International Core Journal of Engineering Volume 6 Issue 12, 2020
ISSN: 2414-1895 DOI: 10.6919/ICJE.202012_6(12).0015

Research on Digital Watermarking Algorithm based on Homomorphic Ciphertext Domain

Lijiao Fang

Beijing Institute of Graphic Communication, Beijing 102600, China.

jao_flj@163.com

Abstract

Aiming at the needs of multimedia information security and copyright protection, this paper combines ElGamal public key cryptography system with Patchwork digital watermarking algorithm, and proposes a homomorphic ciphertext domain digital watermarking algorithm. The new algorithm, using the multiplicative homomorphism of ElGamal, maps the operation of embedding watermark in plaintext domain to embedding watermark in ciphertext domain, and realizes that the operation of image encryption and watermark embedding can be exchanged. In the process of watermark extraction, the ciphertext domain is implemented. The experimental results show that the order of embedding watermark and data encryption does not affect the generation of encrypted watermark, and the extraction of watermark in ciphertext also ensures the security of multimedia data distribution management.

Keywords

ElGamal password algorithm, homomorphic encryption, Patchwork watermarking algorithm, Information security.

1. Introduction

With the development and popularization of Internet technology and cloud computing technology, users can upload massive amounts of data to the cloud for storage, and it is also convenient to download and use it when needed [1]. However, while cloud technology brings convenience to people, people also have to consider and attach importance to data security and privacy protection.

People can encrypt their data before uploading it to the cloud, which can effectively protect private data and reduce the risk of privacy infringement or theft. But how to effectively manage the uncountable multimedia data in the cloud is another issue that needs to be considered. After the user's data is encrypted, the cloud administrator can protect and manage the user's data by embedding the user's relevant information in the ciphertext, and pass Extracting embedded user information for ciphertext retrieval and identification greatly improves the efficiency of cloud management. In fact, encryption technology and watermarking technology are often combined when protecting multimedia data [2]. However, in some areas with relatively high security requirements, such as medicine, military, legal documents, and copyright protection, it is necessary to encrypt data first[3], and then embed watermark information to ensure the security of the information to be processed. In 1988, people have proposed many feasible ciphertext domain watermarking technologies. In the encryption algorithm, due to the unique homomorphic characteristics of the homomorphic cipher algorithm, the algebraic relationship between the plaintext data and the ciphertext data before and after encryption [4 -6], researchers have been constantly looking for more suitable homomorphic encryption domain watermarking algorithms to provide more secure and feasible algorithms for signal processing.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202012_6(12).0015

Literature [7] first proposed a framework for performing discrete wavelet transform and inverse discrete wavelet transform (IDWT) in the encryption domain. Taking two-dimensional Haar wavelet transform as an example, experiments were carried out to prove the effectiveness of this method in secure image processing. Superiority, and provides computational complexity analysis and comparison. In [8], a public key homomorphism-based RDH-EI (Reversible Data Hiding in Encrypted Image) scheme was proposed. Literature [9] improves the robust performance of the encryption domain watermarking scheme based on hybrid discrete wavelet transform (DWT) and discrete cosine transform (DCT), and gives an estimate of the expansion factor after the watermark is embedded in the encryption domain. In the literature [10], a reversible data hiding scheme based on the public key cryptosystem in the homomorphic encryption domain is proposed, which can complete the data extraction and image recovery operations well. Compared with the literature [10], the hidden data proposed in the literature [11] has stronger security performance in the encryption domain, because the data is embedded in the plaintext by homomorphism, rather than embedded in the bit plane of the ciphertext. Literature [12] uses the homomorphic characteristics of the cryptographic algorithm to realize the mapping between the ciphertext domain and the plaintext domain of the multimedia information data. The sequence of watermark embedding and data encryption does not affect the generation of the same encrypted watermark data, The watermark can be extracted without decryption. Nevertheless, it is still possible to improve the performance of the watermarking scheme in the encryption domain, increase the robustness of the watermark and seek efficient algorithms that can reduce the computational cost of the watermarking scheme.

At present, there are many existing watermarking algorithms, but there are still some key problems in the research of digital watermarking algorithms that need to be solved. If the multimedia data is encrypted before the watermark is embedded, the embedded watermark will cause the ciphertext to be unable to be decrypted to the plaintext, which affects the usability of the data. On the other hand, embedding the watermark first, and then encrypting the data, this sequence cannot directly extract the watermark information in the ciphertext domain. Therefore, it not only makes the decryption step of watermark extraction redundant, but also exposes the plaintext data to the detection environment. Reduce the security in the process of multimedia data distribution. Based on the above problems, this paper uses the multiplicative homomorphism of the ElGamal cipher algorithm on the basis of previous research, and combines the Patchwork watermark algorithm [13] to map the plaintext watermark algorithm to the ciphertext domain through homomorphism to realize the ciphertext watermark The embedding and the plaintext watermark are embedded in the ciphertext domain. Solve the problem of mutual influence between encryption and digital watermark, ensure that when data is distributed and managed, the encryption and watermark embedding operations of the data copyright owner and the manager are not restricted by the sequence, and the function of directly detecting the watermark in the ciphertext state of the data is realized. To ensure that the algorithm has higher security.

2. Basic theory of algorithm

This section will introduce the homomorphic characteristics of the ElGamal cryptosystem, the Patchwork watermarking algorithm, and the basic principles of the homomorphic encryption domain watermarking algorithm designed in this article.

2.1 ElGamal homomorphic encryption algorithm

ElGamal algorithm is a relatively common encryption algorithm, which is based on the public key cryptosystem and elliptic curve encryption system proposed in 1984. It can be used for both data encryption and digital signature, and its security depends on the difficulty of computing discrete logarithms in a finite field.

Parameter generation: Let G be the multiplicative group of finite field Z_p, p is a prime number, and g is a generator on Z_p,And $g \in Z_P^*$.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202012_6(12).0015

Key generation: Select x∈[1,p-1], Calculation $y \equiv g^x (modp)$.Then get the private key as*x*, The public key is （y, g, p）.

Encryption process: For encrypted message *m*, Random numbers can be selected arbitrarily r∈[1,p−1], Calculation $c_1 = g^r (modp)$ and $c_2 \equiv mg^r (modp)$, Can get ciphertext $E(\mathrm{m}) = (c_1, c_2)$

Decryption process: After the receiver receives the ciphertext $E(\mathrm{m}) = (c_1, c_2)$, Use private key *x*, Calculation:

$$m = D(E(m)) = c_2 (c_1^{\ x})^{-1} modp$$

Encrypt the plaintext m_1、m_2, Calculated $E(m_1) = (g^{r_1} modp, m_1 y_1^{r_1} modp)$,

$E(m_2) = (g^{r_2} modp, m_2 y_1^{r_2} modp)$, then $E(m_1)E(m_2) = (g^{r_1+r_2} modp, m_1 m_2 y_1^{r_1+r_2} modp)$

$D(E(m_1)E(m_2)) = m_1 m_2 y_1^{r_1+r_2} \ [(g^{r_1+r_2})^x]^{-1} (modp) = m_1 m_2$

Therefore, the ElGamal cryptosystem has multiplicative homomorphism.

2.2 Patchwork watermarking algorithm

The Patchwork watermarking algorithm uses the statistical characteristics of the carrier data to embed the watermark. The core idea is to select some data from the carrier data to form two sets, and then embed the watermark by modifying a certain relationship between the two sets. The two sets can be two/ group coefficients or two feature quantities. The relationship between the sets can be the size/ energy/parity relationship, and the watermark information is extracted according to the corresponding relationship when extracting the watermark. The steps of the general Patchwork watermarking algorithm can be described as follows:

(1) Select some data from the carrier data to form two sets $A = \{a_i\}$, $B = \{b_i\}$, $i \in [1, N]$. It is required that A and B contain the same image coefficient, set to N.

(2) Increase all pixels in set A by d, and decrease all pixels in set B by d.

(3) Detect whether there is a watermark by calculating the mean *Exp*[*s*]:

$$Exp[s] = \sum_{i}^{N} (Exp[a_i^w] - Exp[b_i^w]) \tag{1}$$

among them, $a_i^w = a_i + d$, $b_i^w = b_i - d$

The watermark extraction algorithm is:

$$s = \sum_{i}^{N} (a_i^w - b_i^w) \tag{2}$$

Among them, when s≈2N, it means the carrier with watermark; s≈0, it means the watermark is embedded in the carrier without watermark。

2.3 Basic principles of the algorithm

On this basis, combined with the ElGamal algorithm, the improved Patchwork watermark algorithm steps can be designed as follows:

(1) Select some data from the carrier data to form two sets $A = \{a_i\}$, $B = \{b_i\}$, $i \in [1, N]$,and it is required that A and B contain the same image coefficient, set to N.

(2) Change all the pixels in the set A by λ times, and change all the pixels in the set B by λ^{-1} times. Here λ is close to 1, which is obtained by balancing the watermark extraction rate and the image quality after embedding the watermark.

(3) Watermark detection

ISSN: 2414-1895 DOI: 10.6919/ICJE.202012_6(12).0015

$$Exp[s]=\sum_{i}^{N}\frac{Exp[a_i^w]}{Exp[b_i^w]}\approx\lambda^2 N \tag{3}$$

The value of $Exp[s]$ will determine whether there is a carrier watermark, $Exp[s]\approx\lambda^2 N$, contains a watermarked carrier, $Exp[s]\approx N$, means a non-watermarked carrier.

3. Homomorphic encryption watermarking algorithm based on ElGamal and Patchwork

By combining the homomorphic encryption algorithm with the Patchwork watermark algorithm, the extraction of the watermark after decryption is realized, and the watermark in the ciphertext domain can also be extracted. The overall structure of the algorithm is shown in Figure 1 below.

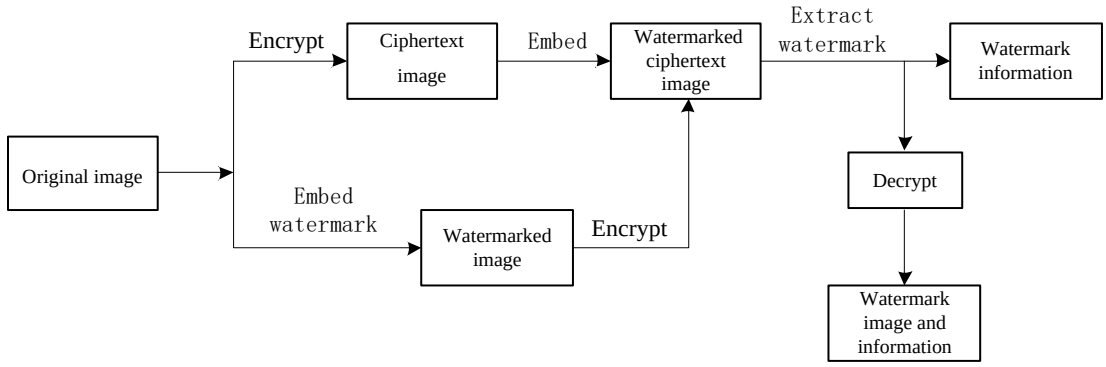


Figure 1 The overall structure of the algorithm

3.1 Algorithm overall scheme

When designing the algorithm, we can use the multiplicative homomorphism feature of the ElGamal homomorphic encryption algorithm to construct a scheme to embed a plaintext watermark in the paired ciphertext,

$$\begin{cases}\bar{a}_i^{ew}=mod(1/\lambda E(a_i^w,r_{a_1}^i k_p),p)\\ \bar{b}_i^{ew}=mod(\lambda(E(b_i^w,r_b^i,k_p)),p)\end{cases} \tag{4}$$

Among them a_i^w, b_i^w are plaintext with watermark, $\bar{a}_i^{ew}, \bar{b}_i^{ew}$ are the encrypted watermark carrier. E(·) is an encryption function.

The extraction algorithm after decryption is:

$$s=\sum_{i=1}^{N}[D(\bar{a}_i^{ew},k_S)/D(\bar{b}_i^{ew},k_S)] \tag{5}$$

When $s\approx\frac{1}{\lambda^2}N$, it means there is a carrier watermark, when s≈N, it means there is no carrier watermark.

Since the same plaintext corresponds to multiple ciphertexts in the ElGamal cryptosystem, the following situations exist:

$$s=\sum_{i=1}^{N}[(\bar{a}_i^{ew},k_S)/(\bar{b}_i^{ew},k_S)]\approx N \tag{6}$$

Therefore, the watermark cannot be extracted directly in the ciphertext state. In order to successfully extract the watermark in the ciphertext state, adjust the value of r to obtain a suitable ciphertext that satisfies the established relationship, and realize the watermark from the plaintext domain to the ciphertext domain watermark. Mapping, adjust the watermark embedding algorithm to:

$$\begin{cases}\hat{a}_i^{ew}=(E(a^w,r_a^i,k_p)\\ \hat{b}_i^{ew}=(E(b^w,r_b^i,k_p)\end{cases}\Big|\hat{a}_i^{ew}>\hat{b}_i^{ew} \tag{7}$$

ISSN: 2414-1895 DOI: 10.6919/ICJE.202012_6(12).0015

The basis for judging whether the watermark is embedded is

$$s = \sum_{i=1}^{N} \delta(\hat{a}_i^{ew}, \hat{b}_i^{ew}) = 0 \quad (8)$$

among them $\delta(a,b) = \begin{cases} 0, & a > b \\ 1, & a \le b \end{cases}$ Indicates that it contains a carrier watermark. Due to the randomness of encryption, no watermark is added $s \approx N$.

Based on the above discussion, the CEW scheme can be written as follows:

Watermark embedding algorithm:

$$\begin{cases} a_i^{ew} = mod(1/\lambda(E(a, r_a^i, k_p)), p) \\ b_i^{ew} = mod(\lambda E(b, r_b^i, k_p), p) \end{cases} | a_i^{ew} > b_i^{ew} \quad (9)$$

Watermark extraction algorithm:

$$s = \sum_{i=1}^{N} \delta(\hat{a}_i^{ew}, \hat{b}_i^{ew}) = 0 \text{ and } s = \sum_{i=1}^{N} (\bar{a}_i^{w} / \bar{b}_i^{w}) \approx \frac{1}{\lambda^2} N \quad (10)$$

In this way, by adjusting the random number r so that $a_i^{ew} > b_i^{ew}$, a solution is designed that can extract the watermark after decryption and also in the encrypted state.

3.2 Detailed steps of the algorithm

After the discussion of 3.1 overall scheme and algorithm principle, this section will introduce in detail the generation of watermark ciphertext, watermark embedding and watermark extraction.

3.2.1 Generation of watermark ciphertext

(1) Plaintext watermarking algorithm

Step 1: In order to obtain the watermark ciphertext, first, under the control of K_c, divide the plaintext into 2 data sets of the same size and non-overlapping each other m_1,m_2。

$$M = \left\{ (m_1^i, m_2^i) \middle| \begin{matrix} m_1^i \in M_1, m_2^i \in M_2, M_1 \cap M_2 = \emptyset, \\ 0 < i < floor(len(M)/2) \end{matrix} \right\} \quad (11)$$

Step 2: M_1/λ,$M_2 \times \lambda$, get the plaintext M_w with watermark.

$$M_w = \begin{cases} m_1^i/\lambda, m_1^i \in M_1 \\ m_2^i \times \lambda, m_2^i \in M_2 \end{cases} \quad (12)$$

Step 3: Under the control of K_p, change the random number R, encrypt M_w to obtain an encrypted watermark carrier.

$$M_{ew} = E(M_w, R, k_p) = \begin{cases} E(m_{w1}^i, r_1^i, k_p), m_{w1}^i \in M_1 \\ E(m_{w2}^i, r_2^i, k_p), m_{w2}^i \in M_2 \end{cases} \quad (13)$$

The value of r is changed to ensure $m_{ew1}^i = E(m_{w1}^i, r_1^i, k_p) > m_{ew2}^i = E(m_{w2}^i, r_2^i, k_p)$。

(2) Ciphertext domain watermarking algorithm

Step 1: Under the control of K_c, divide the ciphertext $M_e = (M_w, R, K_p)$ into two data sets m_{e1} and m_{e2} of the same size and non-overlapping each other.

$$M_e = \left\{ (m_{e1}^i, m_{e2}^i) \middle| \begin{matrix} m_{e1}^i \in M_1, m_{e2}^i \in M_2, M_1 \cap M_2 = \emptyset, \\ 0 < i < floor(len(M_e)/2) \end{matrix} \right\} \quad (14)$$

Step 2: Use ElGamal's multiplicative homomorphism feature to embed the watermark in the ciphertext.

$$M_{ew} = \begin{cases} m_{ew1}^i = mod(m_{e1}^i/\lambda, p), m_{e1}^i \in M_1 \\ m_{ew2}^i = mod(\lambda m_{e2}^i, p), m_{e2}^i \in M_2 \end{cases} | m_{ew1}^i > m_{ew2}^i \quad (15)$$

Due to the randomness of encryption, the appropriate ciphertext m_{ew1}^i and m_{ew2}^i are obtained by adjusting the value of the random number r, and satisfies $m_{ew1}^i > m_{ew2}^i$. Here also because of $\lambda \in (0,1)$, there is a great possibility that $m_{ew1}^i > m_{ew2}^i$. After the watermark is embedded in the ciphertext, the

ISSN: 2414-1895 DOI: 10.6919/ICJE.202012_6(12).0015

encrypted watermark carrier can be decrypted under the control of the private key K_s. The decryption algorithm is as follows:

$$M_w=\mathrm{D}(M_{ew},\ k_s)=\left\{m_w=\frac{m_{ew}y^r modp}{g^{rx}modp}modp|m_{ew}\epsilon M_{ew}\right\} \quad (16)$$

3.2.2 Extract watermark

(1) Extraction of watermark after decryption:

Step 1: Under the control of K_c, the decrypted watermark carrier M_w can be divided into two data sets M_{w1}, M_{w2} .

$$M_w=\left\{(m_{w1}^i,m_{w2}^i)\middle|\ {}^{m_{w1}^i\in M_{w1},m_{w2}^i\in M_{w2},M_{w1}\cap M_{w2}=\emptyset,}_{0<i<floor(N/2)}\right\} \quad (17)$$

Step 2: Calculate the statistics SUM_p.

$$SUM_p=\frac{1}{len(M_{w1})}[\sum_i m_{w1}^i/m_{w2}^i] \quad (18)$$

among them $0<i\le len(M_{w1})$,Calculation results $SUM_p\approx\frac{1}{\lambda^2}$,Indicates a plaintext watermark, SUM_p=1, Indicates no clear text watermark.

(2) Extraction of watermark in ciphertext:

Step 1: Under the control of K_c, divide the encrypted watermark carrier M_{ew} into two data sets M_{ew1} and M_{ew2} of the same size and non-overlapping each other.

$$M_{ew}=\left\{(m_{ew1}^i,m_{ew2}^i)\middle|\ {}^{m_{ew1}^i\in M_{ew1},m_{ew2}^i\in M_{ew2},M_{ew1}\cap M_{ew2}=\emptyset,}_{0<i\le floor(N/2)}\right\} \quad (19)$$

Step 2: Calculate the statistical quantity SUM_e

$$SUM_e=\frac{1}{len(M_{ew1})}\sum_i\delta(m_{ew1}^i/m_{ew2}^i) \quad (20)$$

Among them, $0<i\le len(M_{ew1})$, the calculation result SUMe$\approx$0, indicating that it contains a ciphertext watermark, and SUMe$\approx$1, indicating that it does not contain a ciphertext watermark.

3.3 Pixel overflow problem analysis

In the field of digital watermarking image processing technology, generally the pixel value of the original carrier image is [0,255]. Due to the encryption operation in the encrypted domain watermarking algorithm, the pixel value may exceed 255. In order to avoid pixel overflow, Usually people will preprocess the image [14]. In this algorithm, due to the ElGamal algorithm encryption operation, pixel value overflow will occur. In order to reduce the overflow, in the simulation experiment process, the prime number p=293 is selected, so that when the pixel value is modulo operation, as much as possible So that the range of the calculation result is 0~255, there is a very small possibility that the pixel value will overflow. In the actual experiment, we can scan and mark all pixels. After encryption and before embedding the watermark, modify the pixel value greater than 255 to 255 in advance, and mark the modified bit. Then in the process of decrypting and extracting the watermark, the pixel value is modified to the original pixel value according to the identification bit. Sometimes image preprocessing is cumbersome, but image preprocessing can ensure that the algorithm is carried out accurately.

4. Analysis of experimental results

4.1 Experimental setup

The experiment first selects the standard Lena gray image for experiment. Take Lena (256×256) gray image as an example to verify the feasibility of the algorithm. In the ElGamal encryption algorithm, the large prime number p=293 is selected, and the random number k=5 is selected during encryption. The private key a=3 when decrypting.

In order to measure the performance of the watermarking algorithm, the experiment uses peak signal-to-noise ratio (PSNR), bit error rate (BER) and embedding rate (BR) as objective evaluation criteria

ISSN: 2414-1895 DOI: 10.6919/ICJE.202012_6(12).0015

to test the performance of the watermarking algorithm. The higher the peak signal-to-noise ratio (PSNR) value, the smaller the difference between the embedded image and the original carrier image. Bit error rate BER∈[0,1], the closer the BER value is to 0, the higher the accuracy of watermark extraction. For carriers of the same size, the higher the embedding rate (BR), the larger the corresponding embedding capacity. Suppose I represents the original image, I'represents the watermarked image after decryption, (i, j) represents the pixel coordinates of the image, where h represents the height of the image, w represents the width of the image, and M is the total number of bits embedded in the watermark.

The formula for calculating the peak signal-to-noise ratio (PSNR) is:

$$PSNR = 10 \times \lg \frac{h \times w \times 255^2}{\sum_{i=1}^{h} \sum_{j=1}^{w} [I(i,j) - I'(i,j]^2} \tag{21}$$

The calculation formula of the correlation coefficient (NC) is:

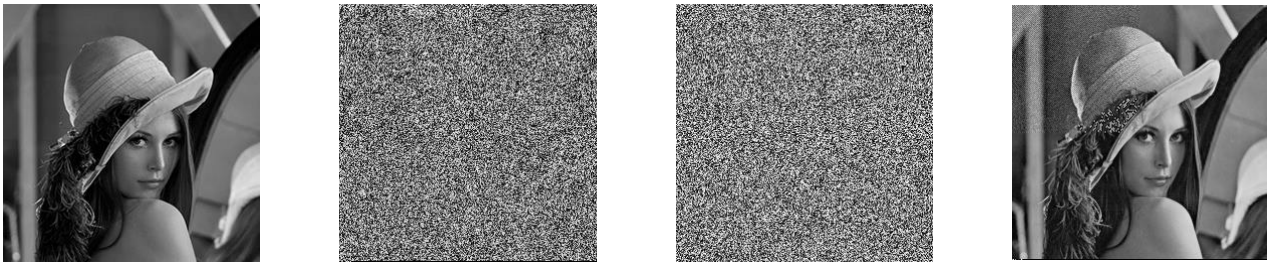
$$NC = \frac{\sum_{i=1}^{h} \sum_{j=1}^{w} I(i,j) I'(i,j)}{\sqrt{\sum_{i=1}^{h} \sum_{j=1}^{w} I(i,j)^2} \times \sqrt{\sum_{i=1}^{h} \sum_{j=1}^{w} I'(i,j)^2}} \tag{22}$$

The calculation formula of bit error rate (BER) is:

$$BER = \frac{1}{M} \sum_{i=1}^{M-1} X_i, X_i = \begin{cases} 1, I(i,j) = I'(i,j) \\ 0, I(i,j) \neq I'(i,j) \end{cases} \tag{23}$$

4.2 Result analysis

Figure 1(a) is the original image, Figure 1(b) is the ciphertext image encrypted by the ElGamal encryption system, Figure 1(c) is the ciphertext watermark image with embedded watermark, and Figure 1(d) is the decryption The extracted watermark image has a PSNR value of 39.80dB and a BER value of 0.0038. In addition, the experimental algorithm has a large embedding capacity, and the embedding rate can reach up to 0.25bpp.



(a) Original image (b) Ciphertext image (c) Watermarked ciphertext image (d) The extracted watermark image psnr=39.80dB

Figure 2 The experimental results in the encrypted domain

4.3 Performance Testing

Table 1 PSNR under different embedding capacity of ciphertext domain

Original image	PSNR under different embedding capacity (dB)			
	0.25	0.0625	0.0156	0.0039
Lena	25.71	26.68	33.98	39.80
peppers	25.39	27.86	28.35	29.17
cameraman	21.59	24.10	33.70	40.23
plane	24.28	28.03	36.10	41.98
baboon	21.68	25.81	33.56	38.11

In order to further evaluate the trial and robustness of this algorithm, two more images (peppers, baboon) with a size of 256×256 are selected for attack experiments and performance evaluation. According to formula (21) and formula (22), calculate the peak signal-to-noise ratio (PSNR) of the

ISSN: 2414-1895 DOI: 10.6919/ICJE.202012_6(12).0015

original watermark image under different embedding capacities that are first encrypted and then embedded in the watermark and then decrypted, and the original watermark image is embedded in the watermark and then encrypted and decrypted. The specific results are shown in Table 1.

5. Conclusion

This paper adopts the method of combining the homomorphic encryption algorithm ElGamal and Patchwork watermarking algorithm to construct a homomorphic ciphertext domain watermarking scheme, which can use the one-to-one mapping between the plaintext domain and the ciphertext domain to realize the extraction of the ciphertext domain watermark, To ensure the security of private data and reduce cloud data distribution management. Experimental results show that the algorithm can still extract watermark information when the embedding capacity is as high as 0.25bpp, and the peak signal-to-noise ratio of the directly extracted watermark image is 41.98 dB, which is higher than other encryption domain watermarking algorithms. The quality is high and the security of the watermarking algorithm is guaranteed. In addition, the next step of research can try to find more efficient methods to solve the problem of image overflow in the homomorphic encryption domain, and further improve the overall performance of the algorithm.

References

[1] Feng DG, Zhang M, Zhang Y, Xu Z. Study on cloud computing security. Ruan Jian Xue Bao/Journal of Software, 2011, 22(1) 71-83 (in Chinese with English abstract). -http:/ www. jos. orgcn/ 1000-9825/ 3958. htm [doi:10.3724/sp.j.1001.2011.03958]

[2] Otto-von-Guericke University at Magdeburg (GAUSS) (2005) First summary report on hybrid systems, TR:IST-2002-507932 [R]. ECRYPT European Network of Excellence in Cryptology. http:// www. ecrypt. eu. org/ecrypt1/documents.htm

[3] KAMSTRA L,HEIJMANS H. Reversible data embedding into Images using wavelet techniques and soaing[J]. IEEE Transactions Image Processing, 2005. 14(12)2082-2090.

[4] R.L.Rivest,L.Adleman,M.L.Dertouzos,On data banks and privacy homomorphisms, Found. Secure Comput. 4 (11) (1978) 169-180.

[5] Gamal T E . A public key cryptosystem and a signature scheme based on discrete logarithms[J]. IEEE Transactions on Information Theory, 1985, 31:469-472.

[6] P. Paillier, Public-key cryptosystems based on composite degree residuosity classes, in: Advances in Cryptology -EUROCRYPT'99, 1592, 1999, pp. 223-238 .

[7] Zheng Peijia,Huang Jiwu. Discrete wavelet transform and data expansion reduction in homomorphic encrypted domain.[J]. IEEE transactions on image processing : a publication of the IEEE Signal Processing Society,2013,22(6).

[8] Y.-C. Chen, C.-W. Shiu, G. Horng, Encrypted signal-based reversible data hiding with public key cryptosystem, J. Vis. Commun. Image Represent. 25 (5) (2014) 1164–1170 .

[9] Jianting Guo,Peijia Zheng,Jiwu Huang. Secure watermarking scheme against watermark attacks in the encrypted domain[J]. Journal of Visual Communication and Image Representation,2015,30.

[10] Shijun Xiang, Xinrong Luo, Efficient reversible data hiding in encrypted image with public key cryptosystem, Signal Process. 10.1186/s13634-017-0496-6

[11] X Zhang, J Long, Z Wang, H Cheng, Lossless and reversible data hiding in encrypted images with public key cryptography. IEEE Trans. Circ. Syst. for Video Technol. 26(9), 1622-1631 (2016)

[12] Jiang, Li.The identical operands commutative encryption and watermarking based on homomorphism[J]. Multimedia tools and applications,2018,77(23):30575-30594.

[13] Bender W, Gruhl D, Morimoto N (1995) Technique for data hiding. Proceeding of the SPIE 2420, storage and retrieval for image and video database III: 164–17

[14] Wu H T, Mai W, Meng S, et al. Reversible Data Hiding With Image Contrast Enhancement Based on Two-Dimensional Histogram Modification[J]. IEEE Access, 2019.