

RESEARCH

Open Access



Self-feedback image retrieval algorithm based on annular color moments

Ying Deng¹ and Yuanhui Yu^{2*}

Abstract

Content-based image retrieval (CBIR) extracts visual content features (such as color, texture, and shape) of a sample image to retrieve another similar image. Due to the existence of the semantic gap, retrieval results are often unsatisfactory. A CBIR method based on relevance feedback (RF) can reduce the semantic gap and achieve a high-retrieval accuracy by establishing a correlation between low-level image features and high-level semantics via human-computer interaction. However, the complicated human-computer interface of RF increases the burden on users; hence, some scholars have proposed the pseudo-relevance feedback (PRF) technology. To further contribute to the research, this paper proposes a self-feedback image retrieval algorithm based on annular color moments. In this approach, hashing sequences of color moments based on annular segmentation are extracted to be used as feature vectors for initial retrieval. Based on this result, improved subtractive clustering and correlation feedback techniques are used for extended queries. Thus, a self-feedback method without user participation is realized. The experimental results show that the accuracy of image retrieval can be improved, and the proposed algorithm is robust to image rotation, scaling, and translation.

Keywords: Image retrieval, Annular color moment, Fuzzy clustering, Self-feedback

1 Introduction

In the era of Web 2.0, especially with the popularity of social networking sites such as Flickr and Facebook, unstructured data such as images, videos, and audios are growing at an alarming rate every day. How to quickly and accurately find the images one needs from a large-scale database has become the focus of research. Techniques for content-based image retrieval (CBIR) use only visual features of an image as a query, storing them in an image feature library [1]. Since these features are usually high-dimensional, storing and retrieving massive visual features are the main challenges for developing the CBIR technology. Hashing is one of the emerging technologies for supporting fast and accurate image retrieval that can be applied as an effective technique for CBIR [2]. The core idea is to map high-dimensional visual features to compact binary codes in the low-dimensional Hamming space, so that visual similarities of images can be efficiently measured using simple yet efficient bit operations. Using hashing as underlying indexing, the storage

can be significantly reduced and the retrieval process can be completed rapidly [3].

The traditional cryptographic hashing method can map any input data to a fixed-length digital sequence; even if only one bit of the input data is changed, the output value will change drastically. In the field of image processing, image enhancement and compression are very common operations. These operations can change the specific representation data of an image, but cannot change the visual essence of the image, that is, the image hashing is maintained constant. Therefore, the hashing function in traditional cryptography is not suitable for image hashing calculation. Hashing functions for images exhibit two basic properties: perceptual robustness and uniqueness. Perceptual robustness means that visually similar images generate the same or a similar hashing value. Uniqueness means that the hashing value is completely different for images with different content [4, 5].

Recently, several image hashing methods have been proposed. The previous image hashing methods extracted some simple and effective global statistical characteristics to represent image hashes, including various histograms of images (e.g., luminance histogram, height

* Correspondence: 200261000045@jmu.edu.cn

²Computer Engineering College of JiMei University, Xiamen 361021, China
Full list of author information is available at the end of the article

histogram, color histogram, cumulative histogram, and cross histogram), mean, variance, color coherence vector, and color moments [6]. The image hashing methods based on global statistical characteristics can obtain a certain robustness of hashing, that is, the hashing extraction method is robust against a series of conventional operations such as image JPEG compression and filtering. However, this method has a serious flaw; it is not particularly sensitive to some minor illegal disturbances, which can lead to serious safety hazards. For example, the image content may completely be altered by malicious attackers during network transmission. Simultaneously, the tampered illegal images may maintain the same histogram, mean, and variance characteristics as those before tampering. Therefore, approaches to extract hashes based on this method are less secure. Later studies have proposed image hashing algorithms based on the transformation and time domains. Among them, the algorithms based on transformation domain can be divided into the discrete Fourier transform (DFT) domain [5, 7], discrete wavelet transform (DWT) domain [2], discrete cosine transform (DCT) domain [8], the integral transform Radon domain [9], etc. The image transformation domain can represent both image detailed characteristics and global contours. Different transformation domains can characterize different characteristics of images, and image characteristics extracted based on different domains can be adapted to different scenarios. Therefore, an appropriate transformation domain for image feature extraction can be selected as needed.

While many effective hashing algorithms have been proposed, there are still many problems to be solved in practical applications. For example, a hashing algorithm based on DWT can resist JPEG compression better, while that based on DCT has a better uniqueness; at the same time, their rotation robustness still needs to be improved. In addition, the existing hashing techniques suffer from two major limitations. First, most existing hashing algorithms are based on gray images. If a color image is input, it is represented by the luminance component. Because the algorithms do not consider information such as hue and saturation, the uniqueness is limited. Second, most hashing schemes employ only low-level visual features, and the well-known semantic gap degrades the CBIR performance [3]. To solve these problems, this paper uses the annular geometric segmentation method to extract the color moments of color images and combines fuzzy clustering and virtual correlation feedback technology to realize the self-feedback of image retrieval [10, 11]. The experimental results show that the proposed method is robust to rotation, scaling, and noise and exhibits good uniqueness.

The remaining paper is organized as follows. Section 2 describes the proposed color image hashing algorithm

based on annular color moments. Section 3 describes the self-feedback image retrieval algorithm based on fuzzy clustering. Section 4 presents the experimental results and their discussion. Section 5 draws a conclusion of this paper.

2 Methods

Currently, most image hashing algorithms mainly deal with gray images. If a color image is input, it is represented by the luminance component of the YCbCr color space, ignoring the hue and saturation information, thus limiting the uniqueness of the algorithm. To solve this problem, annular color moments are proposed to represent hashing values of color images. This method is divided into three steps: preprocessing, feature extraction, and hashing generation.

2.1 Preprocessing

Considering that the input image size may vary, we first use bilinear interpolation to adjust the input image to a fixed size of $N \times N$ (256×256 in this paper) to ensure that the hashing generated by the algorithm has a fixed length. Next, the input RGB color image is converted to the CIE $L^*a^*b^*$ color model representation. The RGB color model is the most commonly used, but because of its non-uniform and unintuitive shortcomings, it has a certain distance from human visual perception. Therefore, the CIE $L^*a^*b^*$ color model is used in this paper as it is more suitable for color vision [1]. The $L^*a^*b^*$ model consists of the lightness (L^* , Luminance) and color components (a^* and b^*) of the relevant color. L^* represents the change in brightness from bright ($L^* = 100$) to dark ($L^* = 0$). The value of a^* indicates the change in color from green ($-a^*$) to red ($+a^*$), and that of b^* indicates the change in color from yellow ($+b^*$) to blue ($-b^*$). The general formula for converting an image from the RGB space to the $L^*a^*b^*$ space is as follows:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.607 & 0.174 & 0.201 \\ 0.299 & 0.587 & 0.114 \\ 0.000 & 0.006 & 1.117 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (1)$$

$$\begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \end{bmatrix} = \begin{bmatrix} 0.607 & 0.174 & 0.201 \\ 0.299 & 0.587 & 0.114 \\ 0.000 & 0.006 & 1.117 \end{bmatrix} = \begin{bmatrix} 255 \\ 255 \\ 255 \end{bmatrix} \quad (2)$$

$$L^* = \begin{cases} 116 \left(\frac{Y}{Y_0} \right)^{1/3} - 16, & \frac{Y}{Y_0} > 0.008856 \\ 903.3 \left(\frac{Y}{Y_0} \right)^{1/3}, & \frac{Y}{Y_0} \leq 0.008856 \end{cases} \quad (3)$$

$$a^* = 500 \left[f\left(\frac{X}{X_0}\right) - f\left(\frac{Y}{Y_0}\right) \right] \quad (4)$$

$$b^* = 200 \left[f\left(\frac{Y}{Y_0}\right) - f\left(\frac{Z}{Z_0}\right) \right] \quad (5)$$

$$f(t) = \begin{cases} t^{\frac{1}{3}} & , t > 0.008856 \\ 7.787t + 116 & , t \leq 0.008856 \end{cases} \quad (6)$$

where X , Y , and Z are the XYZ color space components, and X_0 , Y_0 , and Z_0 are the components of the reference white point [12].

2.2 Feature extraction

The color moment proposed by Stricker and Orengo is a very simple and effective color feature. The mathematical basis of this method is that any color distribution in an image can be represented by its moments. Because the color distribution information is mainly concentrated in the low-order moments, only the mean, variation, and skewness of a color are sufficient to express the color distribution of the image. The mathematical expressions of the three low-order color moments are

$$\mu_i = \frac{1}{N} \sum_{j=1}^N p_{ij} \quad , i = 1, 2, 3 \quad (7)$$

$$\sigma_i = \left(\frac{1}{N} \sum_{j=1}^N (p_{ij} - \mu_i)^2 \right)^{1/2} \quad , i = 1, 2, 3 \quad (8)$$

$$s_i = \left(\frac{1}{N} \sum_{j=1}^N (p_{ij} - \mu_i)^3 \right)^{1/3} \quad , i = 1, 2, 3 \quad (9)$$

where N represents the total number of pixels in the image and p_{ij} is the i th color component of the j -th pixel in the image. Therefore, the total color moment of the image requires only nine components: three color components (L^* , a^* , and b^* in this paper) and three low-order moments per component.

The spatial information of the image is also a very important feature in image retrieval. Because color moments only reflect the overall characteristics of an image and do not express the spatial position of image colors, we use the annular geometrical segmentation method to record the color spatial information [13]. The process of extracting color moments based on annular geometric segmentation involves dividing an image into a circle, an $M-1$ annular, and a residual part at equal intervals according to the coordinates of the center point and then calculating the $L^*a^*b^*$ color moment feature vector of the segmented regions separately. The annular geometric segmentation of the image is shown in Fig. 1. Assuming



Fig. 1 Annular geometric segmentation diagram

W and H to be the width and height of an image, the radius of the circle and the annulus can be expressed as

$$r = \begin{cases} W/2M, & W \leq H \\ H/2M, & W > H \end{cases} \quad (10)$$

The image is divided into M annular parts according to its size. The color moment eigenvectors of the central annular region, the middle $M-1$ annular region, and the segmented remainder are calculated. The relation between the coordinates of an image pixel and the region number can be expressed as

$$ZN = \frac{\sqrt{(x-O_x)^2 + (y-O_y)^2}}{r} + 1 \quad (11)$$

In the above equation, ZN denotes the region number, O_x denotes the abscissa of the center point of the part of the image that is of interest, O_y denotes the vertical coordinate of the center point of the image part of interest, and r denotes the image segmentation radius [13].

The final image hashing value can be obtained by connecting the color moments of three different color components in turn. In image retrieval, the results can be obtained by calculating and comparing the similarity between the corresponding color moment feature vector sequences of two images.

2.3 Color image hashing algorithm based on annular color moments

For a color RGB image, the hashing algorithm based on annular color moments is described as follows.

1. Image preprocessing: First, bilinear interpolation is used to adjust the size of the input image to $N \times N$ (256×256 in this paper). Then, the input RGB color image is converted to the CIE $L^*a^*b^*$ color model representation.

2. Feature extraction: First, an image is divided into $M + 1$ parts: a circle, $M-1$ annuluses, and one division of the remaining part (Fig. 1). Then, the $L^*a^*b^*$ color moment eigenvectors of these divided regions are calculated.

3. Hashing generation: To ensure the validity of the similarity measure, the Gaussian normalization method is first used to process the different components of the feature vector, which may have different physical meanings and value ranges [1]. After normalization, the values of the feature components lie between 0 and 1. Then, each feature component is binarized, and if the vector value is greater than or equal to 0.5, the vector value is set to 1; otherwise, it is set to 0. Finally, by connecting the three color moment components of the three color channels (i.e., L^* , a^* and b^*) in sequence, a binary string h_i with a length of nine bits for each divided region is obtained.

4. Similarity measure: To determine the similarity of two images, we measure the similarity between their indices. In this paper, we use the normalized Hamming distance to measure the similarity between two hashing sequences h_1 and h_2 :

$$d(h_1, h_2) = \frac{1}{L_h} \sum_{i=1}^n |h_1(i) - h_2(i)| \quad (12)$$

where L_h is the length of hashing sequence and $h_1(i)$ and $h_2(i)$ are the binary bits in the corresponding hashing sequences.

Assuming that the image to be retrieved is Q and I is any target image in the image database, we use the weighted Hamming distance of the corresponding vectors in the color moment hashing sequences of the two images to measure the similarity between the feature vectors. We define the similarity as

$$D(Q, I) = \sum_{k=1}^{M+1} W_k \cdot d_k(h_1, h_2) \quad (13)$$

where w_k ($1 \leq k \leq M + 1$) is the user-specified weight of each annular region, which shows the visual importance of the respective regions. In general, people are more interested in the central region of an image; therefore, the values of the weights decrease in the order from inside to outside, emphasizing the central circle and annular regions closer to the center of the target area [9]. The smaller the distance, the more similar two images are, and the larger the distance, the greater is the difference.

3 Self-feedback image retrieval algorithm based on fuzzy clustering

At present, most image retrieval systems are based on low-level visual features (such as color and texture), and the retrieval process is computer-centric, which makes its retrieval performance unsatisfactory. This is mainly due to the gap between low-level visual features and high-level semantic concepts. In addition, the standards of judging the similarity between images by human and computer image retrieval systems are very different. To solve these problems, a CBIR method based on relevance feedback (RF) is proposed. This method embeds the user model in the retrieval system and establishes a correlation between the low-level features of the image and its high-level semantics via human-computer interaction, thereby reducing the semantic gap and achieving a high retrieval accuracy [14, 15]. However, RF's unique complex human-computer interaction interface increases the burden on users. In each feedback, the user must indicate which images are related and score the degree of correlation between these images and the query image. In some cases, the user will be confused and face difficulty in giving an appropriate judgment [16]. Therefore, some scholars have proposed the pseudo-relevance feedback (PRF) technique, which is also called local feedback or blind feedback. The main idea is to improve the retrieval performance of a system by extending the query on the basis of a reasonable assumption that some documents ranked at the top are assumed to be related to the query documents and related documents are selected automatically. The strategies of selecting positive examples and the query expansion method are the key issues in applying the virtual relevance feedback technology [15–17]. Based on the above research, this paper proposes a method for image retrieval self-feedback based on fuzzy clustering and the virtual relevance feedback technology. This method is mainly based on the similarity measure between the image content features; it applies the fuzzy clustering algorithm to automatically expand the query image features without user intervention and to improve the search performance.

3.1 Related technologies

Content-based relevance feedback retrieval is a process of gradual refinement. The basic idea is that the user is allowed to evaluate and mark the retrieval results in the search process, indicating which of the results are related to the query image and which are irrelevant, i.e., “feedback positive examples” and “feedback negative examples.” The system receives the user feedback on the current retrieval results and then automatically adjusts the query based on the feedback information. Finally, the optimized query is used to recalculate the retrieval results. In this way, the system and user gradually make

the retrieval result to move toward the direction of the user's expectation through the interaction and to finally reach the user's request.

At present, the vector model is often used in image retrieval. That is, the image is represented in the vector form in the feature space; hence, the essence of image retrieval is to find the images corresponding to those closest to the query vector in the feature space. From the viewpoint of the vector model, the correlation feedback techniques can be divided into two categories: the point moving algorithm based on the query vector and the feature weight adjusting algorithm. These correlation feedback methods are based on the premise that the image visual features can completely describe the image semantics. Under this assumption, the user's retrieval target can be described by a global distribution function. That is, similar images in the feature space are clustered around the central point. Therefore, the retrieval effect can be improved by moving the query points and modifying the distance measurement functions.

In practical applications, however, there are significant differences between the low-level features of images in the same category due to the diversity of semantically similar images. For example, the elephant class in the Corel image library includes elephants with different postures (e.g., standing and lying) and in different environments (e.g., forest, grassland, and water edge), as shown in Fig. 2. In terms of low-level features, the assumption that semantically similar images cluster around the central point in the

feature space is not always true. As shown in Fig. 3, where the dot Q represents the query image, the solid dot represents the retrieval target, the hollow dot denotes irrelevant images, and $L(Q)$ indicates the distribution of the user's retrieval targets; clearly, the distribution does not have an ideal global distribution center. It is difficult to describe it with some kind of canonical geometry. In this case, if the query image is taken as the center in the feature space and the image in the hypersphere with a constant radius is returned as the retrieval result, it is obvious that some of the related images will be missed, while some uncorrelated images will be misdirected. However, as shown in Fig. 4, when $L(Q)$ is divided into three sub-clusters, each cluster is uniformly distributed. The model can reflect the distribution of the correlated images in the feature space well; it also overcomes the shortcomings of the previous correlation feedback methods, which assume that semantically similar images cluster around the central point in the feature space, so it is helpful in finding more correlated images [15].

Based on the above ideas, a self-feedback strategy based on fuzzy clustering is proposed in this paper. First, the initial retrieval results generate multiple substitution queries using fuzzy clustering to form a new query. To some extent, this compensates for the shortage of image information in a single query sample. Then, these substitution queries are retrieved and merged with the initial retrieval results to form the final retrieval results.

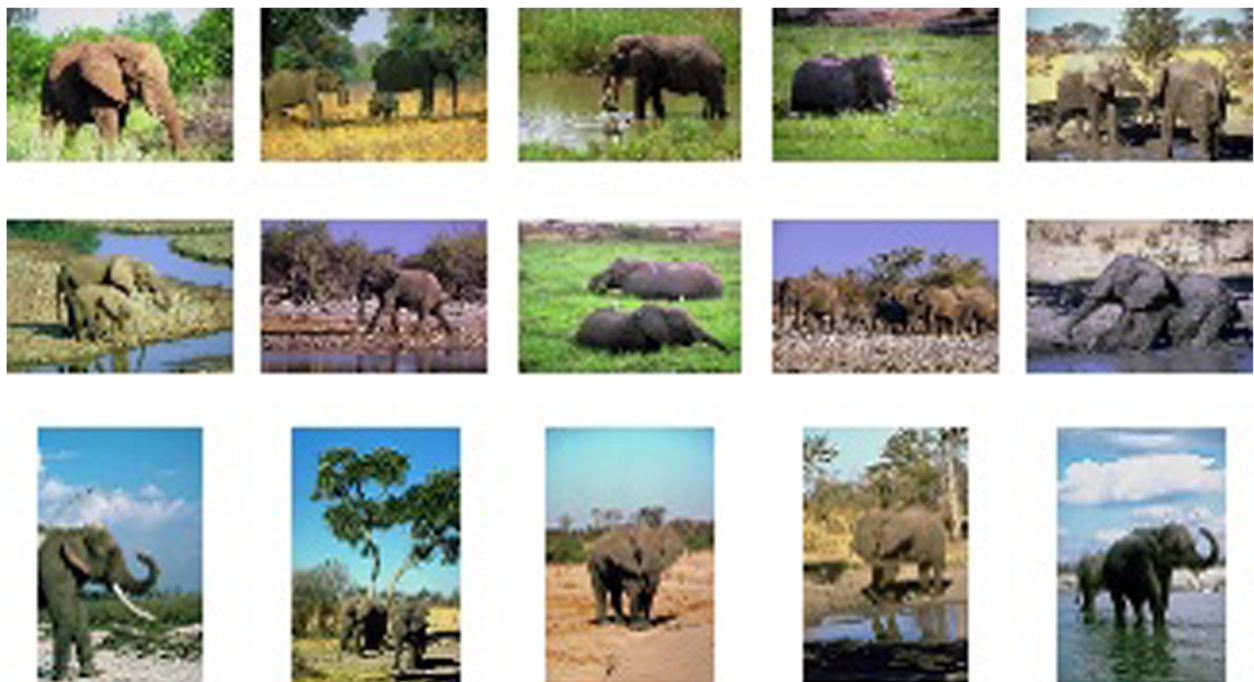


Fig. 2 Elephants in different environments and different poses

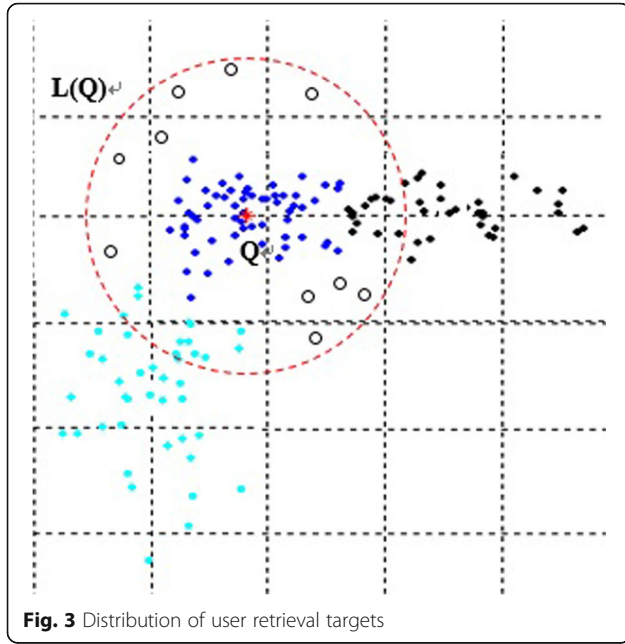


Fig. 3 Distribution of user retrieval targets

3.2 Design of the correlation feedback algorithm

Images with the same semantics may have greater differences in their low-level features. The traditional image retrieval methods based on a single query image usually can only retrieve images that are very similar to the query images in their low-level features and inevitably lose other images that have similar semantics but large differences in their low-level features. Therefore, if there are many different query images, rather than just relying on a query image, more relevant images can be retrieved in order to narrow the semantic gap [14, 18].

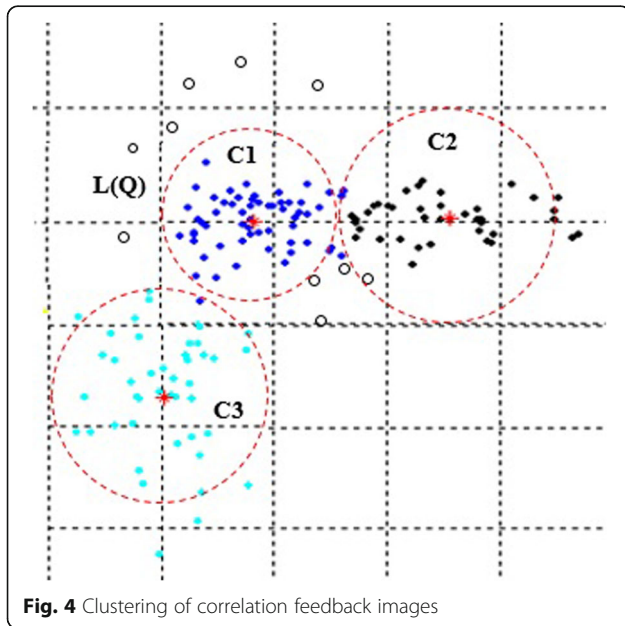


Fig. 4 Clustering of correlation feedback images

The key to using PRF technology to solve image retrieval problems is how to select a positive image. The commonly used precision-recall curve shows a nonlinear inverse relation between precision and recall in the image retrieval algorithm: a lower recall, for example, 0.1–0.2, corresponds to a higher precision, such as 0.8 or more. This means that in the initial retrieval results, the images that are ranked higher in the foreground are more likely to be related to the query image and are potentially relevant. In CBIR, the closest image to the query image is also the most similar image. Therefore, our proposed method automatically selects a group of images that are closer to the query image as positive examples in the initially retrieved results, clusters them according to the low-level features of the images, and selects each cluster center as a substitute for the query feedback to improve the retrieval performance.

3.3 Improved non-parametric fuzzy clustering algorithm

3.3.1 Variable description

The simultaneous clustering and attribute discrimination (SCAD) algorithm is designed to simultaneously search for the optimal cluster center C and the optimal feature weight set W . Each class has its own set of feature weights: $W_i = [w_{i1}, w_{i2}, \dots, w_{iD}]$. Its objective function is defined as follows

$$J(C, U, W; x) = \sum_{i=1}^C \sum_{j=1}^N u_{ij}^m \sum_{k=1}^D w_{ik} (x_{jk} - c_{ik})^2 + \sum_{i=1}^C \delta_i \sum_{k=1}^D w_{ik}^2 \quad (14)$$

where

$$\begin{cases} u_{ij} \in [0, 1], \forall i \\ 0 < \sum_{j=1}^N u_{ij} < N, \forall i, j, \\ \sum_{i=1}^C u_{ij} = 1, \forall j \end{cases}$$

and

$$w_{ik} \in [0, 1], \forall i, k; \sum_{k=1}^D w_{ik} = 1, \forall i.$$

For w_{ik} , the definition is as follows

$$w_{ik} = \frac{1}{n} + \frac{1}{2\delta_i} \sum_{j=1}^N (u_{ij})^m \left[\frac{\|x_j - c_i\|^2}{n} - (x_{jk} - c_{ik})^2 \right] \quad (15)$$

The definition of δ_i is

$$\delta_i = K \frac{\sum_{j=1}^N (u_{ij})^m \sum_{k=1}^D w_{ik} (x_{jk} - c_{ik})^2}{\sum_{k=1}^D (w_{ik})^2} \quad (16)$$

The modified membership formula is

$$u_{ij} = \frac{1}{\sum_{k=1}^c \left(\frac{d_{ij}^2}{d_{kj}^2} \right)^{\frac{1}{m-1}}} \quad (17)$$

The cluster centers can be expressed as

$$C_{ik} = \begin{cases} 0 & \text{if } w_{ik} = 0 \\ \frac{\sum_{j=1}^N (u_{ij})^m x_{jk}}{\sum_{j=1}^N (u_{ij})^m} & \text{if } w_{ik} > 0 \end{cases} \quad (18)$$

3.3.2 Initialization of cluster centers

The SCAD algorithm first needs to initialize the clustering center. The algorithm is sensitive to the initialization of the clustering center. If the initialization is not appropriate, it will lead to convergence of the algorithm to the local extreme point without the optimal fuzzy division of the dataset. At the same time, the algorithm is time-consuming when the data are large. In this paper, the subtraction clustering algorithm is used to initialize the clustering center [19, 20].

Assuming $X = \{x_1, x_2, \dots, x_n\} \subset \mathbb{R}^d$ be the sample point set and n be the number of samples, the process of initializing the cluster center by subtractive clustering is as follows:

1. For each x_i in X , its density index is calculated according to formula (19) and the data point x_{c1} with the highest density index is selected as the first cluster center

$$D_i = \sum_{j=1}^n \exp \left[\frac{-\|x_i - x_j\|^2}{(0.5r_a)^2} \right] \quad (19)$$

2. Assuming that x_{ck} is the k -th selected cluster center and the corresponding density index is D_{ck} , the density index of each data point is modified according to formula (20), and the data point x_{ck+1} with the highest density index is selected as the new clustering center

$$D_i = D_i - D_{ck} \sum_{j=1}^n \exp \left[\frac{-\|x_i - x_{ck}\|^2}{(0.5r_b)^2} \right] \quad (20)$$

3. Calculate D_{ck+1}/D_{c1} ; if the result is less than δ , the algorithm ends; otherwise, go to step 2

The parameters r_a , r_b , and δ need to be predetermined. The parameter δ ($0.5 \leq \delta < 1$) defines the number of initialized cluster centers that are ultimately generated. The smaller the δ is, the larger is the number of clusters generated, while the larger the δ is, the smaller is the number of clusters generated. In this paper, let $\delta = 0.5$, and the values of r_a and r_b are as shown in formula (21).

$$r_a = r_b = \frac{1}{2} \min_k \left\{ \max_i \{ \|x_i - x_k\| \} \right\} \quad (21)$$

3.4 Algorithm description

The improved non-parametric fuzzy clustering algorithm is described as follows:

1. The initial cluster center C is obtained using the algorithm described in Section 3.3.2; let $C_{\max} = k$;
2. For $i = C_{\max}$ to C_{\min} :
 - 2.1 Select the first i clustering centers in C as the new initial centers $C^{(0)}$;
 - 2.2 Update U , C , and W using formulas (17), (18), and (15);
 - 2.3 Determine its convergence; if not, go to 2.2; otherwise perform 2.4;
 - 2.4 Calculate the index value $V_d(c)$ using the effectiveness indicator function;

$$V_d(c) = \frac{\sum_{i=1}^c \frac{1}{n_i} \sum_{k=1}^n u_{ik}^m \cdot \text{dist}_{D_i}(x_k, c_i)^2}{\min_{i=1, \dots, c} \left\{ \min_{j=1, \dots, c, i \neq j} \text{dist}_{D_i}(c_i, c_j)^2 \right\}} \quad (22)$$

3. Compare the validity index values, where c_{i0} corresponds to the maximum (or minimum) index value and $V_d(c_{i0})$ is the optimal number of clusters

3.5 Self-feedback algorithm based on fuzzy clustering

The process of self-feedback based on fuzzy clustering is as follows. First, an initial search is performed. Then, the P-images closer to the query image are selected as positive examples from the initial retrieval results, clustered according to the low-level features of the images, and selected. Multiple cluster centers are used as substitute queries for feedback retrieval. Finally, these retrieval results are merged with the initial retrieval results to form the final retrieval results. This method extracts the target features from multiple related images instead of relying solely on a single query image. The algorithm is designed as follows:



1. The user submits query image Q ;

2. The user performs an initial search based on the annular moment hashing method described in Section 2.3 and arranges the retrieval results in

descending order of the similarity and the output;

3. If the user is satisfied with the search result, the query is ended; otherwise, step 4 is followed;

Table 1 Similarity measure statistics for various digital operations

Operation type	Min	Max	Standard deviation	Average value
Scaling transformation	0.9937	0.9986	0.0042	0.9971
Rotation transformation	0.9826	0.9978	0.0017	0.9942
Gaussian filter	0.969	0.973	0.0014	0.9713
Brightness adjustment	0.9918	0.9961	0.0016	0.9945
Contrast adjustment	0.9927	0.9974	0.0013	0.9957
Gamma correction	0.9731	0.9953	0.0073	0.9851
JPEG compression	0.9815	0.9963	0.0065	0.9898

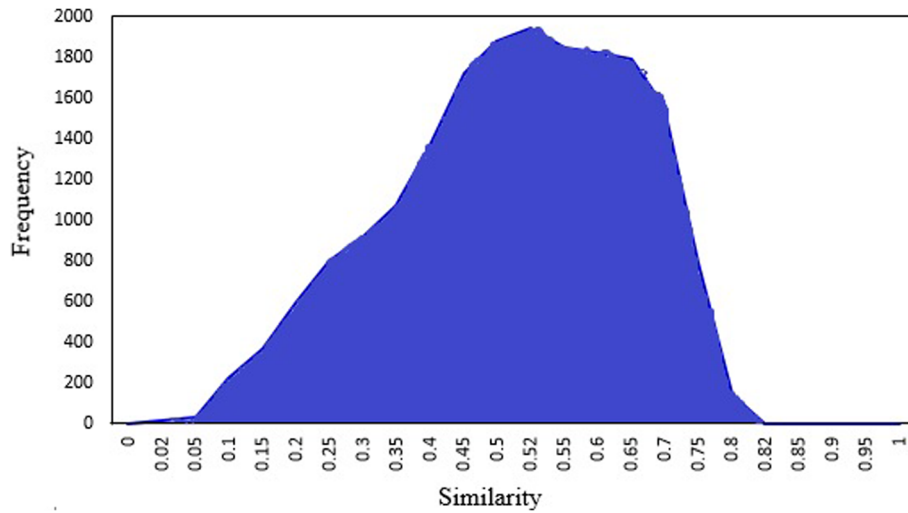


Fig. 6 Distribution of similarities between hashes of different images

4. The number of images participating in clustering is determined and P-images with the highest similarity from the above initial retrieval results $L(Q)$ are selected:

$$L(Q) = \{I_i | i = 1, 2, \dots, P\} \quad (23)$$

5. P-images are clustered using the improved non-parametric fuzzy clustering algorithm described in Section 3.4. The optimal clustering number k^* is the number of classifications of the related images. In this case, the corresponding clustering centers $C = \{C_1, C_2, \dots, C_{k^*}\}$ are the obtained substitution query vectors.
6. Calculating the similarity $L(C, D_j)$ of each image I_j in the image library to C , where D_j is the feature vector of I_j ,

$$L(C, D_j) = \min\{L(C_i, D_j) | i = 1, 2, \dots, k^*\} \quad (24)$$

7. The feedback search result $L(C, D_j)$ is combined with the initial search result to form the final search result and output it; then, step 3 is followed

4 Discussion and experimental results

4.1 Perceptual robustness

All experiments in this study are run using Win7 and MATLAB2010a software; the system memory is 4G. The

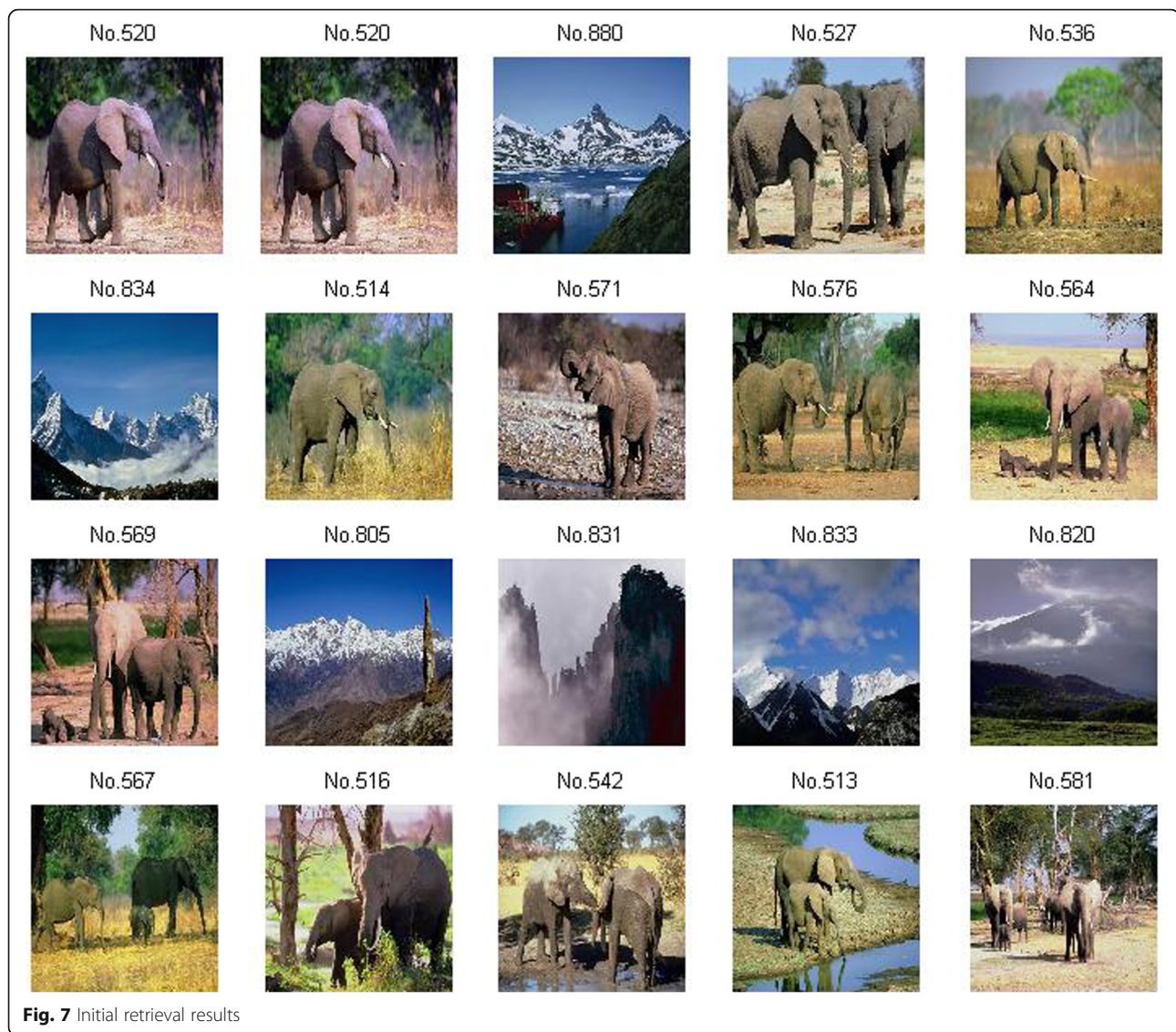
dataset used in the experiment includes 1000 colored images selected from the professional collection photo database of Corel Company as the original images. The dataset is divided into 10 categories: African life, beaches, buildings, buses, dinosaurs, elephants, flowers, horses, mountains, and food. Each category contains 100 images. The image size is 384×256 or 256×384 pixels. To verify the robustness of the algorithm, some common operations such as image scaling, rotation transformation, and noise addition are used to obtain the corresponding visually similar images.

Figure 5 is a part of the experimental results obtained using the proposed algorithm. The first image in the first row is a query image, while the remaining images are the retrieval results arranged according to the similarity measure from large to small. The figure shows that the image obtained by scaling, rotating, and adding noise is at the front of the retrieval result, which shows that the proposed method is robust to rotation, scaling, and noise.

Table 1 lists the performance of the proposed algorithm against various digital operations. It is found that the average similarity measure for all operations is greater than 0.96, which shows that the proposed algorithm can resist the above digital operations and has good robustness. In addition, except for the rotation transformation, the minimum similarity measures for all operations are greater than 0.98. Therefore, when the threshold is 0.9, the proposed algorithm can correctly recognize 87.8% of similar images.

4.2 Uniqueness

To verify the uniqueness measure obtained by the proposed method, a database containing 200 different color images is constructed as a test set. Among



them, 100 images are taken from Corel's professional image library and 100 are downloaded from the Internet. The contents of these images include buildings, animals, plants, and fish, and their size ranges from 256×256 to 3788×9254 pixels. The hashing values of these 200 images are extracted and the similarity between them is calculated. Figure 6 shows the distribution of the similarity of different image hashing sequences. The abscissa is the similarity, and the ordinate is its frequency. The results indicate that the largest similarity measure is 0.8291, the smallest is 0.02, and its mean and standard deviation are 0.0086 and 0.2304, respectively. When the threshold is set to 0.82, no image is misinterpreted as a similar image, which shows that the uniqueness measure obtained by the algorithm is good.

4.3 Performance comparison

Figures 7 and 8 represent the initial retrieval results and the retrieval results after a relevant feedback, respectively. The results in Fig. 10 show that the number of related images in the first 19 images returned for the query image after a relevant feedback is increased from 13 to 18. This is because by expanding the query vector, more query sample image feature information is provided; hence, after a relevant feedback, the precision rate is significantly improved.

To compare the retrieval performance of different methods, we employ a widely used measurement method, the precision-recall curve, as the evaluation criterion of the algorithm retrieval effect. The precision rate refers to the ratio of the number r of relevant images in the query result returned by the system to the

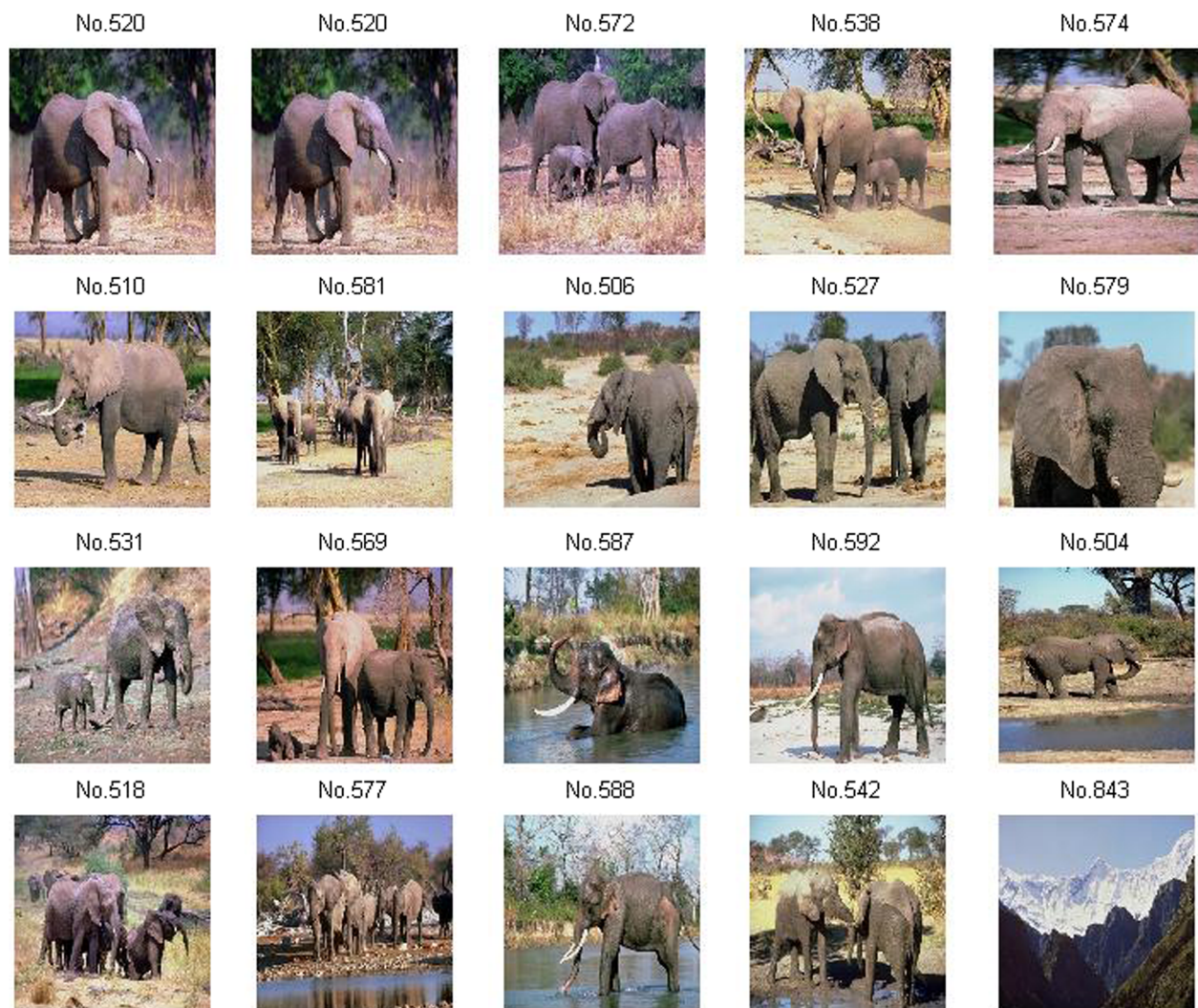


Fig. 8 Retrieval result following a relevant feedback

number N of all returned images during a query. The recall rate refers to the ratio of the number r of relevant images in the query result returned by the system to the number of all relevant images R (including returned and not returned) in the image library during a query. In general, the curve that is closer to the top of the graph indicates a better retrieval performance of an algorithm. In the experiment, 10 types of images are selected from the image library first, and then, 10 images are randomly selected for each type to constitute 100 queries. The average precision of the system is calculated by combining the results of the 100 queries.

Figure 9 shows the performance comparison between the initial retrieval results and the retrieval results obtained after one correlation feedback. It can be seen that by extending the query vector, the accuracy of the query is improved after a correlation feedback; and the average increase is about 7.15%.

When comparing this algorithm to the RT-DCT algorithm [21] and the HSV-GLCM [22] algorithm, the image library introduced in Section 4.1 is used. Figure 10 shows the comparison of the recall and precision of the method proposed in this paper and the retrieval results achieved by the RT-DCT and HSV-GLCM algorithms. It can be seen that the precision-recall curve of the method presented in this paper is at the top of the figure, indicating that this method has the best retrieval performance. Moreover, the method has a fast retrieval speed with the average time for returning 100 images being 1.498 s, while the average retrieval time for the other two algorithms is 1.988 s and 3.15 s.

5 Conclusions

This paper proposed an image hashing method combining annular color moments and virtual correlation

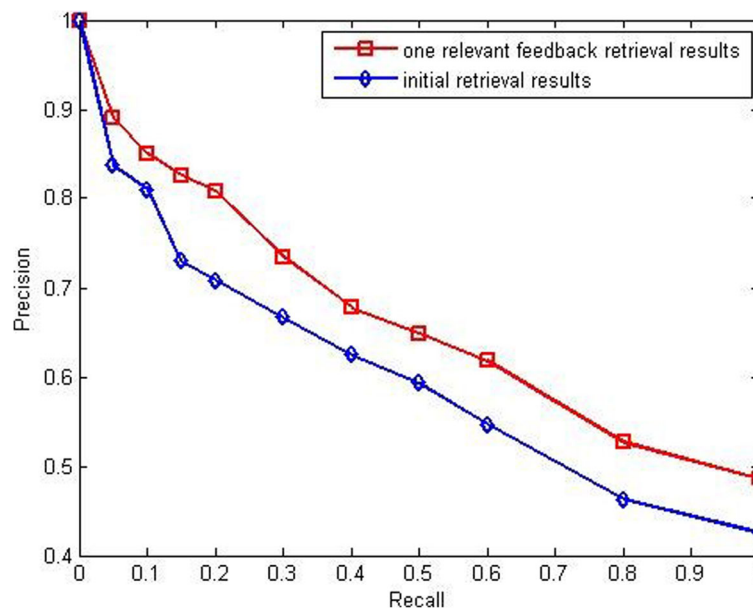


Fig. 9 Comparison of retrieval performance

feedback. The color moments based on annular segmentation were used to describe the color images, and the red, green, and blue components of the pixels were fully utilized to improve the performance of the method. Experimental results showed that the proposed method is robust to JPEG compression, brightness and contrast adjustment, image scaling, and rotation. Furthermore, this paper proposed a self-feedback method for image retrieval based on fuzzy clustering and virtual correlation feedback. This

method is mainly based on the similarity measurement between image content features and applies the fuzzy clustering algorithm to automatically extend query image features without user participation to improve retrieval performance. The experimental results showed that the method can improve the retrieval accuracy by using a limited number of feedbacks.

The current algorithm only considers the color feature, and the future research will include designing a hashing algorithm that can extract the image texture,

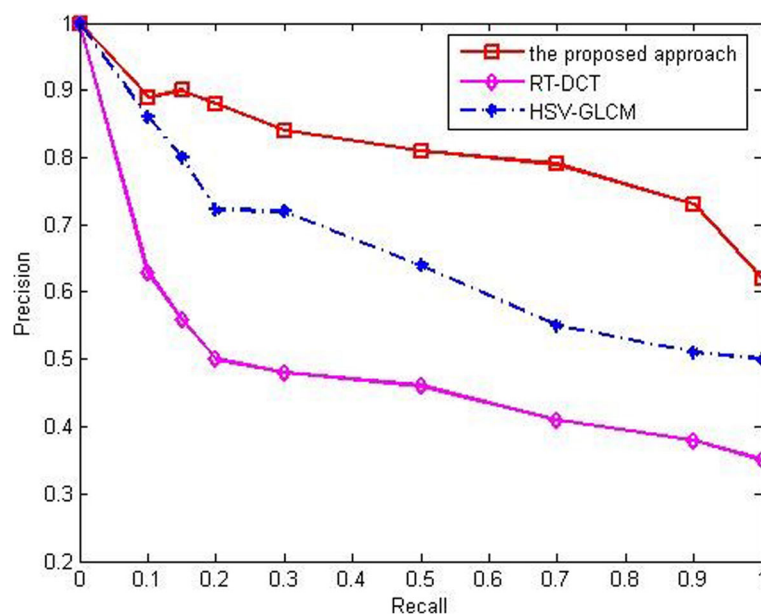


Fig. 10 Comparison of average recall and precision

its shape, and other related features to further optimize the algorithm and improve its retrieval performance. At the same time, we will combine hashing and other machine learning methods to solve the large-scale search problems effectively.

Abbreviations

CBIR: Content-based image retrieval; DCT: Discrete cosine transform; DFT: Discrete Fourier transform; DWT: Discrete wavelets transform; PRF: Pseudo-relevance feedback; RF: Relevance feedback

Acknowledgements

The authors thank the editor and anonymous reviewers for their helpful comments and valuable suggestions.

Funding

The research work was supported by the Education and Scientific Research Project for Middle-aged and Young Teacher of Fujian Province (No. JAT160631), in part by the Natural Science Foundation of Fujian Province, China (No. 2015J01288), in part by the Cooperative Education Project of Ministry of Education of the People's Republic of China (No. 201702038005).

Availability of data and materials

We can provide the data.

Authors' contributions

YD wrote the whole paper. YHY(*corresponding author) performed a part of the experiments of the paper and checked the entire article. Both authors read and approved the final manuscript.

Authors' information

Dengying received her M.S. degree in computer software and theory from Xiamen University in 2010. She is currently an associate professor of Xiamen Institute of technology. Her research interests are image processing, pattern recognition and data mining.

Yuyuanhui, associate professor, master tutor, received his M.S. degree from University of Electronic Science and technology of China in 2002 and worked in Jimei University. His research interests include image processing, intelligent information processing, mobile agent computing and data fusion.

Competing interests

The authors declare that they have no competing interests.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Author details

¹Xiamen Institute of technology, Xiamen 361021, China. ²Computer Engineering College of Jimei University, Xiamen 361021, China.

Received: 1 September 2018 Accepted: 21 December 2018

Published online: 08 January 2019

References

1. Y.T. Zhuang, Q. Pan, F. Wu, *Online Multimedia Information Analysis and Retrieval* (Tsinghua University Press, Beijing, 2002).
2. R. Venkatesan, S.-M. Koon, M.H. Jakubowski, P. Moulin, *Robust image hashing* (Proceedings of IEEE International Conference on Image Processing (ICIP2000), Vancouver, 2000), pp. 664–666.
3. L. Zhu, J. Shen, L. Xie, Z. Cheng, Unsupervised visual hashing with semantic assistant for content-based image retrieval. *IEEE Trans. Knowl. Data Eng.* **29**(2), 472–486 (2017).
4. Q. Chuan, C.C. Chen, G. Cheng, Perceptual robust image hashing scheme based on secret sharing. *J. Comput. Res. Dev.* **49**(8), 1690–1698 (2012).
5. Z. TANG, F. YANG, L. HUANG, et al., Image hashing with dominant DCT coefficients. *Opt. Int. J. Light Electron. Opt.* **125**(18), 5102–5107 (2014).
6. S. Xiang, H.J. Kim, J. Huang, *Histogram-Based Image Hashing Scheme Robust against Geometric Deformations*. *Workshop Multimedia Security* (2007), pp. 121–128.
7. M. Wu, A. Swaminathan, Y. Mao, *A Signal Processing and Randomization Perspective of Robust and Secure Image Hashing* (Proceedings of IEEE 14th Workshop on Statistical Signal Processing (SSP'07), Madison, 2007), pp. 166–170.
8. F. LEFEBVRE, B. MACQ, L.E.U.A.T. JD, *RASH: Radon Soft Hash Algorithm*. *Proceedings of 11th European Signal Processing Conference* (IEEE Press, Piscataway, 2002), pp. 299–302.
9. S. KOZAT S, R. VENKATESAN, M.K. MIHCAK, *Robust Perceptual Image Hashing via Matrix Invariants*. *Proceedings of 2004 International Conference on Image Processing* (IEEE Press, Piscataway, 2004), pp. 3443–3446.
10. L. Zhu, J. Shen, L. Xie, Z. Cheng, Unsupervised topic hypergraph hashing for efficient mobile image retrieval. *IEEE Trans. Cybernetics* **47**(11), 3941–3954 (2017).
11. D.C. Guimarães Pedronette, R.T. Calumby, R.d.S. Torres, A semi-supervised learning algorithm for relevance feedback and collaborative image retrieval. *J. Image Video Process.* (2015). <https://doi.org/10.1186/s13640-015-0081-6>.
12. R.C. Veltkamp, M. Tanase, *Content-Based Image Retrieval Systems: A Survey*. *Technical Report UU-CS-2000-34*, Dept. of Computing Science, Utrecht University (2000).
13. Y.U. Yuanhui, D.E.N.G. Ying, Research of marine organism image retrieval approach based on multi-feature. *J. Henan Univ. (Nature Science)* **5**(2), 217–222 (2015).
14. X. ZHOU, L. ZHANG, Q. ZHANG, et al., A relevance feedback method based on local distribution center in image retrieval. *Pattern Recognit. Artif. Intell.* **16**(2), 152–157 (2003).
15. X. WANG, K. XIE, Research on pseudo-relevance feedback and clustering-based image retrieval. *Comput. Eng. Des.* **29**(6), 1465–1471 (2008).
16. L. ZHANG, F. LIN, B. ZHANG, A forward neural network based relevance feedback algorithm design in image retrieval. *Chin. J. Comput.* **25**(7), 673–680 (2002).
17. X. Zhou, X. Liang, X. Du, J. Zhao, Structure based user identification across social networks. *IEEE Trans. Knowl. Data Eng.* **30**(6), 1178–1191 (2018).
18. D.H. KIM, C.H.U.N.G. CW, K. BARNARD, Relevance feedback using adaptive clustering for image similarity retrieval. *J. Syst. Softw.* **78**(1), 9–23 (2005).
19. H. Sun, M. Sun, *Trail-and-Error Approach for Determining the Number of Clusters* (Proceedings of 4th International Conference on Advances in Machine Learning and Cybernetics (ICMLC 2005), Guangzhou, 2005), pp. 229–238.
20. D. Lu, X. Huang, G. Zhang, X. Zheng, H. Liu, Trusted device-to-device based heterogeneous cellular networks: a new framework for connectivity optimization. *IEEE Trans. Veh. Technol.* **67**(11), 11219–11233 (2018).
21. Q. Ming, Research of color and texture based image retrieval technique [J]. *Sci. Technol. Eng.* **19**(15), 1301–1304 (2009).
22. C.Y. LIN, S.F. CHANG, A key-dependent secure image hashing scheme by using Radon transform. A robust image authentication method distinguishing JPEU compression from malicious manipulation. *IEEE Trans. Circuits Syst. Video Technol.* **11**(2), 153–168 (2001).

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► [springeropen.com](https://www.springeropen.com)