Multimed Tools Appl (2016) 75:2435–2452
DOI 10.1007/s11042-015-2472-1

Efficient near-duplicate image detection with a local-based binary representation

Fudong Nian · Teng Li · Xinyu Wu · Qingwei Gao · Feifeng Li

Received: 29 September 2014 / Revised: 1 December 2014 / Accepted: 15 January 2015 /
Published online: 30 January 2015

Abstract Efficient near-duplicate image detection is important for several applications that feature extraction and matching need to be taken online. Most image representations targeting at conventional image retrieval problems are either computationally expensive to extract and match, or limited in robustness. Aiming at this problem, in this paper, we propose an effective and efficient local-based representation method to encode an image as a binary vector, which is called Local-based Binary Representation (LBR). Local regions are extracted densely from the image, and each region is converted to a simple and effective feature describing its texture. A statistical histogram can be calculated over all the local features, and then it is encoded to a binary vector as the holistic image representation. The proposed binary representation jointly utilizes the local region texture and global visual distribution of the image, based on which a similarity measure can be applied to detect near-duplicate image effectively. The binary encoding scheme can not only greatly speed up the online computation, but also reduce memory cost in real applications. In experiments the precision and recall, as well as computational time of the proposed method are compared with other state-of-the-art image representations and LBR shows clear advantages on online near-duplicate image detection and video keyframe detection tasks.

Keywords Near-duplicate image detection · Binary representation · Similarity matching

F. Nian · T. Li (✉) · Q. Gao
College of Electrical Engineering and Automation, Anhui Univeristy,
HeFei, Anhui Province, 230601, China
e-mail: tenglwy@gmail.com

X. Wu
Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China

F. Li
Huainan Union University, Huainan, Anhui Province, 232001, China

1 Introduction

With the rapid developments of internet and image processing software, a number of online images are increasing day-by-day, and images can be altered more and more easily during transmission (e.g., resizing, cropping, changing luma and chroma, adding text and watermark). This brings the challenge of matching slightly altered image copies to their originals, which is termed "near-duplicate image detection" [9].

In this paper, we focus on the online near-duplicate images detection. Figure 1 gives an exemplar application scenario of our interested problem. Figure 1a shows some resulting images from Google image search engine using query keywords "Golden Gate Bridge". Quite obviously, it displays some images of the same content in one page, which may result in search experience reduction. To improve the user experience, near-duplicate images can be removed to yield more diverse results as shown in Fig. 1b. In this application both feature extraction and similarity matching are required to be conducted online, with high efficiency. One second delay would defect user experience much. Other similar cases include online advertisements detection for TV program, and etc. These applications drive us to develop a feature representation that is efficient and compact enough, to detect near-duplicate images with high speed and low memory cost.

Near-duplicate image detection has played an important role in a lot of applications including storage optimization, copyright enforcement, video copy detection, and improving image search engine [29, 30]. For a long time though, many techniques have been proposed addressing this problem [1, 2]. As a traditional solution, watermarking-based method inserts information into the image prior to distribution, for later extraction for identification. However, it needs to alter the image content, and it is not applicable for images from different distributors. Recent developments in image retrieval field have driven many methods being proposed purely based on visual content of the image. The content-based methods do not need to add any watermark to the image but the image itself is used as the watermark.

State-of-the-art near-duplicate image detection systems mostly rely on the Bag-of-Word (BoW) representation [9, 15, 34]. The BoW is a local feature based method, which usually includes the following basic steps: local feature detection and description, e.g., scale-invariant feature transform (SIFT) [17] and speed-up robust features (SURF) [3],

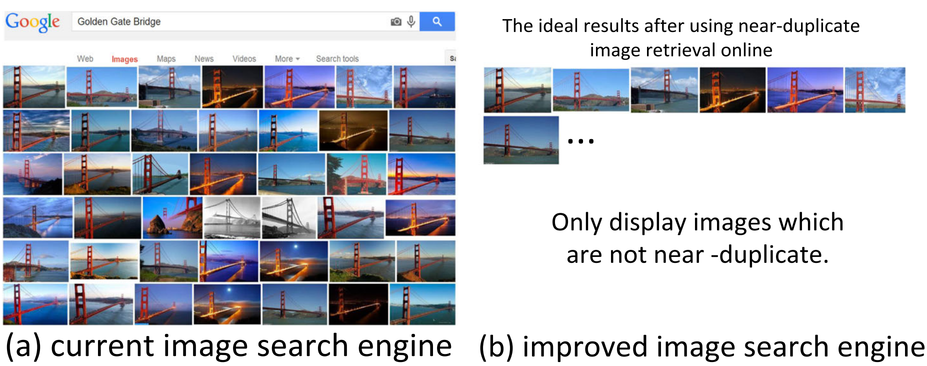


Fig. 1 An exemplar application of our target online near-duplicate image detection: **a** results from Google image search by query keywords "Golden Gate Bridge"; **b** the ideal results after online removing near-duplicate images

codebook generation, and image representation calculation. Comparing to early global-based image representation such as color histogram [26], the BoW shows more robustness on several variations, and achieves success in several related tasks [35, 36]. However, for our interesting online near-duplicate task, the codebook training process in BoW requires a large amount of training data. Besides, the BoW representation requires substantial computation for local feature detection, and it actually neglects the geometric relationships among local features. Some previous works have been done to improve the efficiency of BoW representation. In [13, 23], the binary descriptors of oriented FAST, rotated BRIEF (ORB) and binary robust invariant scalable keypoints (BRISK) have been used as alternatives to SIFT or SURF in BoW. The min-Hash and tf-idf algorithms have also been introduced in [6]; However, these methods are not designed for online tasks, and they are computationally expensive to extract and match in our interested applications.

In this paper, a highly efficient and effective local-based binary image representation is proposed for online near-duplicate image detection. The target near-duplicate images are slightly different on color scheme, size, contrast, luminance, translation, storage format, or have slightly change in the layout or background, excised portions, shift and text-added. Figure 2 shows some samples of near-duplicate images that the proposed method is to handle.

We densely sample the local regions from the image, and convert them to a block-based local binary pattern (LBP) alike features. The local feature extraction step is fast and the region description is discriminative. Then all of these local features are counted together to form a histogram based on specific rules. Finally, the histogram is encoded to a 64-bit binary vector to further improve the efficiency for matching and reducing the online memory cost. Previous works show that binary coding can be efficient and robust for fast visual matching and retrieval [36]. This proposed Local-based Binary Representation (LBR) jointly utilizes the local region texture and global visual distribution of the image, based on which online near-duplicate image detection can be achieved successfully.

The contributions of this paper can be summarized as the follows.

- We propose a novel local-based image representation especially for online near-duplicate image detection tasks. The proposed representation is efficient to compute as well as robust in performance. It is also highly compact and it does not require any training phase.
- The local region of image is described with a block-based binary pattern, which is computationally efficient and discriminative.
- The holistic histogram representation is encoded to a binary vector. It is not only highly compact for memory consumption and computation speed, but also robust to imaging variations.

Fig. 2 Some samples of near-duplicate images that the proposed method is to handle: the *left* two images have serious watermarking and lighting intensity change, the *right* two images have local text insertion

The remainder of this paper is organized as follows. Section 2 discusses some related works in the field of near-duplicate image detection. In Section 3, we present the details of the proposed image representation, as well as its application in near-duplicate image detection. In Section 4, we report our experimental results and make comparisons with state-of-the-art methods [11, 26, 27]. We also conduct an experiment on the application of video keyframe detection. Finally, we draw our conclusions in Section 5.

2 Related work

There have been many image representation techniques proposed for near-duplicate image detection and related tasks [19]. They usually fall into two categories according to the orientation : 1) global-based representation, and 2) local-based representation. Comparatively, the global-based representations are generally more efficient to compute and more compact to storage. However, they are less robust to image transformations, such as occlusion, cropping, etc. Extensive comparison in near duplicate detection application [28] has shown that local-based image representations are in general superior to global-based image representations in terms of accuracy, while at the cost of higher complexity of computing the features and more space for storing them. In the following we briefly review the related works in terms of these two categories.

2.1 Global-based representation

Early methods usually represent images globally. Color histogram maybe the most widely used global feature. Swain et al. [26] first proposed color histogram intersection as a similarity measure to index images. They demonstrated that the color histogram is faster to compute comparing to other invariants. It is also relatively invariant to translation, rotation, and view angle changes. However, the color histogram ignores the shape and the texture. As a result, different images with similar color components may be indistinguishable based solely on color histogram. Another problem is that color histogram is highly sensitive to image noise such as lighting intensity changes and quantization errors.

In [18], Meng et al. employed global image representation, e.g. color/texture feature, for duplicate detection in a large data collection. Although their system is able to handle large-scale database, it is not fully optimized for near-duplicate detection problem. A global descriptor based on average hash algorithm is introduced in [11]. It has been used in some content-based image retrieval systems. The computing steps are as follows: reduce size, reduce color, average the colors, compute the bits, and construct the hash. While the average hash is quick and easy, it may cause miss matches if there is a gamma correction or a color histogram is applied to the image.

Discrete cosine transform (DCT) [10] and discrete wavelet transform (DWT) [4] have also been adopted to represent images globally. In [10], the 8×8 2D DCT transform is applied and the coefficients are then ranked by the AC magnitudes. Near duplications are detected by comparing the L_1 distance between the rank matrices. In [4], for each of the three color channels, the resulting low frequency coefficients of DWT are used as color filter, while the horizontal, vertical and diagonal high frequency coefficients are separately summed and thresholded and used as shape filter. Duplicated images are detected by first ensuring that the values of the shape filter are identical and then the L_2 distance is compared among the color filters.

2.2 Local-based representation

Extracting local features to represent multimedia data for content based image and video search has gained more attention [12, 25]. BoW representation is currently a popular scheme for its good performance and flexibility [9, 15, 34].

Considering that the local feature extraction step of BoW requires much time and storage costs, some methods have been developed to improve the efficiency. In [6], the authors introduced the min-Hash and tf-idf algorithms that can be used with the Haar wavelet (or the histogram). They use a visual vocabulary of quantized local feature descriptors and exploit enhanced min-Hash techniques for retrieval. Zhang et al. [36] proposed a method by extracting the ultrashort binary descriptor (USB) and a compact auxiliary spatial feature from each keypoint detected in images. These methods are more efficient than the BoW model to retrieve images from an pre-built index. However, the complex computing steps are time-consuming for online matching.

Ojala et al. [21] introduced the feature extraction method of LBP. In [27], the image is divided into equal-size blocks from which LBPs are extracted. So they form a lot of local histograms and use a dissimilarity measure based on the Minkowski distances to compare these histograms. Their experiments demonstrate the accuracy of the method. However, the computing process of the feature histogram and the measurement method are complex and time-consuming. Yang et al. [33] proposed Local-Difference-Pattern (LDP) as the unified feature to describe both images and videos. The extraction and encoding of LDP are rather complex for online near-duplicate images detection.

The authors of [16] proposed a computational geometry approach for video duplicate detection. In their proposed method, video clips are modeled as curves in a scaled appearance space. Curves are simplified through spline modeling and distortion threshold controlled point selection. Duplicate detection is therefore achieved by a fast line segment matching algorithm. However, compact curve features are very effective for video duplicate detection while they are not sufficient to near-duplicate images detection (information in a single image is very sparse when compared with video clips).

A method to combine global and local features has been proposed in [31]. Color histograms are first used to detect near-duplicate videos with high confidence and filter out these dissimilar videos. Then a local-based method is utilized to identify the uncertain videos. This hierarchical method is infeasible for our interested online near-duplicate images detection, since it requires extracting both global and local features.

This paper focuses on online near-duplicate image detection, and it is impossible to build up an index at first. In this paper, we propose an effective new representation which is based on local features, with low cost of computation and memory for online near-duplicate images detection.

3 The method

The following parts of this section will detail the extracting process of the proposed LBR representation, and show how to apply it to online near-duplicate image detection.

3.1 The LBR extraction

The flowchart of extracting process of the proposed LBR from a color image is shown in Fig. 3. The basic LBR is designed for a single channel image. For the color image, we first split it into three channels and compute local descriptors based on the dense sampled blocks on each channel; then we form a histogram based on these local region features computed on each channel; finally the histograms of the three channels are concatenated to form a representation for the image. The steps for extracting LBR from a single channel image will be detailed as follows.

To obtain local features, we scan the whole image with a sliding window. The window crops the image to local regions and our local features are obtained from these regions. In general, our local feature is a kind of texture feature. Previous works show that LBP has excellent performance in representing image local feature [21, 27]. However, the traditional pixel-based LBP extraction takes much time for the whole image.

$$blockcodes = \begin{cases} 0, & (A < E) \\ 0, & (B < E) \\ 0, & (C < E) \\ 0, & (D < E) \\ 1, & (F > E) \\ 1, & (G > E) \\ 0, & (H < E) \\ 0, & (I < E) \end{cases} \tag{1}$$

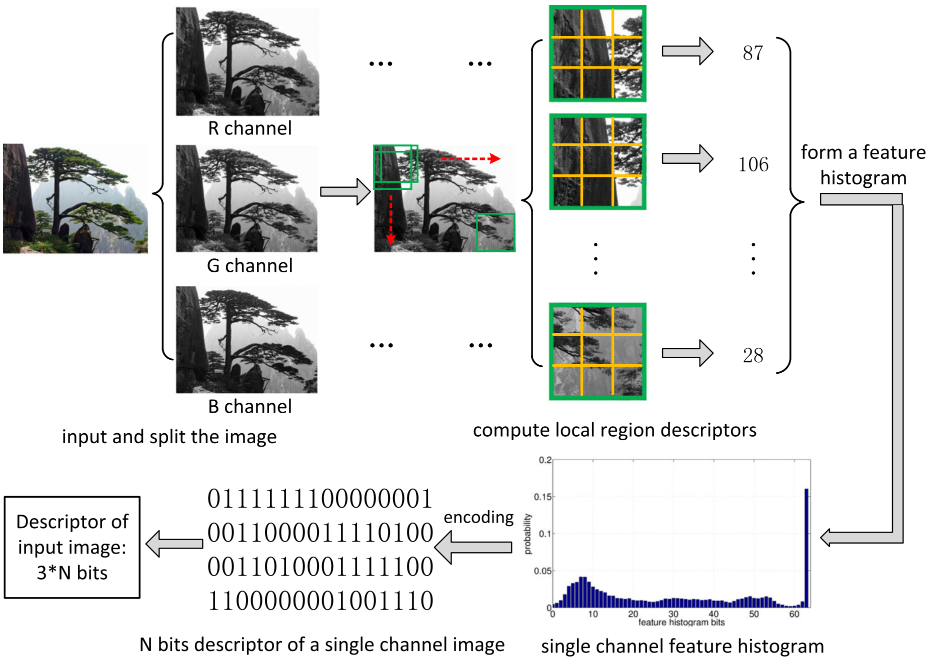


Fig. 3 The flowchart of the LBR extracted process from a color image

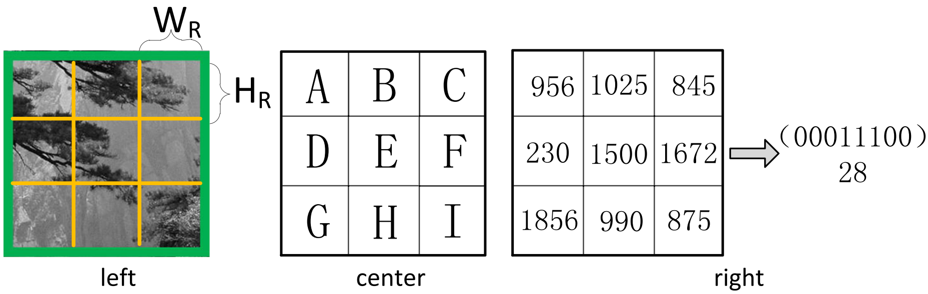


Fig. 4 Process of local region description

In order to resolve this problem, we adopt the block-based LBP. As shown in Fig. 4(left), a region is divided into nine blocks and each block have the same size, assuming that the block's width and height are W_R and H_R respectively. Then we name these nine blocks from "A" to "I", the block in the center of the region is named "E". The sketch map is shown in Fig. 4 (center). We calculate the sum of pixels' value in each block, named from $Total(A)$ to $Total(I)$. These values will be encoded according to Algorithm 1.

Therefore, we will obtain an eight-dimension binary features for a region according to the above rules. The binary features are converted to a decimal number as the local representation. An example is shown in Fig. 4. Assuming the value in each region represents the sum of pixels, the binary codes can be computed as (1). The corresponding decimal number is 12 for this region, and we use it as a local feature. In order to further accelerate the computing speed, skills such as integral image [7] can be applied, so that we only need to query the bottom right corner's value from the integral image in each operation.

Algorithm 1 Algorithm for extracting local region features.

Initialize 8 binary bits $bitsets = \{00000000\}$;
for $iter$ = A to I(except E) **do**
    if $Total(iter) \geq Total(E)$ **then**
        $bitsetiter = 1$;
    end if
end for

Next we extract many local features with the method introduced above, based on which to represent the whole image. The dense sampling strategy has proved to be effective for local regions sampling. Therefore, we move the slide window at a fixed horizontal step x and vertical step y to scan the whole image to get all local regions' feature, as illustrated in Fig. 3. It is similar to the image spatial domain filtering and division method introduced in [27].

From the above steps we will get a decimal number for each local region, and a number of local features for an image. So all these local feature numbers are counted and form a histogram. Because the local feature for each region is a 8-bit binary number, so this histogram has 256 bins. This histogram is still of high dimension, and it is sensitive to changes in local feature. For example, some bins may change a lot if only a small part of local features change. Therefore, in order to improve the robustness and efficiency, we first reduce the dimension of the feature histogram. Assuming that the reduced dimension of the histogram is N, we then normalize this N-dimension histogram and denote it as H.

Algorithm 2 Algorithm for encoding the feature histogram.

```
Initialize N binary bits bitsets = {00000000...};
for iter = 1 to N-1 do
    if H(iter − 1) ≤ H(iter) then
        bitset[N − 1 − iter] = 1;
    end if
end for
```

Finally, the feature histogram H is encoded following Algorithm 2. We compare the adjacent bins of the histogram H and output a binary value accordingly. The binary vector calculated from the histogram H is the image representation called LBR. It describes the shape characteristics of the original histogram. There are two reasons for using this binary coding algorithm, and it will also be verified in Section 4: (1) Converting the histogram feature to a binary descriptor can save memory cost significantly and speed-up feature matching. (2) The binary representation would be robust to variations such as cropping, adding text and layout change situations. Binary features have draw attention in some recent works due to the emerging demands of mobile and embedded vision systems, and have shown their robustness in applications [5, 36].

3.2 Near-duplicate image detection using LBR

Our target is online near-duplicate image detection. It is different from traditional content-based image retrieval (CBIR) systems that the images to be compared can not be given in advance for pre-computing. The detection steps are given as follows.

Firstly, all images are normalized so that the width of each image is changed to a fixed value. This step is very important to minimize the adverse impact of the image scale variation. Next, LBR representation is extracted for the image. For color image, the LBRs are extracted from its R, G, B channels respectively, and are concatenated together to form a complete representation of the original image. Therefore, all color images are represented as a set of $3 * N$ bits LBR vectors.

The task of comparing two images can be as simple as counting the number of bit positions that are different. This is the Hamming distance [8] and it measures the minimum number of substitutions required to change one string into the other, which can be computed very efficiently with a bitwise XOR operation followed by a bit count. A distance of zero indicates that they are very similar or have little variation. The function for comparing the similarity score S of two images is defined as:

$$S = \frac{D_h}{3 * N}, \tag{2}$$

where D_h is the hamming distance of two LBR representations. The smaller the S value, the more similar of the two images.

4 Experiments

In this section, we systematically evaluate the proposed method on both online near-duplicate image detection task and video keyframe detection task. Firstly several experiments are carried on to determine optimal parameters for the proposed LBR. Then comparisons with other state-of-the-art algorithms are taken regarding both accuracy and

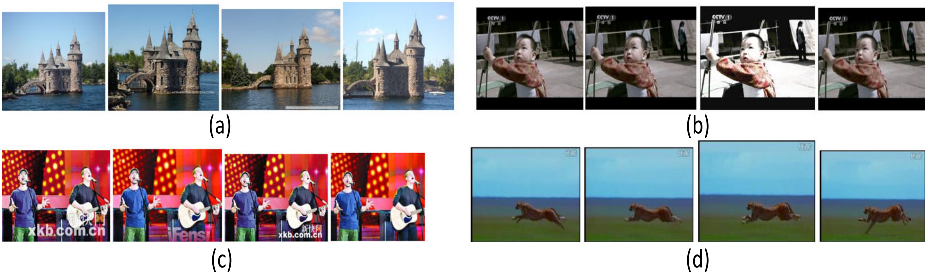


Fig. 5 Sample images from database #1, and each group contains some near-duplicate images

efficiency. Discussions are given about the advantages and limitations of the proposed LBR representation.

4.1 Datasets

Two datasets are constructed to evaluate the proposed method objectively and accurately. There are many public datasets for image retrieval (e.g., Flickr website, Oxford5K [22], UKbench [20], CityU [32]). However, these datasets are not suitable for our target online applications, since their definitions of near-duplicate images are different from ours. In this paper, the near-duplicate images are defined with those variations: resizing, cropping, changing luma and chroma, adding text or watermark, changing layout slightly. Therefore, we build up two our own datasets in experiments.

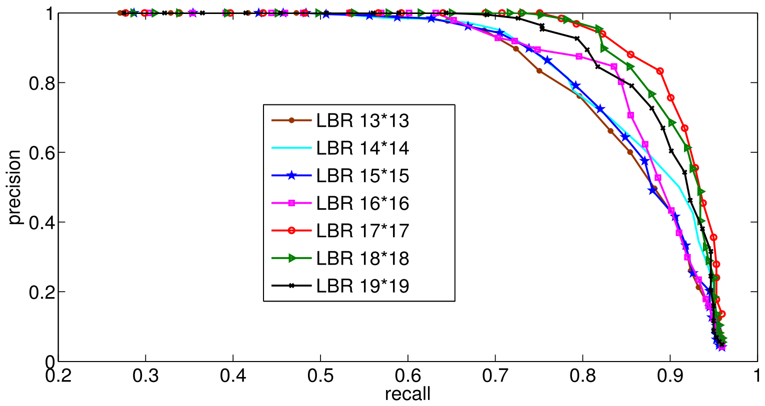
The first dataset contains images collected from Google image search engine through particular queries. Near-duplicate image detection experiments are carried on this database. It contains 1260 images, named as dataset #1. This dataset consists of many groups, each group have some near-duplicate images, some example images are shown in Fig. 5. The near-duplicated images contain variations of resizing, cropping, changing luma and chroma, adding text or watermark, and changing image content slightly.

The second dataset contains several TV program videos collected from different Chinese channels. To prove the effectiveness of the proposed method in large scale video database, the test videos are gathered from four Chinese satellite channels: “CCTV-2”, “SCTV”, “LNTV”, and “ZJTV”, and each has 24 hours data. It is named as dataset #2. All frames in this dataset have 640×480 pixels. Though the content of some video subsequences are same, the related frames’ color and appearance may change in different satellite channels or different times in the same channel (as shown in [14]), so it is an effective and dataset for our experiments.

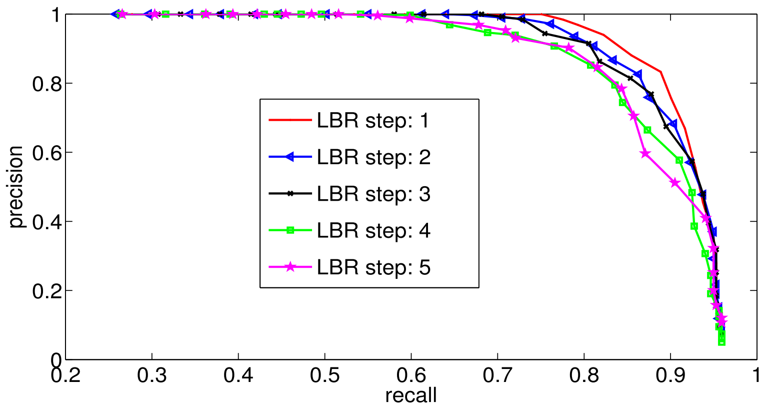
4.2 Experimental setup

Experiments are performed on a 3.3GHz Core-i3 PC (4GB RAM). To get the running time of the tested methods, we used the $\_ftime$ function on Windows platform.

In our experiments, firstly, we use half of the images in dataset #1 (630 images, denoted as training set) to determine the optimal parameters of the proposed LBR, then we compare with previous methods by doing experiments on the other half of the images in dataset #1 (630 images, denoted as test set) and the video data of dataset #2. Three state-of-the-art near-duplicate image retrieval representations, which stand for three categories mentioned



(a) P-R curves using different regions size with the same step (1 pixel).



(b) P-R curves using different steps with the same region size (17*17).

Fig. 6 The P-R curves for determining optimal parameters. **a** P-R curves using different regions size with the same step (1 pixel). **b** P-R curves using different steps with the same region size (17*17).

in Section 2, are compared here regarding detection accuracy and efficiency: 1) color histogram proposed in [26], 2) the method for image retrieval based on block-based LBP [27], and 3) web search for images introduced in [11] (image fingerprint).

To evaluate the accuracy on near-duplicate image detection and video keyframe detection, the Precision-Recall (P-R) curve is used. It is defined as below:

$$Recall = \frac{N_{TruePositive}}{N_{AllTrue}}, \tag{3}$$

$$Precision = \frac{N_{TruePositive}}{N_{AllPositive}}, \tag{4}$$

where $N_{TruePositive}$, $N_{AllTrue}$, $N_{AllPositive}$ represent the number of correctly recalled images, the number of all images, and the number of all recalled images, respectively.

Table 1 The time cost for matching two images

Methods	Time
LBR with binary encoding	2.24 ms
LBR without binary encoding	4.33 ms
LBP-based with binary encoding	6.96 ms
LBP-based without binary encoding	12.12 ms

4.3 Experiments on near-duplicate images

Our experimental strategy to evaluate near-duplicate image detection is fair and effective. Each image in dataset is compared with all other images in the same set to judge whether the image pairs are near-duplicate. A detected pair of images are considered as positive results if they belongs to the same pre-labeled near-duplicate group. Otherwise, we think the results are negative.

Firstly the parameters for LBR are to be set using the training set of dataset #1. From previous sections, it appears that LBR has the following parameters: scale normalized image width: W_w, the size of each block: W_R, H_R, the step size of the sliding window to detect local regions: x, y, and the number of each channel descriptor's bins: N.

In our experiments, we use $W_w = 160$, $N = 64$ and assume $W_R = H_R$, $x = y$ in each image. It should be noted that W_w and N are very important to the efficiency. If increase these two parameters, the LBR descriptors will be more precise. Correspondingly, it needs more computing time and storage space. However, if these two values are too small, the LBR descriptors may lose the representative capability. In our experiments, $W_w = 160$ and $N = 64$ leads to excellent results in every situations. To determine the optimal values for W_R, H_R, a set of experiments are taken. We range W_R, H_R from 1 to 30, fixing other parameters to $x = 1$, $y = 1$. The resulted P-R curves are shown in Fig. 6a (in order to facilitate viewing, we only show a part of the results). It can be seen that the best result is obtained with $W_R = 17$, $H_R = 17$. To determine the optimal values for x and y, we range the values of x, y from 1 to 5, fixing other parameters to $W_R = 17$, $H_R = 17$, and test the performance. The resulted P-R curves are shown in Fig. 6b. It can be seen that the best result is obtained with $x = 1$, $y = 1$. These optimal parameters are used for LBR in the following experiments.

Figure 7 show the performance comparison of the proposed method with three previous methods. For fair judgement, all of these methods mentioned in Fig. 7 utilize their own similarity measurements. In Fig. 7, we could see that the LBR and LBP-based method clearly outperform the other two techniques. It validates the performance priority of local-based methods comparing to global-based methods. The proposed LBR achieves the highest accuracy, showing that it is robust enough to the variations in near-duplicated images considered in this work.

And we could see the advantages of the proposed binary encoding step from Fig. 8 and Table 1. As shown in Fig. 8, we use the same method to get LBP histogram representation as proposed in [27], it is obviously that the result is better when applying the histogram binary encoding step. We also consider the proposed LBR with binary encoding step without binary encoding. For the LBR without binary encoding, histogram intersection is used as the similarity measurement. It can be seen from the results that the binary encoding plays an important role for the accuracy and speed of online near-duplicate images detection. The

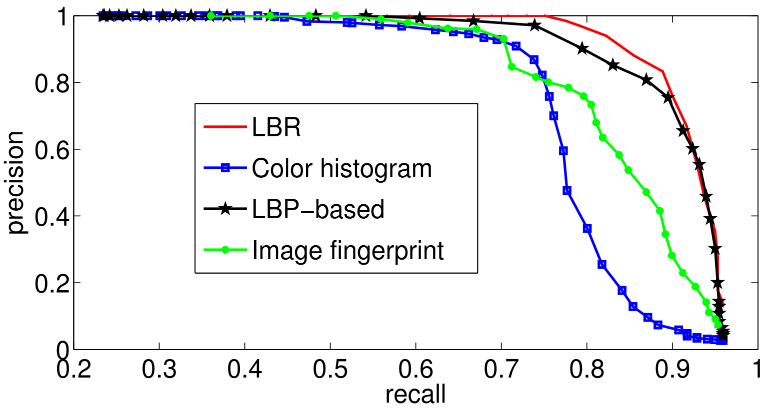


Fig. 7 The comparison of P-R curves of different methods

LBP with binary encoding achieves similar P-R curve to the proposed LBR. However, LBR shows clear priority in terms of computational time.

Table 1 shows the average computing times of different methods for extracting features and matching two images, the LBR is much faster than LBP-based methods and our encoding algorithm is real time-saver. Note that all of our codes are not optimized and the encoding algorithm could be boosted twice on efficiency. The proposed LBR has 192 bits for a color image, which only occupies 24 bytes memory. So the LBR representation has the priority for applications of detecting near-duplicate images in large data corpus.

4.4 Experiments on video keyframe detection

The near-duplicate video keyframe detection is very useful in several interesting applications, eg. online advertisements detection in TV program or online videos. Given an advertising frame, we need to judge whether it appears and the appearing time in a lot of videos. For example, we could statistic the appearing number of an advertisement in a time

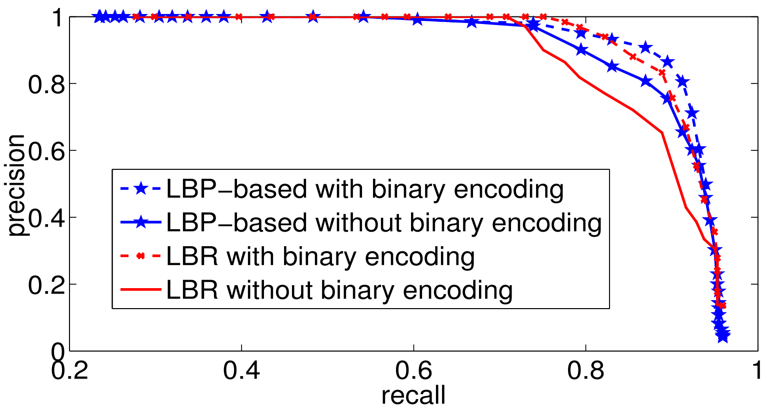


Fig. 8 The comparison of P-R curves of different methods, to validate the effectiveness of the proposed binary encoding

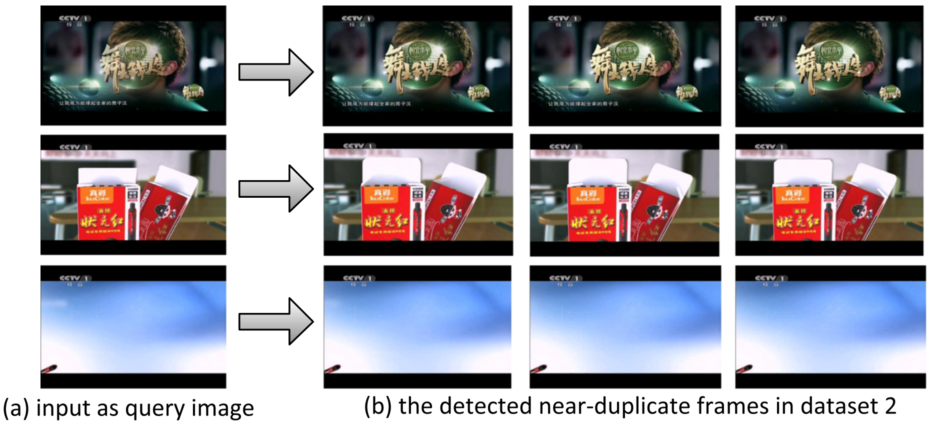


Fig. 9 Examples of video keyframe query. The *first column* shows the query images and the subsequent columns are near-duplicate frames detected in dataset #2

period of TV programs, providing some keyframes in this advertisement video. The same advertisements appearing in different periods of the TV program may contain several types of variations, which are usually caused in video editing and redistribution processes [14], some examples are shown in [14] and the LBR representation we proposed here is applicable for these situations.

In our experiments based on dataset #2, we use a series of advertisement frames as query keyframes to detect their appearance in the test videos. The examples are shown in Fig. 9. We input an advertisement frame as the query keyfame and compare its LBR feature with all frames' LBR features in dataset #2. We could obtain some near-duplicate frames and their locations in dataset #2. According to the manually labeled data we could know our results are positive or negative. We use fifteen images as query data to compute the P-R curves. The parameters of LBR are set as the same as those presented in the previous section. The precision and recall curve is shown in Fig. 10, this figure shows the effectiveness of our proposed method. We can also get the computational time in these experiments. The experiments result shows that our method could process about 900 frames per second. It should be noticed that, due to the temporally continuity of video frames, it is very difficult to recall all of the related frames for a query frame.

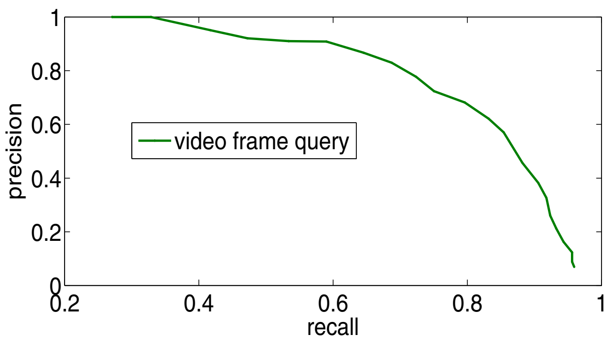


Fig. 10 The P-R curve of video frame query (computed by fifteen query images)

Fig. 11 Examples of false alarms: the left and right pictures in each group are matched based on the proposed LBR

Although the performance of the proposed LBR is very good, it still has some shortcomings. Figure 11 shows two groups of false alarms yielded in experiments using LBR. The group 1 shows two wrong examples in database #1 and the group 2 shows two wrong examples in video frame query experiment. Although they are visually distinct, their feature histograms are similar, without considering the spatial information of local features. As a result, their LBR representations are similar. To further improve the accuracy of near-duplicate image detection, the spatial layout information of local features could be incorporated, which will be considered for the future work.

5 Conclusions

In this paper, we present an efficient and compact image representation for online near-duplicate image detection called LBR, which is based on the collective information from local regions. There are two main novelties in the designation. First, the binary patterns of local features extracted from local regions are efficient and discriminative. Second, the global histogram representation is encoded to a binary vector which is compact enough and robust in performance. The proposed methods can be well applied our interested applications. Experiments have been taken on the near-duplicate image detection and video keyframe detection tasks. The results show that the proposed representation has a good performance regarding both accuracy and efficiency. Moreover, the proposed representation can be also applied in video copy detection [14, 24] by using a special sequence match algorithm, it can be a possible future work to consider.

Acknowledgments This work is supported by the National Natural Science Foundation (NSF) of China (No. 61300056), the Ph.D. Programs Foundation of Ministry of Education of China (No. 20133401120005), the Anhui Provincial Natural Science Foundation of China (No. 1408085QF118), the Open Project Program of the National Laboratory of Pattern Recognition (NLPR) (No. 201306282) and a grant from Shenzhen Science and Technology Project (No. ZDS Y20120617113312191).

References

1. Bao BK, Zhu G, Shen J, Yan S (2013) General subspace learning with corrupted training data via graph embedding. IEEE Trans Image Process 22(11):4380–4393
2. Bao BK, Zhu G, Shen J, Yan S (2013) Robust image analysis with sparse representation on quantized visual features. IEEE Trans Image Process 22(3):860–871
3. Bay H, Ess A, Tuytelaars T, Van Gool L (2008) Speeded-up robust features (surf). Comp Vision Image Underst 110(3):346–359
4. Chang EY, Wang JZ, Li C, Wiederhold G (1998) Rime: A replicated image detector for the world wide web. In: Photonics East (ISAM, VVDC, IEMB), pp 58–67. International Society for Optics and Photonics
5. Choi Y, Park C, Lee JY, Kweon IS (2014) Robust binary feature using intensity order. In: The 12th Asian Conference on Computer Vision (ACCV)
6. Chum O, Philbin J, Zisserman A (2008) Near duplicate image detection: min-hash and tf-idf weighting. In: BMVC, vol 810, pp 812–815
7. Crow FC (1984) Summed-area tables for texture mapping. ACM SIGGRAPH Comput Graph 18(3):207–212
8. Hamming RW (1950) Error detecting and error correcting codes. Bell Syst Tech J 29(2):147–160
9. Ke Y, Sukthankar R, Huston L (2004) An efficient parts-based near-duplicate and sub-image retrieval system. In: Proceedings of the 12th annual ACM international conference on multimedia, pp 869–876. ACM
10. Kim C (2003) Content-based image copy detection. Signal Process Image Commun 18(3):169–184
11. Krawetz N (2011) Image indexing. http://www.hackerfactor.com/blog/index.php?/archives/432-Looks-Like-It.html
12. Law-To J, Chen L, Joly A, Laptev I, Buisson O, Gouet-Brunet V, Boujemaa N, Stentiford F (2007) Video copy detection: a comparative study. In: Proceedings of the 6th ACM international conference on image and video retrieval, pp 371–378. ACM
13. Leutenegger S, Chli M, Siegwart RY (2011) Brisk: binary robust invariant scalable keypoints. In: IEEE international conference on computer vision (ICCV), 2011, pp 2548–2555. IEEE
14. Li T, Nian F, Wu X, Gao Q, Lu Y (2014) Efficient video copy detection using multi-modality and dynamic path search. Multimedia Systems 1–11
15. Li Z, Feng X (2013) Near duplicate image detecting algorithm based on bag of visual word model. J Multimedia 8(5):557–564
16. Liu B, Li Z, Wang M (2010) Efficient video duplicate detection via compact curve matching. In: IEEE international conference on multimedia and expo (ICME), 2010, pp 100–105. IEEE
17. Lowe DG (2004) Distinctive image features from scale-invariant keypoints. Int J Comput Vis 60(2):91–110
18. Meng Y, Chang E, Li B (2003) Enhancing dpf for near-replica image recognition. In: Proceedings of 2003 IEEE computer society conference on computer vision and pattern recognition, 2003, vol 2, pp II–416. IEEE
19. Mishra P, Sonam M, Vijayalakshmi MS (2014) Content based image retrieval using clustering technique: a survey. Int J Res Comput Eng Electron 3(2)
20. Nister D, Stewenius H (2006) Scalable recognition with a vocabulary tree. In: IEEE computer society conference on computer vision and pattern recognition, 2006, vol 2, pp 2161–2168. IEEE
21. Ojala T, Pietikainen M, Maenpaa T (2002) Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Trans Pattern Anal Mach Intell 24(7):971–987
22. Philbin J, Chum O, Isard M, Sivic J, Zisserman A (2007) Object retrieval with large vocabularies and fast spatial matching. In: Proceedings of the IEEE conference on computer vision and pattern recognition
23. Rublee E, Rabaud V, Konolige K, Bradski G (2011) Orb: an efficient alternative to sift or surf. In: IEEE international conference on computer vision (ICCV), 2011, pp 2564–2571. IEEE
24. Shang L, Yang L, Wang F, Chan K. P, Hua X. S (2010) Real-time large scale near-duplicate web video retrieval. In: Proceedings of the international conference on multimedia, pp 531–540. ACM
25. Sivic J, Zisserman A (2003) Video google: a text retrieval approach to object matching in videos. In: Proceedings of 9th IEEE international conference on computer vision, 2003, pp 1470–1477. IEEE
26. Swain M. J, Ballard D. H (1991) Color indexing. Int J Comput Vis 7(1):11–32
27. Takala V, Ahonen T, Pietikäinen M (2005) Block-based methods for image retrieval using local binary patterns. In: Image analysis, pp 882–891. Springer
28. Thomee B, Huiskes M. J, Bakker E, Lew M. S (2008) Large scale image copy detection evaluation. In: Proceedings of the 1st ACM international conference on multimedia information retrieval, pp 59–66. ACM

29. Wang M, Li H, Tao D, Lu K, Wu X (2012) Multimodal graph-based reranking for web image search. IEEE Trans Image Process 21(11):4649–4661
30. Wang M, Ni B, Hua X. S, Chua T. S (2012) Assistive tagging: a survey of multimedia tagging with human-computer joint exploration. ACM Comput Surv (CSUR) 44(4):25
31. Wu X, Hauptmann A. G, Ngo C. W (2007) Practical elimination of near-duplicates from web video search. In: Proceedings of the 15th international conference on multimedia, pp 218–227. ACM
32. Wu X, Ngo C. W, Hauptmann A. G, Tan H. K (2009) Real-time near-duplicate elimination for web video search with content and context. IEEE Trans Multimedia 11(2):196–207
33. Xin Y, Qiang Z, Kwang-Ting C (2009) Near-duplicate detection for images and videos. In: In 1st ACM workshop on large scale multimedia retrieval and mining, pp 73–80. ACM
34. Yang C, Peng J, Fan J (2013) Speed-up multi-modal near duplicate image detection. Open J Appl Sci 3:16–21
35. Yang J, Jiang Y. G, Hauptmann A. G, Ngo C. W (2007) Evaluating bag-of-visual-words representations in scene classification. In: Proceedings of ACM SIGMM workshop on multimedia information retrieval
36. Zhang S, Tian Q, Huang Q, Gao W, Rui Y (2014) Usb: Ultra short binary descriptor for fast visual matching and retrieval. Image Process

Fudong Nian received B.S. from Anhui University (AHU), China in 2013. He is currently a master candidate with College of Electrical Engineering and Automation, Anhui University. His research interests are in the areas of computer vision.

Dr. Teng Li received B.S. from University of Science and Technology of China (USTC) in 2001, M.S. from Institute of Automation, Chinese Academy of Sciences (CASIA) in 2004, and Ph.D. from Korea Advanced Institute of Science and Technolgy (KAIST) in 2010. He received IEEE T-CSVT best paper award in 2014 and ICIMCS best paper award in 2009. He is currently a tenure-track professor with Anhui University.

Dr. Xinyu Wu is now an associate director of Center for Intelligent and Biomimetic systems, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, China. He received his BE and ME degrees from the Department of Automation, University of Science and Technology of China in 2001 and 2004, respectively, and his PhD degree was awarded at the Chinese University of Hong Kong in 2008.

Dr. Qingwei Gao received his Ph.D. degree in information and communication engineering from University of Science and Technology of China (USTC), in 2002. He is currently a professor with the School of electrical engineering and automation, Anhui University. His research interests include wavelet analysis, image processing and signal processing.

Feifeng Li recieved the Master degree from Anhui University in 2013. She is currently a lecturer with Huainan Union University, Anhui Province, China.