Research Article

International Journal of Distributed Sensor Networks

International Journal of Distributed Sensor Networks
2017, Vol. 13(2)

DOI: 10.1177/1550147717694172
journals.sagepub.com/home/ijdsn
SAGE

Fast and accurate near-duplicate image elimination for visual sensor networks

Zhili Zhou[1,2], QM Jonathan Wu[2], Fang Huang[1] and Xingming Sun[1]

Abstract

Currently, a huge amount of visual data such as digital images and videos have been collected by visual sensor nodes, that is, camera nodes, and distributed on visual sensor networks. Among the visual data, there are a lot of near-duplicate images, which cause a serious waste of limited storage, computing, and transmission resources of visual sensor networks and a negative impact on users' experience. Thus, near-duplicate image elimination is urgently demanded. This article proposes a fast and accurate near-duplicate elimination approach for visual sensor networks. First, a coarse-to-fine clustering method based on a combination of global feature and local feature is proposed to cluster near-duplicate images. Then in each near-duplicate group, we adopt PageRank algorithm to analyze the contextual relevance among images to select and reserve seed image and remove the others. The experimental results prove that the proposed approach achieves better performances in the aspects of both efficiency and accuracy compared with the state-of-the-art approaches.

Keywords

Visual sensor networks, Internet of Things, near-duplicate image elimination, image copy detection, near-duplicate detection

Date received: 7 October 2016; accepted: 17 January 2017

Academic Editor: Xuyun Zhang

Introduction

With the rapid developments of sensor technology, wireless networking, and distributed computing, visual sensor networks (VSNs) have emerged as an important part of Internet of Things (IOT), which can support many visual applications such as security surveillance and environmental monitoring.[1] Generally, similar to the other networks researched in Ma et al.,[2] VSNs contain a number of distributed visual sensor nodes, that is, camera nodes, which have appeared in many products such as mobile phones, drones, and robots. These visual sensor nodes are commonly used to collect, process, and transmit visual data including digital images and videos on VSNs.

However, as the number of camera nodes increases significantly, the amount of visual data increases explosively on VSNs, which can be managed by various cloud computing systems.[3–12] Among the visual data, there are a lot of near-duplicate images, which are usually defined as the images derived from the same digital source by various copy attacks or the ones captured from the same scene by different cameras and/or different conditions.[13–15] Two examples of near-duplicate images are shown in Figure 1. The existence of those

[1]Jiangsu Engineering Center of Network Monitoring & School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing, China
[2]Department of Electrical and Computer Engineering, University of Windsor, Windsor, ON, Canada

Corresponding author:
QM Jonathan Wu, Department of Electrical and Computer Engineering, University of Windsor, 401 Sunset Avenue, Windsor, ON N9B 3P4, Canada.
Email: jwu@uwindsor.ca

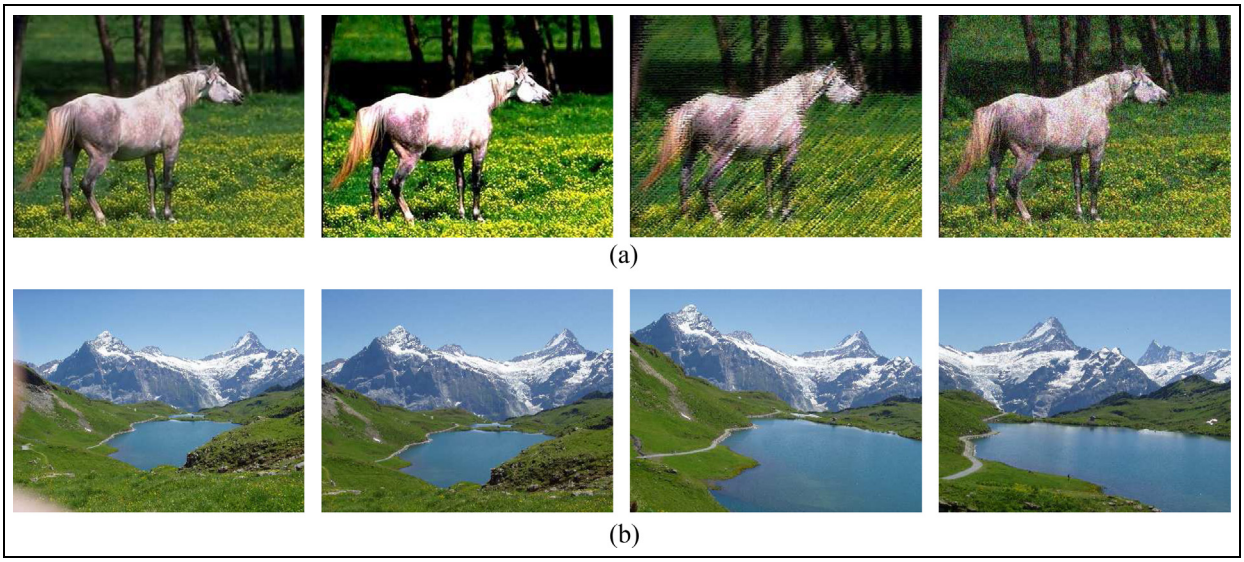

Figure 1. Two examples of near-duplicate images: (a) the original image and its near-duplicates generated by different copy attacks and (b) the original image and its near-duplicates captured by different conditions.

near-duplicate images causes a serious waste of limited storage, computing, and transmission resources of VSNs, and it also leads to a low users' experience since the users have to go through a large amount of near-duplicate images before finding the images that they are interested in. Therefore, near-duplicate image elimination has become an important issue of VSNs.

Since the simplified hardware structure is used in VSNs, the storage, computing, and transmission capabilities are generally limited, which makes the task of near-duplicate image elimination much more challenging. An ideal near-duplicate image elimination approach for VSNs is expected to have high efficiency while maintaining good accuracy. The existing approaches related to near-duplicate elimination are designed for traditional computer-based networks but not for VSNs, which require relatively low computation cost without a significant decrease in accuracy.

To meet the efficiency and accuracy requirements of VSNs, in this article, we propose a fast and accurate approach to efficiently and effectively eliminate near-duplicate images. It mainly consists of two key stages: near-duplicate clustering and seed image selection. The main contributions of our work are summarized as follows:

(1) Near-duplicate clustering is the first and basic step of near-duplicate elimination. Considering both efficiency and accuracy, we propose a near-duplicate clustering scheme based on a combination of global and local features to obtain near-duplicate clusters in a coarse-to-fine manner. First, near-duplicate images are clustered based on global feature to obtain initial clusters, that is, near-duplicate image groups (NIGs). Then a clustering optimization based on local feature is employed with a proposed "nearest expansion" strategy to optimize the clustering results.

(2) The seed image is expected to be the most representative image that has the highest relevance to the others in a NIG.[16] The existing seed selection methods usually choose the top-quality image of each NIG or the image nearest to the computed centroid of the group as seed image. However, those images are not the most representative images. By taking into account the contextual relevance among the images in each NIG, we propose a novel seed image selection method based on PageRank algorithm, which can accurately select the most representative images as seed images and remove the other redundant images to finish the near-duplicate elimination.

The remainder of this article is organized as follows. We review the related work of near-duplicate image elimination in section "Related work." In section "The proposed near-duplicate elimination approach for VSNs," the proposed near-duplicate image elimination approach is detailed. In section "Experiments," the experimental results are presented and discussed. Finally, we draw the conclusion of the proposed approach in section "Conclusion."

Related work

Due to the fact that near-duplicate elimination is quite related to near-duplicate detection and copy detection,

in this section, we review not only the existing near-duplicate image elimination approaches but also the near-duplicate image detection and image copy detection approaches.

Different from steganography techniques researched in Xia et al.,[17,18] which embed mark into digital source beforehand to indentify the copies, near-duplicate detection and copy detection focus on how to find the near-duplicates or copy versions of a given query (original) image. The existing near-duplicate detection and copy detection approaches can be roughly classified into two main categories: global feature–based approaches[19–21] and local feature–based approaches.[14,15,22–26] The global feature–based approaches extract single or several global features from the whole or nearly whole image region and then match those features between images to realize near-duplicate detection or copy detection. Typical global features include the feature based on discrete cosine transformation (DCT),[19] edge-based feature,[20] and color feature.[21,27] Although the global features are efficient to compute and match, they are sensitive to some relatively serious changes and attacks such as the viewpoint changing, capture position changing, and cropping. Consequently, the global feature–based approaches cannot effectively detect the images after those changes and attacks. To address this issue, many local feature–based near-duplicate and copy detection approaches have been proposed.[14,15,22–26] Different from global feature–based approaches, the local feature–based approaches extract hundreds of high-dimensional local features from each image, such as scale-invariant feature transform (SIFT),[28] principal component analysis (PCA)-SIFT,[26] and bag-of-visual-words (BOW).[15,22–25] However, the matching of those high-dimensional local features between images is quite time-consuming. Thus, local feature matching of those approaches is generally accompanied by the construction of index structure to improve the detection efficiency. Locality sensitive hashing (LSH)[29] is one of the most popular index structures for local feature matching.

Most of the recent approaches focus on the studies of near-duplicate detection and copy detection. Only a few approaches introduce some techniques for near-duplicate image elimination. Generally, the existing near-duplicate elimination approaches consist of two key stages: near-duplicate clustering and seed image selection. First, images are clustered into a number of NIGs. Then in each NIG, only the seed image, that is, the most representative image that has the highest relevance to the others,[16] is selected and reserved and the other images are removed.

Chen et al.[30] proposed an image deduplication approach. In this approach, the near-duplicate images are clustered into a number of NIGs by the global feature matching. Then by evaluating image quality considering several factors including size, resolution, and clarity, the top-quality image of each NIG is selected as seed image. However, in this approach, the near-duplicate clustering based on global feature will miss the near-duplicates captured by different viewpoints and positions, due to the week robustness of global feature. Moreover, since top-quality images do not mean that they have the highest relevance to the others, this approach is not accurate enough to select seed images, causing some seed images to be falsely removed. Liu et al.[31] proposed an approach to filter the near-duplicates of geo-tagged photographs. In this approach, the geographic tags embedded in the photographs are used to cluster photographs into initial photograph groups. For seed photograph selection, it simply chooses the photograph nearest to the centroid of each group as seed photograph, where the centroid is computed by the mean values of visual features of all the photographs in the group. However, the near-duplicate clustering cannot be implemented when the images do not have geographic tags. Moreover, choosing the photograph nearest to the group centroid as seed image is not accurate enough. Yang et al.[32] proposed a junk image filtering approach to remove the duplicate images and the irrelevant images. It randomly chooses one image as seed image from each image group, which will result in low accuracy for junk image filtering.

In conclusion, most of the existing near-duplicate elimination approaches only use one type of simple feature, which results in low accuracy for near-duplicate clustering. Also, they ignore the contextual relevance among the images in NIGs, and thus the seed images cannot be accurately selected. Consequently, the near-duplicate images cannot be efficiently and effectively eliminated. More importantly, all of them are originally designed for traditional computer-based networks but not for VSNs, which requires relatively low computation cost without a significant decrease in accuracy for near-duplicate elimination.

Because the global features are more efficient while the local features have better robustness, by combining both their advantages, we propose a near-duplicate clustering scheme based on a combination of global and local features for VSNs to obtain near-duplicate clusters in a coarse-to-fine way. On the other hand, since the seed images are the most representative images that have the highest relevance to the others, the contextual relevance among images in NIG needs to be explored for seed image selection. Inspired by Xie et al.[33] and Jing and Baluja,[34] which use graph-based models to capture the contextual relevance among images for image ranking, we attempt to capture the relevance among images in such a way. Thus, in each NIG we use PageRank algorithm to capture the visual relevance among the images to select seed image to realize near-duplicate elimination.

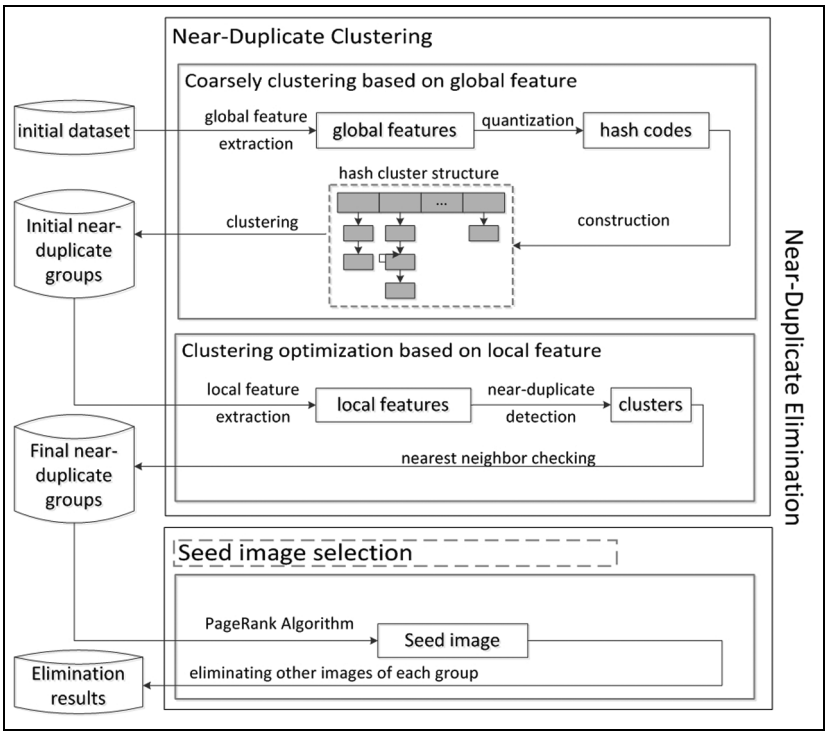

Figure 2. The framework of the proposed near-duplicate image elimination approach for VSNs.

The proposed near-duplicate elimination approach for VSNs

In this section, we introduce the proposed near-duplicate elimination approach for VSNs in detail. Figure 2 shows the framework of the proposed approach. Similar to the existing near-duplicate elimination approaches, our approach includes the following two major stages:

(1) By combining global and local features, near-duplicate images are clustered in a coarse-to-fine way to obtain the NIGs. First, quantized global features of images are used to efficiently cluster near-duplicates to obtain initial NIGs. Then a clustering optimization based on local feature is employed with a proposed nearest expansion strategy to further optimize clustering results.

(2) In each NIG, seed image is selected and reserved based on PageRank algorithm while the other images are removed. First, PageRank algorithm is used to capture contextual relevance among near-duplicate images in each NIG. Then the image that has the greatest share of the visual authorities after iterations on weight propagation is selected as the seed image of the group.

Near-duplicate clustering

At this stage, we describe the near-duplicate image clustering method in detail. As mentioned above, global features have high efficiency but low robustness, while local features have good robustness but high computational complexity. To obtain high efficiency while maintaining good accuracy, we combine the advantages of both global features and local features to cluster near-duplicates in a coarse-to-fine way.

First, images are divided into a number of equal-sized blocks, and the mean gray value of each block is calculated to generate the global features, that is, hash codes. Then according to the hash bins which the hash codes belong to, the dataset images can be divided into different groups, that is, initial groups. To refine the clustering results, we also employ a nearest expansion strategy under the presumption that near-duplicate images are located not only in the same hash bin but also may in nearby hash bins. Thus, we also search for the near-duplicates in the nearest neighbors of each hash bin to improve the recall of near-duplicate clustering. To this end, we extract robust local features from the images in each group. Then to improve the efficiency of local feature matching, an inverted index is constructed and a similarity table (called SimTable) recording the similarities between images is created offline. Finally, by local feature–based matching, we

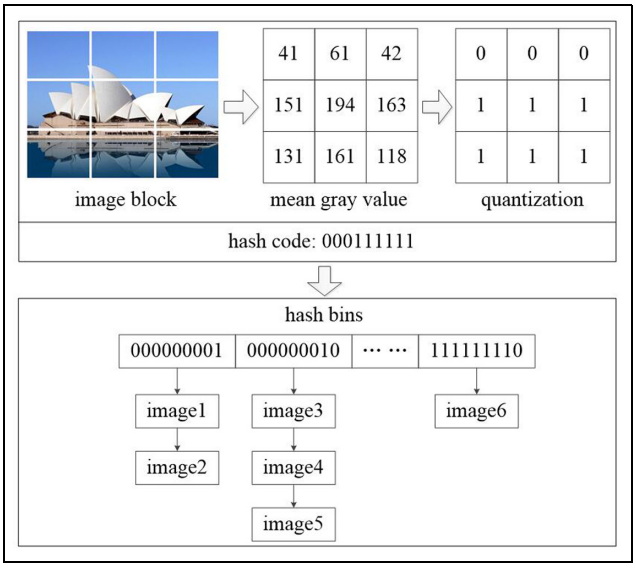

Figure 3. The procedure of coarsely clustering based on global feature.

search for near-duplicates in the nearest neighbors of each hash bin to obtain the final NIGs. After traversing all hash bins and searching in their nearest neighbors, all the final NIGs are generated.

Coarsely clustering based on global feature. First, we coarsely cluster images based on their global features. Global feature can reflect the global appearance of an image and has high efficiency to compute and match. As we know, most near-duplicate images usually have similar global content but some differences in local appearance. Therefore, after coarsely clustering based on global feature, it is likely that most near-duplicate images will be grouped together in the same group or be located in the nearby groups.

The extraction of global feature is detailed as follows. First, we divide an image I into 3×3 equal-sized blocks. Then, we calculate the mean gray value of each block to generate a vector $g = (g_1, g_2, \ldots, g_9)$ by

$$g_i = \frac{3 \times 3}{M \times N} \sum_{x,y \in K_i} f(x,y) \tag{1}$$

where $M \times N$ is the resolution of the image, $f(x,y)$ is the gray value of the pixel located in (x,y), and K_i is the ith image block. Next, vector g is quantified into a binary hash code $h = (k_1, k_2, \ldots, k_9)$ by

$$k_i = \begin{cases} 1, & \text{if } g_i \geq \bar{K} \\ 0, & \text{if } g_i < \bar{K} \end{cases} \quad 1 \leq i \leq 9 \tag{2}$$

where $\bar{K}$ is the mean gray value of image I and it is defined as

$$\bar{K} = \frac{1}{M \times N} \sum_{x=1}^{M} \sum_{y=1}^{N} f(x,y) \tag{3}$$

This procedure of coarsely clustering based on global feature is illustrated in Figure 3. However, although the global feature can be used to efficiently cluster the near-duplicates, the initial clustering results are not accurate enough and many near-duplicates that suffer relatively serious changes or attacks will be missed.

Clustering optimization based on local feature. As mentioned above, only using simple global feature for clustering is not accurate enough to cluster all kinds of near-duplicate images. Compared with global features, local features are generally more robust and stable.[35] Thus, we use local feature to further optimize the initial clustering results. From Bay et al.,[36] speed up robust feature (SURF) is one of the most popular local features, due to its relatively low dimensionality (64 dimensions) and high efficiency of extraction and matching. In this section, we adopt SURF to describe image local regions. Figure 4 illustrates an example of local feature matching results based on SURF after filtering the false matches. In this figure, each line represents a true matching.

Although SURF has relatively high efficiency, hundreds of SURF features will be extracted from each image. If those SURFs are directly used for near-duplicate detection, it will lead to an intensive computational cost. To improve efficiency, it is necessary to construct an inverted index. For a SURF denoted as $f_i = (f_{i,1}, f_{i,2}, \ldots, f_{i,n})$, where $f_{i,j}$ represents the jth dimension of the feature vector f_i and $n = 64$. We quantify f_i into a bit vector $v_i = (v_{i,1}, v_{i,2}, \ldots, v_{i,n})$ by

$$v_{i,j} = \begin{cases} 1, & \text{if } f_{i,j} \geq f_{i,j+1} \\ 0, & \text{if } f_{i,j} < f_{i,j+1} \end{cases} \quad 1 \leq j \leq n-1 \tag{4}$$

$$v_{i,n} = \begin{cases} 1, & \text{if } f_{i,n} \geq f_{i,mean} \\ 0, & \text{if } f_{i,n} < f_{i,mean} \end{cases} \tag{5}$$

where $f_{i,mean}$ is computed as follows

$$f_{i,mean} = \frac{1}{n} \sum_{j=1}^{n} f_{i,j} \tag{6}$$

LSH[29] is a famous technique of building inverted index file for the matching of high-dimensional features. In the technique, each feature is indexed according to its hash value to build the inverted index file, and the feature and its image ID are recorded in each entry of inverted index file. During the feature matching, the hash values of a given query feature are looked up from the entries of the inverted index file, and then an additional comparison between the original features is implemented to further confirm whether two features

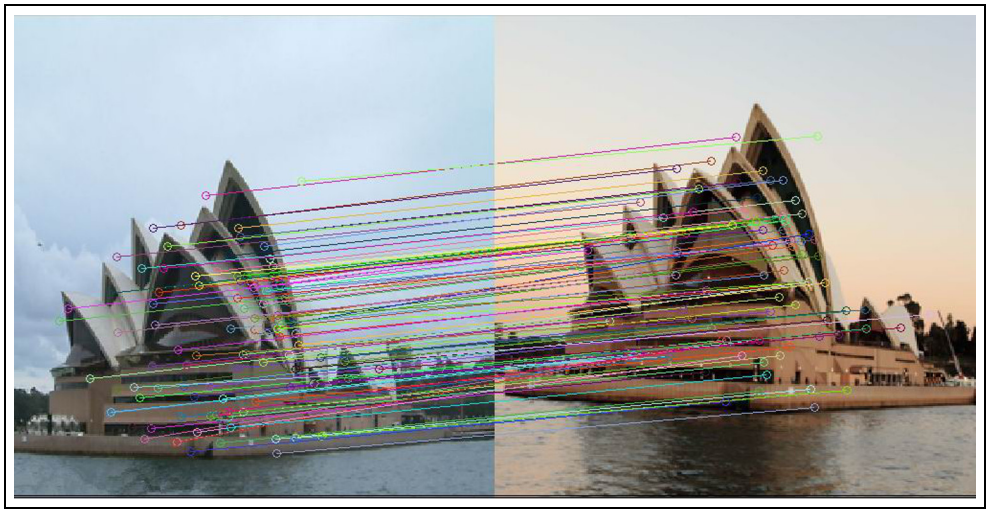

Figure 4. An example of near-duplicate image matching based on SURF.

Table 1. An example of the SimTable.

ID	Features	Near-duplicate IDs	Shared visual words	Similarity
1	159	7, 13, 14, 19	152, 112, 99, 124	0.95, 0.82, 0.75, 0.8
2	87	6, 11, 25	49, 84, 55	0.67, 0.91, 0.74
3	325	15, 27	296, 254	0.92, 0.83
4	93	8, 10, 12	69, 90, 74	0.78, 0.93, 0.81
5	247	16, 21, 22	222, 198, 156	0.91, 0.86, 0.69
…	…	…	…	…

are a match or not. Similar to LSH, we also index each SURF according to its vector to establish an inverted index structure, in which each entry records the ID of the image that the feature belongs to. Then we treat each entry as a visual word. If two SURF features fall into the same entry, they can be directly regarded as a match. From the above, it is clear that our method has some differences from the traditional LSH. Our method is much more efficient than LSH in the aspects of both space and time, because we do not need to store the original SURF features and compare them.

For two images I_i and I_j, suppose that the number of the visual words they shared is m, and m_i and m_j are the total number of features in I_i and I_j, respectively. Then the similarity S_{ij} between image I_i and I_j is defined as

$$S_{ij} = \frac{2m}{m_i + m_j} \tag{7}$$

When checking two images with their visual words, we use a threshold $\boldsymbol{\lambda}$ to determine whether two images are near-duplicates or not. If $S_{ij} \geq \boldsymbol{\lambda}$, image I_i and I_j are determined as near-duplicates of each other. Via the inverted index and threshold $\boldsymbol{\lambda}$, we can record some information about images with their near-duplicates into an offline similarity table, called as SimTable. The table includes the ID of each image, the number of its features, its near-duplicate IDs, the number of visual words it shares with its near-duplicates, and the similarity between the image and its near-duplicates. Table 1 shows an example of the SimTable.

In the coarsely clustering stage, near-duplicate images may also be located in neighbor hash bins. To improve the clustering recall, using the constructed inverted index file and the SimTable, we propose a nearest expansion strategy based on local feature to optimize the initial cluster results. The "nearest neighbor" strategy is detailed as follows:

For a hash bin h_i, in the proposed approach, its nearest neighbors are defined as those bins of which hash codes satisfy $h_j \oplus h_i \leq 3$, where $j \neq i$ and $\oplus$ means exclusive OR operation. Consequently, there will be in total $C_9^0 + C_9^1 + C_9^2 + C_9^3 = 130$ neighbors of each bin. Actually, not all of those hash bins really exist. For example, the codes, that is, "000000000" and "111111111," are impossible, thus the total number of traversed bins is generally much less than 130. The clustering optimization process based on this nearest neighbor strategy is illustrated in Algorithm 1, as shown in Figure 5.

Algorithm 1 Cluster optimization with local feature

Input: The hash bin set $\boldsymbol{H} = \{h_1, h_2, \ldots, h_m\}$; The SimTable.
Output: $\boldsymbol{C}$, the final NIG set.

```
Initialize a NIG set C = { φ };
while H ≠ { φ } do
    for each hash bin h_i do
        Randomly select one image I_i in the hash bin h_i;
        Initialize a NIG c_i and move I_i into c_i;
        Refer to the SimTable, record all near-duplicates of I_i
        into a set c̃_i ;
        for each I_j in c̃_i do
            h_j is the hash bin of I_j;
            if h_j ⊕ h_i ≤ 3 then
                Move I_j from h_j into c_i;
            end if
        end for
        if h_i = { φ } then
            Remove h_i from the hash set H;
        end if
        Add c_i into the NIG set C;
    end for
end while
```

Figure 5. The algorithm of clustering optimization based on local feature.

Seed image selection

After the near-duplicate clustering, we implement seed image selection to reserve the most representative image and remove the other redundant images in each NIG. First, we use PageRank to analyze the contextual relevance among images based on their visual similarity. Then the image with the highest relevance value will be selected as the seed image to represent the group.

The seed image is selected as the most representative image of a NIG, which should have the highest relevance to the others.[16] Thus, in each NIG, we select seed image utilizing contextual relevance among the images, rather than only relying on the image quality or the centroid of the group. Therefore, we need a proper link analysis algorithm to deeply explore the contextual relevance among images in each NIG. As we know, Hyperlink-Induced Topic Search (HITS) is a query-dependent algorithm while PageRank is query-independent, and the seed image selection in each NIG is independent of any specified queries. Thus, we use PageRank to find the one with the highest relevance as the seed image, that is, to find most representative image of each NIG.

PageRank is a ranking algorithm that can identify the importance of a webpage, and it is the only standard of "Google" to measure the importance of a website. In this algorithm, the more popular website will receive more links from other websites. It treats each website as a node, and the link weight between every two nodes is the click rate that one will arrive from one website to another. Each node collects weights from others and distributes its weight to others. After several iterations, each node will have a convergent authority for ranking. Similarly, the idea of PageRank can be cited to vote the most relevant image as the seed image in a NIG. We treat each image as a node and establish links between every two images of the group. Then we use PageRank to iteratively update the authority of each node with link weight propagation. Since the link weight is based on the visual similarity, the image that has the highest visual authority after iteration will be regarded as the image that has the highest relevance to the others, and thus it will be reserved as the "seed image" to represent its group.

In our image seed selection method, the link weights between image nodes in a NIG are assigned into an asymmetric matrix a, where $a[i, j]$ is the similarity from image I_j to image I_i (note that $a[i, j] \neq a[j, i]$ because the reference image is different). Inevitably, the aforementioned near-duplicate clustering method may lead to some false positives, and thus there will be some exceptional image nodes in NIGs. To avoid false elimination, we remove the exceptional image that has smallest similarity to the others beforehand. We iterate this process until the similarity between every two images is larger than the predefined threshold $\boldsymbol{\lambda}$. Then, each column of a is normalized so that the sum of weights of the column equals to 1. Next, we adopt PageRank to iteratively update the visual authority of each image node for R times. By the experiments, $R = 5$ can get a balance between convergence and time cost, and thus we choose $R = 5$ in our method. Finally, we obtain the seed image, which has the highest authority in each NIG. The algorithm of seed image selection is described in Figure 6.

Experiments

To evaluate the performance of our approach, we conduct the experiments on an artificial near-duplicate image dataset. We also implement intensive experiments on a real-world image dataset, in which images are downloaded from web image search engine. In addition, we compare our approach with several state-of-the-art approaches in the aspects of both accuracy and efficiency.

Datasets

The details of the two datasets adopted in our experiments are given as follows:

(1) *CopyDays3K.*[37] It is the INRIA Copydays dataset, which consists of 3000 near-duplicate images, including 150 original images and 2850 near-duplicates. There are 19 near-duplicates

Algorithm 2 Seed image selection

Input: A NIG c; The SimTable.
Output: s, the seed image.

```
int R = 0, num = the total number of images in c;
Initialize a 1×num node vector v for c, where the initial
value of each element is set as 1/num;
Construct a num×num link matrix a according to the SimTable,
where a[i, j] is the similarity between image I_i and I_j;
for each a[*, j] do
    Reserve top-K elements of the j-th column;
    Set other elemets as 0;
    Normalize the j-th column so that all elements sum to 1;
end for
while R<5 do
    Update v with the normalized a;
    R++;
end while
Compute the maximum value of v;
Return the corresponding image of the maximum value.
```

Figure 6. Seed selection algorithm based on PageRank.

of each original image in average. These near-duplicates are generated via some typical copy attacks such as scaling, JPEG compression, rotation, caption adding, and their combinations. In this dataset, the ground truth of images in each NIG is the original image and the corresponding near-duplicates, and the ground truth of the representative image is the original image.

(2) *Web3K*. This dataset contains 3000 images downloaded from the Internet. We search on the image search engine of Google by 10 random keywords, and then select the top 300 results for each keyword. As a result, there are 10 NIGs and 300 images for each NIG. It is necessary to note that all the images in this dataset are tagged by 10 volunteers, and the ground truth of images in each NIG and the corresponding representative images can be determined by their tagging results.

Reference works

In our experiments, we compare our approach with the following approaches:

(1) *Geo-tag*. This is a framework of near-duplicate elimination based on the geo-tagged photographs.[31] The approach proposes a hybrid index structure to store the useful image information during the stage of offline, so that the online retrieval has less computational complexity. Moreover, some strategies are proposed to quickly skip some unnecessary computations. Then seed photograph selection is determined by the distance between each photograph and its group centroid.

(2) *FAIDA*. This approach aims to eliminate redundant copies of duplicate images.[30] In this approach, duplicate image clustering is implemented using three global feature-based filters: perceptual hashing filter, gray block filter, and Haar wavelet filter. During the stage of duplicate elimination, a fuzzy logic reasoning system based on the image quality is adopted to determine whether each image needs to be reserved or removed.

(3) *K-way*. This approach is devoted to filter junk images on the Internet.[32] Junk images in this approach refer to irrelevant images and duplicate images. For image clustering, web near-duplicate image clusters are indentified by integrating bilingual image search results of the same keyword-based query. Then the duplications are removed under a coarse-to-fine structure.

Evaluation criteria

We use precision, recall, and $F1$-measure to evaluate the performances of different approaches for near-duplicate image clustering. The accuracy of near-duplicate image elimination is evaluated by redundancy elimination accuracy, denoted as *mRea*. The total time cost is used to evaluate the efficiency of our approach. The definitions of precision, recall, F-measure, and mRea are given as follows:

1. *Precision, recall, and F1-measure*. In this section, we use precision, recall, and $F1$-measure to evaluate the performance of our clustering method. The mean precision mp and the mean recall mr in our article are defined as

$$mp = \frac{1}{m}\sum_{i=1}^{m} p(c_i) \tag{8}$$

$$mr = \frac{1}{m}\sum_{i=1}^{m} r(c_i) \tag{9}$$

where m is the number of detected near-duplicate image clusters; $p(c_i)$ and $r(c_i)$ are the precision and recall of cluster c_i, which are defined by

$$p(c_i) = \frac{|\text{correctly detected near-duplicates in } c_i|}{|\text{all detected images in } c_i|} \tag{10}$$

$$r(c_i) = \frac{|\text{correctly detected near-duplicates in } c_i|}{|\text{ground truth near} - \text{duplicates in } c_i|} \tag{11}$$

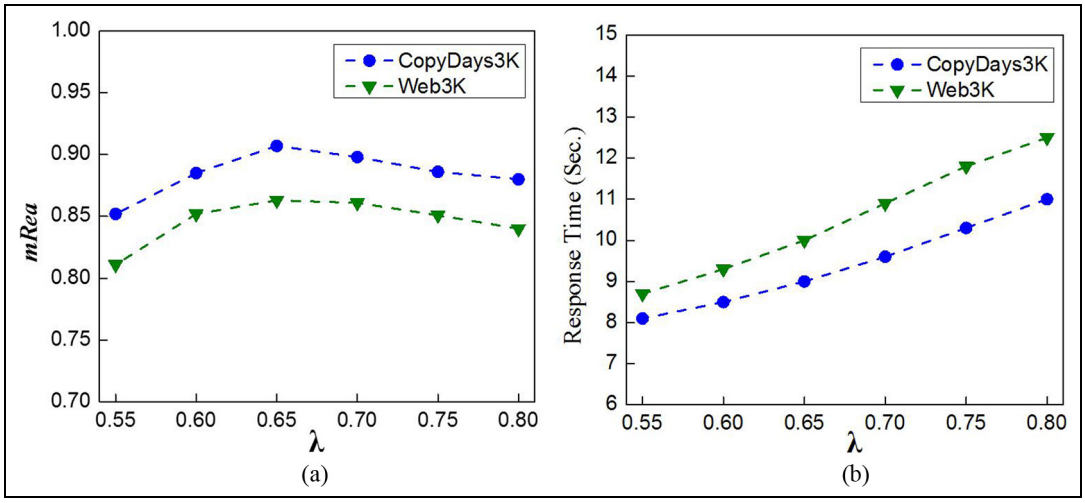

Figure 7. Effect of λ on the performance of near-duplicate image elimination: (a) *mRea* and (b) efficiency.

where |*| is the number of *. Then *F*-measure is defined by

$$F(a) = \frac{(\alpha^2 + 1)pr}{\alpha^2 p + r} \tag{12}$$

where α is a constant. When $\alpha = 1$, *F*-measure will change to the most common form: *F*1-measure, as

$$F1 = \frac{2 * pr}{p + r} \tag{13}$$

2. *Redundancy elimination accuracy*. The performance of our near-duplicate elimination method is evaluated by the mean redundancy elimination accuracy, denoted as *mRea*, which is defined by

$$mRea = \frac{1}{m}\sum_{i=1}^{m} Rea(c_i) \tag{14}$$

where m is the number of detected near-duplicate image clusters; $Rea(c_i)$ is the redundancy elimination accuracy of cluster c_i, and it is defined as

$$Rea(c_i) = \sum_{r=1}^{|c_i|} \frac{1}{r} \times gt(r) \tag{15}$$

where r (from 1 to $|c_i|$) represents the ranking results of cluster c_i; $gt(r)$ is a binary function. For example, the real representative image in cluster c_i is ranked at $r = 2$, and thus $gt(2) = 1$ and $gt(r) = 0$, when $r \neq 2$. Thus, when the real representative image is ranked at the top of NIG c_i, $Rea(c_i)$ has the largest value.

Parameter evaluation

The similarity threshold **λ** has some effects on the performance of near-duplicate image elimination of our approach. If **λ** is too small, some irrelevance images will be falsely eliminated, resulting in low *mRea*. Conversely, if **λ** is too large, some real near-duplicate images will not be eliminated effectively, also resulting in a low *mRea*. Figure 7(a) illustrates the *mRea* values with different **λ** on the two datasets. From this figure, it is clear that when **λ** is too small or too large, the accuracy of our approach degrades. On the other hand, **λ** also plays an important role in the total time cost of the proposed approach, since the size of NIG is determined by **λ**. A larger **λ** will result in an increase in the total cost time, because more near-duplicates in NIGs need to be processed. Figure 7(b) shows the effect of **λ** on the total response time. The increase in **λ** results in an increase in the total time cost. From Figure 7(a) and (b), we can see that when **λ** equals 0.65, a good tradeoff between accuracy and efficiency can be obtained. Thus, we set **λ** as 0.65 in the following experiments.

Experimental results and analysis

We first evaluate and compare the effectiveness of our approach and the other three approaches for near-duplicate clustering. Then we evaluate and compare the accuracy of those approaches for redundancy elimination. In addition, we use total time cost and memory usage to evaluate and compare the efficiency of those approaches:

(1) *The effectiveness of near-duplicate clustering*. We evaluate and compare the effectiveness of our clustering method and the three aforementioned methods. The precision, recall, and *F*1-measures of those methods on the two datasets are listed in Tables 2 and 3, respectively. From the two tables, it can be observed that our method outperforms the other three methods

Table 2. Comparison of clustering accuracy on dataset Copydays3K.

Method	Precision	Recall	*F*1-measure
Geo-tag	0.9005	0.8961	0.8983
FAIDA	0.8796	0.6974	0.7780
K-way	0.9613	0.9307	0.9458
Our method	**0.9878**	**0.9569**	**0.9721**

The highest values are highlighted in bold.

Table 3. Comparison of clustering accuracy on dataset Web3K.

Method	Precision	Recall	*F*1-measure
Geo-tag	0.8251	0.7988	0.8117
FAIDA	0.8274	0.6563	0.7320
K-way	0.9126	**0.9075**	0.9100
Our method	**0.9331**	0.8972	**0.9148**

The highest values are highlighted in bold.

both on Copydays3K and Web3K. Moreover, it is clear that the results of all of those methods on Copydays3K are better than those on Web3K, because the real-world images usually suffer more serious changes and attacks.

(2) *The effectiveness of redundancy elimination accuracy*. the results of those approaches for near-duplicate clustering are illustrated in Figure 8(a) and (b). From the two figures, we can observe that our clustering method has the highest redundancy elimination accuracy, and thus the final selected seed images will be most close to the ground truth of representative images. It is also clear that the *mRea* values of the latter three methods on Copydays3K are better than on Web3K. That is because artificial near-duplicate images are more regular than real-world near-duplicate images, and thus they are easier to be identified.

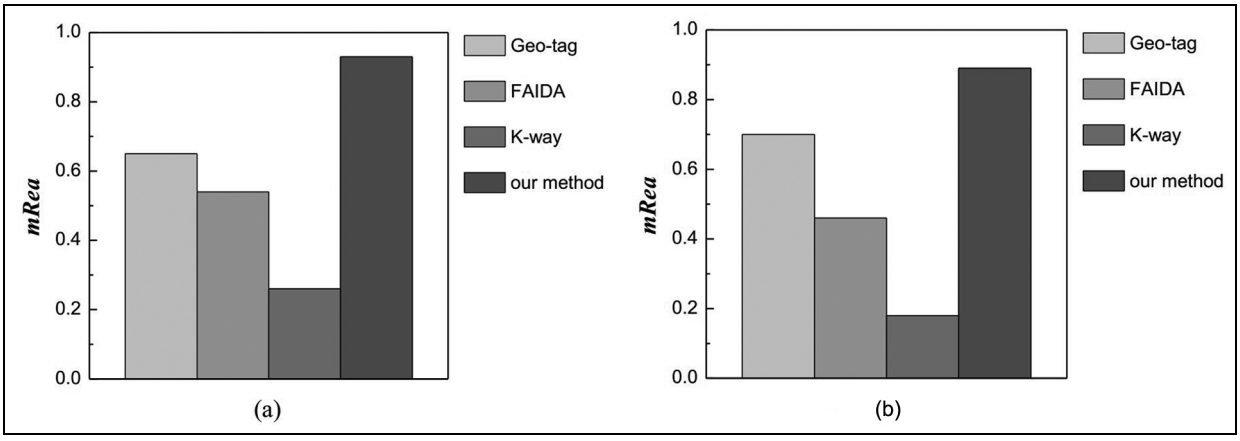

Figure 8. Comparison of the mean redundancy elimination accuracy: (a) *mRea* of Copydays3K and (b) *mRea* of Web3K.

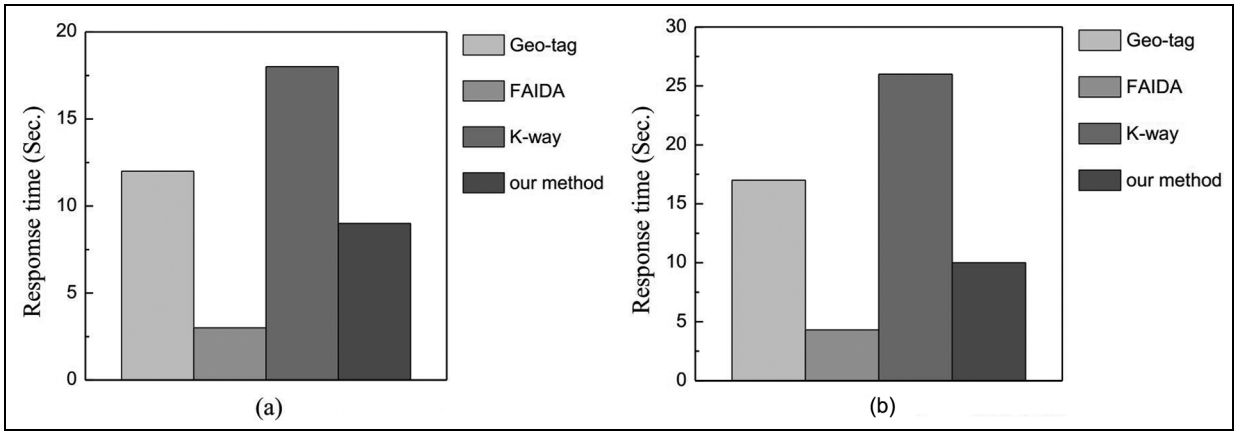

Figure 9. Comparison of total time cost: (a) *mRea* of Copydays3K and (b) *mRea* of Web3K.

Table 4. Comparison of efficiency of memory usages.

	Memory usage (MB)	
	CopyDays3K	Web3K
Geo-tag	0.63	0.75
FAIDA	0.01	0.01
K-way	0.41	0.51
Our approach	0.37	0.46

(3) *Time cost and memory usage*. We use total time cost and memory usage to evaluate and compare the efficiency of our approach and the other three approaches. Figure 9 shows the comparison results of the total time costs on the two datasets. From the figure, we can observe that the K-way has the highest time cost because it directly adopts pairwise image matching based on local feature. The efficiency of our approach is slightly lower than that of FAIDA, because FAIDA is only based on global feature and thus it can achieve rapid near-duplicate identification. Moreover, the efficiency of our approach is higher than those of the other two approaches.

As memory usage is another significant factor of efficiency, we also compare the memory usages of different approaches. Table 4 shows the comparison results of memory usages of the four approaches. In this table, the memory usage of FAIDA is the lowest. Our approach is second to FAIDA and outperforms the other two approaches. Geo-tag has the highest memory usage, due to the extensive memory usage during the building of the hybrid index structure. Our approach has less memory usage than the other two approaches, mainly because we do not store the original global and local features in the index files. Memory usages of those approaches on the Web3K dataset are always higher than those on the CopyDays3K dataset, because NIGs in the former dataset contain more near-duplicates than the latter one.

Conclusion

In this article, we present a fast and accurate approach for near-duplicate image elimination for VSNs. At the stage of near-duplicate clustering, we combine the advantages of global feature and local feature to efficiently cluster the NIGs. Moreover, we adopt PageRank algorithm to select seed images. This PageRank algorithm can effectively capture contextual relevance between near-duplicate images based on their visual similarity, and thus the most representative image in each NIG will be accurately selected and reserved. The experiments conducted on artificial near-duplicate image dataset and real-world near-duplicate image dataset demonstrate that our approach can achieve desirable performance in the aspects of both efficiency and accuracy for VSNs.

Acknowledgements

The authors would like to thank Prof. Ching-Nung Yang from National Dong Hwa University for reviewing this article and giving some valuable advices.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported, in part, by the National Natural Science Foundation of China under grant numbers 61602253, U1536206, 61232016, U1405254, 61373133, 61502242, and 61572258; in part, by the Jiangsu Basic Research Programs-Natural Science Foundation under grant numbers BK20150925 and BK20151530; in part, by the Priority Academic Program Development of Jiangsu Higher Education Institutions (PAPD) fund; in part, by the Collaborative Innovation Center of Atmospheric Environment and Equipment Technology (CICAEET) fund; and, in part, by College Students Practice Innovation Training Program under grant number 201610300022, China.

References

1. Zhang Y, Sun X and Wang B. Efficient algorithm for K-barrier coverage based on integer linear programming. *China Commun* 2016; 13(7): 16–23.
2. Ma T, Zhou J, Tang M, et al. Social network and tag sources based augmenting collaborative recommender system. *IEICE T Inf Syst* 2015; E98.D(4): 902–910.
3. Xia Z, Zhang L, Qin Z, et al. A privacy-preserving and copy-deterrence content-based image retrieval scheme in cloud computing. *IEEE T Inf Foren Sec* 2016; 11(11): 2594–2608.
4. Fu Z, Guan C, Sun X, et al. Towards efficient multi-keyword fuzzy search over encrypted outsourced data with accuracy improvement. *IEEE T Inf Foren Sec* 2016; 11(12): 2706–2716.
5. Qi L, Xu X, Zhang X, et al. Structural balance theory-based e-commerce recommendation over big rating data. *IEEE T Big Data*. Epub ahead of print 16 September 2016. DOI: 10.1109/TBDATA.2016.2602849.
6. Fu Z, Shu J, Sun X, et al. Enabling personalized search over encrypted outsourced data with efficiency improvement. *IEEE T Parall Distr* 2016; 27: 2546–2559. DOI: 10.1109/TPDS.2015.2506573.
7. Xia Z, Wang X, Sun X, et al. A secure and dynamic multi-keyword ranked search scheme over encrypted cloud data. *IEEE T Parall Distr* 2016; 27: 340–352.

8. Fu Z, Sun X, Liu Q, et al. Achieving efficient cloud search services: multi-keyword ranked search over encrypted cloud data supporting parallel computing. *IEICE T Commun* 2015; 98(1): 190–200.
9. Qi L, Dou W, Hu C, et al. A context-aware service evaluation approach over big data for cloud applications. *IEEE Transactions on Cloud Computing*. Epub ahead of print 23 December 2015. DOI: 10.1109/TCC.2015.2511764.
10. Kong Y, Zhang M and Ye D. A belief propagation-based method for task allocation in open and dynamic cloud environments. *Knowl: Based Syst* 2017; 115(1): 123–132.
11. Liu Q, Cai W, Shen J, et al. A speculative approach to spatial-temporal efficiency with multi-objective optimization in a heterogeneous cloud environment. *Secur Commun Netw* 2016; 9(17): 4002–4012.
12. Qi L, Dou W and Chen J. Weighted PCA-based service selection method for multimedia services in cloud environment. *Computing* 2016; 98(1): 195–214.
13. Zhou Z, Yang C, Chen B, et al. Effective and efficient image copy detection with resistance to arbitrary rotation. *IEICE T Inf Syst* 2016; E99D(6): 1531–1540.
14. Li J, Li XL, Yang B, et al. Segmentation-based image copy-move forgery detection scheme. *IEEE T Inf Foren Sec* 2015; 10(3): 507–518.
15. Zhou Z, Wu QMJ, Yang C, et al. Effective and efficient global context verification for image copy detection. *IEEE T Inf Foren Sec* 2017; 12(1): 48–63.
16. Li J, Qian X, Li Q, et al. Mining near duplicate image groups. *Multimed Tools Appl* 2015; 74(2): 655–669.
17. Xia Z, Wang X, Sun X, et al. Steganalysis of LSB matching using differences between nonadjacent pixels. *Multimed Tools Appl* 2016; 75(4): 1947–1962.
18. Xia Z, Wang X, Sun X, et al. Steganalysis of least significant bit matching using multi-order differences. *Secur Commun Netw* 2014; 7(8): 1283–1291.
19. Kim C. Content-based image copy detection. *Signal Process: Image* 2003; 18(3): 169–184.
20. Lin C and Wang S. An edge-based image copy detection scheme. *Fundam Inform* 2008; 83(3): 299–318.
21. Yue J, Li Z, Liu L, et al. Content-based image retrieval using color and texture fused features. *Math Comput Model* 2011; 54(3–4): 1121–1127.
22. Wan-Lei Z and Chong-Wah N. Scale-rotation invariant pattern entropy for keypoint-based near-duplicate detection. *IEEE Trans Image Process* 2009; 18: 412–423.
23. Xu D, Cham TJ, Yan S, et al. Near duplicate identification with spatially aligned pyramid matching. *IEEE T Circ Syst Vid* 2010; 20(8): 1068–1079.
24. Zhou W, Li H, Lu Y, et al. SIFT match verification by geometric coding for large-scale partial-duplicate web image search. *ACM T Multim Comput* 2013; 9(1): 4.
25. Yao J, Yang B and Zhu Q. Near-duplicate image retrieval based on contextual descriptor. *IEEE Signal Proc Let* 2015; 22(9): 1404–1408.
26. Ke Y and Sukthankar R. PCA-SIFT: a more distinctive representation for local image descriptors. In: *Proceedings of the IEEE computer society conference on computer vision and pattern recognition*, Washington, DC, 27 June–2 July 2004, pp.506–513. Washington, DC: IEEE.
27. Chen B, Shu H, Coatrieux G, et al. Color image analysis by quaternion-type moments. *J Math Imaging Vis* 2014; 51(1): 124–144.
28. Lowe D. Distinctive image features from scale-invariant keypoints. *Int J Comput Vision* 2004; 60(2): 91–110.
29. Indyk P and Motwani R. Approximate nearest neighbors: towards removing the curse of dimensionality. In: *Proceedings of ACM symposium on theory of computing*, Dallas, TX, 24–26 May 1998, pp.604–613. New York, NY: ACM.
30. Chen M, Wang S, Yun X, et al. FAIDA: a fast and accurate image deduplication approach. *J Comput Res Dev* 2013; 50(1): 101–110.
31. Liu J, Huang Z, Cheng H, et al. Presenting diverse location views with real-time near-duplicate photo elimination. In: *Proceedings of the IEEE international conference on data engineering*, Brisbane, QLD, Australia, 8–12 April 2013, pp.505–516. New York: IEEE.
32. Yang C, Peng J, Feng X, et al. Integrating bilingual search results for automatic junk image filtering. *Multimed Tools Appl* 2014; 70(2): 661–688.
33. Xie LX, Tian Q, Zhou WA, et al. Fast and accurate near-duplicate image search with affinity propagation on the ImageWeb. *Comput Vis Image Und* 2014; 124: 31–41.
34. Jing Y and Baluja S. VisualRank: applying PageRank to large-scale image search. *IEEE T Pattern Anal* 2008; 30(11): 1877–1890.
35. Mikolajczyk K, Tuytelaars T, Schmid C, et al. A comparison of affine region detectors. *Int J Comput Vision* 2005; 65(1–2): 43–72.
36. Bay H, Ess A, Tuytelaars T, et al. Speeded-up robust features (SURF). *Comput Vis Image Und* 2008; 110(3): 346–359.
37. Copydays. http://lear.inrialpes.fr/~jegou/data.php, 2008.