CMC, vol.55, no.2, pp.285-295, 2018

A Cryptograph Domain Image Retrieval Method Based on Paillier Homomorphic Block Encryption

Wenjia Xu[1], Shijun Xiang[1, *] and Vasily Sachnev[2]

Abstract: With the rapid development of information network, the computing resources and storage capacity of ordinary users cannot meet their needs of data processing. The emergence of cloud computing solves this problem but brings data security problems. How to manage and retrieve ciphertext data effectively becomes a challenging problem. To these problems, a new image retrieval method in ciphertext domain by block image encrypting based on Paillier homomophic cryptosystem is proposed in this paper. This can be described as follows: According to the Paillier encryption technology, the image owner encrypts the original image in blocks, obtains the image in ciphertext domain, then passes it to the third party server. The server calculates the difference histogram of the image in ciphertext domain according to the public key and establishes the index database. The user passes the retrieved image to the server. The server computes the differential histogram of the retrieved image by public key. Then, compares the similarity of it with the histogram in index database and selects larger similarity images in ciphertext and send them to the user. The user obtains the target image with the private key. The experimental results show that the method is feasible and simple.

Keywords: Paillier cryptosystem, homomorphic encryption, image retrieval, feature extraction, difference histogram.

1 Introduction

With the advent of big data's era, the computing resources and storage capacity of ordinary users cannot meet their needs of data processing, and heterogeneous data such as images, video, audio, text and so on are growing at an astonishing rate every day. The emergence of cloud computing is a good solution to this problem. So, more and more users are uploading their files to cloud. For these massive images which contain rich visual information [Cao, Zhou, Sun et al. (2018)], how to conveniently, quickly and accurately query and retrieve the images that users need or are interested in in these vast image databases has become a hot topic in the field of multimedia information retrieval [Xiang and He (2018)]. At the same time, the creation, processing and sharing of personal data such as user information, photos and video are becoming easier and easier, but the privacy

[1] School of Information Science and Technology, Jinan University, No. 601, West Huangpu Avenue, Guangzhou and 510632, China.

[2] Department of Information, Communications and Electronic Engineering, The Catholic University of Korea, Seoul and 02841, South Korea.

* Corresponding author: Shijun Xiang. Email: Shijun_Xiang@qq.com.

CMC. doi:10.3970/cmc.2018.01719

and security of data have raised concerns [Cheng (2016)]. Data stored in the cloud may suffer from malicious use by cloud service providers since data owners have no longer direct control over data [Fu, Ren, Shu et al. (2016)]. At present, the problem of user privacy has not only attracted the attention of relevant departments of various countries, but also caused huge repercussions among the users [Pradeep, Mridula and Mohanan (2016)]. Encryption, which is one of the most powerful methods for content protection, can convert original data into unintelligible ciphertext data [Xiang and Luo (2017)]. Traditional encryption technology can guarantee privacy by converting plaintext into ciphertext, but it limits further information processing such as data compression, data transformation and data retrieval. And downloading all the data from the cloud and decrypt locally is obviously impractical [Xia, Wang, Sun et al. (2015)]. For the signal processing on the basis of privacy protection, scholars put forward ‘how can process the encrypted signal and obtain similar or same to operating on the plaintext processing’. This problem becomes a new direction in the field of information security. Its achievement involves cryptography [Tang, Zhu, Wang et al. (2014)], privacy protection of data mining [Schonberg, Draper, Yeo et al. (2008)], image/video compression in ciphertext domain [Zhang (2012)], information hiding in ciphertext domain [Bianchi, Piva and Barni (2009)], and signal transformation in ciphertext domain [Zhu, Li and Guo (2014)] and so on.

Chor et al. [Chor, Goldreich, Kushilevitz et al. (1995)] first proposed the concept of private information retrieval in 1995, whose algorithm can protect the user's privacy by downloading the database to local retrieval. Paper [Song and Wagner (2000)] first proposed a kind of text retrieval in ciphertext domain scheme based on word-for-word encryption and accurate matching. In the earlier years, the scheme of image retrieval in ciphertext domain realized by adding image keywords and translating image retrieval into text retrieval in ciphertext domain. In order to improve the efficiency of retrieval, the paper [Goh (2003)] puts forward the idea of establishing safety index. The real technique of image retrieval in ciphertext domain is first proposed by Lun et al. It is an image retrieval method that supports feature protection.

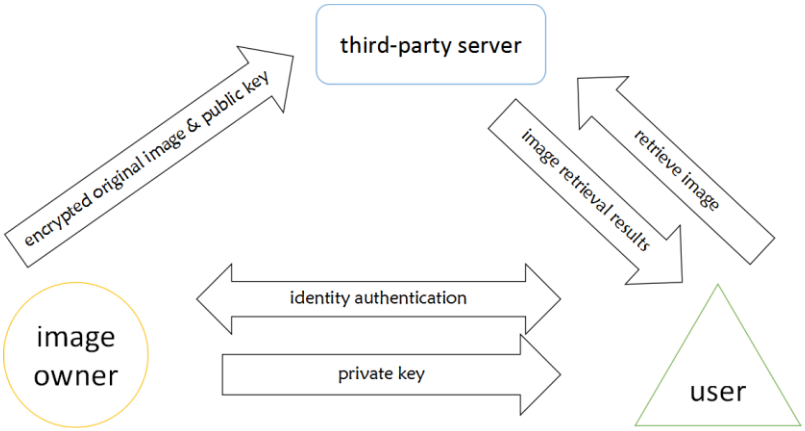


Figure 1: The basic structure of image retrieval in ciphertext domain

The system structure adopts the design framework as shown in Fig. 1. This composed of image owner, third-party server and user. The image owner uploads the encrypted original image and public key to a third party server. The server stores the secret image and uses the public key to extract the image feature (difference histogram), then establishes the feature index. The user sends the retrieve image to the server. The server calculates the difference histogram of the retrieved image, retrieves the similar secret image by the feature index, and sends the similar secret image to the user. The user and the image owner verify the identity. Then the user can get the private key, decrypt the secret image with the private key, and get the target image.

2 The paillier cryptosystem

In 1978, Rivest [Rivest (1978)] first propose the homomorphic encryption system. The homomorphic encryption system uses different keys for encrypt and decrypt process: using the public key to encrypt, and then the encrypted information sent to the user who have the corresponding private key, which can achieve privacy protection. In addition, ciphertext can perform arithmetic operation directly, and the result of decryption is consistent with that of corresponding operation in plaintext domain. It provides a reliable technical means for encrypting domain information processing. In order to achieve semantic security, Goldwasser et al. [Goldwasser and Micali (1984)] proposed a public key cryptosystem with probability characteristics in 1984. In 1999, Paillier [Pailler (1999)] proposed an encryption technique with homomorphism and probabilistic. In this paper, the Paillier encryption algorithm is used to propose a new image retrieval method in ciphertext domain.

2.1 Key generation

In order to encrypt massage using the Paillier cryptosystem, the keys must first be established. The algorithm for generating keys is as follows: One must choose two large primes p and q, then calculate N and λ:

$$N = p \times q \tag{1}$$

$$\lambda = lcm(p-1, q-1) \tag{2}$$

Where, the function $lcm(a,b)$ calculate least common multiple of a and b. Then a semi-random, nonzero integer, g must be selected, it must satisfy the following two conditions:

$$g \in Z^*_{N^2} \tag{3}$$

$$\gcd(L(\mathrm{mod}(g^\lambda, N^2)), N) = 1 \tag{4}$$

Where, $Z^*_{N^2}$ is the set of integers relatively prime with N^2 in Z_{N^2}, Z_{N^2} is the set of integers which less than N^2. Function $\gcd(a,b)$ calculate the greatest common divisor of a and b. Function $\mathrm{mod}(a,b)$ calculates the a modulus b. The expression of function $L(u)$ is:

$$L(u) = \frac{u-1}{N} \tag{5}$$

In this paper, the difference histogram is used as a feature to retrieve images. In order to calculate the difference histogram comparison table between ciphertext and plaintext domain, the public key r_1 is also needed:

$$r\_1 = \mathrm{mod}(g^{\lambda-1}, N) \tag{6}$$

In which, (N, g, r_1) represents the public key and λ represents the private key.

2.2 Encryption

Let m be a plaintext message to be encrypted, where

$$m \in Z_N \tag{7}$$

Select random r where

$$r \in Z_N^* \tag{8}$$

Compute ciphertext c as:

$$c = E[m,r] = \mathrm{mod}(g^m \times r^N, N^2) \tag{9}$$

Where $E[m,r]$ is encryption function. Since the selection of r is random, for the same plaintext m, it can be encrypted into different ciphertext c, thus ensuring the semantic security of ciphertext.

2.3 Decryption

Let c be the ciphertext to decrypt need private key λ. Compute the plaintext message as:

$$m = D[c] = \mathrm{mod}(\frac{L(\mathrm{mod}(c^{\lambda}, N^2))}{L(\mathrm{mod}(g^{\lambda}, N^2))}, N) \tag{10}$$

In the actual calculation,

$$\mathrm{mod}(a \div b, p) = \mathrm{mod}(a \times [b]^{-1}, p) \tag{11}$$

Where $[b]^{-1}$ represents the modular multiplicative inverse of b with respect to p. In mathematics, in particular, the area of number theory, a modular multiplicative inverse of an integer a is an integer x such that the product ax is congruent to 1 with respect to the modulus f. In the standard notation of modular arithmetic this congruence is written as $ax \equiv 1 \mathrm{mod}\, f$.

2.4 Homomorphic properties

Homomorphic encryption is a cryptography technique based on the theory of mathematical problem with complexity computing. The homomorphic encryption of data processing to obtain an output, decrypt the output. The result is the same as that obtained by processing the unencrypted raw data by the same method. Homomorphism includes addition homomorphism, multiplicative homomorphism, mixed multiplicative homomorphism, subtraction homomorphism and division homomorphism. This paper focuses on the homomorphism addition and homomorphism subtraction of Paillier algorithm.

Addition homomorphism: Without knowing the specific values of $m1$ and $m2$, the Paillier cryptosystem can calculate $E[m1+m2,r]$ through the values of $E[m1,r]$ and $E[m2,r]$:

$$\begin{aligned}&D(\mathrm{mod}(E[m1,r1]\times E[m2,r2],N^2))\\&=D(\mathrm{mod}(g^{m1}r1^N\times g^{m2}r2^N,N^2))\\&=D(\mathrm{mod}(g^{m1+m2}\times(r1*r2)^N,N^2))\\&=\mathrm{mod}(m1+m2,N)\end{aligned} \tag{12}$$

Subtraction homomorphism: In theory, it is not necessary to know the values of $m1$ and $m2$, value of $E(m1-m2,r)$ can be calculated with $E(m1,r)$ and $E(m2,r)$:

$$\begin{aligned}&D(\mathrm{mod}(E(m1,r1)\div E(m2,r2),N^2))\\&=D(\mathrm{mod}(\frac{g^{m1}r1^N}{g^{m2}r2^N},N^2))\\&=D(\mathrm{mod}(g^{m1-m2}\times(\frac{r1}{r2})^N,N^2))\\&=\mathrm{mod}(m1-m2,N)\end{aligned} \tag{13}$$

Through the above formula (11), it can be deduced that:

$$D(\mathrm{mod}(E(m1,r1)\times[E(m2,r2)]^{-1},N^2))=\mathrm{mod}(m1-m2,N) \tag{14}$$

3 Block encryption

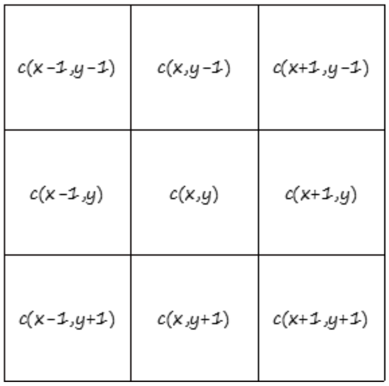
In order to calculate the difference histogram of plaintext or ciphertext image, the image is divided into blocks and then encrypted by Paillier encryption system. In the encryption process, the key is first generated according to the formula (1-6), then the image is divided into several modules of size 3×3 pixels, and the extra part can be ignored.

Figure 2: Flower close-up picture

Take the flower picture shown in Fig. 2 as an example. Fig. 2 shows the image size of 768×1024 pixels, which can be divided into 256×341 modules and the extra parts of 768×1 pixels. Each module is selected randomly a r by formula (8), and then the value of each pixel m in the module is encrypted using this r by formula (9). Each value in the extra part is also selected a r for encryption. That is, in the same module, all pixels are encrypted using the same parameter, so the processing can calculate the difference between pixels in the same module using the modular multiplication inverse element in the encryption domain. Then the difference histogram is constructed for image retrieval.

4 Difference histogram of the encryption matrix

Image features generally include global image feature and local image feature. In early image retrieval techniques, global image features such as color, texture, shape and spatial relationship are often used. Later, some scholars proposed the feature extraction technique based on image segmentation technique. In recent years, some feature descriptors with local invariant properties have been proposed, such as SIFT, etc. And it has been widely used in image retrieval [Mei and Ye (2018)]. In 2012, Hsu et al. [Hsu, Lu and Pei (2012)] proposed a SIFT image feature extraction algorithm based on paillier encryption system. However, this algorithm has the disadvantages of high computational complexity and low efficiency, so it is difficult to be implemented in practical applications. The samples selected in this paper are all grayscale images of flowers. Therefore, texture is used as the retrieval feature, and the difference histogram is used to describe it. The encrypted image can be regarded as an encryption matrix, and the difference histogram of the encrypted matrix can be calculated according to the algorithm in reference Wang et al. [Wang and Zhang (2005)]. For an encryption matrix and its public key, the encryption matrix should be divided into M modules with size of 3×3 firstly. Then, $cd1$, $cd2$, $cd3$ and $cd4$ in each module should be calculated according to formula (15-18). Each module is shown in Fig. 3, and (x, y) represents its location.



c(x-1,y-1)	c(x,y-1)	c(x+1,y-1)
c(x-1,y)	c(x,y)	c(x+1,y)
c(x-1,y+1)	c(x,y+1)	c(x+1,y+1)

Figure 3: Schematic diagram of a single module of an encrypted matrix

According to Fig. 3, we can obtain:

$$cd1=\mathrm{mod}(c(x-1,y-1)\times[c(x+1,y+1)]^{-1},N^2) \tag{15}$$

$$cd2=\mathrm{mod}(c(x,y-1)\times[c(x,y+1)]^{-1},N^2) \tag{16}$$

$$cd3=\mathrm{mod}(c(x+1,y-1)\times[c(x-1,y+1)]^{-1},N^2) \tag{17}$$

$$cd4=\mathrm{mod}(c(x-1,y)\times[c(x+1,y)]^{-1},N^2) \tag{18}$$

Due to use the same r in one module, we can get the difference Tab. (1) between -255 to 255 in ciphertext domain:

$$\operatorname{mod}(E(m1,r)\times[E(m2,r)]^{-1},N^2)=\begin{cases}\operatorname{mod}(g^{m1-m2}\times1^N,N^2)\cdots\cdots\cdots m1>m2\\1\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots m1=m2\\\operatorname{mod}(g^{N-|m1-m2|}\times r\_1^N,N^2)\cdots m1<m2\end{cases}\tag{19}$$

Table 1: Mapping table between ciphertexts and plaintexts in [-255,255]
（$N=551,g=21020,r\_1=168$）

-255	-254	…	-1	0	1	…	254	255
183760	223278	…	271811	1	21020	…	223818	53264

The specific difference between $m1$ and $m2$ can be obtained without the specific value of $m1$ and $m2$ through Eq. (19). Using Eq. (19), find the corresponding value in the range of [-255,255], and the corresponding table is made. For example, the corresponding Tab. 1 when $N=551,g=21020,r\_1=168$.

Finally, the histogram Hc1, Hc2, Hc3 and Hc4 of each module are calculated according to the $cd1$, $cd2$, $cd3$ and $cd4$ in each module according to corresponding Tab. 1. The four histograms are the difference histograms of the encryption matrix. Fig. 4 is the corresponding histogram Hc1, Hc2, Hc3 and Hc4 of the difference values after the encryption of Fig. 2.

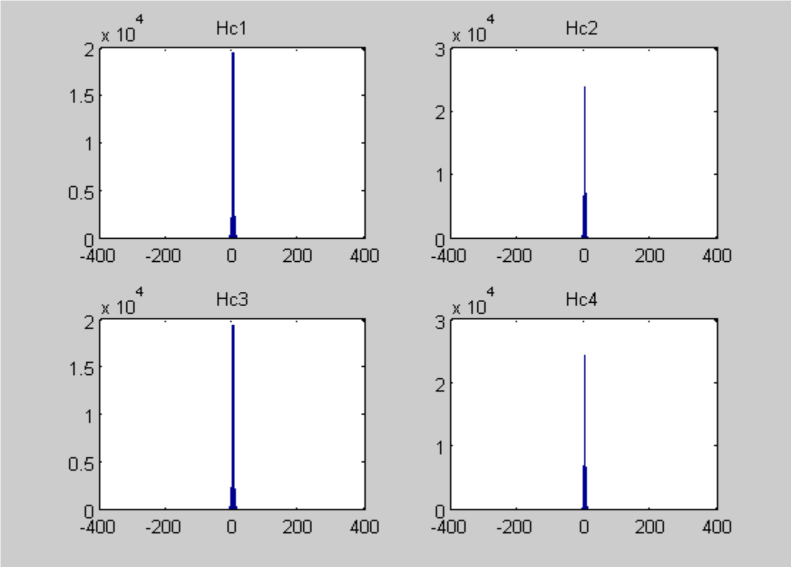


Figure 4: Difference histogram of Fig. 2 after encryption

5 Calculate histogram similarity

In this paper, histogram similarity is used to determine whether the retrieval image is similar to the original image. For a retrieval image, it is divided with size 3×3 into N modules and calculate the four difference values $d1, d2, d3, d4$ of each module according to the formula (20-23), the excess parts do not need to be calculated.

$$d1 = m(x-1, y-1) - m(x+1, y+1) \tag{20}$$

$$d2 = m(x, y-1) - m(x, y+1) \tag{21}$$

$$d3 = m(x+1, y-1) - m(x-1, y+1) \tag{22}$$

$$d1 = m(x-1, y) - m(x+1, y) \tag{23}$$

The histogram H1, H2, H3 and H4 of $d1, d2, d3, d4$ in each module on [-255, 255] are respectively calculated. Set these four histograms as the histogram of the difference of the retrieved image. Fig. 5 shows the histogram H1, H2, H3 and H4 of the image Fig. 2.

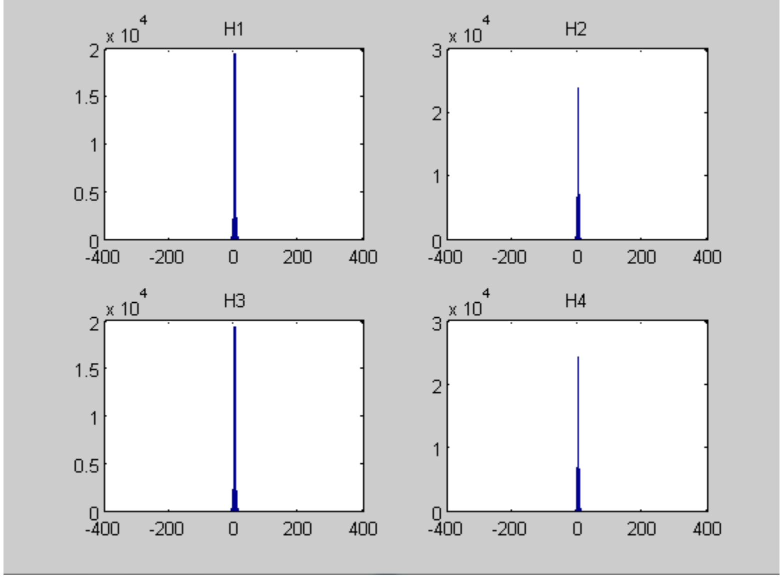


Figure 5: Difference histogram of Fig. 2 before encryption

Then the similarities of four retrieval histograms and the corresponding four encrypted histograms are calculated according to the formula (24).These four similarities are represented by $dis1$, $dis2$, $dis3$ and $dis4$.

$$dis = \sum_{i=-255}^{255} \min(\frac{H(i)}{N}, \frac{Hc(i)}{M}) \tag{24}$$

The function $\min(a, b)$ returns the smaller one between a and b. And then we calculate the average of the four degrees of similarity. Using this average as the similarity between the retrieval image and the encryption matrix. The six matrices with the largest similarity were selected as the similarity matrix, and the similar images were obtained after decryption.

6 Test results and analysis

In this paper, 50 experimental images are selected, including 3 kinds of flowers related images: Flowering shrubs picture, flowering close-up picture and hand drawing picture. Limited space of this paper, only one sample image is selected to illustrate the retrieval results. The image is entered separately, and 6 images with the highest similarity in the database are returned. After decryption, six target images are obtained. Fig. 6 is flowering shrubs picture, Fig. 7 is decrypt the array returns from Fig. 6, Fig. 8 is flowering close-up picture, Fig. 9 is decrypt the array returns from Fig. 8, Fig. 10 is hand drawing picture, Fig. 11 is decrypt the array returns from Fig. 10. The retrieval effect is shown in Figs. 6-11.

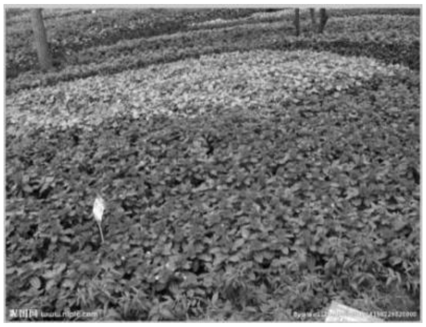


Figure 6: Flowering shrubs picture

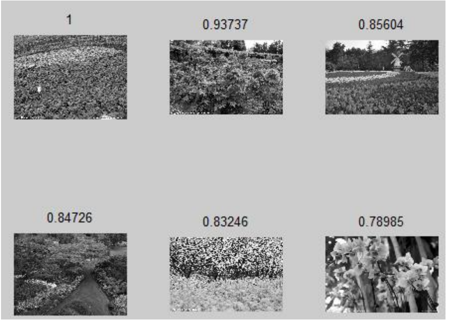


Figure 7: Decrypt the array returns from Fig. 6

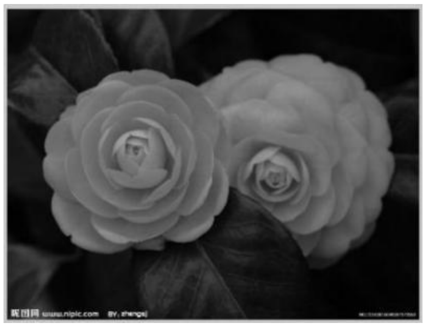


Figure 8: Flowering close-up picture

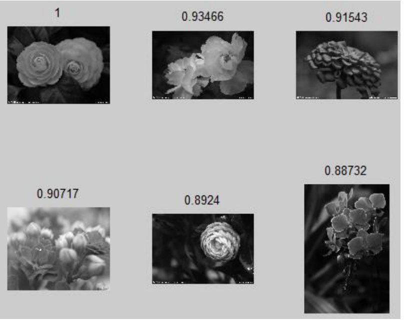


Figure 9: Decrypt the array returns from Fig. 8

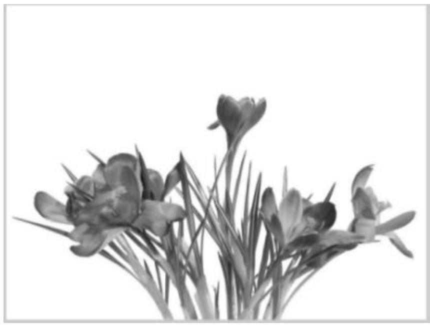


Figure 10: Hand drawing picture

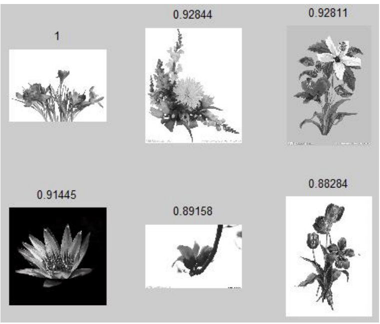


Figure 11: Decrypt the array returns from Fig. 10

Comparing Fig. 6 and Fig. 7, it can be seen that the similarity of the first diagram in Fig. 7 is 1, which is exactly the same as the original graph. The similarities of other five images are 0.94, 0.86, 0.85, 0.83 and 0.79, respectively, and they are all in flowering shrubs picture class. In Fig. 8 and Fig. 9 contrast, the highest similarity returns is the original image, image similarity is 1. Although the rest of the five pictures' similarity is below 1, but also belong to flowering close-up picture class. And it can be observed that the image with higher similarity is similar to the texture distribution of the original image. However, in Fig. 11, only four pictures in the hand drawing picture class, and the similarity of the other two images is 0.91 and 0.89, but they are not hand drawing pictures.

It can be seen from the retrieval results that the proposed algorithm can achieve the goal of image retrieval in ciphertext domain. This method has the advantages of simple algorithm and high efficiency. This algorithm is only suitable for image texture retrieval. The precision rate still needs to be improved.

For example, in Figs. 10 and 11, two pictures with a similarity of 0.91 and 0.89 can have such a high similarity to the original image, largely because the processing of the background in the photo takes a large area of white space similar to the hand-drawn image. The texture changes detected by this algorithm are similar. Because the background occupies a large area and the weight of similarity calculation is also large, the retrieval effect is affected.

7 Conclusion

This paper presents a method of image retrieval in ciphertext domain based on Paillier homomorphic encryption system. The image owner uploads the encrypted original image and public key to a third party server. The server stores the encrypted image, uses the public key to extract the image feature (differential histogram), and establishes the feature index. The user sends the retrieve image to the server. The server calculates the difference histogram of the retrieved image, retrieves the similar encrypted image by the feature index, and sends them to the user. The user and the image owner verify the identity. Then the user can get the private key, decrypt the secret image with the private key, and get the target image. The experimental results show that the method is feasible and effective. But the accuracy rate is still to be improved. At present, the method is only applicable to texture retrieval, and other kinds of retrieval methods need to be further studied.

Acknowledgement: This work was supported in part by the Natural Science Foundation of China (No.61772234, 61272414).

References

Bianchi, T.; Piva, A.; Barni, M. (2009): On the implementation of the discrete Fourier transform in the encrypted domain. *IEEE Transactions on Information Forensics and Security*, vol. 4, no.1, pp. 86-97.

Cheng, H. (2016): *Research on encrypted JPEG image retrieval*. School of Communication and Information Engineering.

Chor, B.; Goldreich, O.; Kushilevitz, E.; Sudan, M. (1995): Private information retrieval. *IEEE Symposium on Foundations of Computer Science*, vol. 36, pp. 41-51.

Cao, Y.; Zhou, Z.; Sun, X.; Gao, C. (2018): Coverless information hiding based on the molecular structure images of material. *Computers, Materials & Continua*, vol. 54, no. 2, pp. 197-207.

Fu, Z.; Ren, K.; Shu, J.; Sun, X.; Huang, F. (2016): Enabling personalized search over encrypted outsourced data with efficiency improvement. *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 9, pp. 2546-2559.

Goh, E. J. (2003): Secure indexes. *IACR Cryptology ePrint Archive.*

Goldwasser, S.; Micali, S. (1984): Probabilistic encryption. *Journal of Computer and System Sciences*, vol. 28, no. 2, pp. 270- 299.

Hsu, C.; Lu, C.; Pei, S. (2012): Image feature extraction in encrypted domain with privacy-preserving SIFT. *IEEE Transactions on Image Processing*, vol. 21, no. 11, pp. 4593- 4607.

Mei, Y.; Ye, P. (2018): Research and development of encryption domain image retrieval technology.

Pailler, **P.** (1999): Public-key cryptosystems based on composite degree residuosity classes. *Germany: Springer Berlin Heidelberg*, pp. 223-238.

Pradeep, A.; Mridula, S.; Mohanan, S. (2016): High security identity tags using spiral resonators. *Computers, Materials & Continua*, vol. 52, no. 3, pp. 187-196.

Rivest, R. L.; Shamir, A.; Adleman, L. M. (1978): On data banks and privacy homomorphisms. *Foundations of secure computation*, vol. 4, no. 11, pp. 169-180.

Schonberg, D.; Draper, S. C.; Yeo, C.; Ramchandran, K. (2008): Toward compression of encrypted images and video sequences. *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 4, pp. 749-762.

Song, D.; Wagner, D.; Perrig, A. (2000): Practical techniques for searches on encrypted data. *Proceedings of IEEE Synposium on Security and Privacy*, pp. 44-55.

Tang, D.; Zhu, S; Wang, L; Yang, H. (2014): FAN Jia. Fully homomorphic encryption scheme from RLWE. *Journal on Communications*, vol. 35, no. 1, pp. 173-182.

Wang, C. R.; Zang, T. (2005): A fast image texture analysis algorithm. *Opto-Electronic Engineering*, vol. 32, no. 1, pp. 74-77.

Xiang, S.; He, J. (2018): Database authentication watermarking scheme in encrypted domain. *IET Information Security*, vol. 12, no. 1, pp. 42-51.

Xiang, S.; Luo, X. (2017): Reversible data hiding in homomorphic encrypted domain by mirroring ciphertext group. *IEEE Transactions on Circuits and Systems for Video Technology.*

Xia, Z.; Wang, X.; Sun, X.; Wang, Q. (2016): A secure and dynamic multi-keyword ranked search scheme over encrypted cloud data. *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 2, pp. 340-352.

Zhang, X. (2012): Separable reversible data hiding in encrypted image. *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 2, pp. 526-532.

Zhu, X.; Li, H.; Guo, Z. (2014): Privacy-preserving query over the encrypted image in cloud computing. *Journal of Xidian University*, vol. 41, no. 2, pp. 151-158.