Image Deduplication Based on Hashing and Clustering in Cloud Storage

Lu Chen[1,2,3], Feng Xiang[4], and Zhixin Sun[1,2,3*]
[1] Engineering Research Center of Post Big Data Technology and Application of Jiangsu Province Nanjing University of Posts and Telecommunications, Nanjing, 210003, China
[2] Research and Development Center of Post Industry Technology of the State Posts Bureau (Internet of Things Technology), Nanjing University of Posts and Telecommunications, Nanjing, 210003, China
[3] Engineering Research Center of Broadband Wireless Communication Technology of the Ministry of Education, Nanjing University of Posts and Telecommunications, Nanjing, 210003, China
[e-mail:2018070261@njupt.edu.cn, sunzx@njupt.edu.cn]
[4] National Engineering Laboratory for Logistics Information Technology, Shanghai 200000, China
[e-mail: xiangfyto@126.com]
*Corresponding author: Zhixin Sun

Received August 13, 2020; revised October 23, 2020; March 26, 2021; published April 30, 2021

Abstract

With the continuous development of cloud storage, plenty of redundant data exists in cloud storage, especially multimedia data such as images and videos. Data deduplication is a data reduction technology that significantly reduces storage requirements and increases bandwidth efficiency. To ensure data security, users typically encrypt data before uploading it. However, there is a contradiction between data encryption and deduplication. Existing deduplication methods for regular files cannot be applied to image deduplication because images need to be detected based on visual content. In this paper, we propose a secure image deduplication scheme based on hashing and clustering, which combines a novel perceptual hash algorithm based on Local Binary Pattern. In this scheme, the hash value of the image is used as the fingerprint to perform deduplication, and the image is transmitted in an encrypted form. Images are clustered to reduce the time complexity of deduplication. The proposed scheme can ensure the security of images and improve deduplication accuracy. The comparison with other image deduplication schemes demonstrates that our scheme has somewhat better performance.

Keywords: Cloud Storage, Clustering, Image Deduplication, Perceptual Hash, Feature Extraction

This work is supported by the National Nature Science Foundation of China (No. 61972208, No.61672299), Postgraduate Research &Practice Innovation Program of Jiangsu Province (SJKY19_0770). The authors thank the sponsors for their support and the reviewers for helpful comments.

http://doi.org/10.3837/tiis.2021.04.014 **ISSN : 1976-7277**

1. Introduction

In recent years, cloud storage technology has developed rapidly, and more and more users outsource their data to the cloud for storage and management [1]. Cloud storage management has the advantages of automation and intelligence, high storage efficiency, and low cost. With the explosive growth of digital information, more and more data is stored in the cloud, which generates a large amount of redundant data. Therefore, eliminating duplicate data in cloud storage and increasing storage efficiency is an urgent issue.

The concept of deduplication was first proposed in 2000 to support global compression in large-scale storage systems with a coarser granularity [2]. Deduplication is a data reduction technology that reduces storage space and transmission bandwidth consumption. It computes a secure, hash-based fingerprint of a file or block and then identifies duplicates by matching their fingerprints. Deduplication is now widely used in cloud storage data redundancy, eliminating duplicate copies in the cloud and replacing them with pointers to copies [3]. Deduplication currently faces many challenges, and many technologies are evolving. For example, users often want to encrypt data before uploading it, which is contradictory to deduplication. The Convergent Encryption (CE) algorithm is proposed to solve this problem, which uses the hash value of the file or data block as the encryption key. Based on this, M. Bellare et al. [4] proposed Message-Locked Encryption (MLE), which calculates the encryption key from the plaintext and system parameters, ensuring that the same ciphertext is generated from the same plaintext. It is widely used in encrypted deduplication.

Depending on the type of data, deduplication can be divided into text-based technology and multimedia-based technology. Text-based deduplication primarily detects duplicate data in the binary encoding of files, which performs accurate hash matching based on the bitstream. For example, the early EMC Centera system [5] and Windows' single-instance storage system [6] both used file-level deduplication. Block-level deduplication is further divided into fixed-length block technology and variable-length block technology. Typical applications for variable-length block-level deduplication include the P2P file system Pasta [7], the archive storage system Deep Store [8], and the low-bandwidth network system LBFS [9].

However, traditional deduplication technology cannot achieve a good deduplication effect for multimedia data, such as images. In many cases, images are more intuitive and vivid than text in representing information, and the number of images stored on a cloud server will increase at an alarming rate [10]. Users usually pay attention to the content of the image rather than the specific details of the image. The existing deduplication methods of the regular file are very precise and strict for the definition of repetition, so they cannot be applied to the deduplication of the images [11-13]. Moreover, the security of the image in deduplication needs to be ensured. Currently, deduplication mainly uses content-based repetitive image detection technology. Many scholars have paid attention to the secure and accurate deduplication of images. Hash of images can reflect the content of images well and is widely used in duplicate image detection. At present, the algorithms used to generate image hashes mainly include average hash algorithm (a-Hash), perceptual hash algorithm (p-Hash), and difference value hash algorithm (d-Hash). According to the accuracy and speed, now the performance of the perceptual hash algorithm is better. Perceptual hash generates a string of codes for the image through the characteristics of the image. Perceptual hashing can significantly improve computational efficiency. Similar images have similar hash values. The existing perceptual hash algorithm performs DCT transformation [14] on the image, then calculates the average value and generates the perceptual hash of the image. The DCT transformation transforms the image from the spatial domain to the frequency domain and can

extract the image's global features.

Although there have been some studies on image deduplication, the security and accuracy of image deduplication need to be improved. The most critical step in the image hash calculation is the feature extraction of the image, including color, texture, shape, etc. DCT is an efficient method for describing global image features, which is widely used in image hashing. However, the method of generating hash values using DCT transformation alone is often not accurate enough. In addition, many image feature description methods focus on the local features of the image [15-17].

In this paper, we propose an image deduplication scheme based on hashing and clustering under group applications, and it combines a novel LBP-based perceptual hash algorithm(LBPH). In our scheme, the image is digitized by the LBPH algorithm. The duplicate image is detected and eliminated in an encrypted form. The clustering algorithm is used to classify images and reduce the time for deduplication. Our work makes the following contributes:

1) A novel image hashing algorithm is proposed and used in image deduplication. The algorithm combines the LBP features and DCT features of the image to improve the discrimination of the hash value and to improve the accuracy of image deduplication.
2) In order to ensure the privacy of images, deduplication is performed after encrypting the image. In our scheme, we encrypt the image and then perform duplicate image detection based on hash values which ensure image security in user uploads and downloads.
3) In our scheme, the K-means algorithm is adopted for similar image clustering. Our solution reduces the comparison time of image hash values, which reduces the time complexity of image deduplication.

Compared with existing methods, the proposed scheme performs duplicate image detection based on visual content. It improves the accuracy of image deduplication through a novel image hashing algorithm. It improves the discrimination of the hash value, and it also performs duplicate image detection based on hash values which ensures image security. The proposed scheme reduces the comparison time of image hash values, which reduces the time complexity of image deduplication.

The rest of the paper is organized as follows: the second part discusses the related work and points out the problems that need to be solved; the third part gives essential preliminaries; the fourth part and the fifth part describe the proposed secure deduplication scheme of images in detail; the sixth part analyzes the security, and the seventh part analyzes the performance of the proposed scheme; the eighth part concludes the content of this paper and discusses the future work.

2. Related Work

With the advent of the Internet, smartphones, and social networking sites, users worldwide share a large number of images and videos. The copying and dissemination of multimedia data are becoming more and more convenient and effective. There are plenty of redundant multimedia data in cloud storage, especially image files, including many duplicates.

Existing data deduplication includes file level and block level. File-level deduplication uses files as a unit and uses hash functions to obtain the hash value of each file. It can eliminate duplicate files and save storage space [18]. Block-level deduplication divides the file into multiple data blocks according to a certain method, calculates the hash value of each data block, and detects the same data block through the hash value [19]. These methods are more effective for text files, and images are often based on visual content to determine whether they are

duplicates. Traditional methods are too strict, and two images with different hash values may not have different content. Therefore, content-based image deduplication is necessary. Researchers have proposed hash algorithms for duplicate image deletion, such as mean hash, difference value hash, and perceptual hash. The first two methods are fast and straightforward to calculate based on the gray value of the image, but the anti-rotation and anti-scaling ability are poor. Perceptual hashing is a one-way mapping that transforms multimedia data sets into perceptual summary sets [20]，so it can detect images with the same perception content. The model is shown in **Fig. 1**. It can detect images with the same perceptual content. Among them, the perceptual feature extraction primarily uses the image DCT coefficient [21]. However, the perceptual hashing method that only uses DCT transformation usually emphasizes low-frequency information, ignores texture details, and is not accurate enough.

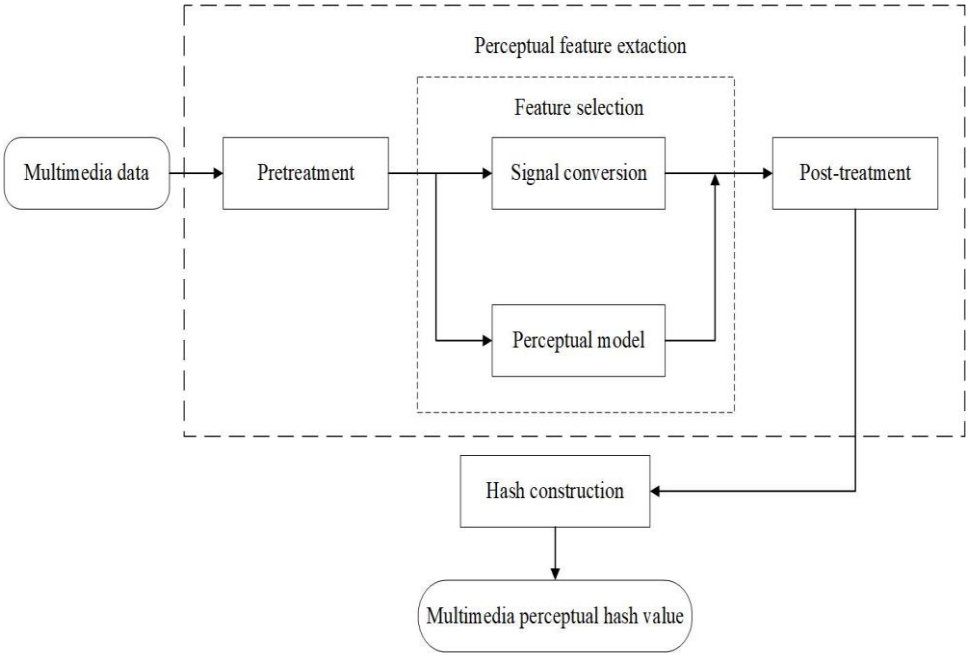

Fig. 1. Perceptual hashing process

On the other hand, to ensure data privacy when deduplicating, many researchers proposed secure deduplication methods based on content encryption. The methods include Convergent Encryption Algorithm (CE) [22] and the Message-Locked Encryption Algorithm (MLE) [4]. In the deduplication system based on CE and MLE, the image will be encrypted or decrypted with a convergent encryption key which is derived by computing the hash value of the image content [23-27]. These methods perform poorly in image deduplication because visually similar images may generate the same key, resulting in the same ciphertext. Many scholars have paid attention to the high accuracy and secure methods which support image deduplication. Gang et al. [28] proposed a secure cloud storage system with image deduplication. They calculated the hash value of the image, exporting the convergent encryption key to encrypt and decrypt the image. So the same image copy can generate the same ciphertext. The same ciphertext will be generated for checking duplicate image copies, and the owner can download the ciphertext again and retrieve the image using the key. However, this method can only eliminate the same image.

Rashid et al. [29] proposed an image hashing method that embeds partial encryption and unique image hashing into a hierarchical tree set partitioning (SPIHT) compression algorithm. Partial encryption methods are designed to ensure the security of the proposed method for semi-honest CSPs. Image hashing methods are used to classify the same compressed and encrypted images so that they can be deduplicated. This solution does not increase the extra computational overhead of image encryption, hashing, and deduplication. However, these methods can only perform deduplication on the same images and cannot be applied to similar images. So image deduplication faces new challenges.

Some scholars proposed a deduplication method (SPSD) for similar images in the paper [30], which introduces a hash algorithm (a-Hash) to generate signatures for images. The hash algorithm performs well in measuring the similarity of perceived similar images. Images and signatures stored in cloud services are encrypted using a shared group key through the symmetric cryptosystem to prevent data leakage. It calculates the hash distance and performs a duplicate check on the encrypted hash to determine if a new image is to be uploaded. Based on the SPSD method, Li et al. [31] proposed a method called CSPD, which utilizes a DCT-based perceptual hash algorithm to improve the accuracy of duplicate image detection. It can be seen that different hash algorithms affect the accuracy of image deduplication. These methods support fuzzy deduplication, but the accuracy of deduplication still needs to be improved. Image deduplication currently faces new challenges in terms of security and accuracy.

3. Preliminaries

3.1 Hamming Distance

Hamming distance refers to the number of different characters in corresponding positions in two equal-length strings, which is a way to compare the distance between strings or numbers. Let L, M be two strings of length n: $L = \{l_1, l_2, \dots, l_n\}$, $M = \{m_1, m_2, \dots, m_n\}$. We use $d(L, M)$ to represent the Hamming distance between them. The Hamming distance measures the minimum number of replacements required to change the string L to M by replacing characters. As shown in (1), for strings of length n, the Hamming distance can be further normalized :

$$d(L, M) = \frac{\sum_{k=1}^{n} (L_k \oplus M_k)}{n} \tag{1}$$

The Hamming distance obtained by the calculation can be used to detect the similarity. The smaller the Hamming distance, the higher the similarity and vice versa. In order to quickly detect similar strings, we need to set a Hamming distance threshold.

3.2 K-Means Clustering Algorithm

Clustering refers to the process of dividing a target object set into multiple classes consisting of similar objects. At present, the main clustering methods include hash clustering and K-means clustering. K-means clustering randomly selects any K samples from the sample set as the centers of the initial cluster, then calculates the distance from each sample in the sample set to the K centers and adds the current sample to the nearest class. After this, it calculates the mean of each class to get k new centers.

4. Proposed Scheme

In this section, we have developed a system model for the deduplication scheme. It includes the following three participants:

User Group: This is a collection of users that can share data between group members. Furthermore, users in the same group have the same private key, which is used to encrypt the hash of the image. The image hash value encrypted with the key is called the fingerprint of the image.

Image set: This is a collection of images stored on a cloud storage server. Images are uploaded and shared by group users. Each image is accompanied by a cryptographic hash value, which is calculated by the LBPH algorithm.

Cloud Storage Server: Users outsourcing and storing images in the cloud storage server.

4.1 LBP-Based Perceptual Hash Algorithm (LBPH)

Many operations do not change the main content of the image, such as the transformation of the storage format, scaling, color-changing, and rotating. It is possible to detect different images with different encodings. To detect duplicate images, we first need to digitize the image content and use the hash algorithm to generate the "fingerprint" of the image.

Many duplicate images have no effect on the main content of the image. At present, the commonly used hash algorithms include meaning hash algorithm (aHash), difference value hash algorithm (dHash), and DCT-based perceptual hash algorithm. Most of the existing deduplication methods for images adopt these algorithms, but their accuracy of deduplication still needs improvement. LBP value is a good reflection of the characteristics of the local image texture. The texture details described by the LBP feature of the image often appear as high-frequency information, which is complementary to low-frequency DCT coefficients, so it is beneficial to improve the deduplication accuracy. Therefore, we propose an LBP-based perceptual hash algorithm (LBPH).

The detailed process is as follows:

1) Image pretreatment

In order to generate fixed-length hashes for images of different sizes, all input images P are converted to $N \times N$ by the method of Bilinear Interpolation; Gaussian low pass filtering is used for the image in order to mitigate the effect of noise interference on the final hash value; Convert the image to a grayscale image. Finally, a processed grayscale thumbnail P_1 is obtained.

2) Local feature processing

The LBP operator can effectively describe the texture features of the image. In order to improve the accuracy of LBP features, Ojala et al. proposed the circular LBP operator. It allows for as many pixel points as there are circular neighborhoods of the radius R. The transformation first obtains the gray value of the central pixel point, calculates the coordinates of the $k-th$ sampling point. Then obtains the gray value of the $k-th$ sampling point by bilinear interpolation, and finally obtains the LBP value. The sampling point is calculated as follows:

$$x_{sp} = x_0 + Rcos\left(\frac{2\pi_{sp}}{SP}\right), y_{sp} = y_0 - sin\left(\frac{2\pi_{sp}}{SP}\right) \quad (2)$$

Where R is the sampling radius, sp is the $sp-th$ sampling point, and SP is the number of samples.

The LBP transformation effect of the image is shown in **Fig. 2**:

Fig. 2(a). Original images.

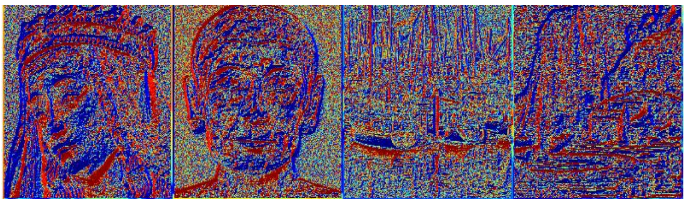

Fig. 2(b). LBP transformed images.

As shown in **Fig. 2**, select R=1, P=8, and get the LBP image through the LBP operation.

3) Global feature processing

DCT is an image compression algorithm that can transform an image from a pixel domain to a frequency domain. For an image of size $N \times N$, the DCT transformation formula is:

$$F_{(u,v)} = C(u)C(v)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)cos\frac{\pi(2x+1)u}{2N}cos\frac{\pi(2y+1)v}{2N} \tag{3}$$

Among the formula, $C(u) = C(v) = \begin{cases} \sqrt{\frac{1}{N}}, & u,v = 0 \\ \sqrt{\frac{2}{N}}, & other\ cases \end{cases}$

In the above formula, $f(x,y)$ is the pixel point of the image, and $F(u,v)$ is the DCT threshold matrix of the image. The DCT transform filters out high-frequency details of the image, leaving only the global features of the image.

4) Generate the hash of the image

Suppose the obtained DCT matrix is 32×32. First, we select the upper left corner of the DCT matrix: the low-frequency information of the image, which is an 8×8 sub-matrix. And then, we calculate the mean of the sub-matrix elements, that is, the average value of the sub-matrix elements, and set the hash value of each pixel according to (4):

$$\mathrm{h(x)} = \begin{cases} 1, & x \geq mean \\ 0, & x \leq mean \end{cases} \tag{4}$$

Finally, the image hash sequence H is generated.

4.2 Basic Idea

This section will describe the basic idea of the proposed secure image deduplication based on perceptual hash and image clustering. The hash value of the image is represented by H_I, and the encrypted image is represented by C_I. The scheme is mainly divided into three parts:

1) *Initial upload.*

The client calculates the perceptual hash values of images and encrypts the images using the symmetric encryption algorithm. Then upload the encrypted images and the perceptual hash values to the cloud storage server. The cloud storage server stores encrypted images with their hash values.

2) *Complete the users' uploading process.*

The user uploads the hash of the image. Then the server-side performs duplicate data detection and image clustering.

3) *Complete the user's downloading process.*

Duplicate images have similar fingerprints, and by comparing the Hamming distance of fingerprints, we perform deduplication. The image deduplication is performed in an encrypted form, which protects the confidentiality of the image. We encrypt original images using the symmetric encryption algorithm. **Fig. 3** illustrates the basic idea of our image deduplication scheme and what each of the three parts does.

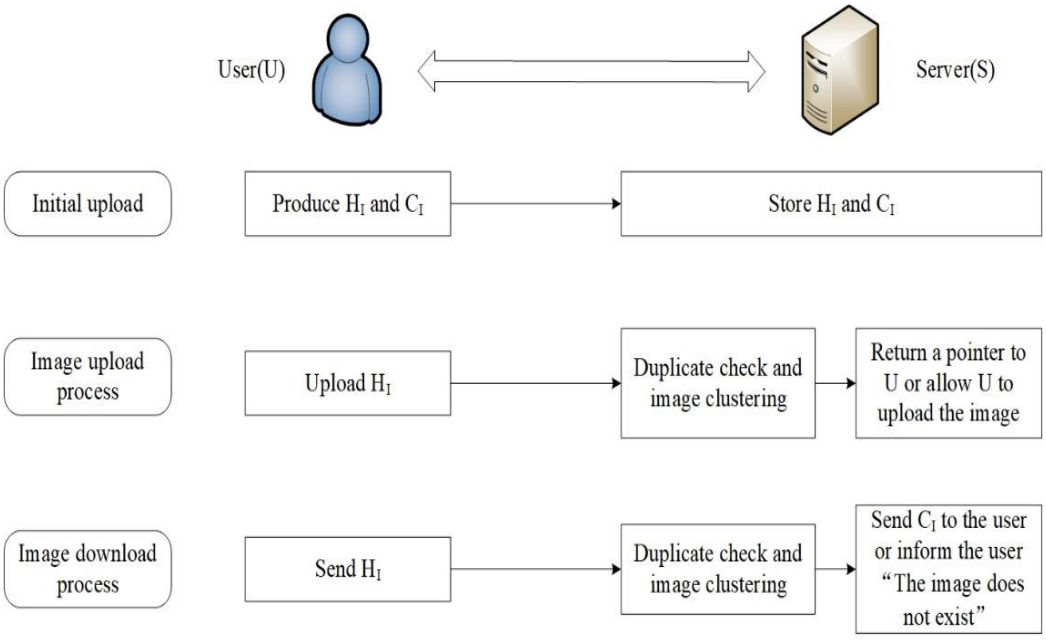

Fig. 3. The basic idea of the proposed scheme

5. Detailed Description of Our Scheme

In this section, we describe our image deduplication scheme based on hashing and clustering in detail. Our solution is applied to the user group. Users in the same group have the same group key. The scheme includes three participants: user group, image set, and cloud storage server. As shown in **Table 1**, we first define some parameters.

Table 1. Meaning of some parameters

Parameter	Meaning
H_I	The hash value calculated by LBPH of image I.
C_I	The encrypted form of image I.
U	Users in the group who want to upload or download images.
S	Cloud storage server.
k	The number of classes in the image set.
I_{center}	The central image of the class.
D	The distance between H_I and the central image.
d	The Hamming distance between the H_I and the hash values of all images in the class.
t	The threshold for deduplication is t.
K_I	The class with the shortest distance from the image I.
N	The distance threshold between H_I and the central image.

The following is the specific process of initial upload, user uploading images, and downloading images.

(1) Initial upload

The images in the image set are pre-clustered on the cloud storage server. For the image I, the hash value of the image is calculated by the LBP-based perceptual hash (LBPH) algorithm. It is called the fingerprint, which is denoted as H_I. The image is encrypted by the client with the symmetric encryption algorithm. The encrypted images denoted as C_I with their fingerprints will be uploaded. If the image is the first time to upload, the cloud storage server will receive the C_I and H_I.

(2) Upload an image

The process of uploading an image from the user denoted as U to the cloud storage server denoted as S is shown in **Fig. 4**:

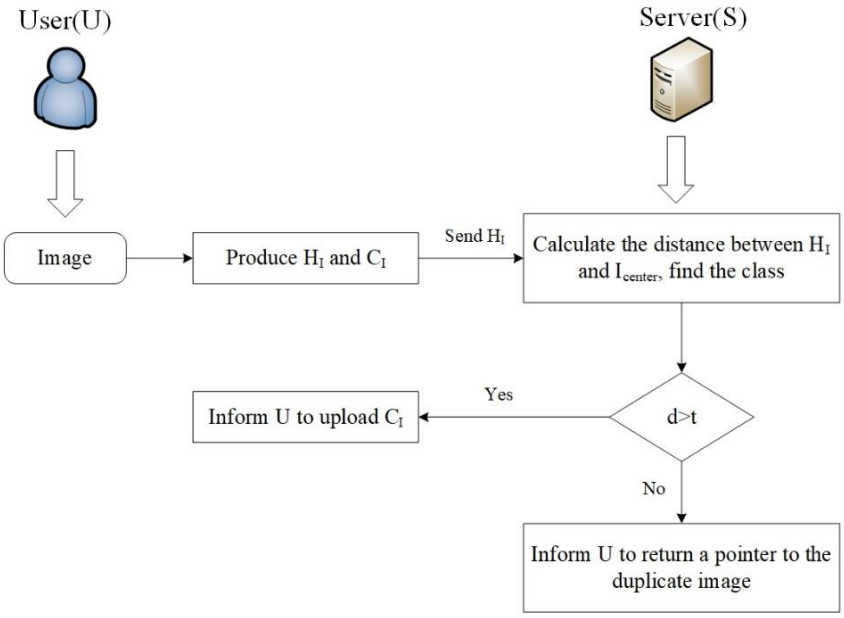

Fig. 4. The process of uploading an image

In the initial state, there are k classes in the image set. Each class has a central image denoted as I_{center}. When a user wants to upload an image, the process is as follows:

Step 1. U uploads the hash value H_I of the image I, and S calculates the Hamming distance from the H_I to the k centroid images I_{center}, respectively.

Step 2. Suppose the distance threshold between H_I and the central image is N. Let D be the distance between H_I and the central image.

If the distance D is less than or equal to the threshold N, we select the nearest class denoted as K_I, then calculate the Hamming distance d between the H_I and the hash values of all images in the class. Assume that the threshold for deduplication is t.

1) $d \leq t$

When the distance d is less than or equal to the specified threshold t, the image I and the image I' in the class are considered to be duplicate images, and the cloud storage server S informs U to return a pointer to the copy, and the user has no need to upload the image.

2) $d > t$

When the distance d between H_I and the hash value of images in the class K is greater than the specified threshold t, the encrypted image C_I will be uploaded and added to the class K_I. Then the new center of the class K_I is recalculated.

If the distance between H_I and every centroid image is greater than the threshold N, we treat it as a new class, then upload the encrypted image C_I and re-cluster.

(3) Download an image

When a user of the user group wants to download images from the image set, the process of downloading an image by the user is as follows:

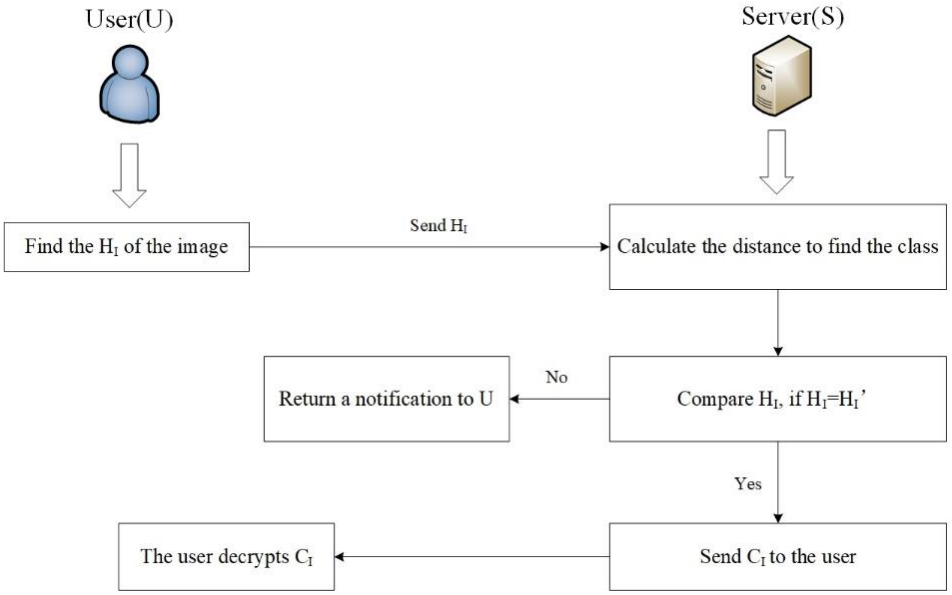

Fig. 5. The process of downloading an image

U finds the hash value H_I of the image I and sends it to S, S first finds the class in which I is located in the storage, and then compares the image fingerprint. If H_I=H_I', the corresponding encrypted image C_I is returned to the user. The user decrypts the image with the private key. If no matching hash value H_I is found，S returns a notification to the user: "The image does not exist."

6. Security Analysis

On the client-side, many transformations of the image can change its binary representation but maintain its human visual perception. The main image modification operations include scaling, color transformation, compression, and rotation. These images can actually be considered duplicate images. Therefore, these duplicate images occupy a large amount of storage space and resources.

In the above solution, the client uses the LBPH algorithm to calculate the image fingerprint H_I, which is the hash value of the original image I. Cloud storage server performs the image detection according to the hash value, so it cannot get image content. During the process of the user uploading and downloading, the attacker will not obtain the original image because the image is encrypted using the secret key. Only users of the user group can decrypt the image, which guarantees the confidentiality of the image.

7. Experiment and Performance Analysis

This section will compare the proposed scheme with the existing schemes SPSD and CSPD, and the performance evaluation of our scheme is also given. In our experiment, we use images in the IVC-LAR database. It includes 8 original color images (4 natural images and 4 art images), 120 distorted images generated from 3 different processing, and 5 compression rates.

For example, in **Fig. 6**, the transformations are JPEG Compression, JPEG2000 Compression, LAR coding, and rotating.

Fig. 6(a). JPEG compression

Fig. 6(b). JPEG2000 compression

Fig. 6(c). LAR coding

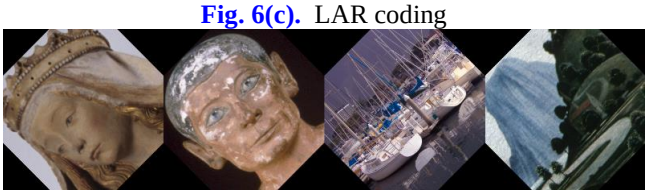

Fig. 6(d). Rotating

The experiment results are shown in **Fig. 7** and **Fig. 8**. Our scheme uses the proposed LBPH algorithm. SPSD adopts the a-Hash algorithm, and CSPD uses the p-Hash algorithm. Hamming distance is used for the comparison between hash values.

First, we define image deduplication accuracy. We use n to indicate the number of duplicate images in the image set. The number of duplicate images actually eliminated in the scheme is denoted as m. The accuracy of image deduplication is calculated by (5):

$$Accuracy = \frac{n - |n - m|}{n} \times 100\% \tag{5}$$

It can be seen from **Fig. 7** that when the threshold of the hash value is small, the deduplication accuracy of the a-Hash and p-Hash algorithms is relatively high. When the threshold is selected larger, the LBPH algorithm in our scheme has higher deduplication accuracy. When the threshold is selected larger, the deduplication accuracy of our scheme does not exceed 100%. That is, the number of duplicate images actually eliminated does not exceed the total number of duplicate images. So it is more accurate and not very sensitive to the threshold. Therefore, the robustness of our scheme is better.

In addition, we adopt the K-means algorithm to cluster images. **Fig. 8** shows the deduplication accuracy of our scheme under different cluster numbers. When the number of clusters is small, the deduplication accuracy is high. So our scheme is more suitable for situations when there are fewer clusters. In our scheme, each image hash value is only compared to the image hash values in the nearest class. For n sample images and k classes,

the time complexity of the deduplication of the CSPD and SPSD is $O(n)$, and in our scheme is $O(n/k)$.

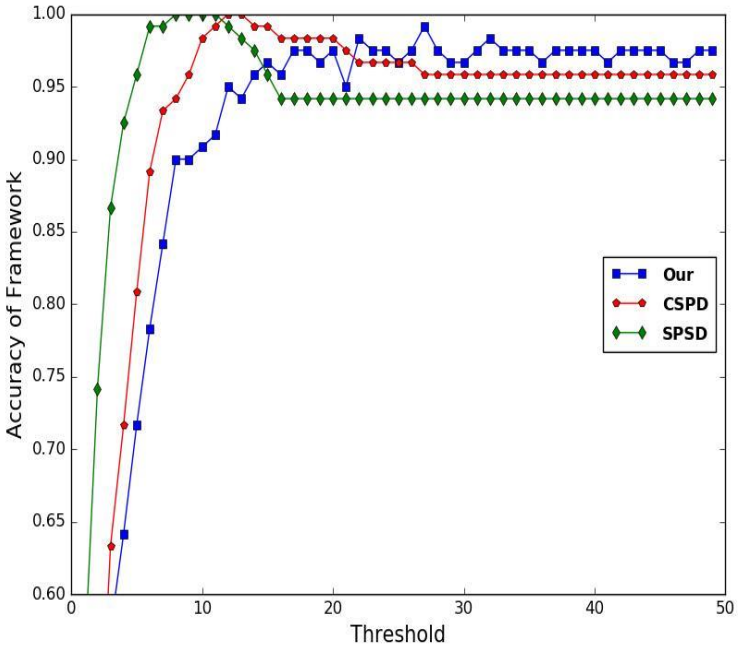

Fig. 7. Framework accuracy under different thresholds

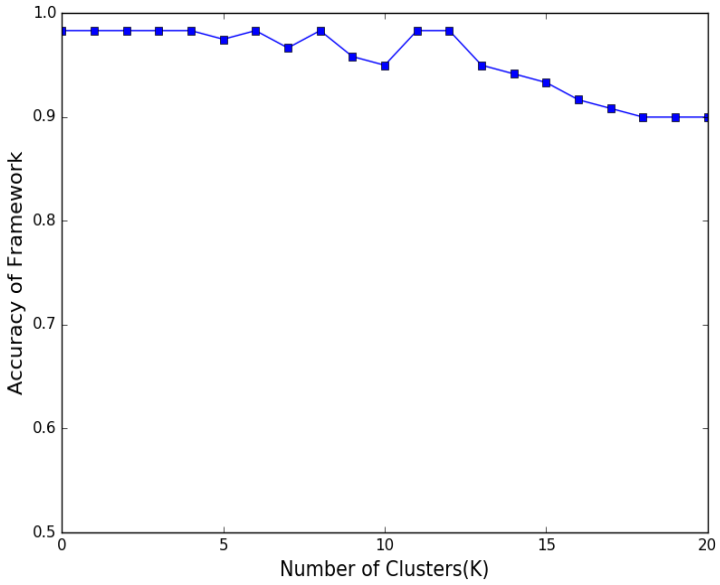

Fig. 8. Framework accuracy under different cluster numbers

Table 2 shows the hash value changes of the data set under the rotation transformation. It can be seen from the table that under the same transformation, our scheme has an enormous amount of hash value change, so it has higher discrimination of images, and the deduplication effect is better than SPSD and CSPD.

Table 2. Hash value changes of data set under rotation transformation

	Mean	Max	Min
Ours	116.12	240.00	84.00
SPSD	11.00	16.00	2.00
CSPD	92.16	128.00	27.00

8. Conclusion

In this paper, we proposed an image deduplication scheme based on LBPH and clustering in cloud storage. We proposed an algorithm called LBPH to detect duplicate images for deduplication check, and we applied it to our deduplication scheme. Moreover, images will be encrypted during transmission. The proposed scheme enables users of the same user group to perform deduplication when uploading images and cluster images on the server-side. This solution eliminates duplicate images from a visual point of view, and the image is encrypted by the client before uploading, thus protecting image confidentiality, saving storage space, increasing storage utilization, and reducing deduplication time. We provide security analysis and performance evaluation to describe that our scheme can ensure certain security and improve the accuracy of deduplication. In addition, although the LBPH algorithm is more accurate than some methods, it is slower. The future work mainly includes the following two aspects: 1) Optimize the image hash algorithm, improve its accuracy and speed to make it more suitable for large-scale data. 2) Study a broader image deduplication scheme to make it suitable for users outside the user group and to ensure the security of the solution.

References

[1] J. Shen, T. Zhou, D. He, Y. Zhang, X. Sun, and Y. Xiang, "Block Design-Based Key Agreement for Group Data Sharing in Cloud Computing," *IEEE Transactions on Dependable and Secure Computing*, vol. 16, no. 6, pp. 996-1010, July 2019. Article (CrossRef Link)

[2] W. Xia, H. Jiang, D. Feng, F. Douglis, P. Shilane, Y. Hua, M. Fu, Y. Zhang, and Y. Zhou, "A Comprehensive Study of the Past, Present, and Future of Data Deduplication," in *Proc. of the IEEE*, vol. 104, no. 9, pp.1681-1710, Aug. 2016. Article (CrossRef Link)

[3] Y. Fu, N. Xiao, H. Jiang, G. Hu, and W. Chen, "Application-Aware Big Data Deduplication in Cloud Environment," *IEEE Transactions on Cloud Computing*, vol. 7, no. 4, pp. 921-934, May 2019. Article (CrossRef Link)

[4] M. Bellare, S. Keelveedhi, and T. Ristenpart, "Message-Locked Encryption and Secure Deduplication," in *Proc. of Annual International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT)*, pp. 296-312, 2013. Article (CrossRef Link)

[5] H. S. Gunawi, N. Agrawal, A. C. Arpaci-Dusseau, J. Schindler, and R. H. Arpaci-Dusseau, "Deconstructing commodity storage clusters," in *Proc. of the 32nd International Symposium on Computer Architecture (ISCA'05)*, pp. 60-71, 2005. Article (CrossRef Link)

[6] W. J. Bolosky, S. Corbin, D. Goebel, and J. R. Douceur, "Single instance storage in Windows® 2000," in *Proc. of the 4th Conference on Usenix Windows Systems Symposium*, pp. 13-24, 2000. Article (CrossRef Link)

[7] T. D. Moreton, I. A. Pratt, and T. L. Harris, "Storage, Mutability and Naming in Pasta," in *Proc. of International Conference on Research in Networking*, pp. 215-219, 2002. Article (CrossRef Link)

[8] L. L. You, K. T. Pollack, and D. D. E. Long, "Deep Store: An Archival Storage System Architecture," in *Proc. of the 21st International Conference on Data Engineering (ICDE'05)*, pp. 804-815, 2005. Article (CrossRef Link)

[9] A. Muthitacharoen, B. Chen, and D. Mazières, "A low-bandwidth network file system," in *Proc. of the 18th ACM Symposium on Operating Systems Review*, vol. 35, no. 5, pp. 174-187, Oct. 2009. Article (CrossRef Link)
[10] L. Zhang and J. Ma, "Image Annotation by Incorporating Word Correlations into Multi-class SVM," *Soft Computing*, pp. 917-927, Feb. 2009. Article (CrossRef Link)
[11] C. Yan, B. Gong, Y. Wei, and Y. Gao, "Deep Multi-View Enhancement Hashing for Image Retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 43, no. 4, pp. 1445-1451, Apr. 2021. Article (CrossRef Link)
[12] C. Yan, B. Shao, H. Zhao, R. Ning, Y. Zhang, and F. Xu, "3D Room Layout Estimation from a Single RGB Image," *IEEE Transactions on Multimedi*a, vol. 22, no. 11, pp. 3014-3024, Jan. 2020. Article (CrossRef Link)
[13] C. Yan, Z. Li, Y. Zhang, Y. Liu, X. Ji, and Y. Zhang, "Depth image denoising using nuclear norm and learning graph model," *ACM Transactions on Multimedia Computing Communications and Applications*, vol. 16, no. 4, pp. 1-17, Dec. 2020. Article (CrossRef Link)
[14] S. Chatterjee and K. Sarawadekar, "An Optimized Architecture of HEVC Core Transform Using Real-Valued DCT Coefficients," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 65, no. 12, pp. 2052-2056, Dec. 2018. Article (CrossRef Link)
[15] Y. Zhuang and L. Liang, "A Novel Local Invariant Feature Extraction Method for High-dynamic Range Images," in *Proc. of the 2nd International Conference on Safety Produce Informatization (IICSPI)*, pp. 307-310, 2019. Article (CrossRef Link)
[16] B. Xiao, K. Wang, X. Bi, W. Li, and J. Han, "2D-LBP: An Enhanced Local Binary Feature for Texture Image Classification," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 9, pp. 2796-2808, Sep. 2019. Article (CrossRef Link)
[17] W. Mao and X. Peng, "WLIB-SIFT: A Distinctive Local Image Feature Descriptor," in *Proc. of IEEE 2nd International Conference on Information Communication and Signal Processing (ICICSP)*, pp. 379-383, 2019. Article (CrossRef Link)
[18] K. W. Su, J. S. Leu, M. C. Yu, Y. T. Wu, E. C. Lee, and T. Song, "Design and implementation of various file deduplication schemes on storage devices," *Mobile Networks and Applications*, vol. 22, no. 1, pp. 40-50, Jan. 2017. Article (CrossRef Link)
[19] R. Chen, Y. Mu, G. Yang, and F. Guo, "BL-MLE: Block-Level Message-Locked Encryption for Secure Large File Deduplication," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 12, pp. 2643-2652, Dec. 2015. Article (CrossRef Link)
[20] Y. Li, D. Wang, and J. Wang, "Perceptual image hash function via associative memory-based self-correcting," *Electronics Letters*, vol. 54, no. 4, pp. 208-210, Feb. 2018. Article (CrossRef Link)
[21] Z. Zhang, Y. Liu, Z. Xiong, J. Li, and M. Zhang, "Focus and Blurriness Measure Using Reorganized DCT Coefficients for an Autofocus Application," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 1, pp. 15-30, Jan. 2018. Article (CrossRef Link)
[22] N. R. A. Rani, S. K. R. Kumar, and P. P. Kumar, "A Survey on Data Redundancy Check in a Hybrid Cloud by using Convergent Encryption," *Indian Journal of Science and Technology*, vol. 9, no. 4, pp. 1-5, Jan. 2016. Article (CrossRef Link)
[23] X. Guo, A. Fu, B. Kuang, and W. Ding, "Secure deduplication and integrity audit system based on convergent encryption for cloud storage," *Journal on Communications*, vol. 38, no. Z2, pp. 156-163, June 2017. Article (CrossRef Link)
[24] Y. Zhou, D. Feng, W. Xia, and M. Fu, "A twice-Hash based convergent encryption strategy for data deduplication," *Computer Engineering & Science*, vol. 38, no. 9, pp. 1755-1762, Sep. 2016. Article (CrossRef Link)
[25] J. Li, Y. K. Li, X. Chen, P. P. C. Lee, and W. Lou, "A Hybrid Cloud Approach for Secure Authorized Deduplication," *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 5, pp. 1206-1216, May 2015. Article (CrossRef Link)
[26] F. Yan, Y. Tan, Q. Zhang, F. Wu, Z. Cheng, and J. Zheng, "An effective RAID data layout for object-based deduplication backup system," *Chinese Journal of Electronics*, vol. 25, no. 5, pp. 832-840, Sep. 2016. Article (CrossRef Link)

[27] H. Shin, D. Koo, Y. Shin, and J. Hur, "Privacy-Preserving and Updatable Block-Level Data Deduplication in Cloud Storage Services," in *Proc. of IEEE 11th International Conference on Cloud Computing (CLOUD)*, pp. 392-400, 2018. Article (CrossRef Link)
[28] H. Gang, H. Yan, and L. Xu, "Secure Image Deduplication in Cloud Storage," in *Proc. of the 3rd International Conference on Information and Communication Technology-EurAsia (ICT-EURASIA) and the 9th International Conference on Research and Practical Issues of Enterprise Information Systems (CONFENIS)*, pp. 243-251, 2015. Article (CrossRef Link)
[29] F. Rashid and A. Miri, "Secure image data deduplication through compressive sensing," in *Proc. of 14th Annual Conference on Privacy, Security and Trust (PST)*, pp. 569-572, 2016. Article (CrossRef Link)
[30] X. Li, J. Li, and F. Huang, "A secure cloud storage system supporting privacy-preserving fuzzy deduplication," *Soft Computing*, vol. 20, no. 4, pp. 1437-1448, Jan. 2016. Article (CrossRef Link)
[31] D. Li, C. Yang, C. Li, Q. Jiang, X. Chen, J. Ma, and J. Ren, "A client-based secure deduplication of multimedia data," in *Proc. of IEEE International Conference on Communications (ICC)*, pp. 1-6, 2017. Article (CrossRef Link)

Lu Chen received the B.Eng. degree in network engineering from Nanjing University of Posts and Telecommunications, Nanjing, China. She is currently pursuing the M.D.-Ph.D. degree in information network from the Nanjing University of Posts and Telecommunications. Her primary research interests are in cloud storage, network security, and blockchain technology.

Feng Xiang received his B.A. degree from Nanjing University and Master of Public Administration from JFK School of Government, Harvard University. He is currently CEO of YTO Express Company Ltd and Director of National Engineering Laboratory for Logistics Information Technology. His research interests include logistics engineering and strategic management of business.

Zhixin Sun is the dean of School of Modern Posts, Nanjing University of Posts and Telecommunications. He received his PHD degree in Nanjing University of Aeronautics and Astronautics, China in 1998 and worked as a post doctor in Seoul National University, South Korea between 2001 and 2002. He has published more than 90 literatures on journals worldwide. His research area includes information security, computer networks, and computer science.