J Real-Time Image Proc (2017) 13:557–570
DOI 10.1007/s11554-016-0632-9

SPECIAL ISSUE PAPER

Real-time, large-scale duplicate image detection method based on multi-feature fusion

Ming Chen[1] · Yuhua Li[1] · Zhifeng Zhang[1] · Ching-Hsien Hsu[2,3] · Shangguang Wang[4]

Received: 5 March 2016 / Accepted: 18 August 2016 / Published online: 19 October 2016

Abstract Recently, using old or irrelevant images in microblogs to spread false rumors has become increasingly rampant. Therefore, tracking and verifying the sources of images has become essential. In order to solve this problem, this paper provides a real-time, large-scale duplicate image detection method based on multi-feature fusion. This method firstly uses multi-feature fusion to improve retrieval accuracy. Then, by Hbase optimization, it uses a bloom filter and range query to improve retrieval efficiency. Experimental results show that, compared with existing algorithms, this method has higher precision and recall rates. Meanwhile, real-time responsiveness and scalability of the approach also meet real-world needs.

Keywords Duplicate image detection · Multi-feature fusion · Image retrieval

1 Introduction

On the Internet, copying and forwarding images is very easy, not only providing convenience for users [1], but also bringing numerous security problems [2, 3]. For example, in March 2013, one media outlet reported that the transgenic corn products supplied by the Monsanto Company might cause cancer. To prove its credibility, the report cited partial CCTV screenshots. This led to a large number of reprints and mass panic. However, it was later confirmed that the report was an old news story from September 2012 whose conclusions had been denied by the Supreme Council of the French biotechnology. Recently, using old or irrelevant images on microblogs to spread rumors has become increasingly rampant. Hence, tracking and verifying images' sources has become essential.

In the process of tracking and verifying image sources, the most critical problem is large-scale duplicate image detection. Generally speaking, "duplicate" is a precise term. Namely, data content is exactly same in a duplicate. However, for practical multimedia applications, "duplicate" is more focused on external form rather than data content, so the definition of "duplicate" also includes different copies of an image. These copies come from a set of tolerable transforms. By analyzing microblog images, we find that there are three main tolerable transformation types:

1. Scaling transformations: An image is scaled or stretched. This situation is very common in microblog images.
2. Watermark transformations: The same image has different watermark information added. For example, when users upload images to Sina Weibo or Sohu Weibo, the microblog ISPs automatically add watermark information to images.
3. Storage format transformations: An image is stored in different formats, such as JEPG, PNG, or TIFF. The changes in image content are small.

For more complex transformations such as light transformations, rotation transformations, and other optical or

✉ Ching-Hsien Hsu
chh@chu.edu.tw

[1] Software Engineering College, Zhengzheng University of Light Industry, Zhengzhou, China

[2] School of Mathematics and Big Data, Foshan University, Foshan, China

[3] Department of Computer Science and Information Engineering, Chung Hua University, Hsinchu, Taiwan

[4] State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, China

geometric transformations, the occurrence probability in microblog images is very small, and hence, they are not considered in this article.

The existing literature has provided a number of duplicate image detection methods. Although these methods are able to reasonably deal with their own design scenarios, for very large image collections, those methods suffer from either intensive computation complexity or degraded performance [4]. So they have difficulty satisfying real-time responsiveness and accuracy in large-scale image retrieval. For example, Changick [5] used an ordinal measure of discrete cosine transformation (DCT) coefficients to detect duplicate images. The method inherited good resistance to noise attack and horizontal or vertical flip attacks. Li et al. [6] again proposed a robust image copy detection algorithm based on an ordinal measure of the full DCT coefficients. Because the middle- and low-frequency coefficients of the full DCT transformation were changed regularly by a scaling transformation, their proposed method could resist scaling attack. However, when describing images, using a single feature tends to miss some detailed information. Hence, these methods' universality is not strong. In this case, in order to detect more duplicate images, we have to relax the retrieval threshold, which reduces retrieval accuracy. For large image collections, smaller errors lead to larger error return. This will reduce user experience. Meanwhile, existing algorithms are mainly based on the single-machine environment, and it is difficult to directly extend them to distributed environments. Therefore, for large-scale microblog images, this study proposes a real-time duplicate image detection method based on multi-feature fusion. This method has the following advantages:

1. By multi-feature fusion, this method uses spatial structure features, color features, and texture features from multiple angles to filter non-duplicate images. This can effectively improve retrieval accuracy.
2. Through its Hbase design, this method uses a bloom filter and range query to ensure real-time performance of hadoop clusters. A bloom filter is used to eliminate partially extended signatures, which can reduce disk IO. Range query is used to replace prefix query, which can effectively reduce the amount of data loading with Hbase as well as the number of comparisons of the next filtering stage.

2 Related work

At present, according to different features, content-based copy detection (CBCD) can be divided into three research directions:

2.1 Global feature-based CBCD

Global features extract global statistics information to represent images. This can be divided into color features, shape features, texture features, and spatial structure features. The advantages of global features are their simple calculation and less space required. However, due to focusing on the overall information of images, global features tend to ignore local information in images.

The first near-duplicate image search engine was RIME [7]. RIME used Daubechies' wavelet transformation to extract image features and a multi-dimensional extensible hashing scheme to create a high-dimensional index. This method could effectively detect slightly modified images, but it faced difficulty in identifying severely distorted images. Subsequently, Wu et al. [8] proposed an ellipse track-division-based image copy detection method. However, due to the ellipse shape without the rotation invariance property, this method did not really solve the rotation distortion problem. On this basis, Zhou et al. [9] proposed a cirque division strategy to detect duplicate images. It was well known that the content of a cirque track region was almost invariant under rotation and scale attacks; therefore, the proposed method could effectively resist the above two attacks. However, it could not solve the circle shift issue caused by non-proportional scaling transformations. In addition, Hesiao et al. [10] used an extended feature set to detect image copies. The method estimated in advance all possible transformations by prior knowledge and then searched the transformed images one by one. However, in real life, it was not only difficult to estimate all transformation types, but the extended feature sets incurred additional computational costs, as well. This would affect the real-time performance of the system. In recent years, people gradually realize a single feature is difficult to ensure retrieval accuracy. So Feng et al. [11] proposed a novel image descriptor, called global correlation descriptor (GCD), to extract color and texture feature, respectively, so that these features have the same effect in CBCD. For addressing the problem of large-scale image search, Jegou et al. [12] propose aggregating local image descriptors into a global vector. The experiment shows that the image representation can be reduced to a few dozen bytes while preserving high accuracy.

2.2 Local feature-based CBCD

Local feature-based CBCD first detects the stable regions of an image and then extracts high-dimensional feature vectors in the vicinity of each region. Hence, the image is represented as a set of feature vectors. With respect to global features, local features are more suitable to working with local changes in image content. However, local features have difficulty distinguishing similar images and duplicate images. Meanwhile, the matching algorithms of local features are more complex and face difficulty in meeting the real-time requirement in large-scale image retrieval.

The landmark work on local features was the scale-invariant feature transform (SIFT) feature-based method [13]. The SIFT had good local invariance, and it received widespread attention in academia. On its basis, many scholars proposed improved algorithms for the SIFT feature-based approach. In order to improve discrimination and robustness, Mikolajczyk and Schmid [14] proposed the gradient location orientation histogram (GLOH) feature. To improve illumination invariance, many scholars began to study color SIFT, which let SIFT features from the grayscale space extend to the color space [15]. Although the above methods could solve their respective problems, they need a large amount of time overhead. Therefore, to reduce computational complexity, Ke and Sukthankar [16] proposed PCA-SIFT features. However, experiments showed that the discrimination of PCA-SIFT decreases with decreasing feature dimensions. Bay et al. [17] also proposed an accelerated algorithm for SIFT features: the speed-up robust feature (SURF). Experiments showed the extraction rate of the SUFT feature was three times faster than that of the SIFT feature [18]. Although the above algorithms had good scale invariance and rotation invariance, they had difficulty resisting affine transformations. Given this situation, according to “watershed” concept, Matas et al. [19] proposed the maximally stable extremal regions (MSER) detection algorithm. In addition, the fast and compact binary robust independent elementary feature (BRIEF) was gaining more and more attention [20]. The advantage of BRIEF was its computation speed, but BRIEF lacked the rotation invariance and scale invariance properties. Therefore, the oriented brief (ORB) [21] was proposed by Rublee et al. Although ORB also lacked the scale invariance property, in practice the problem could be solved by heuristic strategies. Experimental results showed the calculation speed of ORB to be 100 times faster than that of SIFT and 10 times faster than SURF. Therefore, ORB was suitable for processing real-time video data. Combined the characteristics of the human retina and Daisy descriptor [22], Alahi et al. [23] propose Fast Retina Keypoint (FREAK), which has a density distribution of the human retina. Yang and Cheng [24] propose a highly efficient and distinctive binary descriptor, called local difference binary (LDB). LDB directly computes a binary string for an image patch using simple intensity and gradient difference tests on pairwise grid cells within the patch. A multiple-gridding strategy and a salient bit selection method are applied to capture the distinct patterns of the patch at different spatial granularities. Compared with other descriptors, LDB has higher robustness and computational efficiency.

2.3 Perception hash-based CBCD

Computational complexity is the key problem in large-scale image retrieval. A perception hash can solve the problem by compressing image features into binary signatures. Binary signatures generated by a perception hash have two advantages: They can preserve the similarity of the original feature space, and they can greatly improve computational efficiency. However, because of containing less image information, binary signatures have difficulty in ensuring retrieval precision.

At present, perception hashes can be divided into three categories: unsupervised hashes, supervised hashes, and semi-supervised hashes. Unsupervised hashes refer to the generation process of perception hashes that do not need training data. The typical representatives of an unsupervised hash are LSH [25], minhash [26], and simhash [27]. However, a large number of experimental results show that measuring similarity is not able to guarantee semantic similarity. In real life, images often contain a large number of tagged data; in order to take full advantage of these existing semantic data, a supervision hash is proposed. Compared with unsupervised hashes, supervised hashes have higher retrieval precision so that, in recent years, supervised hashes have had greater development. The typical representatives of supervised hashes are similarity-sensitive coding (SSC) [28], restricted Boltzmann machines (RBMs) [29], self-taught hashes (STH) [30], nonnegative spare coding-induced hashes (NSCIH) [31], and histograms of sparse codes (HSC) [32]. Although supervised hashes can preserve the semantic similarity of images, when the amount of tagged data is too small, it will lead to over-fitting phenomena. Therefore, in order to solve this problem, Wang et al. [33] proposed a semi-supervised learning hash (SSH). And Bauml et al. [34] propose a unified learning framework for multi-class classification which incorporates labeled and unlabeled data.

3 Algorithm description

Duplicate image detection implies finding images that have the same visual perception but different codes. At present, due to the existence of semantic gaps, the accuracy of similar matching is unable to reach 100 %. In large-scale image retrieval, smaller errors lead to larger error return. This will significantly affect the user experience. Therefore, in order to improve retrieval accuracy, this section selects a perception hash, a block-average grayscale feature, and a Haar wavelet feature to implement multi-feature fusion. The features can effectively compensate for each other. The fusion process divides into two phases: the rapid filtration stage and the precise filtration stage.

3.1 Rapid filtration stage

A large-scale image retrieval system is put forward for higher real-time and scalable requirements. Therefore, a rapid feature comparison method is needed to meet system requirements.

3.1.1 Random projection as perception hash

In this stage, this study uses a random projection algorithm as a perception hash to implement real-time responsiveness and scalability. A random projection algorithm is used to generate binary signatures for images, and comparison of signatures can be used to judge the similarity between two images.

A random projection algorithm can preserve the similarity of data. Namely, similar data in high-dimensional space have smaller Hamming distance in Hamming space. The dissimilar data in high-dimensional space have larger Hamming distance in Hamming space. The advantage of this property is that the computation of binary signatures is simple and fast, and signatures are easy to compare and expand. However, the drawback is lack of accuracy.

The specific algorithm process as follows:

1. Image feature extraction
   Firstly, the input image is converted into grayscale K, whose size is scaled to $M \times M$. Then, K is evenly divided into n blocks. The average grayscale value v_i of each block K_i is computed, and a block-average grayscale feature $V_K = (v_1, v_2, \ldots, v_n)$ is generated. If the gray value of the image pixel is represented by $g(x, y)$, then formula (1) is established.
   $$v_i = \frac{n}{M^2} \sum_{x,y \in K_i} g(x,y). \tag{1}$$
2. Image signature generation

Definition 1 Random projection hash $h(V)$: a nonzero vector $X = (x_1, x_2, \ldots, x_n)$ is randomly selected from n-dimensional space, where each dimensional component x_i is drawn from the standard normal distribution $N(0, 1)$. Considering the angle between the feature vector V and the vector X, if it is an acute angle, then $h_X(V) = 1$; otherwise, $h_X(V) = 0$. If the inner product of vectors is presented by °, then formula (2) is established as follows:

$$h_X(V) = \begin{cases} 1, & V \circ X \geq 0 \\ 0, & V \circ X \prec 0 \end{cases} \tag{2}$$

Definition 2 Random projection signature $sig(V)$: the vectors $X_1, X_2, \ldots, X_f$ are randomly generated, where $f \leq n$. Then, the inner products between V and X_i are calculated. If $V \circ X_i \geq 0$, then the ith hash bit is $h_{X_i}(V) = 1$; otherwise, $h_{X_i}(V) = 0$, namely

$$\mathrm{sig}(V) = h_{X_f}(V) h_{X_{f-1}}(V) \cdots h_{X_1}(V) \tag{3}$$

Theorem 1 *In a random projection signature, the appearance probability of 0 or 1 is equal and independent* [35].

Proof Suppose that V is a vector in n-dimensional real space, then we have the following conclusions.

Independence If $S_i = V \circ X_i$, $X_i = (x_{i,1}, x_{i,2}, \ldots, x_{i,n})$, and $\forall x_{i,j} \sim N(0, 1)$, then $S_i = \sum_{j=1}^{n} v_j \times x_{i,j}$ and $v_j \times$ -$x_{i,j} \sim N(0, v_j^2)$. According to the additivity of normal distributions, we can infer that random variable $S_i \sim N(0, \sum_{j=1}^{n} v_j^2) = N(0, \delta^2)$. This indicates that each component generated by the random projection algorithm belongs to a normal distribution whose expectation is 0 and whose variance is $\sum_{j=1}^{n} v_j^2$.

For any two variables S_p and S_q, the covariance is cos $(S_p, S_q) = E(S_p \times S_q) - E(S_p) \times E(S_q)$. Here S_p and S_q belong to a normal distribution whose expectation is 0. Therefore, $\mathrm{cov}(S_p, S_q) = E(S_p) \times E(S_q) = E((\sum_{i=1}^{n} v_i x_{p,i}) \times (\sum_{j=1}^{n} v_j x_{q,j}))$. According to the distributive law of multiplication, $\mathrm{cov}(S_p, S_q) = E(\sum_{i=1}^{n} \sum_{j=1}^{n} v_i v_j x_{p,i} x_{q,j}) = \sum_{i=1}^{n} \sum_{j=1}^{n} v_i v_j E(x_{p,i} x_{q,j})$. Here $x_{p,i}$ and $x_{q,j}$ belong to the standard normal distribution and are independent. Therefore, $\mathrm{cov}(S_p, S_q) = 0$.

Equality By definition 1, we can obtain $h_X(V) = \mathrm{sign}(S) = \mathrm{sign}(V \circ X) = \begin{cases} 1, & V \circ X \geq 0 \\ 0, & V \circ X \prec 0 \end{cases}$. Here sign() represents a sign function. Combined with the conclusion of independence, $S_i \sim N(0, \sum_{j=1}^{n} v_j^2) = N(0, \delta^2)$, we can infer formulas (4) and (5). Namely, the appearance probability of [0, 1] is equal in a random projection signature.

$$\Pr(h_X(V) = 1) = \Pr(\mathrm{sign}(S) = 1) = \Pr(S \geq 0) = \frac{1}{2} \tag{4}$$

$$\Pr(h_X(V) = 0) = \Pr(\mathrm{sign}(S) = 0) = \Pr(S \prec 0) = \frac{1}{2} \tag{5}$$

Theorem 2 *In a random projection signature, the exactly equal probability of f bits of signatures is* $\Pr(s) = (1 - \frac{\arccos(s)}{\pi})^f$, *where s represents the similarity of two vectors.*

Proof Suppose that two n-dimensional vectors are U and V. As shown in Fig. 1, if the angle between U and V is θ, then only when the random vector X_i falls in the intersection angle between the normal vectors U and V, there is $h_{X_i}(U) \neq h_{x_i}(V)$. At this time, the unequal probability of the corresponding signature bit is $\frac{\theta}{\pi}$. Combined with Theorem 1, the exactly equal probability of an f-bit signature is $\Pr(s) = (1 - \frac{\theta}{\pi})^f$. According to the cosine similarity principle $s = \frac{U \circ V}{|U||V|} = \cos\theta$, we can infer $\Pr(s) = (1 - \frac{\arccos(s)}{\pi})^f$.

As shown in Theorem 2, $\Pr(s)$is a monotonically increasing function of s. The character of $\Pr(s)$ can keep the similarity of data. Namely, similar data in high-dimensional space have smaller Hamming distance in Hamming space. Therefore, if two vectors have higher similarity, then the probability of having equal hash values is higher.

3. Image signature comparison
   Image signatures generated by random projection may be affected by noise. Therefore, in order to improve retrieval recall, this paper not only considers the situation where image signatures are exactly equal, but also considers the situation where image signatures are similar. Namely, when the Hamming distance of two image signatures is not greater than the threshold H, we think the two images may be duplicates.

$$D_{\text{ham}}(\text{sig}(U), \text{sig}(V)) = \sum_{i=1}^{f} (h_{X_i}(U) \oplus h_{X_i}(V)) \le H \tag{6}$$

3.2 Precise filtration stage

From the point of view of containing information, the original image contains the richest content, but the information is hidden in the overall structure of the image and cannot be directly understood by computers. Therefore, features must be extracted to describe images. However, in each process of feature extraction, there is a certain degree of information loss, which makes it difficult for retrieval accuracy to meet user needs. Here the information loss of a perception hash is particularly obvious. Therefore, a more sophisticated exact match is indispensable. In this stage, this study uses a block-average grayscale feature and a Haar wavelet feature to conduct multiple filtering. Multiple filtering from the angle of color and texture can effectively improve retrieval accuracy.

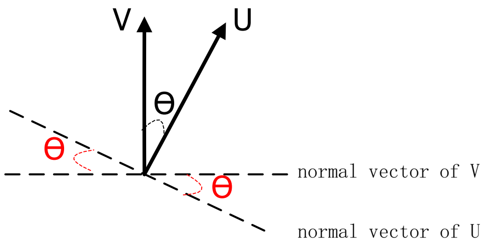


Fig. 1 A random projection

3.2.1 *Block-average grayscale feature*

Due to hash collisions, non-duplicate images may fall into the same bucket using a perception hash; namely, they may have the same signature. Therefore, perception hashes can only play the function of fast indexing and preliminary filtering. We need to further filter non-duplicate images. The analysis reveals, using a block-average grayscale feature, that the same dimension values of duplicate images are very close. On this basis, we use the Manhattan distance to calculate the similarity of two images.

$$D_{\text{man}}(V_X, V_Y) = \sum_{i=1}^{N} |V_{X_i} - V_{Y_i}| \tag{7}$$

Here V_X and V_Y represent the block-average grayscale features of images X and Y. V_{X_i} represents the ith component of a block-average grayscale feature V_X. If D_{man}(-V_X, V_Y) is not greater than threshold T, then we think images X and Y may be duplicates.

3.2.2 *Haar wavelet feature*

The block-average grayscale feature uses the intermediate results of the perception hash, so the processing speed is very fast. However, the processing units of the block-average grayscale feature are blocks, which ignore a great deal of detailed information. Therefore, this paper uses a Haar wavelet feature to filter further. An important character of the Haar wavelet is that it not only can reflect the global information of images, but also reflects the local information of images.

The specific algorithm is as follows:

1. The input image is scaled to 64 × 64 pixels and converted to grayscale.
2. Haar wavelet decomposition is used on the grayscale image. Sixty elements with the largest absolute values from the image matrices are extracted.
3. Sixty elements are replaced by their one-dimensional subscripts ($V[i, j] = i \times 64 + j$). Meanwhile, if an element is less than 0, the corresponding one-dimensional subscript is multiplied by -1. Then, the one-dimensional subscripts are sorted, and the left 10-element sequence of sorted subscripts is selected to generate a Haar wavelet feature.

4. When the Manhattan distance of feature vectors is not greater than threshold *t*, we think the images are duplicate images.

3.3 Algorithm analysis

3.3.1 Image grayscale information analysis

As shown in Fig. 2, image transformation can lead to image pixel changes. This will affect the matching of similar images. However, the analysis shows that changes in pixels are small for the above transformation. Especially for the block-average grayscale value algorithm, the calculation of the average grayscale value of a region could effectively offset the effects of individual pixel changes and thus reduce the effect of random noise.

In addition, before and after transformation, the change in block-average grayscale values is small. Through random projection, images with similar space–color structures are mapped to the same bucket, and images with dissimilar space–color structure are mapped to different buckets. Therefore, this hash algorithm can effectively filter most non-duplicate images and keep just parts of the candidate images. This can greatly improve algorithm efficiency; however, hash collisions are difficult to avoid, which cause non-duplicate images to be mistakenly identified as duplicate images. Therefore, the block-average grayscale value algorithm must be used to implement further precision filtration.

It can be seen that images can be rapidly classified by a perception hash. This can effectively reduce the burden of the next phrase and improve algorithm efficiency. Meanwhile, based on the previous stage, the block-average grayscale value algorithm can improve retrieval accuracy through further comparison, and the block-average grayscale feature uses the intermediate results of the perception hash. The processing speed for this approach is very fast. Therefore, the combination of two algorithms enjoys the effect of complementary advantages.

3.3.2 Image edge information analysis

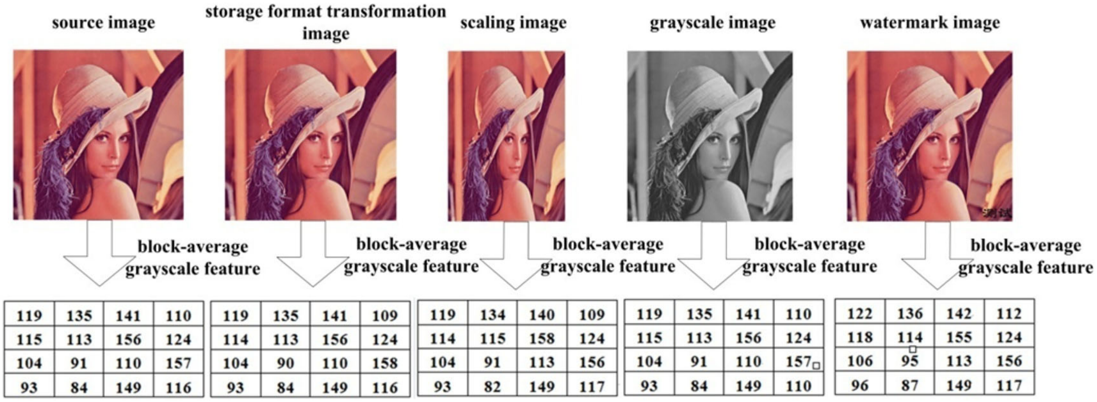
As shown in Fig. 3, for the above transformations, the image edge information shows almost no change. Therefore, the images can be effectively described by extracting their edge detail information. Haar wavelet transform is currently considered to be an effective edge information extraction algorithm.

Haar wavelet features and block-average grayscale features have useful complementary advantages. This is because, on one hand, Haar wavelet transform can extract image edge information by multi-scale analysis, which has nothing to do with the original resolution of images. This is a more detailed characterization of image texture. Block-average grayscale features lack description of detailed information. On the other hand, Haar wavelet transform is based on points to capture image features. However, in fact, the smooth boundaries of natural objects result in the main compositions of images not being points, but rather lines and surfaces. Therefore, Haar wavelet transform shows significant limitations in dealing with images. However, the block-average grayscale algorithm is exactly the opposite. It deals with surfaces; hence, block-average grayscale features can effectively make up for Haar wavelet features.

3.4 Effect analysis

In order to provide the reference to the practical application, this section analyses the effect of multi-feature fusion.

Figure 4 shows the process of multi-feature fusion. Here images A, B, C, and D are test data, and B is forged by using the block-average grayscale feature of image *A*. In perceptual hash filtration stage, although the four images are different, images A, B and images C, D have the same color structure, respectively. After filtration, images A, B, C, and D are divided into two groups and each group is encircled by the red dash. Because signature comparison uses a hash map, the time complexity of this stage is O (1).


source image
storage format transformation image
scaling image
grayscale image
watermark image
测试
block-average grayscale feature
block-average grayscale feature
block-average grayscale feature
block-average grayscale feature
block-average grayscale feature

119	135	141	110
115	113	156	124
104	91	110	157
93	84	149	116

119	135	141	109
114	113	156	124
104	90	110	158
93	84	149	116

119	134	140	109
114	115	158	124
104	91	113	156
93	82	149	117

119	135	141	110
115	113	156	124
104	91	110	157□
93	84	149	110

122	136	142	112
118	114	155	124
106	□ 95	113	156
96	87	149	117

Fig. 2 Analysis of grayscale information of sample image

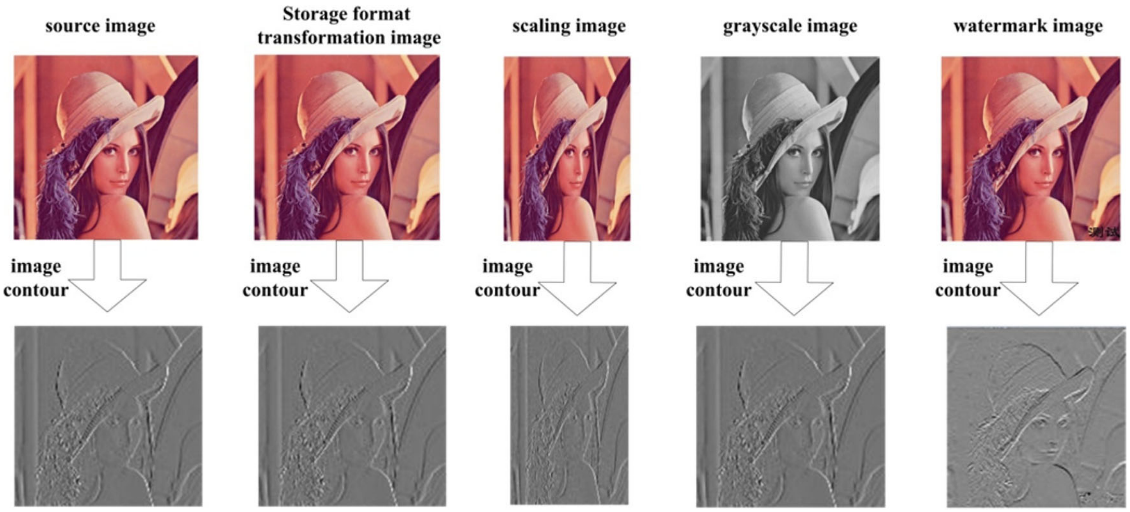


Fig. 3 Image edge information analysis

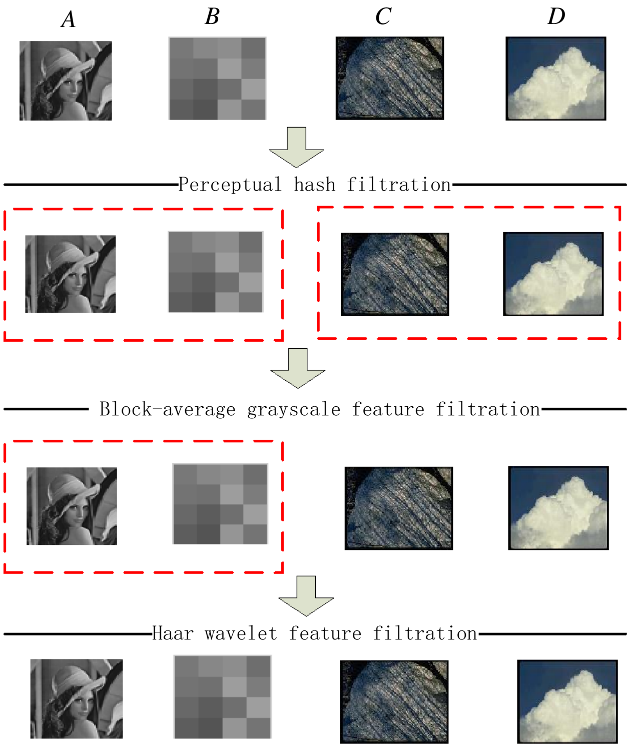

A
B
C
D
Perceptual hash filtration
Block-average grayscale feature filtration
Haar wavelet feature filtration

Fig. 4 Multi-feature fusion

In block-average grayscale feature filtration stage, although images C and D have the same color structure, the color of the corresponding block has larger difference. So they are easy to be distinguished by the block-average grayscale feature. Meanwhile, images A and B have the same block-average grayscale feature. So in this stage, they cannot be distinguished. In term of time, this stage exploits the intermediate results of the previous stage. So the speed is relatively fast. In the worst case, the time complexity of this stage is O (num); num is the number of images. In Haar wavelet feature filtration stage, images A and B have different texture. So they can be easily distinguished by Haar wavelet feature. In the worst case, the time complexity of this stage is O (num). From the overall perspective, due to the good cohesiveness and complementarities between multi-features, this algorithm can not only maintain a high recall rate, but also meet the accuracy requirements of duplicate image detection. Additionally, as shown in Table 1, with increasing number of images, the whole time complexity increases linearly.

Table 1 Time complexity

Multiple filtration	Time complexity
Perception hash filtration	O (1)
Block-average grayscale feature filtration	O (num)
Haar wavelet feature filtration	O (num)
The whole time complexity	O (num)

4 System architecture

In order to ensure the real-time performance of large-scale image retrieval, this paper designs a distributed system architecture based on hadoop. In the design, hadoop clusters provide a distributed parallel processing environment. Hadoop distributed file system (HDFS) is the foundation of a whole structure in support of the superstructure. In the superstructure, Hbase is mainly used to manage image information and image features. The system architecture is shown in Fig. 5. In offline part, images are stored in the underlying hadoop clusters. Image features are extracted by MapReduce and stored in Hbase as given in Table 2. In online part, when a query image comes, the client firstly extracts the image signature and image norm of the query

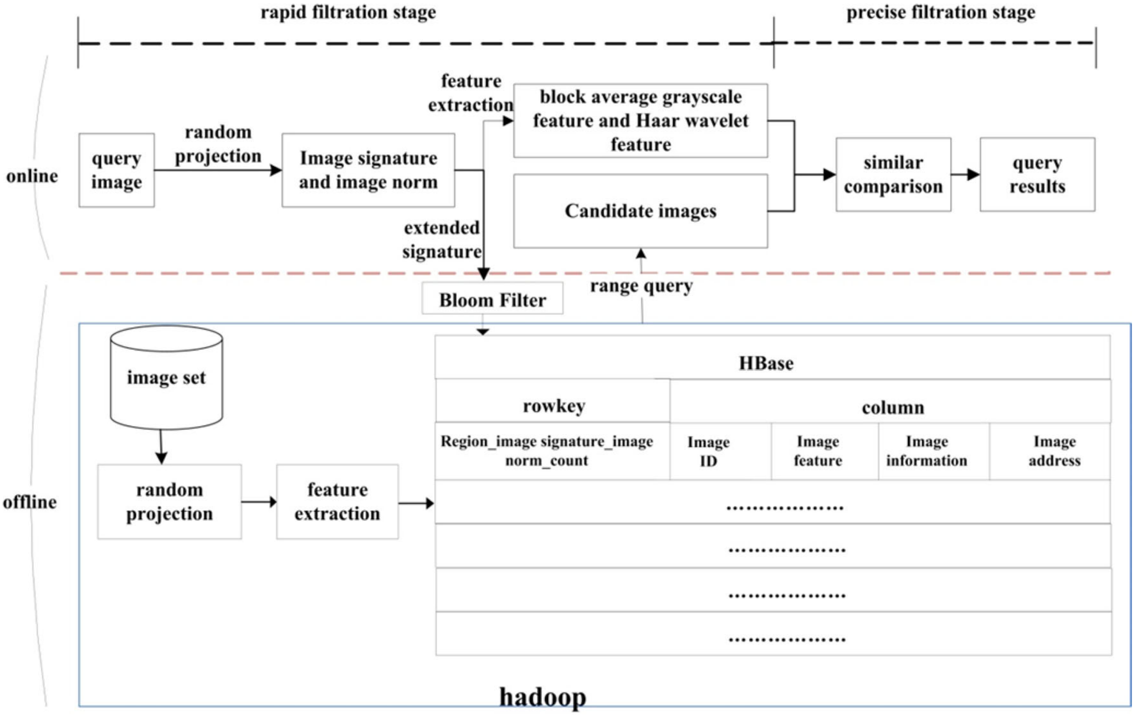


Fig. 5 System architecture

Table 2 Hbase design

Field	Value
Primary key	Region_image signature_image norm_count (stored as a hex string)
C1: image ID	Image name
C2: image feature	Image feature information (stored as a string)
C3: image information	Image source and other related information
C4: image storage address	Image storage position (stored as a string)

image in rapid filtration stage. Then, extended signatures of the query image are generated by changes in image signatures within H bits. The comparison between the extended signatures and image signatures in Hbase can be used to judge the similarity between two images. Here, image signatures can play the function of fast indexing and preliminary filtering. And in order to improve the retrieval efficiency of Hbase, this paper uses a bloom filter to exclude partly extended signatures. This process can reduce the positioning time of a perception hash. Meanwhile, according to formula (11), this paper uses a range query to replace the individual scanning of prefix queries, which can effectively reduce the amount of data loading and the number of comparisons in the next filtering stage. In precise filtration stage, due to hash collisions, we need block-average grayscale feature and Haar wavelet feature to further filter non-duplicate images.

4.1 Hbase design

In the rapid filtration stage, to improve retrieval recall, if the Hamming distance between test images and the query images is not greater than the threshold H, then the test images are seen as candidate images for the next filter stage. For a large number of candidate images, there is a key issue of how to return query results within a tolerable amount of time. A simple global scan is time-consuming and thus cannot meet practical needs. Therefore, we need to design and optimize Hbase according to the application. A specific Hbase structure is shown in Table 2. Here, in primary key, region presents the partition's number of image signatures. Image signature uses hexadecimal representation to improve retrieval speed. Image norm is generated by calculating the 1-norm of a block-average grayscale feature. Count is the number of images, which is used to avoid duplicating prefix fields.

According to Hbase design, the changes in image signatures within H bits are seen as extended signatures. We can then search the signatures using the prefix query. Because just partial data are being dealt with rather than global scanning, this method can effectively improve query efficiency. For example, when $H = 2$ and $f = 32$, we need $C_{32}^{0} + C_{32}^{1} + C_{32}^{2} = 529$ prefix queries. Here, it can be seen that, with increasing H and f, the number of lookups increases dramatically.

4.2 Optimization design

The Hbase prefix query can effectively improve query efficiency. However, in practical applications, image signatures are not uniformly distributed; for the same image signature, there may be many corresponding images or none. So, with increasing H and f, it is difficult for using the simple prefix search to guarantee the real-time performance of the system. In order to solve this problem, this study implements two aspects of optimization: On the one hand, a bloom filter is exploited to eliminate partly extended signatures in order to reduce the number of signatures searched. On the other hand, for two images having the same signature, we use a range query to replace the prefix query. This can decrease the amount of data loading for Hbase in the next filtering stage.

4.2.1 Bloom filter

The bloom filter is a data structure used to determine whether an element is within a collection. This technology was first applied to disk access and later became widely used in database queries, intrusion detection, and other applications.

In this section, before each range query, we first compare image signatures with the storage bit of the bloom filter. If the corresponding bit is 1, it means there may be a corresponding image with the same signature; then, we need to further access Hbase. Otherwise, it means that there are no corresponding images with the same signature, in which case the query is skipped. By using the bloom filter, we can effectively remove part of the extended signatures and decrease disk IO time.

4.2.2 Range query

For any two image features X and Y, when their Manhattan distance is not greater than the threshold T, we believe that they are similar. Formula (8) is established.

$$D_{\mathrm{man}}(X,Y) = \sum_{i=1}^{N} |x_i - y_i| \leq T \tag{8}$$

According to the nature of algebra, X and Y can be regarded as points in space and have formula (9).

$$\left| \sum_{i=1}^{N} |x_i| - \sum_{i=1}^{N} |y_i| \right| \leq \sum_{i=1}^{N} |x_i - x_j| \tag{9}$$

From formulas (8) and (9), we can deduce formula (10). Formula (10) guarantees that if two images are similar, then the corresponding 1-norms are relatively close. Therefore, sequential scans can be transformed into range scans.

$$\sum_{i=1}^{N} |y_i| - T \leq \sum_{i=1}^{N} |x_i| \leq \sum_{i=1}^{N} |y_i| + T \tag{10}$$

Although a perception hash can preserve the similarity of the data, there is a large amount of information loss in the process of generating image signatures, leading to a large number of dissimilar images having the same signature in Hbase. A prefix query based on image signatures will return a large number of irrelevant images along with their information. This would incur a large amount of data loading and increase the number of data comparisons in the precise filtration stage. Therefore, for images with the same signature, it is necessary to do further filtering. Namely, according to formula (10), the prefix query is replaced by a range query for the same signature.

The specific query process is as follows: When the input image A is provided, we first calculate the 1-norm $\|A\|_1$ and all image signatures of the query vector A. We use the bloom filter to eliminate partly extended signatures. Then, according to the remaining signatures, we, respectively, calculate the query range [region_sig(A)_low limit, region_sig(A)_high limit]. Namely, starting with region_sig(A)_low limit and ending with region_sig(A)_high limit, the sequential scan is carried out in Hbase. Finally, the relevant information for query results is returned to proceed to the next filtering.

$$\begin{cases} \text{region\_sig}(A)\text{\_low limit} = \|A\|_1 - T \\ \text{region\_sig}(A)\text{\_high limit} = \|A\|_1 + T \end{cases} \tag{11}$$

5 Experimental results and analysis

In order to verify the proposed method, this study conducts a large number of related experiments, including: (1) parameter estimation. In order to improve the identifiability of image features, we analyze experimental results to select the most suitable parameters. (2) Algorithm comparison. In order to verify the effect of this method, we compare the proposed method with other classic algorithms. (3) Large-scale image retrieval. The purpose of these experiments is to verify the real-time and scalable performance of this method.

Because duplicate images in microblogs are difficult to count, the experiment is divided into two aspects:

1. To ensure the accuracy of retrieval results in a single-machine environment, we randomly selected ten thousand images from the Corel image database as test images to estimate parameters and compare algorithms. Here, Corel images include a large number

of similar images. This can help us to select the proper thresholds to distinguish similar images and duplicate images. An inverted index was used as the index structure.

2. In a distributed environment, in order to more closely approach the real situation, this paper downloads 12 million images as test images from Netease, Sohu, Sina, and other portals. For 12 million images, we construct hadoop clusters to evaluate the real-time and scalability of large-scale image retrieval.

To generate copies, we selected 50 images from the above two test image sets as query images, and each query image was processed by a scaling transformation, watermark transformation, and storage format transformation using Photoshop. Here, the number of the scaling images was 400. The number of watermark images was 100. The number of storage format transformation images was 100.

Here, hadoop clusters included 4 hosts. One was the master, and the others were slaves. The configurations of the four hosts are as follows: Intel(R) Xeon(R) CPUs (E5645) @ 2.40 GHz with 32 GB RAM and 600 GB of hard disk space.

5.1 Parameter estimation

The values of H and T chosen in the experiments are relevant to block-average grayscale features of images saved on hadoop. The value of t chosen in the experiments is relevant to Haar wavelet features of images saved on hadoop. Because the test images are randomly selected from the Internet, the three values chosen are only relevant to the selected features and irrelevant to the scenarios. For different features, the selection of parameter values can be reference to the following method.

5.1.1 *The choice of parameter H*

In order to improve anti-jamming capability for image signatures, this study obtained the optimal threshold H through a receiver's operating characteristic (ROC) curve. The ROC curve represents the relation between recall and precision under different thresholds H. As shown in Fig. 6, when $H = 2$, we have the largest recall rate. Analyzing the undetected duplicate images, the main factors affecting the recall rate were threshold T and t. Continuing to increase H, the impact on the recall rate was not only very limited, but also led to the decline of the precision rate and the increase in retrieval time, so we chose $H = 2$.

5.1.2 *The choice of parameter T*

To obtain the optimal threshold T, a precision–recall (PR) curve was used. Intuitively, the ideal duplicate image detection algorithm should be able to distinguish similar images and resist image transformations. However, in fact, due to the influence of the semantic gap, precision and recall are checks and balances. As shown in Fig. 7, when the threshold T varies between 8 and 10, precision and recall achieve a better balance. We selected $T = 9$.

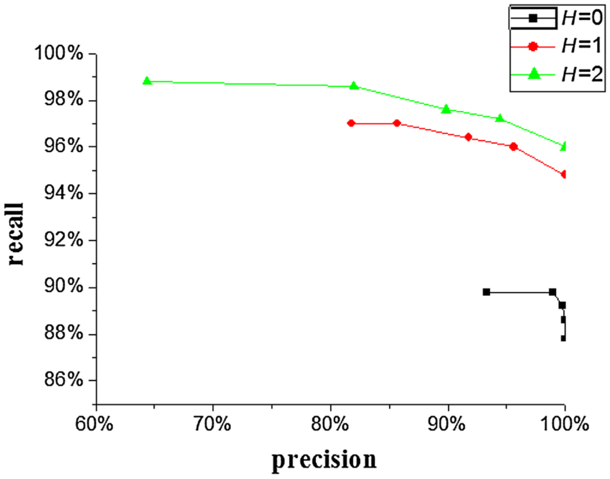


Fig. 6 Threshold *H*

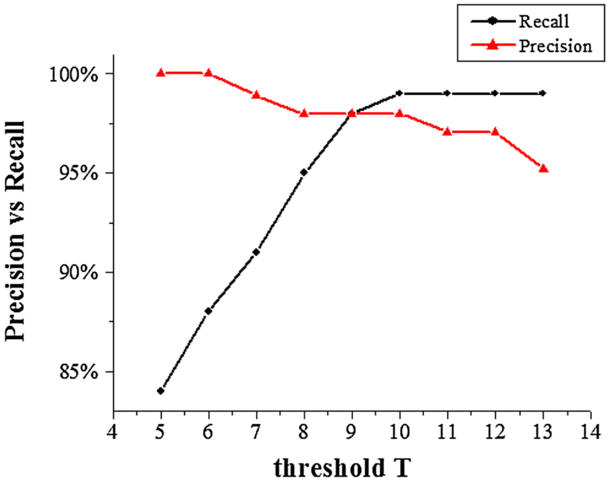


Fig. 7 Threshold *T*

5.1.3 *The choice of parameter t*

$$\begin{cases} \mathrm{fr} = \dfrac{\text{the number of undetected copies}}{\text{the number of total copies}} \\ \mathrm{cr} = \dfrac{\text{the number of detected non-copies}}{\text{the number of non-copies}} \end{cases} \tag{12}$$

When threshold t was too large, it was difficult for it to play a role in distinguishing similar images, which reduced the precision rate. However, when threshold t was

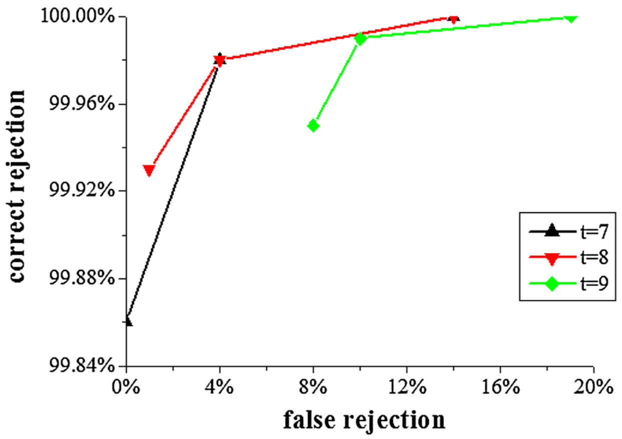


Fig. 8 Threshold *t*

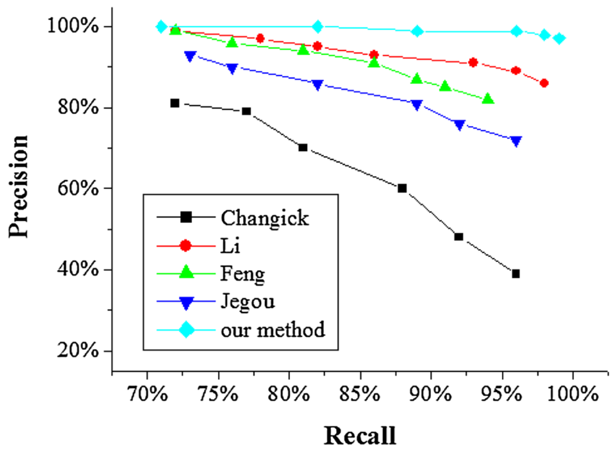


Fig. 9 Algorithm comparison

too small, it led to some duplicate images being perceived as non-duplicate images, which reduced the recall rate. Therefore, we need to choose an appropriate threshold *t* according to experimental results. This study obtained the optimal threshold *t* through the ROC curve, which represents the relation between false rejection (fr) and correct rejection (cr) under different thresholds *t*. As shown in Fig. 8, we selected $t = 8$. The definitions of false rejection and correct rejection are given in formula (12).

5.2 Comparison of algorithms

In order to verify the effectiveness of this method, we compared it to Changick's methods [5], Li's methods [6], Feng's method [11], and Jegou's method [12]. The four methods are classic algorithms for image retrieval and have good robustness to local geometric distortion and scaling transformation.

As shown in Fig. 9, the proposed method has two advantages: (1) with the same precision rate, the recall rate of the proposed method is closest to the ideal value (100 %). (2) When the precision rate increases, the recall rate of the four methods decrease sharply, whereas the recall rate of the proposed method decreases slowly. Hence, this method gives better performance.

5.3 Large-scale image retrieval

5.3.1 Scalability analysis

The average retrieval time under different numbers of images was tested. As shown in Fig. 10, with increasing number of images, the average retrieval time increased linearly. When the number of images was 12 million, the average retrieval time was 0.7 s. With the expansion of hadoop clusters, the average retrieval time can be further reduced. Therefore, this system has good scalability.

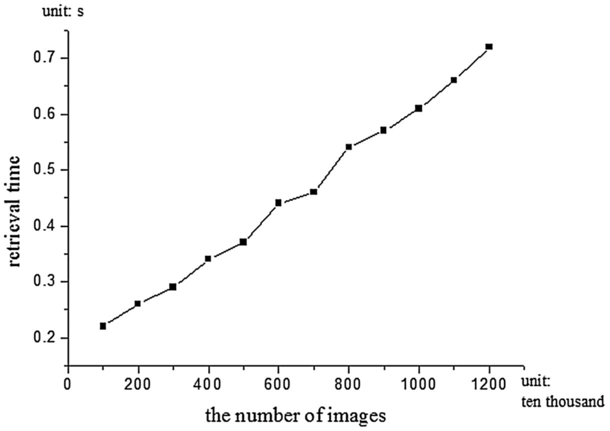


Fig. 10 Comparison of average retrieval time under different numbers of images

5.3.2 Algorithm optimization

The effect of the bloom filter (BF), prefix query (PQ), range query (RQ), and multiple threading (MT) was tested. As shown in Fig. 11, for 12 million images, the average retrieval time of the basic algorithm was 20.3 s. By adding the bloom filter, the average retrieval time shortened to 17.2 s, and the efficiency was improved by 15.3 %. Meanwhile, with the increase in data volume, Hbase distributed these data to each node, and the nodes were divided into different regions. Therefore, the use of multiple threading was shown to improve average retrieval time. However, the most obvious performance improvement was with the range query. The range query greatly reduced the amount of data loading in Hbase and the number of comparisons in the next filter, so the average retrieval time was only 0.72 s.

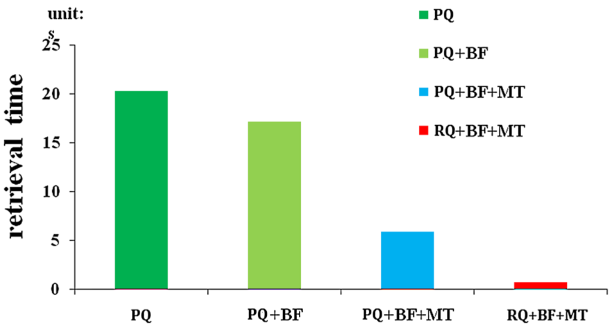


Fig. 11 Algorithm optimization

Table 3 Retrieval results

Recall	Precision
93.8 %	99.1 %

5.3.3 *Retrieval effect*

As shown in Table 3, the experimental results indicate that, when $H = 2, T = 9$, and $t = 8$, the recall rate of the proposed method was 93.8 %; the precision rate was 99.1 %. With further adjustment of threshold T and t, the actual retrieval precision of this method could approach 100 %, while the decrease in the recall rate is limited. This situation demonstrates that the proposed method has a higher accuracy.

6 Conclusion

For tracking and verifying image sources, this paper described a real-time, large-scale duplicate image detection method based on multi-feature fusion. The proposed method used multi-feature fusion to improve retrieval accuracy. Here, multi-feature fusion uses a perception hash to generate image signatures, reducing the search time of images, and uses block-average grayscale features and Haar wavelet features to conduct multiple filtering, thereby improving the retrieval accuracy. Then, using Hbase design, this method utilized the bloom filter and range query to improve the retrieval efficiency. Here, the bloom filter was used to eliminate partly extended signatures, reducing disk IO time. The range query was used to replace individual scanning of the prefix query, effectively reducing the amount of data loading and the number of comparisons required. Experimental results show that, compared to existing algorithms, for scaling transformations, watermark transformations, and storage format transformations, the proposed method exhibited a higher precision rate and recall rate. Meanwhile, the method's real-time responsiveness and scalability also met actual needs. Future work will mainly focus on more complex image transformations, especially cropping transformation.

Acknowledgments This research is supported by the National Natural Science Foundation of China (Grant Nos. U1504608 and 81501548) and the Foundation and Cutting-Edge Technologies Research Program of Henan Province (132300410430).

References

1. Wang, S.G., Liu, Z.P., Sun, Q.B., et al.: Towards an accurate evaluation of quality of cloud service in service-oriented cloud computing. J. Intell. Manuf. **25**, 283–291 (2014)
2. Wang, S.G., Zheng, Z.B., Wu, Z.P., et al.: Reputation measurement and malicious feedback rating prevention in web service recommendation systems. IEEE Trans. Serv. Comput. **8**, 755–767 (2015)
3. Wang, S.G., Huang, L., Hsu, C.-H., et al.: Collaboration reputation for trustworthy web service selection in social networks. J. Comput. Syst. Sci. **82**(1), 130–143 (2016)
4. Wang B., Li Z.W., Li, M.J. et al.: Large-scale duplicate detection for web image search. In: IEEE International Conference on Multimedia and Expo, pp. 353–356 (2006)
5. Changick, K.: Content-based image copy detection. Signal Process Image Commun. **18**(3), 169–184 (2003)
6. Li, H.F., Xu, Z.H., Zhou, F.H., et al.: A robust image copy detection scheme using ordinal measure of full DCT coefficients. J. Comput. Res. Dev. **47**(10), 1812–1822 (2010)
7. Chang, E.Y., Wang, J.Z., Li, C., RIME: A replicated image detector for the world-wide-web. In: Proceedings of SPIE Symposium of Voice, Video, and Data Communications, pp. 68–77, Boston, USA, (1998)
8. Wu, M.N., Lin, C.C., Chang, C.C., Image copy detection with rotating tolerance. In: Proceedings of Computational Intelligence and Security (CIS), pp. 464–469 (2005)
9. Zhou, F.H., Li, X.W., Xu, Z.H., et al.: Image copy detection with rotation and scaling tolerance. J. Comput. Res. Dev. **46**(8), 1349–1356 (2009)
10. Hsiao, J.H., Chen, C.S., Chien, L.F., Chen, M.S.: A new approach to image copy detection based of extended feature sets. IEEE Trans. Image Process **16**(8), 2069–2079 (2007)
11. Feng, L., Wu, J., Sl, Liu, et al.: Global correlation descriptor: a novel image representation for image retrieval. J. Vis. Commun. Image Represent. **33**, 104–114 (2015)
12. Jegou, H., Perronnin, F., Douze, M., et al.: Aggregating local image descriptors into compact codes. IEEE Trans. Pattern Anal. Mach. Intell. **34**, 1704–1716 (2012)
13. Lowe, D.G. Object recognition from local scale-invariant features. In: Proceedings of the 7th International Conference on computer Vision, pp. 1150–1157 (1999)
14. Mikolajczyk, K., Schmid, C.: A performance evaluation of local descriptors. IEEE Trans. Pattern Anal. Mach. Intell. **27**, 1615–1630 (2005)
15. Bosch, A., Zisserman, A., Munoz, X. Scene classification via pLSA. In: Proceedings of the 9th European Conference on Computer Vision, pp. 517–530 (2006)
16. Ke, Y., Sukthankar, R. PCA-SIFT: a more distinctive representation for local image descriptors. In: Proceedings of the 2004 IEEE computer society conference on Computer Vision and Pattern Recognition, pp. 506–513 (2004)
17. Bay, H., Tuytelaars, T., Van Gool, L.: SURF: speeded up robust features. In: Proceedings of the 9th European Conference on Computer Vision, vol 1, pp. 404–417 (2006)
18. Bauer, J., Sunderhauf N., Protzel P. Comparing several implementations of two recently published feature detectors. In: International Conference on Intelligent and Autonomous Systems, pp. 143–148 (2007)

19. Matas, J., Chum, O., Urban, M., et al. Robust wide baseline stereo from maximally stable extremal region. In: Proceedings of the 13th British Machine Vision Conference (BMVC), pp. 384–393 (2002)
20. Calonder, M., Lepetit, V., Strecha, C., et al. BRIEF: binary robust independent elementary features. In: Proceedings of the 11th European Conference on Computer Vision(ECCV), pp. 778–792 (2010)
21. Rublee, E., Rabaud, V., Konolige, K., et al. ORB: an efficient alternative to SIFT or SURF. In: Proceedings of IEEE International Conference on Computer Vision (ICCV), pp. 2564–2572 (2011)
22. Tola, E., Lepetit, V., Fua, P. A fast local descriptor for dense matching. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8 (2008)
23. Alahi, A., Ortiz, R., Vandergheynst, P. FREAK: fast retina keypoints. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 510–517 (2012)
24. Yang, X., Cheng, K.: Local difference binary for ultrafast and distinctive feature description. IEEE Trans. Pattern Anal. Mach. Intell. **36**(1), 188–194 (2014)
25. Indyk, P., Motwani, R. Approximate nearest neighbors: towards removing the curse of dimensionality. In: Proceedings of the 30th Annual ACM Symposium on Theory of Computing, pp. 604–613 (1998)
26. Broder, A.Z. On the resemblance and containment of documents. In: Proceedings of Compression and Complexity of Sequences, pp. 21–29 (1997)
27. Moses, S. Similarity estimation techniques from rounding algorithms. In: Proceedings of the 34th Annual ACM Symposium on Theory of Computing, pp. 380–388 (2002)
28. Shakhnarovich, G., Viola, P.A., Darrell, T. Fast pose estimation with parameter sensitive hashing. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 750–759 (2003)
29. Torralba, A., Fergus, R., Weiss, Y. Small codes and large image databases for recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1–8 (2008)
30. Zhang, D., Wang, J., Cai, D., et al. Self-taught hashing for fast similarity search. In: Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 18–25 (2010)
31. Zou, F.H., Feng, H., Ling, H.F., et al.: Nonnegative spare coding induced hashing for image copy detection. J. Neurocomput. **105**, 81–89 (2013)
32. Ren, X.F., Ramanan, D.: Histograms of sparse codes for object detection. IEEE Conf. Comput. Vis. Pattern **9**(4), 3246–3253 (2013)
33. Wang, J., Kumar, S., Chang, S.F. Semi-supervised hashing for scalable image retrieval. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3424–3431 (2010)
34. Bauml, M., Tapaswi, M., Stiefelhagen, R.: Semi-supervised learning with constraints for person identification in multimedia data. In: Proceedings of the IEEE Conference on Computer Vision and Pattern, pp. 3602–3609 (2013)
35. Yuan, P.S., Sha, C.F., Wang, X.L., et al.: C-approximate nearest neighbor query algorithm based on learning for high-dimensional data. J. Softw. **23**(8), 2018–2031 (2012)

Ming Chen is a lecturer at Software Engineering College, Zheng Zhou University of Light Industry. He received his Ph.D. degree in Beijing University of Posts and Telecommunications in 2014. His research interests include multimedia information retrieval and data mining.

Yuhua Li is a lecturer at Software Engineering College, Zheng Zhou University of Light Industry. He received his Ph.D. degree in Computer Software and Theory from Sun Yat-sen University in 2014. His current research interests include multimedia information retrieval, machine learning and computational intelligence.

Zhifeng Zhang is an associate professor at Software Engineering College, Zheng Zhou University of Light Industry. He received his master degree in Xi'an University of Technology. His research interests include software engineering.

Ching-Hsien Hsu is a professor in Department of Computer Science and Information Engineering at Chung Hua University, Taiwan, and distinguished chair professor in School of Computer and Communication Engineering at Tianjin University of Technology, China. His research includes high-performance computing, cloud computing, parallel and distributed systems, ubiquitous/pervasive computing and intelligence. He has published 200 papers in refereed journals, conference proceedings, and book chapters in these areas. Dr. Hsu is the editor-in-chief of international journal of Grid and High Performance Computing and international journal of Big Data Intelligence and serving as editorial board for a number of prestigious journals, including IEEE Transactions on Service Computing, IEEE Transactions on Cloud Computing. He has been acting as an author/co-author or an editor/co-editor of 10 books from Springer, IGI Global, World Scientific and McGraw-Hill. He has also edited a number of special issues at top journals, such as IEEE Transactions on Cloud Computing, IEEE Transactions on Services Computing, Future Generation Computer Systems, Journal of Supercomputing, International Journal of Communication Systems, Automated Software Engineering, Journal of System Architecture, Concurrency and Computation: Practice and Experience, The Knowledge Engineering Review, Internet Research, Information System Frontiers. He was awarded 6 times distinguished award for excellence in research and annual outstanding research award through 2005–2012 from Chung Hua University. He has been serving as executive committee of Taiwan Association of Cloud Computing (TACC) from 2008 to 2012 and executive committee of the IEEE Technical Committee of Scalable Computing (2008–2012). He is an elected member of the Phi Tau Phi Scholastic honor society; IEEE senior member; and standing director of Taiwan Association of Cloud Computing (TACC).

Shangguang Wang is an associate professor at the State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications. He received his PhD degree in computer science at Beijing University of Posts and Telecommunications of China in 2011. His PhD thesis was awarded as outstanding doctoral dissertation by BUPT in 2012. His research interests include Service Computing, Mobile Services, QoS Management.