Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2023.037134
Article

Secure Content Based Image Retrieval Scheme Based on Deep Hashing and Searchable Encryption

Zhen Wang, Qiu-yu Zhang*, Ling-tao Meng and Yi-lin Liu

School of Computer and Communication, Lanzhou University of Technology, Lanzhou, 730050, China
*Corresponding Author: Qiu-yu Zhang. Email: zhangqylz@163.com

Received: 25 October 2022; Accepted: 07 April 2023

Abstract: To solve the problem that the existing ciphertext domain image retrieval system is challenging to balance security, retrieval efficiency, and retrieval accuracy. This research suggests a searchable encryption and deep hashing-based secure image retrieval technique that extracts more expressive image features and constructs a secure, searchable encryption scheme. First, a deep learning framework based on residual network and transfer learning model is designed to extract more representative image deep features. Secondly, the central similarity is used to quantify and construct the deep hash sequence of features. The Paillier homomorphic encryption encrypts the deep hash sequence to build a high-security and low-complexity searchable index. Finally, according to the additive homomorphic property of Paillier homomorphic encryption, a similarity measurement method suitable for computing in the retrieval system's security is ensured by the encrypted domain. The experimental results, which were obtained on Web Image Database from the National University of Singapore (NUS-WIDE), Microsoft Common Objects in Context (MS COCO), and ImageNet data sets, demonstrate the system's robust security and precise retrieval, the proposed scheme can achieve efficient image retrieval without revealing user privacy. The retrieval accuracy is improved by at least 37% compared to traditional hashing schemes. At the same time, the retrieval time is saved by at least 9.7% compared to the latest deep hashing schemes.

Keywords: Content-based image retrieval; deep supervised hashing; central similarity quantification; searchable encryption; Paillier homomorphic encryption

1 Introduction

Because of the swift rise in the popularity of portable intelligent terminal devices, people are increasingly interested in using images to record information in life. The scale of the generated image data is also increasing. Customers now prefer using the internet to conduct numerous tasks in their daily lives, even those that need sensitive data [1]. In this case, data owners frequently employ the content-based image retrieval (CBIR) technique to do away with laborious storage and administration;

it achieves accurate and efficient retrieval [2]. Privacy is the top concern of image owners when outsourcing images to cloud servers. Still, cloud servers are only partially secure and trustworthy for image owners [3], and there are various malicious external attacks [4]. For this reason, image retrieval in the encrypted domain is produced, which can be retrieved without decrypting the data and avoiding information leakage, and achieves retrieval performance equivalent to that of plaintext retrieval. However, the encrypted image will lose its original plaintext features, and user's storage and retrieval of image data is limited [5]. Therefore, how to achieve efficient retrieval of images in the ciphertext domain has become a hot research direction.

In recent years, scholars have proposed many content-based encryption-domain image retrieval schemes. To retrieve images from massive datasets accurately and efficiently, compact and rich feature representations are the core of CBIR [6]. It has been proved that the expressive power of features of the traditional method is less effective than the deep features obtained by the multi-layer learning of deep learning. Deep features often have higher dimensions and more semantically aware data for better precision. More profound network architecture helps learn prominent abstract features to bridge semantic gaps [7]. In stochastic gradient descent-based network training, training becomes more difficult as the depth of the network increases because multi-layer backpropagation of the error signal can very quickly induce the phenomenon of "gradient dispersion" or "gradient explosion" [8]. From the perspective of ensemble learning, researchers use feature fusion technology and residual network to reduce the negative impact of network depth on features [9–11] so that the network can be deepened to a deeper level and obtain more image information. Essential features extracted from deep networks are usually complex, the retrieval efficiency could be higher, and the in-depth features must be transformed into more compact codes. Compared with in-depth features, feature binary hash codes have lower complexity and lower retrieval overhead [12]. The existing image-supervised hashing algorithms are developing rapidly. At present, supervised hashing can be divided into shallow supervised hashes Iterative Quantization-Canonical Correlation Analysis (ITQ-CCA) [13], Binary Reconstructive Embeddings (BRE) [14], Kernel-Based Supervised Hashing (KSH) [15], Supervised Discrete Hashing (SDH) [16] and deep supervised hashing Convolutional Neural Networks Hashing (CNNH) [17], Deep Neural Networks Hashing (DNNH) [18], Deep Hashing Network (DHN) [19], HashNet [20], Deep Cauchy Hashing (DCH) [21], Central Similarity Hashing (CSH) [22] according to whether the deep learning framework is used. The core of hashing algorithms is to maintain the similarity of images and improve the quality of hash codes when binarizing real-valued features. Due to their economy in terms of computing and storage, hash algorithms are frequently utilized for large-scale image retrieval [23].

Large-scale CBIR services usually generate a lot of storage and computational overhead, and the emergence of cloud services provides convenience for large-scale retrieval. Cloud servers offer exceptional data storage and processing options but must protect image privacy [24]. When the existing research's image index and query trapdoor is directly sent to the cloud server in plaintext, the similarity distance between the image dataset index and the query trapdoor is quickly passed to the cloud server. The standard image features have a limited ability to represent images. Still, when extracting deep features from image data, the deeper depth features extracted by the network, the greater the computational overhead and the lower the efficiency of the deep neural network. In the existing CBIR schemes with privacy protection, there is a problem that the security of the retrieval system and the retrieval efficiency cannot be balanced.

In summary, the deep neural network used in the existing related research has high computational overhead and low feature extraction efficiency. The security of the image retrieval system in the ciphertext domain cannot be guaranteed. To solve these problems, improve the representation of image features, overcome the computational overhead of deep neural networks, and achieve safe and efficient image retrieval. The proposed scheme uses image datasets with different scales as the research object; the obtained images are the same size after preprocessing the data sets. According to a large amount of data in the image dataset and a large amount of calculation for feature extraction, an image deep learning framework based on transfer learning and residual network is designed, combined with the central similarity quantification (CSQ) [22] algorithm and Paillier homomorphic encryption, a secure, searchable index construction algorithm is proposed. A ciphertext similarity calculation scheme is constructed according to the properties of homomorphic addition and scalar multiplication of Paillier homomorphic encryption to realize the ciphertext retrieval of the secure index. The following is a summary of this work's significant contributions:

1) A deep learning framework based on transfer learning and residual networks is designed, which reduces the computational overhead of deep neural networks and effectively improves the extraction speed of image features while ensuring retrieval accuracy.
2) A secure index construction scheme is designed based on CSQ and Paillier homomorphic encryption. Using the central similarity hashing algorithm, a sufficiently discriminative hash code can be generated from the perspective of global data distribution, which can improve the accuracy of the retrieval system.
3) A similarity calculation method suitable for encrypted domain images is designed, and the homomorphic property of Paillier homomorphic encryption is used to solve the problem of the high computational complexity of ciphertext similarity and reduce the complexity of the retrieval system.

The remainder of this paper is structured as follows: The pertinent research is described in Section 2. The proposed safe CBIR methodology's system model is introduced in depth in Section 3, focusing on developing the deep feature extraction strategy, the central similarity metric quantified hash, and the ciphertext similarity measurement technique. The performance of the suggested system is examined in depth in Section 4, along with an experimental simulation and a comparison to other approaches. Ultimately, the work of this study is summarized in Section 5.

2 Related Work

Secure CBIR technology has essential application value for managing and retrieving encrypted data stored in third parties. For example, image Digital Rights Management (DRM), cloud storage, engineering images, medical data, driver's distraction detection [25], etc. Currently, the existing content-based encrypted domain image retrieval technologies mainly include encrypted image-based secure image retrieval and encrypted feature-based secure image retrieval. The primary distinction between the two techniques is whether the feature extraction is done before or after uploading the encoded image [26].

2.1 CBIR Based on Encrypted Image

CBIR schemes based on encrypted images, such as Ferreira et al. [27] proposed a new cryptographic scheme Image Encryption Scheme CBIR (IES-CBIR), specially designed for images, which moved the privacy-preserving index calculation to the cloud server, reducing the computational pressure on the client and improved the efficiency of retrieval systems. Xia et al. [28] proposed a privacy-preserving CBIR method that can directly extract Local Binary Model (LBP) features from encrypted images using block shuffling, intra-block shuffling, and order-preserving pixel replacement for encrypted images. Xia et al. [29] proposed a scheme for secure retrieval of encrypted images in the

Luminance Bandwidth Chrominance (YUV) color space, using stream encryption, value permutation, and position scrambling encryption algorithms to encrypt the image and extract the Absorption Coefficient (AC) of different color components and Manhattan distance of color histogram to measure similarity. Xia et al. [30] proposed a secure LBP and Bag of Words (BoW) model-based CBIR system, which protects images with a replacement table generated by block permutation and order-preserving encryption, which has better security and retrieval accuracy. Awan et al. [31] proposed a neural network architecture based on the spatial convolution attention mechanism for effective malware categorization. Qin et al. [32] proposed a secure image retrieval method based on deep learning and adaptive weighted fusion using the K Nearest Neighbor (KNN) algorithm and logistic encryption method to effectively protect the security of fusion features and improve the retrieval accuracy of encrypted images. Ma et al. [33] proposed a searchable encoded image retrieval approach in the cloud environment that utilized multi-feature adaptive post fusion, using Red, Green, and Blue (RGB) channel replacement and zigzag scanning to scramble encrypted images, which has good retrieval performance. Tang et al. [34] proposed an encryption method compatible with the Joint Photographic Experts Group (JPEG) format; the cloud server extracts Discrete Cosine Transform (DCT) coefficient features from encrypted images and retrieves similar images, which had strong security and good retrieval performance. Xu et al. [35] proposed a cloud-based content-based personal information-protected image retrieval method that used orthogonal decomposition to divide the image into two parts and perform encryption and feature extraction, respectively, so that the cloud server could right away obtain features from the encrypted image. Liu et al. [36] proposed a content-based ciphertext image retrieval scheme, which used value substitution and position scrambling to encrypt the image and extract the encrypted difference histogram as an image feature vector to extract features in the encryption domain. Wu et al. [37] proposed a new confidentiality-aware deoxyribonucleic acid (DNA) computer system for encrypting medical images, ensuring privacy, and establishing a safe medical atmosphere. Anju et al. [38] proposed a quicker, secure CBIR technique that uses asymmetric scalar product preservation to enable privacy-preserving ranking searches and secure indexing updates by clustering global feature Multimedia Content Description Interface (MPEG-7) visual descriptors of images.

2.2 CBIR Based on Encrypted Feature

The retrieval based on the encrypted feature is an encryption link added to the CBIR framework. For example, Cheng et al. [39] proposed a person re-identification-based surveillance video privacy protection strategy, which can protect the privacy of the detected person while providing personal identification services using convolutional neural network (CNN) and Kernel-based Supervised Hashing (KSH) to extract useful personal re-identification features, which can work efficiently while ensuring personal privacy. Li et al. [40] proposed an affinity retrieval of encrypted images in the cloud server, using CNNs to extract deep features of images, and designed a K-means clustering based on affinity propagation (AP) clustering encrypted hierarchical index tree; the scheme has high retrieval accuracy and efficiency. Weng et al. [41] proposed a multimedia retrieval method that encrypts image feature information using a robust hashing algorithm to protect privacy with high retrieval efficiency. Wang et al. [42] proposed a content-based image retrieval scheme to protect the privacy of multiple users. It provides user privacy protection and cloud data security during an image search. Hassan et al. [43] proposed an encrypted image retrieval scheme using deep neural networks to extract image features and also presented a secure image similarity assessment where a cloud server can modify image properties. It has led to ranking protocols that allow images to be compared and ranked without awareness of their content. Baliga et al. [44] proposed a method for searchable encryption of feature-rich data to solve the similarity calculation problem for large-scale image data. Du et al. [45] proposed a secure image retrieval solution based on an index-encrypted deep hash algorithm and an enhanced 4-dimensional hyperchaotic system. Zhang et al. [46] proposed a novel deep hash-based image retrieval method that can protect the privacy and generate high-quality image hashes. This provides an efficient index structure for agile image retrieval in cloud environments. Xia et al. [47] proposed a CBIR that supports encrypted images, which uses feature vectors to represent the corresponding images, then uses a secure KNN algorithm to protect the feature vectors, uses a position-sensitive hash to construct the pre-filtering table, and uses watermark extraction to track the illegal query users of distributed images, which has high retrieval accuracy and security. Wu et al. [48] proposed an end-to-end architecture based on edge computing; the proposed technique considerably enhances detection performance in low-light settings with minimal latency while operating on edge devices. Xia et al. [49] proposed a secrecy-protected CBIR approach that employed local characteristics to describe images and Earth Mover's Distance (EMD) to evaluate image similarity, with locally sensitive hashing to increase retrieval efficiency and accuracy.

To sum up, compared with image retrieval based on encrypted features, image retrieval based on encrypted images needs to encrypt each pixel of the image, resulting in a large amount of computational data, high computational complexity, and poor representation of extracted features, which is not suitable for large-scale image retrieval. Therefore, this study suggests a secure CBIR approach based on deep hashing and searchable encryption.

3 The Proposed Method

3.1 Secure Image Retrieval Model

Fig. 1 shows the proposed secure CBIR model, which consists of three entities: image owner, cloud server, and users.

As shown in Fig. 1, the image owner must encrypt and upload the original dataset to the cloud server. The original image set data the image owner holds can be expressed as $M = \{m_i\}_{i=1}^{n}$. Where i is the image's order and n is the total number of samples in the dataset. Initially, the image owner initializes the original image dataset, processing the image into images of the same size, length, and width, then encrypting them by Lorenz hyperchaos [50] to obtain the encrypted image set data, which is marked as $W = \{w_i\}_{i=1}^{n}$. Then the image owner extracts the image feature set $P = \{p_i\}_{i=1}^{n}$ through the feature extraction module; the image owner uses the central similarity quantization hash algorithm to hash the extracted image feature set obtaining the deep hash feature set $F = \{f_i\}_{i=1}^{n}$ of the image. The Paillier homomorphic encryption encrypts the extracted deep features to get the secure, searchable index set. Finally, the image owner stores the encrypted image set and index set in the cloud server.

Users build the security index of the relevant image as the query trap gate, develop a feature vector for the image to be queried using the feature extraction module, and send it to the cloud server for image retrieval. The cloud server does the image retrieval process to provide encrypted images that are semantically identical. Additionally, to decode the most relevant encrypted image received from the cloud server and download the associated image, the user must obtain access control authorization from the image owner.

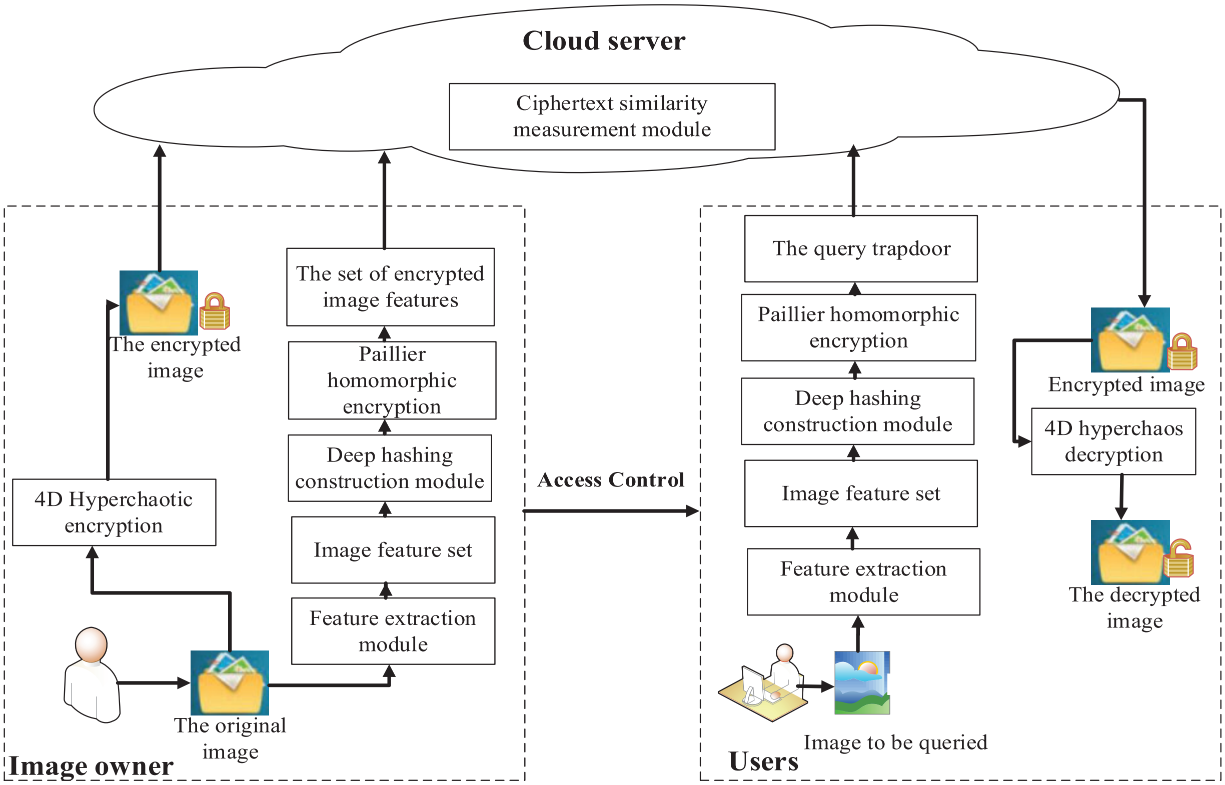

Cloud server
Ciphertext similarity measurement module
The encrypted image
The set of encrypted image features
Paillier homomorphic encryption
Deep hashing construction module
4D Hyperchaotic encryption
Image feature set
Feature extraction module
The original image
Image owner
Access Control
The query trapdoor
Paillier homomorphic encryption
Deep hashing construction module
Image feature set
Feature extraction module
Encrypted image
4D hyperchaos decryption
The decrypted image
Image to be queried
Users

Figure 1: The proposed secure CBIR model

3.2 Deep Learning Framework Based on Transfer Learning and Residual Network

Fig. 2 shows the deep learning framework based on transfer learning and residual network. In this paper, the framework is used to extract features from the original image set and complete the construction of the image feature set.

As shown in Fig. 2, the framework scheme first uses the transfer learning method to extract image features. The source domain dataset is more extensive, and the model's generalization ability in the target domain dataset is better. Therefore, the most significant image dataset, ImageNet, is usually selected as the source domain data. The ResNet50 network model is used to pre-train it to obtain the trained network structure parameters. Then, as the target domain data, a data set of the same type as the source domain data is chosen, and some parameters from the training model are copied to the target model. The knowledge of the source domain data set is transferred to the target domain. With a small amount of training, the retrieval precision is improved, the training cost of the image retrieval system is reduced, and the retrieval efficiency of the image retrieval system is improved. Finally, the result is mapped to the hash layer for the target domain feature. The defined steps for feature extraction using the framework above are:

Step 1: Download the ResNet50 network model that has been trained on ImageNet as a pre-training model;

Step 2: Create a new neural network model as the target model and copy all network structures and model parameters to the target model, except for the last fully connected layer in the pre-trained model;

Step 3: Add a fully connected layer to the target model whose output size is the number of categories in the target domain record, and randomly initialize the model parameters of this layer;

Step 4: Train the target model with the target domain dataset to obtain the newly trained fully connected layer, and the parameters of the remaining layers in the target model are obtained by fine-tuning;

Step 5: Add a ReLU layer to the target model as the hash layer; the hash layer's output is the target domain's eigenvalue.

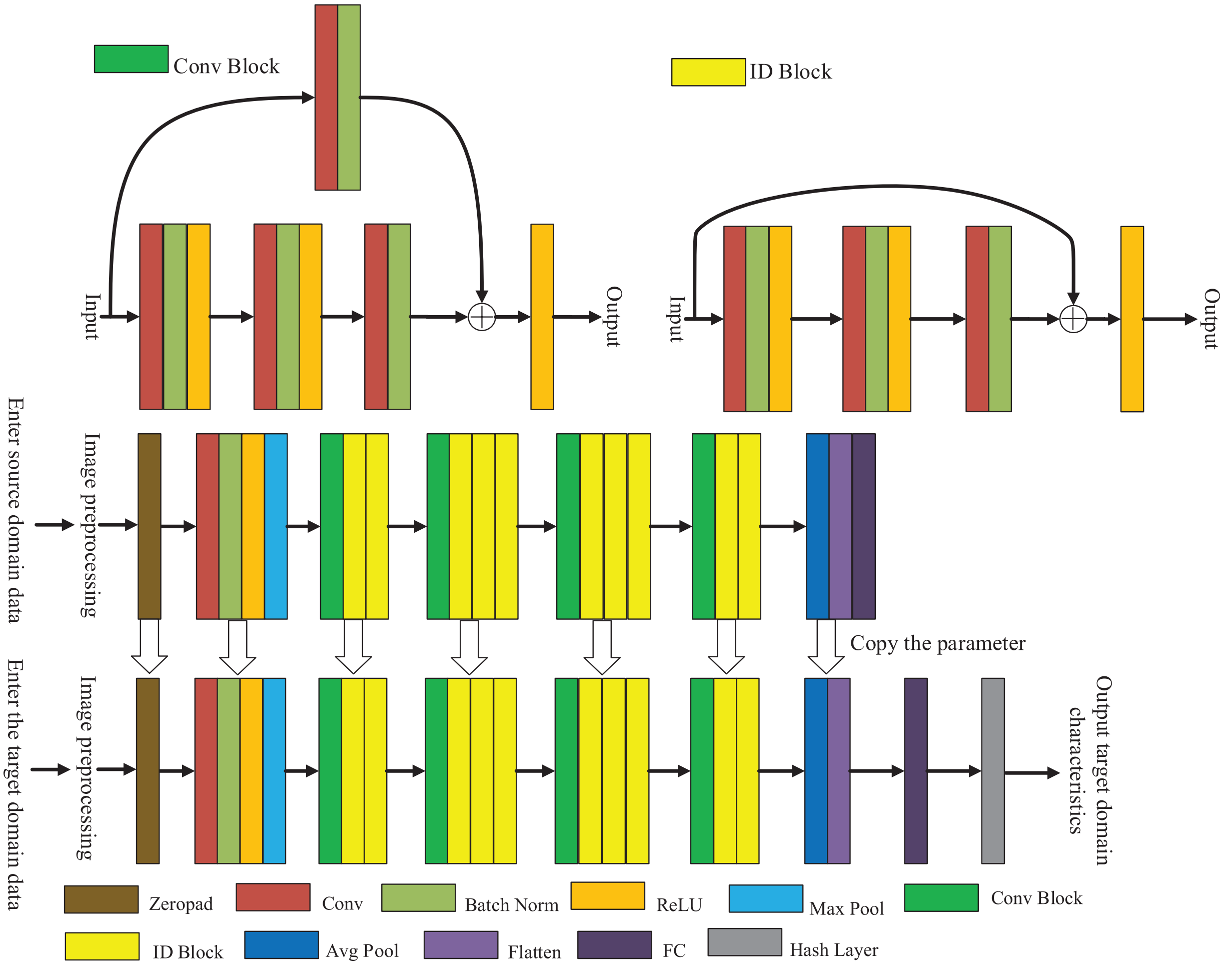


Figure 2: Deep learning framework based on transfer learning and residual network

In Fig. 2, the ResNet50 network model trained on ImageNet is downloaded in the proposed scheme. Some parameters of the trained model are copied to the target model through the fine-tuning technology of transfer learning, which dramatically reduces the retrieval overhead and improves the retrieval efficiency of the target domain dataset to ensure retrieval accuracy. Because the ResNet network model solves the problem of gradient explosion and gradient disappearance faced by traditional networks, the network can be deepened to a deeper level to obtain more information. Moreover, the ResNet network model performs excellently in image classification and recognition. Therefore, download the ImageNet dataset trained the ResNet50 network model as the training model, and the training model, except for the last full connection layer of all network structure and parameters Ψ are copied to the target model, then adding two initialization of all connection layers to the target model network structure, which is marked the final hash layer. Because the source domain dataset ImageNet is large enough, the target model captured by transfer learning has good generalization. When training the target domain dataset, only a few iterations are needed to extract the deep feature set $\boldsymbol{P}$ with a definite meaning in the final fully connected layer.

The feature extraction algorithm is shown in Algorithm 1.

Algorithm 1: Feature extraction algorithm based on transfer learning and residual network

```
Input: Target domain data set $M$; Pre-trained model $\Lambda$; Number of iterations $T_{\mathrm{i}}$; Number of epoch $T_{\mathrm{e}}$;
Output: Target model neural network parameters $\Theta$; Deep feature set $\boldsymbol{P}$;
Initialize the target model $\gamma$; Target model neural network parameters $\Theta$;
All the network structures and model parameters $\Psi$ exclude the final fully connected layer of the before-training model $\Lambda$ are copied to the target model $\gamma$;
Add an initialized fully connected layer and a ReLU layer for the target model $\gamma$;
for $i=1\rightarrow T_{\mathrm{i}}$ do
    Randomly sample $u$ images from $M$ and initialize $\boldsymbol{P}$;
    Randomly extract $m$ images from $M$ as the query set $\boldsymbol{Y}$;
    for $j=1\rightarrow T_{\mathrm{e}}$ do
        1. Select $u$ images from $Y$ as the input of the target model $\gamma$;
        2. Computation with mini-batches in forward propagation $f\ (x;\ \Theta)$;
        3. Get the eigenvalues $\boldsymbol{p_j}$ in the fully connected layer;
        4. Use backpropagation to update the target model neural network parameters $\Theta$;
    end
    Update $\boldsymbol{P}[j]=\boldsymbol{p}_{\mathrm{j}}$;
end
```

3.3 *The Secure Searchable Index Construction Scheme Based on CSQ and Paillier Homomorphic Encryption*

Design idea: The deep features of images extracted by deep learning have higher dimensions, and directly using the deep features of images will have higher computational complexity. Therefore, to reduce the dimensionality of image depth features and give acceptable similarity to high-dimensional data in low-dimensional space, this paper uses a deep hash algorithm based on a central similarity metric to construct global central similarity to optimize and obtain a new loss function, to improve the efficiency of hash learning and retrieval accuracy. Unlike previous hash learning methods, the hash center $\boldsymbol{C}$ based on center similarity quantization is learned from the extracted deep feature $\boldsymbol{P}$. To ensure the security of the image retrieval system, this paper uses Paillier homomorphic encryption to encrypt the extracted deep features before uploading them to the cloud server.

The specific processing steps of the secure, searchable index construction scheme based on central similarity quantization and Paillier homomorphic encryption are as follows:

Step 1: The proposed scheme initially needs to check the hash centroid to construct a deep hash based on the quantization of central similarity. Define hash center $\boldsymbol{C}$ as a set $\boldsymbol{C} = \{\boldsymbol{c}_i\}_{i=1}^{s} \subset \{0, 1\}^{Q}$, where $\boldsymbol{c}_{\mathrm{i}}$ is each hash center, the amount of hash centers is s, while the size of the Hamming space is Q. (Q is set to an even number). In this paper, the hash center is extracted from the Hadamard matrix. Firstly, a Hadamard matrix of $Q\times Q$ is constructed, as shown in Eq. (1):

$$\boldsymbol{H}_Q = \begin{bmatrix} \boldsymbol{H}_{2^{n-1}} & \boldsymbol{H}_{2^{n-1}} \\ \boldsymbol{H}_{2^{n-1}} & -\boldsymbol{H}_{2^{n-1}} \end{bmatrix} = \boldsymbol{H}_2 \otimes \boldsymbol{H}_{2^{n-1}} \tag{1}$$

where stands for Hadamad product. Two initial Hadamard matrices are $\boldsymbol{H}_1$= [1] and $\boldsymbol{H}_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$. When the number of hash centers is $s \leq Q$, each row can be directly selected as the hash center. When the number of hash centers is $Q < s \leq 2Q$, the hash center is constructed using the combination of two Hadamard matrices $\boldsymbol{H}_{2\mathrm{Q}}$=[$\boldsymbol{H}_{\mathrm{Q}}$, $-\boldsymbol{H}_{\mathrm{Q}}$]$^{\mathrm{T}}$. When $s > 2Q$ or Q is not a power of 2, the hash center is determined by random sampling bits from each center vector, and each bit of the hash center $\boldsymbol{c}_{\mathrm{i}}$ is acquired by sampling bits from Bernoulli distribution Bern (0.5).

Step 2: Data sets can be divided into single-label and multi-label data sets according to the label value of each data. For single-label data sets, each category is assigned a hash center. Generate q hash centers $\left\{\boldsymbol{c}_1, \boldsymbol{c}_2, \ldots, \boldsymbol{c}_q\right\}$ corresponding to the set $\left\{\boldsymbol{l}_1, \boldsymbol{l}_2, \ldots, \boldsymbol{l}_q\right\}$ of labels. Data pairs with the equal label share a semantic hash center $\boldsymbol{C}' = \left\{\boldsymbol{c}'_1, \cdots, \boldsymbol{c}'_N\right\}$, where, $\boldsymbol{c}'_i$ is the semantic hash center of training sample $\boldsymbol{p}_{\mathrm{i}}$, and N is the number of data points in training sample $\boldsymbol{P}$. For multi-label datasets, transfer centers are generated by data pairs that share multiple labels. The hash center $\boldsymbol{C} = \left\{\boldsymbol{c}_1, \boldsymbol{c}_2, \ldots, \boldsymbol{c}_q\right\}$ corresponding to q semantic labels $\left\{\boldsymbol{l}_1, \boldsymbol{l}_2, \ldots, \boldsymbol{l}_q\right\}$ is first generated. Then, for data containing more than one category, the centroid $\boldsymbol{c}$ of those centers is computed as the hash center of $\boldsymbol{p}$. To guarantee that the coming about the semantic hash center is binary, the dominant value in the same bit is selected to compute each bit. If the number of zeros equals the number of ones, these bits are sampled from Bern (0.5). Finally, for each $\boldsymbol{p}_{\mathrm{i}} \in \boldsymbol{P}$, the centroid is taken as its semantic hash center $\boldsymbol{C}' = \left\{\boldsymbol{c}'_1, \cdots, \boldsymbol{c}'_N\right\}$.

Step 3: Maximize the logarithmic posterior of the hash code to get the central similarity training objective. The maximum log posterior is used to evaluate the hash code $\boldsymbol{H} = [\boldsymbol{h}_1, \cdots, \boldsymbol{h}_N]$. Since the hash center is a binary vector, the Binary Cross Entropy (BCE) is used to degree the hash center. The distance of Hamming separates the hash code and the semantic hash center $D_H\left(\boldsymbol{c}'_i, \boldsymbol{h}_i\right) = BCE\left(\boldsymbol{c}'_i, \boldsymbol{h}_i\right)$. As a consequence, Eq. (2) illustrates the conditioned probability computation result:

$$\log P\left(c'_i|h_i\right) \propto \frac{1}{K}\sum\nolimits_{k\in K}\left(c'_{i,k}\log h_{i,k} + \left(1 - c'_{i,k}\right)\log\left(1 - h_{i,k}\right)\right) \tag{2}$$

where $P\left(\boldsymbol{c}'_i|\boldsymbol{h}_i\right)$ is the conditional likelihood of the center $\boldsymbol{c}'_i$ given hash code $\boldsymbol{h}_{\mathrm{i}}$. According to the above equation, the conditional probability $P\left(\boldsymbol{c}'_i|\boldsymbol{h}_i\right)$ is more significant, the $\boldsymbol{h}$ is near its semantic hash center $\boldsymbol{c}$', and the Hamming distance between them is smaller. Therefore, the central similarity loss L_C is shown in Eq. (3):

$$L_C = \frac{1}{K}\sum_{i}^{N}\sum_{k\in K}\left[c'_{i,k}\log h_{i,k} + \left(1 - c'_{i,k}\right)\log\left(1 - h_{i,k}\right)\right] \tag{3}$$

Because all hash centers are binary, the current optimization techniques cannot ensure that the hash codes produced are entirely focused on the hash center, so a smooth quantization loss L_Q is presented to improve the produced $\boldsymbol{h}_{\mathrm{i}}$. The measured loss L_Q is shown in Eq. (4) :

$$L_Q = \sum_{i}^{N} \sum_{k=1}^{K} \left(\log\cosh\left(\left|2h_{i,k} - 1\right| - 1\right)\right) \tag{4}$$

where *logcosh* is a smooth function, the central similarity optimization problem can be expressed as Eq. (5):

$$\min_{\Theta} L_T = L_C + \lambda_1 L_Q \tag{5}$$

where, Θ is the collection of all parameters, and λ_1 is the hyperparameter obtained by network look. Depending on the loss function $\min_{\Theta} L_T$, the central similarity is quantified by the standard framework of deep hashing.

Step 4: To improve the security of the image retrieval system and realize searchable encryption, the proposed scheme firstly uses Paillier homomorphic encryption to encrypt the extracted hash code, then send it to the cloud server.

First, generate the public key of Paillier homomorphic encryption. Two prime numbers a and b are randomly selected to meet the definition of Eq. (6):

$$\gcd\left(ab, (a-1)(b-1)\right) = 1 \tag{6}$$

Secondly, calculate $V = ab$ and $\delta = lcm\,(a-1, b-1)$, where *lcm* denotes the least common multiple, the $g \in Z^*_{n^2}$ is selected randomly, satisfying Eq. (7):

$$g = V + 1 \tag{7}$$

Z stands for integer, and the public key (V, g) of Paillier homomorphic encryption is obtained. During encryption, the binary hash code $\boldsymbol{h}$ is converted into A decimal integer $\boldsymbol{h}^t_i$, and $0 \leq \boldsymbol{h}^t < V$, select a random number z, selected as a random number z, which satisfies $0 < z < V$ and $z \in Z^*_{V^2}$ (refer to the existence of multiplicative inverse for z under the remaining system of V^2).

Finally, the ciphertext $\boldsymbol{e}_i = g^{\boldsymbol{h}^t_i} z^V \bmod V^2$ is computed and uploaded to the cloud server. The index construction algorithm based on central similarity quantization and Paillier homomorphic encryption is shown in Algorithm 2.

Algorithm 2: The secure searchable index construction algorithm based on CSQ and Paillier homomorphic encryption

```
Input: The number of hash centers $s$; The dimension of the Hamming space $Q$; The number of the feature set $\boldsymbol{P}$ is $N$; Semantic label $L$; Hyperparameters $\lambda_1$;
Output: Hash center $\boldsymbol{C}$; Semantic hash center $\boldsymbol{C}'$; Security index $\boldsymbol{E}$;
Initialize the size of $Q \times Q$ Hadamard matrix $\boldsymbol{H}_Q = [\boldsymbol{h}^i_a]$, $\boldsymbol{H}_{2Q} = [\boldsymbol{H}_Q, -\boldsymbol{H}_Q]^T = [\boldsymbol{h}^i_{2q}]$;
  for $i = 1 \rightarrow s$ do
    if $\mathrm{s} \leq Q$ & $Q = 2^n$ then
      $\boldsymbol{c}_i = \boldsymbol{h}^i_a$;
    end
    else if $Q < s \leq 2Q$ & $Q = 2^n$ then
      $\boldsymbol{c}_i = \boldsymbol{h}^i_{2Q}$;
```

(Continued)

Algorithm 2: Continued

```
        else
            $\boldsymbol{c}_i$ [random half position] $= 1$;
            $\boldsymbol{c}_i$ [the other posion] $= 0$;
        end
        if $c_i = -1$ then
            $\boldsymbol{c}_i = 0$;
        end
    end
    $\boldsymbol{C} = \{\boldsymbol{c}_1, \ldots, \boldsymbol{c}_s\} \subset \{0, 1\}^Q$;
    Calculate the centroid of the data sample to get the semantic hash center $\boldsymbol{C}' = \left\{\boldsymbol{c}'_1, \cdots, \boldsymbol{c}'_N\right\}$;
    Update $L_C$ according to Eq. (3);
    Update $L_Q$ according to Eq. (4);
    Update $\min_{\Theta} L_T$ according to Eq. (5);
    Take $\min_{\Theta} L_T$ as the loss function of the target neural network;
    Generate hash code $\boldsymbol{H} = [\boldsymbol{h}_1, \cdots, \boldsymbol{h}_N]$;
    Update $a$ and $b$ according to Eq. (6);
    $V = ab$;
    $\delta = lcm\,(a - 1, b - 1)$;
    Update $g$ according to Eq. (7);
    Choose the random number $z$;
    for $i = 1 \rightarrow N$ do
        Convert the obtained binary hash code $\boldsymbol{h}_i$ to decimal $\boldsymbol{h}_i^t$;
        $\boldsymbol{e}_i = g^{\boldsymbol{h}_i^t} z^V \bmod V^2$;
        $\boldsymbol{E}\,(i) = \boldsymbol{e}_i$;
    end
    Return $\boldsymbol{E} = \{\boldsymbol{e}_1, \cdots, \boldsymbol{e}_N\}$
```

3.4 Similarity Measurement Scheme Based on Paillier Homomorphic Encryption Homomorphism

One of the primary methods used in CBIR is similarity measurement. This work develops a solution based on Paillier homomorphic encryption for the similarity assessment module in Fig. 1. The processing flow of Paillier homomorphic encryption similarity measurement is depicted in Fig. 3.

As shown in Fig. 3, the proposed scheme builds the index table with features which are to one by one correspondence encrypted image dataset, return the most similar r image encrypted features via Paillier homomorphic subtraction between feature index table and query trapdoor, then query the index table to obtain the corresponding image serial number; Finally, r ciphertext images are obtained by querying the corresponding image sequence number, it returns to the user.

This paper uses the extracted ciphertext feature hash value $\boldsymbol{e}_i$ as the feature vector of the image for retrieval. The specific processing steps are as follows:

Step 1: Build a feature index table in the cloud server for the feature set $\boldsymbol{E}$ after Paillier homomorphic encryption, build a ciphertext image dataset for the image set $\boldsymbol{W}$ after Lorenz hyperchaotic encryption, and make them correspond to each other one by one.

Step 2: Calculate the distance between the query trapdoor $\boldsymbol{e}_q$ and all features $\{\boldsymbol{e}_1, \cdots, \boldsymbol{e}_N\}$ in the feature index table by Paillier homomorphic subtraction. Homomorphic subtraction is calculated as follows: For arbitrary plaintext $t_1, t_2 \in Z$ and $y_1, y_2 \in Z^*$, the corresponding ciphertext $\boldsymbol{e}_1 = \boldsymbol{E}(t_1), \boldsymbol{e}_2 = \boldsymbol{E}(t_2)$ satisfies Eq. (8):

$$\begin{aligned} \boldsymbol{e}_1 \cdot \boldsymbol{e}_2^{-1} &= \boldsymbol{E}(t_1) \cdot (\boldsymbol{E}(t_2))^{-1} \\ &= g^{t_1} \cdot y_1^V \bmod V^2 \cdot \left(g^{t_2} \cdot y_2^V \bmod V^2\right)^{-1} \\ &= g^{t_1 - t_2} \cdot \left(y_1 \cdot y_2^{-1}\right)^V \bmod V^2 \\ &= \boldsymbol{E}(t_1 - t_2) \end{aligned} \tag{8}$$

Step 3: According to the result of homomorphic subtraction, query the index table, and return r ciphertext images corresponding to the closest r eigenvectors to the user who has obtained the access right.

Step 4: The key K of Lorenz hyperchaotic obtained by the user according to access control AC decrypt the ciphertext image to acquire the decoded retrieval image.

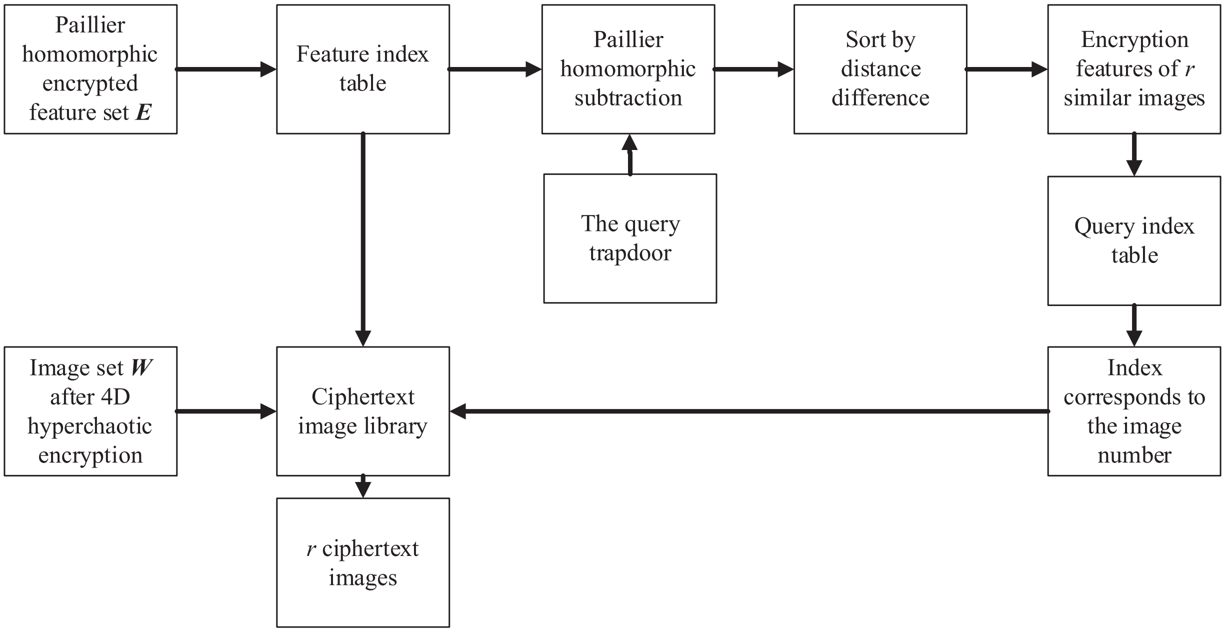

Paillier homomorphic encrypted feature set E
Feature index table
Paillier homomorphic subtraction
Sort by distance difference
Encryption features of r similar images
The query trapdoor
Query index table
Image set W after 4D hyperchaotic encryption
Ciphertext image library
Index corresponds to the image number
r ciphertext images

Figure 3: Similarity measurement scheme based on Paillier homomorphic encryption

4 Experimental Results and Performance Analysis

The experiment focuses on three aspects: security analysis, retrieval accuracy, and retrieval efficiency, and compares the proposed scheme's performance to that of other similar CBIR systems. The proposed scheme was tested using NUS-WIDE [42], ImageNet [43], and MS COCO [44] image datasets. The experimental hardware environment is CPU: Inter(R) Xeon(R) Sliver 4210 CPU @2.20GHz, GPU: NVIDIA GeForce RTX 3080 Ti, memory: 36 GB (2933 MHz). The software environment is Windows 10, PyCharm (Professional Edition) 2020.3.2x64, Anaconda, CUDA 11.4, cuDNN v.8.2.4. The deep learning framework is Pytorch. The pre-training model selects ResNet34, ResNet50, and ResNet101, which have been trained on the ImageNet dataset. The network optimizer selects Adam, the learning rate is 0.0001, the learning rate decay is 0.9, and the weight decay is 0.000001.

4.1 Security and Performance Analysis

Cloud servers are trustworthy and enquiring because they can execute relevant protocol operations properly and attack and examine sensitive user data. In this part, the effectiveness of the suggested picture security retrieval technique is evaluated in terms of three different criteria: image privacy security, image feature privacy security, and decryption-restored image quality.

Image privacy and security: The proposed approach employs the Lorenz hyperchaotic system to encrypt images. The key stream is repeatedly created once the initial parameter values of the hyperchaotic system are established. Then, the image is subjected to modulo diffusion, changing the position and gray value of the pixels to realize the first hiding of image information. Then, after the pixel matrix of the two-dimensional image is changed into a one-dimensional vector, the one-dimensional vector is scrambled without repetition to hide the image's original information and realize again the second image information is hiding. The finite field diffusion technique is used to learn the third image information concealing. Finally, the modular diffusion is repeated twice, and the information of the plaintext image pixels is diffused into the entire ciphertext image to realize the fourth concealment of the image information. The information on the image pixel position and gray value size has been completely changed, and the ciphertext image no longer contains any information about the plaintext image. Therefore, the image encryption method used in this paper has high security.

Image feature privacy and security: The deep hash features of images are first extracted by the proposed scheme's secure, searchable index construction scheme, which then uses the Paillier homomorphic encryption algorithm to create a secure, searchable index before uploading it to the cloud server. Compared with the underlying features of the image, such as color, texture, shape, gradient, and other underlying features, the image information mined by the depth feature of the image is deeper and more abstract. The deep hash algorithm used in the proposed scheme is also a kind of hash algorithm, which is irreversible, and knowing the result of the hash algorithm, there is no way to convert it to the original target. However, the distances of hash values between similar features may be equal. The cloud server may judge whether two images are identical according to the distance between the hash results, so the hash values must be encrypted. To encrypt the hash value, this work uses the Paillier homomorphic encryption technique; the encrypted hash data has a lot of redundant data, so the cloud server cannot speculate what the original hash data is. Therefore, the searchable index constructed in this paper is safe.

Decryption and restoration of image quality: In searchable encryption, peak signal-to-noise ratio (PSNR) can measure the performance of image encryption and the decryption and restoration of image quality. PSNR mainly examines the error between the pixels of the plaintext image and the ciphertext image at the corresponding position. Give a ciphertext image w and a plaintext image m of the same size $X \times Y$, the mean square error (MSE) is defined as Eq. (9):

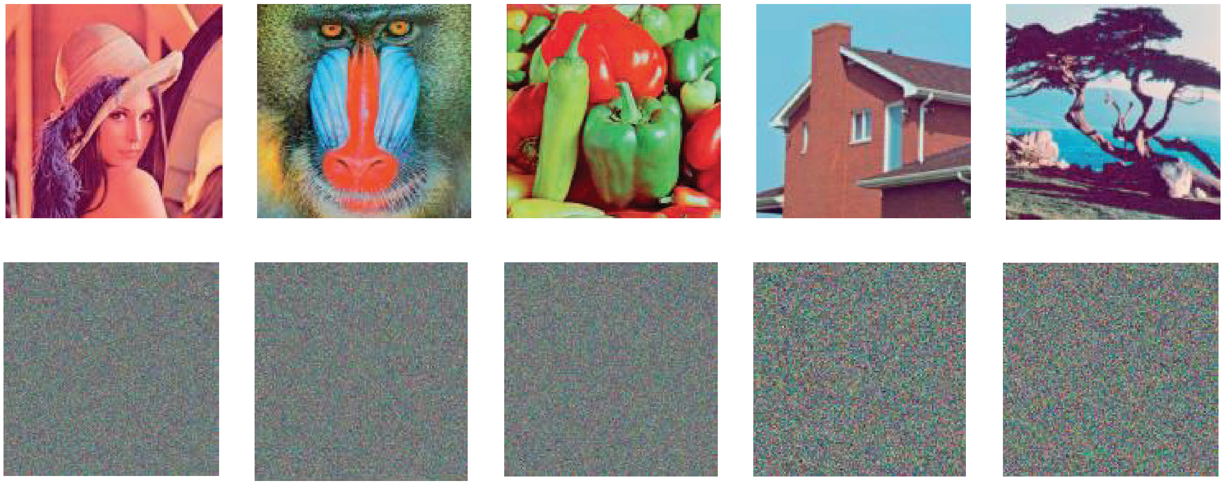
$$MSE = \frac{1}{X * Y} \sum_{i=1}^{X} \sum_{j=1}^{Y} \left(w\left(i,j\right) - m\left(i,j\right) \right)^2 \tag{9}$$

PSNR is calculated using the following Eq. (10):

$$PSNR = 10\log_{10}\frac{(2^n - 1)^2}{MSE} \tag{10}$$

where n is the total number of pixels in the image.

The greater the image distortion, the greater the difference between the plaintext and ciphertext images, and the better the encryption result, the lower the PSNR value. The encryption quality of the image is shown in Fig. 4.



PSNR=8.3731 PSNR=8.5064 PSNR=8.4962 PSNR=8.4637 PSNR=8.3947

Figure 4: Image encryption quality

As shown in Fig. 4, the encrypted ciphertext image is incomprehensible, and the PSNR value is low, which means the image distortion before and after encryption is greater. The malicious attacker cannot infer the image content from the encrypted image. As a result, the encryption system used in this research satisfies all of the demands for image decoding and safe recovery. Fig. 5 shows the PSNR comparison results of the proposed encryption scheme with the other three image retrieval schemes, Aggregate Deep Fast Supervised Discrete Hashing (ADFSDH) [24], Orthogonal Decomposition Project (ODP) [35], and Improved Deep Pairwise Supervised Hashing (IDPSH) [45].

As shown in Fig. 5, the PSNR values of the encryption scheme adopted in this scheme and other schemes for 5 common plaintext images and ciphertext images are all below 10. When the PSNR value is less than 10, It is challenging to distinguish by the eye whether the plaintext image is the same as the ciphertext image. This paper uses an encryption scheme compared with the other three encryption schemes in five common image encryption before and after the PSNR value is the lowest, the corresponding error between pixels is bigger, and the encryption image before and after the gap is relatively obvious, obtain the encryption effect is better.

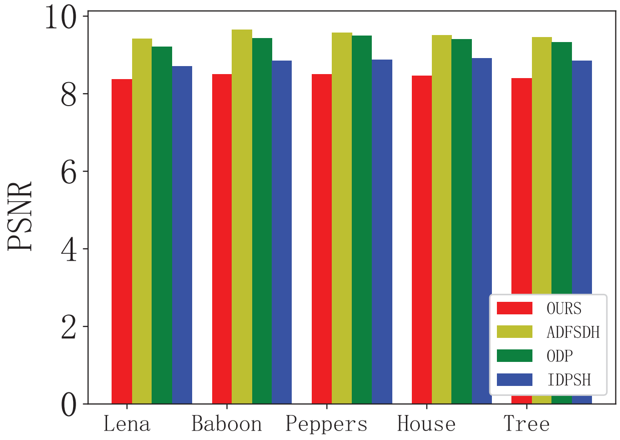


Figure 5: PSNR of encryption schemes in different encryption domain image retrieval schemes

4.2 Image Retrieval Performance Analysis

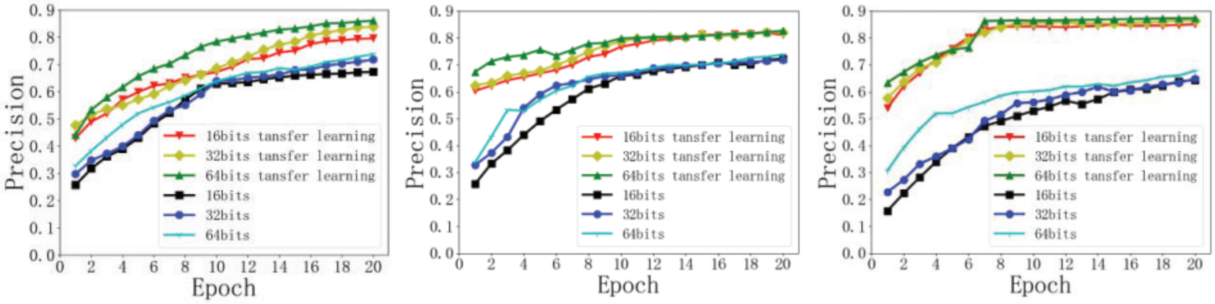
In this paper, the proposed imagined secure retrieval scheme's retrieval capability is verified on three benchmark datasets: NUS-WIDE [51], ImageNet [52], and MS COCO [53]. Among them, the NUS-WIDE multi-label dataset contains 81 categories and 269,648 image data in the dataset. This paper selects 12,600 image data from the original dataset, which belongs to the 21 most common categories, and two images are considered similar when they share at least one label. ImageNet is a single tag data set, it contains 1,000 categories, the data set has more than 1.2 million training images, and there are 50,000 images in the validation set; this paper randomly selects 100 categories, using concentrating all these categories of training image database, and all images in validation used for query, when tags of two images are the same, they are considered as a pair of similar images. MS COCO is a multi-label image recognition, segmentation, and captioning data set. The current version contains 80 categories, 82,783 training images, and 40,504 validation images. After pruning the images without category information, the training and validation images are combined to obtain 122,218 images. In this paper, 5,000 images are randomly selected as queries; the remaining images are used as databases, and then 10,000 images are randomly selected from the database as the training set.

4.2.1 Accuracy Analysis of Network Model

Transfer the knowledge of the largest ImageNet data set to the target data set with transfer learning, make after using the transfer study does not need training for new image data sets from scratch and can greatly save time cost, and computational overhead. To increase the retrieval system's effectiveness, this paper uses ResNet34, ResNet50, and ResNet101 residual network structures as pre-training models and constructs corresponding target models. To obtain the best hash coding length for image representation ability, this paper uses the hash coding with the hash length of 16/32/64 and whether to use the transfer learning model for training respectively to conduct experiments and evaluates their test accuracy and training cost.

Fig. 6 shows the test accuracy curves of the first 20 training epochs using the ResNet50 model alone and the ResNet50 model using transfer learning on different datasets.

As shown in Fig. 6, the network model applying transfer learning in the three benchmark datasets can achieve greater and more consistent test accuracy after a few iterations than the model without transfer learning. The ResNet50 network model, which was trained on the ImageNet dataset, serves as the transfer learning model in this study. The part of parameters and network structure of the pre-trained ResNet50 network model was directly copied to the target model by transfer learning; because the pre-trained model was trained on the largest dataset ImageNet, the training data of the target model was expanded invisibly, which made the whole model more robust and have stronger generalization ability.



(a)ResNet50 @NUS-WIDE (b)ResNet50 @MS COCO (c)ResNet50 @ImageNet

Figure 6: Comparison of test accuracy curves of the ResNet50 model under different data sets

To further evaluate the effectiveness of network models utilizing various transfer learning under various hash coding lengths, mean Average retrieval Precision (mAP) [54] was used to identify the network structure of the best model. The mAP values of hash sequences of various lengths under various data sets for retrieval strategies utilizing various transfer learning models are shown in Table 1.

Table 1: mAP values corresponding to hash sequences with different network models

Model	NUS-WIDE			MS COCO			ImageNet		
	16 bits	32 bits	64 bits	16 bits	32 bits	64 bits	16 bits	32 bits	64bits
ResNet34	0.746	0.761	0.795	0.769	0.792	0.807	0.771	0.821	0.832
ResNet50	**0.821**	**0.833**	**0.843**	**0.812**	**0.841**	**0.872**	**0.856**	**0.867**	**0.878**
ResNet101	0.813	0.831	0.838	0.820	0.834	0.842	0.856	0.868	0.881

On the three benchmark datasets, as shown in Table 1, the network model of transfer learning utilizing ResNet34, ResNet50, and ResNet101 exhibited high retrieval accuracy, demonstrating the viability of the suggested approach. The network model utilizing ResNet50 transfer learning produced improved retrieval accuracy on three distinct length hash codes compared to ResNet34 and ResNet101 models. ResNet50 had a deeper network layer than ResNet34, the extracted features were more meaningful, and the classification accuracy was higher. Compared with ResNet50, ResNet101 performed better on complex ImageNet datasets, but it was lower than ResNet50 on relatively simple and less task size NUS-WIDE and MS COCO datasets. ResNet101 performed better on complex data due to its intricate network structure. However, the over-fitting phenomenon occurs on data sets with less volume, decreasing the retrieval accuracy. Moreover, ResNet101 had deeper network layers, a more complex network structure, and more parameters to be trained, so the training time of ResNet101 was longer. Therefore, the retrieval efficiency of the system was reduced. Therefore, this paper adopts the ResNet50 transfer learning network structure.

4.2.2 Retrieval Performance Analysis

The mAP values of the suggested scheme were compared and analyzed with the current 10 classical hash schemes to confirm the retrieval performance of the new scheme. Include six deep hashing techniques. The mAP values of various length hash sequences with various retrieval strategies are displayed in Table 2.

Table 2: mAP values corresponding to different retrieval schemes and hash sequences of different lengths

Scheme	NUS-WIDE			MS COCO			ImageNet		
	16 bits	32 bits	64 bits	16 bits	32 bits	64 bits	16 bits	32 bits	64 bits
ITQ-CCA	0.435	0.435	0.435	0.566	0.562	0.502	0.266	0.436	0.576
BRE	0.485	0.525	0.544	0.592	0.622	0.634	0.063	0.253	0.358
KSH	0.394	0.407	0.399	0.521	0.534	0.536	0.160	0.298	0.394
SDH	0.575	0.590	0.613	0.554	0.564	0.580	0.299	0.455	0.585
CNNH	0.655	0.659	0.647	0.599	0.617	0.620	0.655	0.659	0.647
DNNH	0.703	0.738	0.754	0.644	0.731	0.745	0.703	0.738	0.754
DHN	0.712	0.759	0.771	0.719	0.773	0.788	0.712	0.759	0.771
HashNet	0.622	0.701	0.739	0.745	0.773	0.788	0.757	0.775	0.790
DCH	0.773	0.795	0.818	0.759	0.801	0.825	0.652	0.737	0.758
CSH	0.810	0.825	0.839	0.796	0.838	0.861	0.851	0.865	0.873
Proposed	**0.821**	**0.833**	**0.843**	**0.812**	**0.841**	**0.872**	**0.856**	**0.867**	**0.878**

Table 2 shows that the proposed approach outperformed the other 10 existing deep hash strategies regarding retrieval accuracy for three different length hash sequences on the NUS-WIDE, MS COCO, and ImageNet datasets. Table 2 used the ResNet50 network model as the backbone of six deep hashing methods. At the same time, it is used as the pre-trained model and target model of this paper. The mAP value of the proposed scheme is much higher than that of the four supervised shallow methods; the proposed scheme increases respectively by 37%, 46.6%, and 50.1% on NUS-WIDE, MS COCO, and ImageNet. Especially on the ImageNet dataset, the improvement was the most obvious. Compared with the four supervised shallow hash schemes, the retrieval accuracy of the proposed scheme, and the six deep supervised hashing schemes, the retrieval accuracy of the proposed scheme has been improved to different degrees. It can be found that deep supervised hashing can directly guide the learning of deep features with supervised information. It was a compatible hash learning architecture with good retrieval accuracy.

Compared with the latest deep supervised hashing methods such as ADFSDH, IDPSH, Deep Fuzzy Hashing Network (DFHN) [55], Joint Learning based Deep Supervised Hashing (JLDSH) [56], and Deep Uncoupled Discrete Hashing (DUDH) [57], the proposed scheme still achieved better retrieval accuracy. Table 3 shows the mAP values of different length hash sequences in different depth-supervised hash retrieval schemes in NUS-WIDE, MS COCO, and ImageNet datasets.

As can be seen from Table 3, the proposed scheme achieved high retrieval accuracy on the three data sets and the highest retrieval accuracy on the ImageNet dataset. Deep supervised hashing can directly guide the learning of deep features with supervised information. It was a compatible hash learning architecture with good retrieval accuracy. By comparing the proposed scheme, JLDSH, and DUDH, it was found that although the proposed scheme achieved the highest retrieval accuracy on the ImageNet dataset, it failed to achieve the highest performance on NUS-WIDE and MS COCO datasets because both NUS-WIDE and MS COCO datasets were multi-label datasets. The DUDH scheme implicitly maintained the similarity between the database and the query through the similarity passing set. It performed well on multi-label datasets where the similarity was more complex. JLDSH sets hyperparameters on supervised information through joint learning, and the hash encoding values obtained by JLDSH were more symbolic, so it also performed well on multi-label datasets. The proposed scheme adopted a transfer learning method to copy the parameters from the pre-training model with strong generalization ability, expand the size of the training set in a disguised way, the extracted features had more substantial meaning, and the model had better convergence after training. In addition, hash coding was carried out from a global perspective, and the retrieval accuracy was higher. On the premise of ensuring retrieval accuracy, if the training time for the training set were shorter, the efficiency of the whole retrieval system would be higher. Table 4 shows the training time of 3 datasets with different hashing lengths under different depth hash schemes when the ResNet50 network model was the backbone.

Table 3: mAP values corresponding to different retrieval schemes and different length hash sequences

Scheme	NUS-WIDE			MS COCO			ImageNet		
	16 bits	32 bits	64 bits	16 bits	32 bits	64 bits	16 bits	32 bits	64 bits
ADFSDH	0.713	0.733	0.749	0.732	0.741	0.748	0.694	0.703	0.713
IDPSH	0.809	0.818	0.829	0.796	0.811	0.826	0.786	0.810	0.811
DFHN	0.528	0.530	0.532	0.521	0.534	0.536	0.483	0.498	0.504
JLDSH	0.862	0.900	0.910	0.874	0.880	0.884	0.831	0.844	0.853
DUDH	0.875	0.900	0.911	0.886	0.901	0.904	0.812	0.859	0.871
Proposed	**0.821**	**0.833**	**0.843**	**0.812**	**0.841**	**0.872**	**0.856**	**0.867**	**0.878**

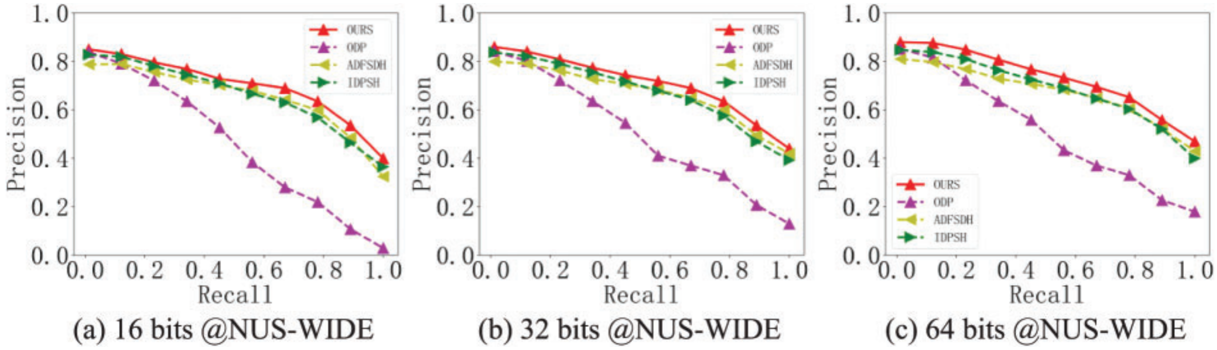
Table 4: Training time of different depth hashing schemes for 3 datasets with different hashing lengths (min)

Scheme	NUS-WIDE		MS COCO		ImageNet	
	32 bits	64 bits	32 bits	64 bits	32 bits	64 bits
DHN	5.31×10^2	5.56×10^2	5.73×10^2	5.86×10^2	5.46×10^2	5.61×10^2
HashNet	8.75×10^2	8.92×10^2	8.11×10^2	8.38×10^2	8.26×10^2	8.41×10^2
CSH	1.62×10^2	1.69×10^2	1.36×10^2	1.41×10^2	1.06×10^2	1.17×10^2
ADFSDH	5.43×10^2	5.63×10^2	5.54×10^2	5.71×10^2	5.78×10^2	5.96×10^2
IDPSH	5.36×10^2	5.59×10^2	5.56×10^2	5.77×10^2	5.76×10^2	5.93×10^2
DFHN	6.73×10^2	6.91×10^2	6.81×10^2	6.98×10^2	6.93×10^2	7.15×10^2
JLDSH	4.36×10^2	4.51×10^2	4.65×10^2	4.76×10^2	4.86×10^2	4.97×10^2
DUDH	1.56×10^2	1.63×10^2	1.22×10^2	1.28×10^2	1.04×10^2	1.13×10^2
Proposed	$\mathbf{1.28 \times 10^2}$	$\mathbf{1.37 \times 10^2}$	$\mathbf{1.08 \times 10^2}$	$\mathbf{1.11 \times 10^2}$	$\mathbf{0.93 \times 10^2}$	$\mathbf{1.02 \times 10^2}$

As seen from Table 4, the scheme in this paper had the shortest training time for the three datasets with different length hash codes, so it had the highest feature extraction efficiency. Compared with the latest deep hash construction scheme, the retrieval time of this paper's three benchmark data sets was improved by at least 9.7%. In this paper, transfer learning and central similarity measure were combined with copying the parameters from the network model with good generalization ability, avoiding a large amount of computation overhead and time cost; hash coding was carried out from a global perspective, optimizing the overall network structure. Hash coding was generated with the iteration of the network. It decreased training time and hash code generation time. It increased the retrieval effectiveness of the retrieval system by avoiding the requirement for further calculation to acquire estimated hash values after acquiring eigenvalues.

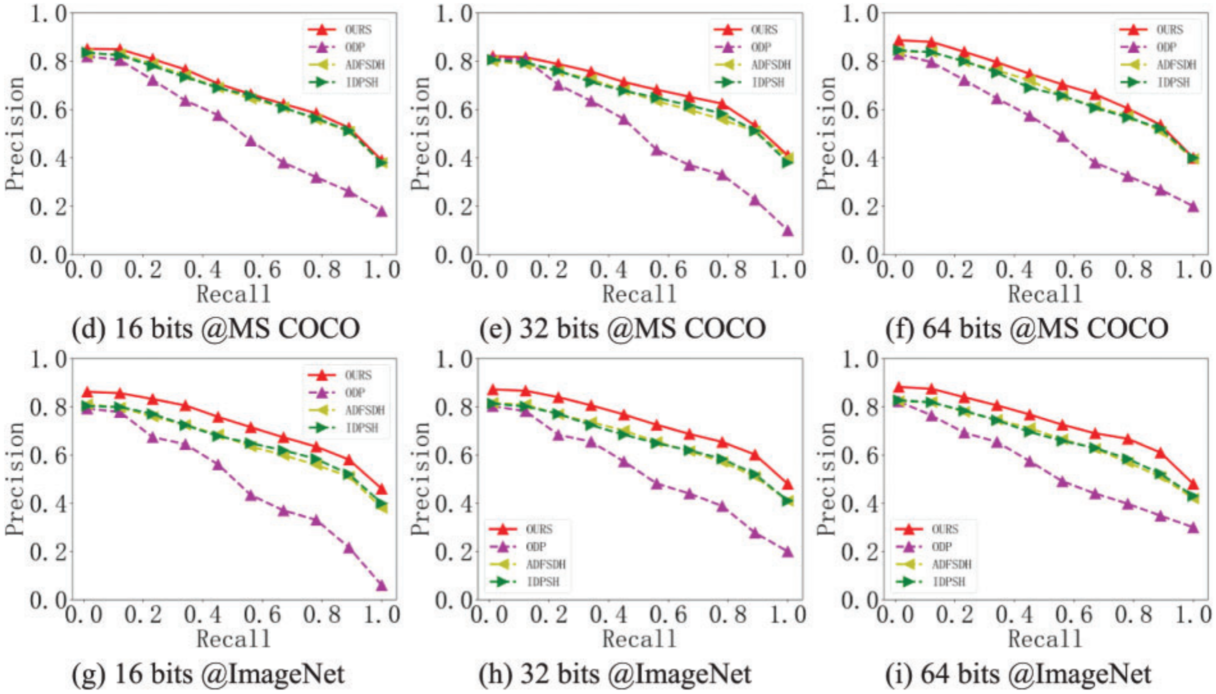
The proposed retrieval scheme was compared with other exting privacy-preserving image retrieval schems to verify the retrieval performance of the proposed retrieval scheme on ciphertext images. This study utilized the recall rate (R) and precision rate (P) to assess how well the system performed while retrieving data. The performance of the retrieval scheme was analyzed by comparing the P-R curve of different schemes and the area enclosed by the coordinate axes. This paper took NUS-WIDE, MS COCO, and ImageNet datasets as examples to compare four privacy-preserving image retrieval schemes with three different length hash codes. Fig. 7 shows the retrieval performance of the retrieval accuracy recall curve on the different datasets.

As shown in Fig. 7, the proposed scheme achieved the best results on 3 datasets. It proved that the scheme in this paper had good retrieval accuracy on the premise of ensuring security. Compared with the ODP scheme, the other three deep hashing schemes had good retrieval performance under different length hash codes, which indicated that the features extracted by deep hashing in privacy-preserving image retrieval were more meaningful and more suitable for large-scale encryption domain image retrieval. Compared with other deep hash schemes ADFSDH and IDPSH, the proposed scheme used the hash coding method of central similarity quantization. Globally, the proposed approach learns the hash coding by minimizing the Hamming distance between hash codes and related hash centers and bringing comparable data pairings' hash codes closer to a common center. As separate data pairings' hash codes converged to distinct centers, the retrieval accuracy was considerably increased. Additionally, the retrieval accuracy and efficiency of the retrieval system may be increased by training a transfer learning model with a good generalization ability on image data.



(a) 16 bits @NUS-WIDE (b) 32 bits @NUS-WIDE (c) 64 bits @NUS-WIDE

Figure 7: (Continued)



(d) 16 bits @MS COCO (e) 32 bits @MS COCO (f) 64 bits @MS COCO

(g) 16 bits @ImageNet (h) 32 bits @ImageNet (i) 64 bits @ImageNet

Figure 7: Performance of retrieval accuracy recall curves on different datasets

5 Conclusion

This research proposes a secure image retrieval technique based on deep hashing and searchable encryption. This approach enhances the searchability of enormous image data on the cloud while maintaining privacy. The suggested technique enhances the significance of image features, retrieval accuracy, and retrieval efficiency by generating hash codes using a transfer learning model and central similarity quantization (CSQ). The following are the key contributions: 1) Transfer learning and ResNet50 are used to create a deep neural network model that has a greater beginning performance, a quicker rate of model improvement, and better model convergence, which lowers the complexity of feature extraction. 2) The CSQ and Paillier homomorphic encryption creates a safe, searchable index with a deeper meaning to improve retrieval precision. 3) A similarity measure approach appropriate for Paillier homomorphic encryption is developed to increase the security of picture retrieval in the encrypted domain.

Experimental results show that the proposed ciphertext domain image retrieval method has a higher mAP value, more extensive P-R curve coverage, and shorter retrieval time than other content-based image retrieval schemes. It is proved that this method has higher security, retrieval precision, and retrieval efficiency.

The content-based ciphertext domain image retrieval approach suggested in this research only partially uses the performance of the cloud server, and feature extraction is still carried out on the client, resulting in a high processing cost for the client. These are some of the drawbacks of this study.

Future scholars must concentrate on these two areas: 1) To decrease client-side computation costs, think about extracting deep characteristics from encrypted photos on cloud servers. 2) Studying more realistic multi-cloud server architectures to boost the encrypted domain picture retrieval system's dependability, availability, and robustness.

Acknowledgement: The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

Funding Statement: This work is supported by the National Natural Science Foundation of China (No. 61862041).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] G. Usha, S. Kannimuthu, P. D. Mahendiran, A. D. Shanker and D. Venugopal, "Static analysis method for detecting cross site scripting vulnerabilities," *International Journal of Information and Computer Security*, vol. 13, no. 1, pp. 32–47, 2020.

[2] S. Gkelios, A. Sophokleous, S. Plakias, Y. Boutalis and S. Chatzichristofis, "Deep convolutional features for image retrieval," *Expert Systems with Applications*, vol. 177, pp. 1–47, 2021.

[3] L. Zheng, Y. Yang and Q. Tian, "SIFT meets CNN: A decade survey of instance retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 5, pp. 1224–1244, 2017.

[4] G. Usha and S. Kannimuthu, "A secure cross-layer AODV routing method to detect and isolate (SCLARDI) black hole attacks for MANET," *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 25, no. 4, pp. 2761–2769, 2017.

[5] B. Li, S. Ding and X. Yang, A privacy-preserving scheme for JPEG image retrieval based on deep learning. In: *Journal of Physics: Conf. Series*, Zhuhai, China: IOP Publishing, vol. 1856, no. 1, pp. 1–6, 2021.

[6] W. Chen, Y. Liu, W. Wang, E. Bakker, T. Georgiou *et al.,* "Deep learning for instance retrieval: A survey," arXiv preprint arXiv, 2101.11282, 2021.

[7] X. Li, T. Uricchio, L. Ballan, M. Bertini, C. Snoek *et al.,* "Socializing the semantic gap: A comparative survey on image tag assignment, refinement, and retrieval," *ACM Computing Surveys (CSUR)*, vol. 49, no. 1, pp. 1–39, 2016.

[8] S. Bharati, P. Podder, M. Mondal and V. Prasath, "CO-ResNet: Optimized ResNet model for COVID-19 diagnosis from X-ray images," *International Journal of Hybrid Intelligent Systems*, vol. 17, no. 1–2, pp. 71–85, 2021.

[9] O. I. Obaid, M. A. Mohammed, A. O. Salman, S. A. Mostafa and A. A. Elngar, "Comparing the performance of pre-trained deep learning models in object detection and recognition," *Journal of Information Technology Management*, vol. 14, no. 4, pp. 40–56, 2022.

[10] A. S. Al-Waisy, D. Ibrahim, D. A. Zebari, S. Hammadi, H. Mohammed *et al.,* "Identifying defective solar cells in electroluminescence images using deep feature representations," *PeerJ Computer Science*, vol. 8, no. 5, pp. 1–18, 2022.

[11] M. A. Mohammed, M. J. Abdulhasan, N. M. Kumar, K. H. Abdulkareem and S. A. Mostafa, "Automated waste-sorting and recycling classification using artificial neural network and features fusion: A digital-enabled circular economy vision for smart cities," *Multimedia Tools and Applications*, vol. 28, pp. 1–16, 2022.

[12] H. Cui, L. Zhu, J. Li, Y. Yang and L. Nie, "Scalable deep hashing for large-scale social image retrieval," *IEEE Transactions on Image Processing*, vol. 29, pp. 1271–1284, 2019.

[13] Y. Gong, S. Lazebnik, A. Gordo and F. Perronnin, "Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 12, pp. 2916–2929, 2012.

[14] B. Kulis and T. Darrell, "Learning to hash with binary reconstructive embeddings," *Advances in Neural Information Processing Systems*, vol. 22, pp. 1–9, 2009.

[15] W. Liu, J. Wang, R. Ji, Y. Jiang and S. Chang, "Supervised hashing with kernels," in *2012 IEEE Conf. on Computer Vision and Pattern Recognition*, Providence, Rhode Island, USA, pp. 2074–2081, 2012.

[16] F. Shen, C. Shen, W. Liu and H. Shen, "Supervised discrete hashing," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, Boston, Massachusetts, USA, pp. 37–45, 2015.

[17] R. Xia, Y. Pan, H. Lai and S. Yan, "Supervised hashing for image retrieval via image representation learning," *Twenty-eighth AAAI Conf. on Artificial Intelligence*, vol. 28, no. 1, pp. 2162–2516, 2014.

[18] H. Lai, Y. Pan, Y. Liu and S. Yan, "Simultaneous feature learning and hash coding with deep neural networks," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, Providence, Rhode Island, USA, pp. 3270–3278, 2015.

[19] H. Zhu, M. Long, J. Wang and Y. Cao, "Deep hashing network for efficient similarity retrieval," *Proc. of the AAAI Conf. on Artificial Intelligence*, vol. 30, no. 1, pp. 2415–2421, 2016.

[20] Z. Cao, M. Long, J. Wang and P. Yu, "Hashnet: Deep learning to hash by continuation," in *Proc. of the IEEE Int. Conf. on Computer Vision*, Venice, Italy, pp. 5608–5617, 2017.

[21] Y. Cao, M. Long, B. Liu and J. Wang, "Deep cauchy hashing for hamming space retrieval," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, New York, NY, USA, pp. 1229–1237, 2018.

[22] L. Yuan, T. Wang, X. Zhang, F. Tay, Z. Jie *et al.,* "Central similarity quantization for efficient image and video retrieval," in *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, Seattle, WA, USA, pp. 3083–3092, 2020.

[23] S. Li, L. Wu, W. Meng, Z. Xu, C. Qin *et al.,* "DVPPIR: Privacy-preserving image retrieval based on DCNN and VHE," *Neural Computing and Applications*, vol. 34, no. 17, pp. 14355–14371, 2022.

[24] S. Cheng, L. Wang, G. Huang and A. Du, "A privacy-preserving image retrieval scheme based secure kNN, DNA coding and deep hashing," *Multimedia Tools and Applications*, vol. 80, no. 15, pp. 22733–22755, 2021.

[25] T. Abbas, S. F. Ali, M. A. Mohammed, A. Z. Khan, M. J. Awan *et al.,* "Deep learning approach based on residual neural network and SVM classifier for driver's distraction detection," *Applied Sciences*, vol. 12, no. 13, pp. 6626, 2022.

[26] W. Ma, T. Zhou, J. Qin, X. Xiang and Y. Tan, "A privacy-preserving content-based image retrieval method based on deep learning in cloud computing," *Expert Systems with Applications*, vol. 203, pp. 1–12, 2022.

[27] B. Ferreira, J. Rodrigues, J. Leitao and H. Domingos, "Practical privacy-preserving content-based retrieval in cloud image repositories," *IEEE Transactions on Cloud Computing*, vol. 7, no. 3, pp. 784–798, 2017.

[28] Z. Xia, L. Jiang, X. Ma, W. Yang, P. Ji *et al.,* "A privacy-preserving outsourcing scheme for image local binary pattern in secure industrial internet of things," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 1, pp. 629–638, 2019.

[29] Z. Xia, L. Lu, T. Qin, H. Shim, X. Chen *et al.,* "A privacy-preserving image retrieval based on AC-coefficients and color histograms in cloud environment," *CMC-Computers Materials & Continua*, vol. 58, no. 1, pp. 27–43, 2019.

[30] Z. Xia, L. Wang, J. Tang, N. Xiong and J. Weng, "A privacy-preserving image retrieval scheme using secure local binary pattern in cloud computing," *IEEE Transactions on Network Science and Engineering*, vol. 8, no. 1, pp. 318–330, 2020.

[31] M. J. Awan, O. A. Masood, M. A. Mohammed, A. Yasin, A. M. Zain *et al.,* "Image-based malware classification using VGG19 network and spatial convolutional attention," *Electronics*, vol. 10, no. 19, pp. 2444, 2021.

[32] J. Qin, J. Chen, X. Xiang, Y. Tan, W. Ma *et al.,* "A privacy-preserving image retrieval method based on deep learning and adaptive weighted fusion," *Journal of Real-Time Image Processing*, vol. 17, no. 1, pp. 161–173, 2020.

[33] W. Ma, J. Qin, X. Xiang, Y. Tan and Z. He, "Searchable encrypted image retrieval based on multi-feature adaptive late-fusion," *Mathematics*, vol. 8, no. 6, pp. 1019, 2020.

[34] J. Tang, Z. Xia, L. Wang, C. Yuan and X. Zhao, "OPPR: An outsourcing privacy-preserving JPEG image retrieval scheme with local histograms in cloud environment," *Journal on Big Data*, vol. 3, no. 1, pp. 21, 2021.

[35] Y. Xu, J. Gong, L. Xiong, Z. Xu and J. Wang, "A privacy-preserving content-based image retrieval method in cloud environment," *Journal of Visual Communication and Image Representation*, vol. 43, no. 2, pp. 164–172, 2017.

[36] D. Liu, J. Shen, Z. Xia and X. Sun, "A content-based image retrieval scheme using an encrypted difference histogram in cloud computing," *Information-an International Interdisciplinary Journal*, vol. 8, no. 3, pp. 96, 2017.

[37] Y. Wu, L. Zhang, S. Berrettiv and S. Wan, "Medical image encryption by content-aware dna computing for secure healthcare," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 2, pp. 2089–2098, 2022.

[38] J. Anju and R. Shreelekshmi, "A faster secure content-based image retrieval using clustering for cloud," *Expert Systems with Applications*, vol. 189, pp. 1–11, 2022.

[39] H. Cheng, H. Wang, X. Liu, Y. Fang, M. Wang *et al.,* "Person re-identification over encrypted outsourced surveillance videos," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 3, pp. 1456–1473, 2019.

[40] Y. Li, J. Ma, Y. Miao, Y. Wang, X. Liu *et al.,* "Similarity search for encrypted images in secure cloud computing," *IEEE Transactions on Cloud Computing*, vol. 10, no. 2, pp. 1142–1155, 2020.

[41] L. Weng, L. Amsaleg and T. Furon, "Privacy-preserving outsourced media search," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 10, pp. 2738–2751, 2016.

[42] X. Wang, J. Ma and Y. Miao, "Multi-user image outsourcing retrieval scheme with efficient privacy protection," *Journal of Communications (Chinese)*, vol. 40, no. 2, pp. 31–39, 2019.

[43] A. Hassan, F. Liu, F. Wang and Y. Wang, "Secure content based image retrieval for mobile users with deep neural networks in the cloud," *Journal of Systems Architecture*, vol. 116, pp. 1–17, 2021.

[44] B. Baliga, R. Medepalli and S. Muralikrishna, "Securing textual and image data on cloud using searchable encryption," *International Journal of Information Technology*, vol. 13, no. 3, pp. 1111–1117, 2021.

[45] A. Du, L. Wang, S. Cheng and N. Ao, "A privacy-protected image retrieval scheme for fast and secure image search," *Symmetry*, vol. 12, no. 2, pp. 282, 2020.

[46] C. Zhang, L. Zhu, S. Zhang and W. Yu, "TDHPPIR: An efficient deep hashing based privacy-preserving image retrieval method," *Neurocomputing*, vol. 406, no. 1, pp. 386–398, 2020.

[47] Z. Xia, X. Wang, L. Zhang, Z. Qin and X. Sun, "A privacy-preserving and copy-deterrence content-based image retrieval scheme in cloud computing," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 11, pp. 2594–2608, 2016.

[48] Y. Wu, H. Guo, C. Chakraborty, M. Khosravi, S. Berretti *et al.,* "Edge computing driven low-light image dynamic enhancement for object detection," *IEEE Transactions on Network Science and Engineering*, vol. 2022, pp. 1, 2022.

[49] Z. Xia, Y. Zhu, X. Sun, Z. Qin and K. Ren, "Towards privacy-preserving content-based image retrieval in cloud computing," *IEEE Transactions on Cloud Computing*, vol. 6, no. 1, pp. 276–286, 2015.

[50] R. Lin and S. Li, "An image encryption scheme based on Lorenz Hyperchaotic system and RSA algorithm," *Security and Communication Networks*, vol. 2021, pp. 1–18, 2021.

[51] T. Chua, J. Tang, R. Hong, H. Li, Z. Luo *et al.,* "Nus-wide: A real-world web image database from national university of Singapore," in *Proc. of the ACM Int. Conf. on Image and Video Retrieval*, New York, NY, USA, pp. 1–9, 2009.

[52] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh *et al.,* "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[53] T. Lin, M. Maire, S. Belongie, J. Hays, P. Perona *et al.,* "Microsoft coco: Common objects in context," in *European Conf. on Computer Vision*, Zurich, Switzerland, pp. 740–755, 2014.

[54] J. Philbin, O. Chum, M. Isard, J. Sivic and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," in *2007 IEEE Conf. on Computer Vision and Pattern Recognition*, Minneapolis, Minnesota, USA, pp. 1–8, 2007.

[55] H. Lu, M. Zhang, X. Xu, Y. Li and H. Shen, "Deep fuzzy hashing network for efficient image retrieval," *IEEE Transactions on Fuzzy Systems*, vol. 29, no. 1, pp. 166–176, 2020.

[56] G. Gu, J. Liu, Z. Li, W. Huo and Y. Zhao, "Joint learning based deep supervised hashing for large-scale image retrieval," *Neurocomputing*, vol. 385, no. 1, pp. 348–357, 2020.

[57] D. Wu, Q. Dai, B. Li and W. Wang, "Deep uncoupled discrete hashing via similarity matrix decomposition," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 19, no. 1, pp. 1–22, 2022.