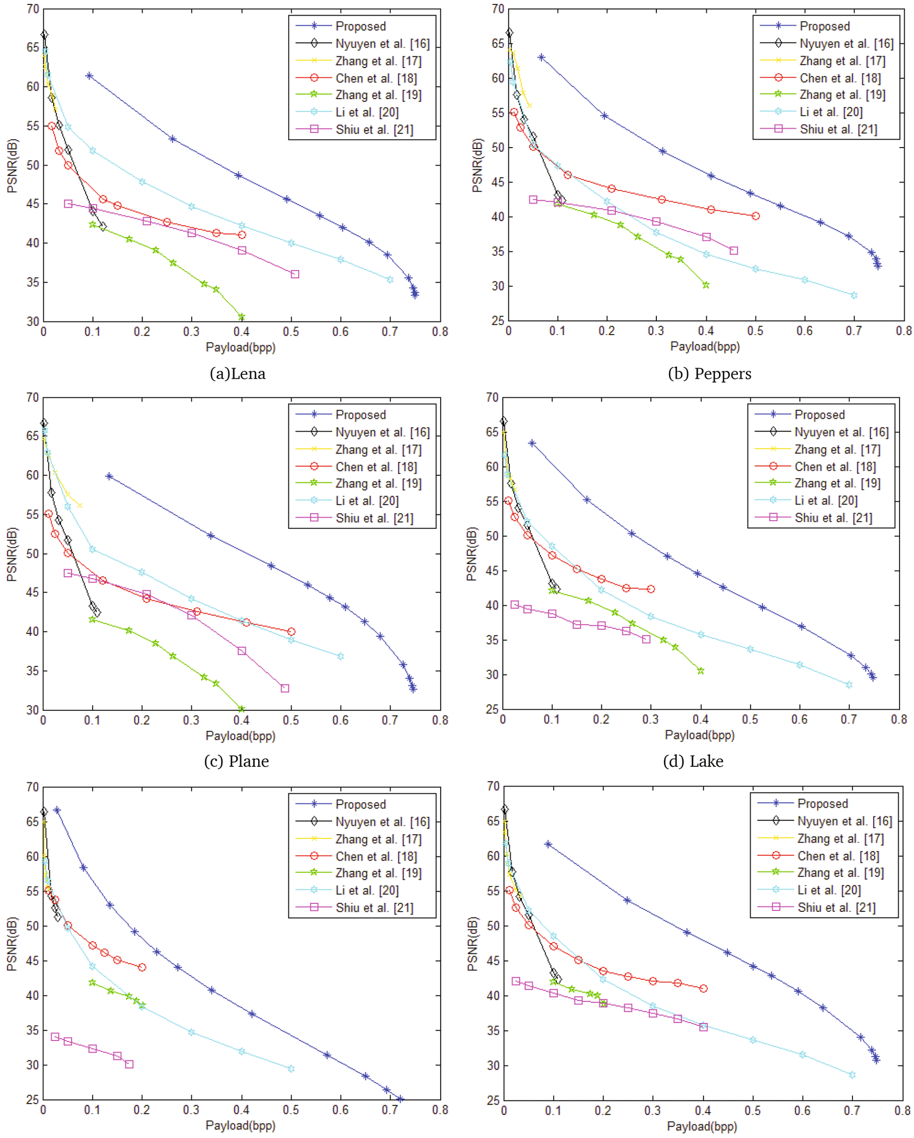
(d) Lake

(e) Baboon

(f) Boat

Fig. 4. Six 512 × 512 grayscale test images

image *Lena* of size 512 × 512 to verify the feasibility of the proposed algorithm. Figure 3 shows the experimental comparisons among original Lena, directly decrypted Lena, and recovered Lena. Here, we set respectively $\theta = 0$, $\theta = 4$, $\theta = 7$, $\theta = 31$ and get different *Payload* and *PSNR*. The notation "+∞" shows that the original image can be reconstructed perfectly without any distortion. Moreover, the receiver can coordinate the relationship between embedding capacity and image quality by parameter θ.



(a)Lena

(b) Peppers

(c) Plane

(d) Lake

(e) Baboon

(f) Boat

Fig. 5. Comparisons of the performances of different schemes on six images

Table 1 lists the *Payload* and *PSNR* of the directly decrypted images when different values of θ are used for six standard gray text images shown in Fig. 4. The parameter θ is used to determine the *Payload* and *PSNR* of the directly decrypted images. As can be seen in Table 1, the higher value of θ, the higher *Payload* and the lower *PSNR* we will get.

Further, we compare the experimental results of the proposed algorithm and six existing schemes in [16–21]. The comparison results are shown in Fig. 5. The parameter θ is used to determine the *Payload* and *PSNR* of the directly decrypted images. By observing the results, the proposed algorithm is better than the existing schemes with respect to the embedding capacity and image quality.

5 Conclusions

This work proposes a EIRDH-P algorithm with interpolation-error expansion and homomorphic encryption. On the one hand, the existing scheme introduces obvious distortion when the embedding date is high while the proposed method improves the embedding capacity and image quality of the directly decrypted image. On the other hand, the proposed method overcomes the overflow or underflow problem and does not need side information. Meanwhile, the additional data can be only extracted after image decryption, which is not flexible enough. In the future, the research on homomorphic encrypted algorithm will be carried on to study separable EIRDH-P algorithm with interpolation-error expansion.

References

1. Shi, Y., Li, X., Zhang, X., et al.: Reversible data hiding: advances in the past two decades. IEEE Access (2016). doi:10.1109/ACCESS.2016.2573308
2. Wang, J., Ni, J., Zhang, X., et al.: Rate and distortion optimization for reversible data hiding using multiple histogram shifting. IEEE Trans. Cybern. (2016). doi:10.1109/TCYB.2015.2514110
3. Ma, B., Shi, Y.: A reversible data hiding scheme based on code division multiplexing. IEEE Trans. Inf. Secur. Forensics **11**(9), 1914–1927 (2016)
4. Qian, Z., Zhang, X.: Reversible data hiding in encrypted images with distributed source encoding. IEEE Trans. Circuits Syst. Video Technol. **26**(4), 636–646 (2016)
5. Zhang, W., Wang, H., Hou, D., et al.: Reversible data hiding in encrypted images by reversible image transformation. IEEE Trans. Multimedia. doi:10.1109/TMM.2016.2569497
6. Wu, H., Shi, Y., Wang, H., et al.: Separable reversible data hiding for encrypted palette images with color partitioning and flipping verification. IEEE Trans. Circuits Syst. Video Technol. (2016). doi:10.1109/TCSVT.2016.2556585
7. Tian, J.: Reversible data embedding using a difference expansion. IEEE Trans. Circuits Syst. Video Technol. **13**(8), 890–896 (2003)
8. Dragoi, L., Coltuc, D.: Local-prediction-based difference expansion reversible watermaking. IEEE Trans. Image Process. **23**(4), 1779–1790 (2014)
9. Jarali, A., Rao, J.: Unique LSB compression data hiding method. Int. J. Emerg. Sci. Eng. **2** (3), 17–21 (2013)

10. Zhang, X.: Reversible data hiding in encrypted image. IEEE Signal Process. Lett. **18**(4), 255–258 (2011)
11. Hong, W., Chen, T., Wu, H.: An improved Reversible data hiding in encrypted images using side match. IEEE Signal Process. Lett. **19**(4), 199–202 (2012)
12. Ma, K., Zhang, W., Zhao, X., et al.: Reversible data hiding in encrypted images by reserving room before encryption. IEEE Trans. Inf. Secur. Forensics **8**(3), 553–562 (2013)
13. Zhou, J., Sun, W., Dong, L., et al.: Secure reversible image data hiding over encrypted domain via key modulation. IEEE Trans. Circuits Syst. Video Technol. **26**(3), 441–452 (2016)
14. Cao, X., Du, L., Wei, X., et al.: High capacity reversible data hiding in encrypted images by patch-level sparse representation. IEEE Trans. Cybern. **46**(5), 1132–1143 (2016)
15. Xu, D., Wang, R.: Separable and error-free reversible data hiding in encrypted imaged. Signal Process. **123**, 9–21 (2016)
16. Nyuyen, T., Chang, C., Chang, W.: High capacity reversible data hiding scheme for encrypted images. Signal Process. Image Commun. **44**, 52–64 (2016)
17. Zhang, W., Ma, K., Yu, N.: Reversibility improved data hiding in encrypted images. Signal Process. **94**(1), 118–127 (2014)
18. Chen, Y., Shiu, C., Horng, G.: Encrypted signal-based reversible data hiding with public key cryptosystem. J. Vis. Commun. Image Represent. **25**, 1164–1170 (2014)
19. Zhang, X., Long, J., Wang, Z., Cheng, H.: Lossless and reversible data hiding in encrypted images with public key cryptography. IEEE Trans. Circuits Syst. Video Technol. (2015). doi:10.1109/TCSVT.2015.2433194
20. Li, M., Xiao, D., Zhang, Y., Nan, H.: Reversible data hiding in encrypted images using cross division and additive homomorphism. Signal Process. Image Commun. **39**, 234–248 (2015)
21. Shiu, C., Chen, Y., Hong, W.: Encrypted image-based reversible data hiding with public key cryptosystem from difference expansion. Signal Process. Image Commun. **39**, 226–233 (2015)
22. Thodi, D.M., Rodriguez, J.: Expansion embedding techniques for reversible watermarking. IEEE Trans. Image Process. **16**(3), 721–730 (2007)
23. Dragoi, I., Coltuc, D.: On local prediction based reversible watermarking. IEEE Trans. Image Process. **24**(4), 1244–1246 (2015)
24. Xiang, S., Wang, Y.: Non-inter expansion embedding techniques for reversible image watermarking. EURASIP J. Adv. Signal Process. **2015**, 56–68 (2015)
25. Aguilar, C., Fau, S., Fontaine, C., Gogniat, G.: Recent advances in homomorphic encryption: a possible future for signal processing in the encrypted domain. IEEE Signal Process. Mag. **30**(2), 108–117 (2013)
26. Wang, W., Hu, Y., Chen, L., Huang, X., Sunar, B.: Exploring the feasibility of fully homomorphic encryption. IEEE Trans. Comput. **64**(3), 698–706 (2015)
27. Paillier, P.: Public-key cryptosystems based on composite degree residuosity classes. In: Proceedings of the International Conference on the Theory and Application of Cryptographic Techniques Prague, Czech Republic, pp. 223–238 (1999)
28. Luo, L., Chen, Z., Chen, M., Zeng, X., Xiong, Z.: Reversible image watermarking using interpolation technique. IEEE Trans. Inf. Secur. Forensics **5**(1), 187–193 (2010)