SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

June 2013, Vol. 56 068104:1–068104:11
doi: 10.1007/s11432-012-4760-y

Robust video hashing based on representative-dispersive frames

NIE XiuShan[1,2], LIU Ju[2,3*], SUN JianDe[2], WANG LianQi[4] & YANG XiaoHui[2]

[1]*School of Information Science and Technology, Shandong University of Finance and Economics, Jinan* 250014*, China;*
[2] *School of Information Science and Engineering, Shandong University, Jinan* 250100*, China;*
[3]*Hisense State Key Laboratory of Digital Multi-Media Technology, Qingdao* 266071*, China;*
[4]*Department of Physics and Astronomy, University of California, Irvine, Irvine, CA* 92697*, USA*

Received February 6, 2012; accepted October 30, 2012; published online January 6, 2013

Abstract This study proposes a robust video hashing for video copy detection. The proposed method, which is based on representative-dispersive frames (R-D frames), can reveal the global and local information of a video. In this method, a video is represented as a graph with frames as vertices. A similarity measure is proposed to calculate the weights between edges. To select R-D frames, the adjacency matrix of the generated graph is constructed, and the adjacency number of each vertex is calculated, and then some vertices that represent the R-D frames of the video are selected. To reveal the temporal and spatial information of the video, all R-D frames are scanned to constitute an image called video tomography image, the fourth-order cumulant of which is calculated to generate a hash sequence that can inherently describe the corresponding video. Experimental results show that the proposed video hashing is resistant to geometric attacks on frames and channel impairments on transmission.

Keywords representative-dispersive frames, video hashing, video tomography, video copy detection

Citation Nie X S, Liu J, Sun J D, et al. Robust video hashing based on representative-dispersive frames. Sci China Inf Sci, 2013, 56, 068104(11), doi: 10.1007/s11432-012-4760-y

1 Introduction

The fast spread of mobile devices, along with higher Internet bandwidth and cheaper storage, enables end users to easily generate, store, and share large amounts of video content via the Internet. It is urgent to secure copyrighted video contents from illegal use and detect them effectively. On the other hand, the redundancy of Web video copies makes users spend significant amount of time searching for the videos they need. Consequently, users have to repeatedly watch similar copies of videos that have been viewed previously. This process is time-consuming as the users need to watch different versions of duplicate or near-duplicate videos that stream over the Internet. In these cases, the detection of video copies in a video database is one of the key issues in multimedia management.

Traditionally, watermarking techniques have been used to detect copies of images or videos [1,2]. These techniques embed watermarks into the media. These watermarks are imperceptible and used for proving

*Corresponding author (email: juliu@sdu.edu.cn)

the authenticity of the media. However, the watermarks that are embedded into the media somewhat distort the media. On the other hand, robust hashing techniques, also called fingerprinting, extract the most important features of the media to calculate compact digests that allow for efficient content identification without modifying the media.

In robust video hashing, the hash sequence is typically a short binary string taken as a persistent fingerprint of the corresponding video. A video hash sequence is a digest of the video content. Moreover, the video hash sequence is useful for the multimedia domain, because the same video content often exists in various forms, e.g., different file formats and quality levels, etc. The ability of video hashing to assign the same hash sequence to all versions makes it a promising solution for multimedia content identification. The hash sequence is compact, and thus, searching through hash sequences is more efficient than comparing video files directly.

A robust video hashing H that is sensitive to a secret key K can be described as follow:

1) $\mathrm{H}(K, V)$ is uncorrelated with $\mathrm{H}(K, V')$ when two videos V and V' are dissimilar;

2) $\mathrm{H}(K, V)$ is strongly correlated with $\mathrm{H}(K, V_a)$ when V and V_a are similar in content; and

3) $\mathrm{H}(K, V)$ is uncorrelated with $\mathrm{H}(K', V)$ when $K \neq K'$.

The key is important for video identification and it is owned by video management institutions.

This study proposes a robust video hashing for video copy detection. The major contributions of this work are as follows:

1) A new notion called representative-dispersive frames (R-D frames) and a method to select these frames are proposed. The video can be equivalent to a complete undirected weighted graph with frames as the vertices and frame similarities as the edges. Hence, some graph algorithms can be applied to the video. R-D frames contain two types of frames. The first type consists of representative frames representing the main information of the video. The second type consists of dispersive frames dispersed in different parts of the video. These dispersive frames represent the local structure of the video. An adjacency matrix is constructed and adjacency numbers of vertices are calculated in the weight graph mapped from the video to select the R-D frames.

2) A new method of hash generation using video tomography image is proposed. The temporal and spatial information of videos are very important, and the video tomography image can mix the spatial and temporal information together. The R-D frames are scanned to construct a video tomography image, and the global feature extracted from the video tomography image generates a hash sequence. This hash sequence is robust to temporal modification of videos because the temporal modifications, such as frame loss and inserting, can only influence a minimum amount of the local information of the tomography image.

2 Related work

Content-based video identification uses the content of the video to calculate a unique signature based on various video features. Joly [3] presented a copy retrieval scheme based on local features and used the features of key frames that had strong intensity in global motion. A video signature based on the centroid of gradient orientations was proposed in [4], which was robust against various common video processings including lossy compression and frame rate change but was vulnerable to geometric transformations. These methods more or less omitted the temporal variation in video to a certain extent. Therefore, some researchers combined the spatiotemporal and motion characteristic, and introduced the concept of robust hashing, wherein the similarity of robust hash sequences was taken as the measure of multimedia file identification, such as images and videos. Zhou et al. [5] partitioned the brightness of the video into blocks and used the partial differential characteristics of blocks with a new similarity measure to generate video hashes. Coskun et al. [6] proposed two robust hash algorithms based on the discrete cosine transform (DCT) for the identification of video copies. A method for modifying Coskun's method was proposed in [7]. This method generated video hashes based on temporally informative representative images-DCT (TIRI-DCT). Law et al. [8] combined spatial feature points with the trajectory of interested points to obtain robust hash. Xiang [9] proposed a robust hashing algorithm by using video luminance

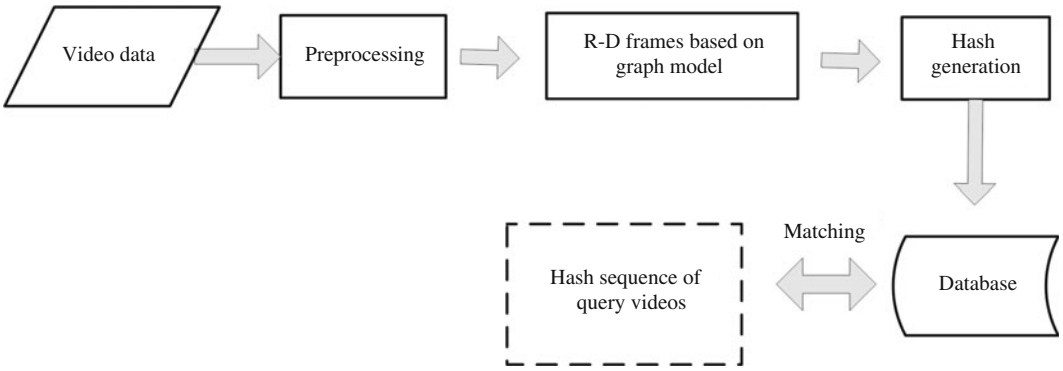

Figure 1 Framework of proposed method.

histogram in shape, which was robust to common geometric distortions and video processing operations. Li [10] extended image hashing to video hashing and proposed a frame hash-based video hash construction framework. Manifold learning-based video identification methods were proposed in [11,12]. Researchers investigated video identification from different perspectives to improve its performance. Zhao et al. [13] presented a frame fusion-based copy detection approach, which converted video copy detection to frame similarity search and frame fusion under the assumption of temporal consistency. Douze [14] introduced a video copy detection system that efficiently matched individual frames and then verified their spatio-temporal consistency. Sun [15,16] proposed a video hashing algorithm with weighted matching based on visual saliency, in which the weighted hash matching was defined in video hashing for the first time.

Current video hashing methods mostly extend hashing techniques that have been developed for images. A video sequence generates hash with large dimensionality because it is composed of many frames, making the database search computationally costly. To overcome this drawback, key frames are selected from video and then fed to video hashing. However, the selection of key frames mainly depends on shot boundary detection and sensitive camera parameters. The current study proposes a graph-based method to select key frames called R-D frames to overcome the drawback discussed above. The R-D frames are different from key frames as they represent not only the main contents of a video but also its local information. Moreover, the selection of R-D frames is based on graph theory and is more robust than the current shot boundary-based key frame selection. Another drawback of key frames-based video hashing is the sensitivity of the key frames to frame drop and noise. In the proposed algorithm, the R-D frames are mixed into a tomography image to generate video hashes. Therefore, a small drop in the number of frames does not change the content of the tomograph image, and hence the hashes can be extracted correctly.

3 Proposed algorithm

Figure 1 illustrates the flowchart of the proposed approach. First, the preprocessing is applied to videos to obtain a fix number of frames. Then, the video is mapped to an undirected weight graph called spatiotemporal graph based on a graph model. In this process, a similarity criterion based on spatiotemporal information is selected to calculate the weights of the edges. Some vertices were selected based on adjacency matrix of the obtained graph. According to the selected vertices, the corresponding frames are selected to constitute R-D frames, and then a tomography image is constructed using these R-D frames. Finally, a robust video hash sequence from the obtained image is generated for matching.

3.1 Preprocessing

Preprocessing is used to resist to the changes in frame size and frame rate. The input video sequence $F(w, h, n)$ is first converted to a standard video signal $F(144, 176, 10)$ in the experiments via smoothing and subsampling, where w is the frame width, h is the frame height ($w = 144$ and $h = 176$ are the sizes of

QCIF sequences widely used in the video processing community), and n is the frame rate. The changes in resolution do not alter the content of video, and thus preprocessing is feasible.

3.2 Representative-dispersive frames

This subsection discusses the R-D frames. The video frames are taken as vertices of a graph, and then the mature graph theory is used to search for the R-D frames.

3.2.1 *Graph model of video*

A video is represented as a weighted, undirected graph called spatiotemporal graph of the video. Let $G = (V, E, W)$ denote the spatiotemporal graph, where V and E are vertex set and edge set, respectively, and $W = (w_{ij})$ is the weight matrix of the edges. Given that spatial similarity and temporal similarity are two important elements used for calculating the weight, an exponential function that measures the weight can be expressed as follows:

$$w_{ij} = \exp\left(-k \cdot \mathrm{sim}(i,j) \cdot |f_j - f_i|\right), \tag{1}$$

where k is an importance factor used to avoid too small value of exponent, and is set at 10 in the experiments, $\mathrm{sim}(i,j)$ is the luminance similarity, and $|f_j - f_i|$ is the time difference between the ith and jth frames.

According to Eq. (1), the weight computation considers both the luminance similarity and temporal frame distance, and thus, it is called the spatiotemporal graph.

Certain features, such as pixel values and texture, are used to calculate $\mathrm{sim}(i,j)$. However, luminance histograms are used to calculate spatial similarity because they can well describe the distribution of pixels and exhibits good robustness. The pixel set of each frame is first denoted by P, and then the luminance similarity $\mathrm{sim}(i,j)$ is calculated as

$$\mathrm{sim}(i,j) = \max \min_{u \in P} \{H_i(u), H_j(u)\}, \tag{2}$$

where $H_i(u)$ and $H_j(u)$ are the normalized luminance histogram values of the ith and jth frames, respectively.

Then, the graph is further simplified by removing edges that exceed a certain threshold. The threshold t is set to the mean of all the weights of the edges in the experiments. The obtained graph is used to select R-D frames.

3.2.2 *Representative-dispersive frame selection*

The selection of R-D frames is based on the spatiotemporal graph model. The frames of a video can be taken as vertices of the corresponding graph; therefore, finding R-D frames in the video is similar to finding representative and dispersive vertices in the obtained graph. Some classic algorithms in graph theory can be used in such process.

Two definitions are listed as follows.

Adjacency matrix: A matrix $\boldsymbol{R} = \{r_{ij}\}_{N \times N}$ is called adjacency matrix of spatiotemporal graph G, where N is the number of vertices in the graph. $r_{ij} = 1$, when $e_{ij} \in E$; otherwise, $r_{ij} = 0$. In particular, $r_{ii} = 1$.

Therefore, the elements of adjacency matrix are 1 or 0.

Adjacency number: Given an adjacency matrix $\boldsymbol{R}$, the adjacency number s_i of the ith vertex in graph G can be expressed as $s_i = \sum_j r_{ij}$.

The selection of representative frames searches for the vertices that are of larger adjacency numbers in the graph, given that more vertices are adjacent to them. The detection algorithm is based on the following two-step model:

1) Construct an adjacency matrix $\boldsymbol{R} = (r_{ij})_{N \times N}$ of the spatiotemporal graph G.

2) Calculate the adjacency number of each vertex, and then save the adjacency numbers as a set $A = \{a_1, a_2, \ldots, a_N\}$. The adjacency number of each vertex is compared with a threshold to determine whether the frame the vertex represents is a representative frame or not. If the adjacency number a_i is larger than a given threshold, then the ith frame is considered as a representative frame. A globally automatic threshold is introduced in the proposed scheme. Let μ_a and σ_a denote the mean and the standard deviation of the adjacency number set, and set the threshold at $\mu_a + a\sigma_a$, where a is a constant, set at 0.3.

Dispersive frame selection aims to locate the frames that are dispersed in different parts of a video. Dispersive frames should be as different from one another as possible because they each represent the local property of a corresponding part of a video. To do so, the vertices that represent dispersive frames in the spatiotemporal graph should be disconnected.

In graph theory, a subset S of V is called an independent set of G if no two vertices of S are adjacent in G. Therefore, the corresponding vertices of the dispersive frames form an independent set. As is known, the independent set is not unique for the graph, neither is the dispersive frames set. Thus, a heuristic method is proposed to obtain a unique set of dispersive frames. This method aims to find an independent set that may not be a maximum but is unique. The process consists of the following two steps:

1) Calculate the adjacency number of each vertex in the adjacency matrix $\boldsymbol{R}$. If the adjacency number of the ith row is the biggest, then convert every element of the ith row and ith column to zero.

2) Repeat 1) until the biggest adjacency number in the matrix $\boldsymbol{R}$ is 1; that is, the corresponding vertices of G whose adjacency numbers are 1 are only joined by themselves. After which, an independent set in the graph is obtained, and the vertices in this set correspond to the dispersive frames.

Consequently, the R-D frames of the video are obtained using the algorithms above. In the R-D frames, the representative frames represent the entire main information of the video, while the dispersive frames reveal the local properties of the video.

The robustness of the selection of the R-D frames depends on the robustness of the edge weights in the spatiotemporal graph. Assume Δs (without loss of generality, let $\Delta s > 0$) is the change after some spatial modifications on the videos. Then, the weight w'_{ij} after spatial modifications can be expressed as follows:

$$w'_{ij} = \exp\left(-k \cdot (\mathrm{sim}(i,j) + \Delta s) \cdot |f_j - f_i|\right). \tag{3}$$

The ratio of change to weight Δr_w is given by

$$\begin{aligned} \Delta r_w &= \frac{w'_{ij} - w_{ij}}{w_{ij}} = \frac{\exp(-k \cdot \mathrm{sim}(i,j) \cdot |f_j - f_i|)(\exp(-k \cdot \Delta s \cdot |f_j - f_i|) - 1)}{\exp(\mathrm{sim}(i,j) \cdot |f_j - f_i|)} \\ &= \exp(-k \cdot \Delta s \cdot |f_j - f_i|) - 1. \end{aligned} \tag{4}$$

Given that $k = 10$ and the minimum of $|f_j - f_i|$ is 0.1, as calculated based on the frame rate, we have

$$\Delta r_w = \exp(-k \cdot \Delta s \cdot |f_j - f_i|) - 1 \leqslant \exp(-\Delta s) - 1 \leqslant \Delta s, \tag{5}$$

where Δs is a small value and the change ratio of weight is even smaller according to (5). Similar results can also be obtained under temporal modifications. Furthermore, the binarization of weights enhances the robustness of the graph as well. Therefore, the selection of R-D frames not only contains the spatiotemporal information of a video but also provides robustness against minor modifications.

The size of the adjacency matrix is somewhat large for computation because videos consist of many frames. However, the adjacency matrix is a sparse matrix with numerous zeros, and thus, the complexity of storage and computation is reduced dramatically.

3.3 Hash generation

This subsection discusses the process of hash generation based on R-D frames, which includes video tomography image generation and hash computation. The feature of video is extracted from the tomography image, and then hash values are generated by this feature through hash computation.

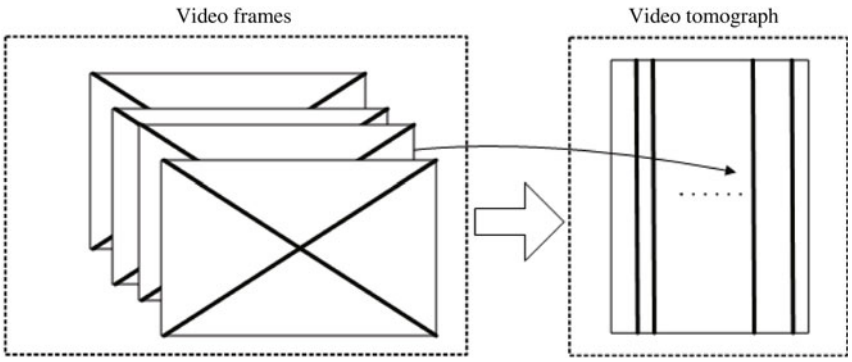

Figure 2 The process of generating a video tomography image.

3.3.1 *Video tomography image*

The generation of a hash sequence from each R-D frame is difficult because the number of R-D frames remains substantial even when it is smaller than the number of frames. Moreover, the hash sequence may vary when some R-D frames are lost under temporal modifications. Thus, a new image called video tomography image is generated from all R-D frames, and the high-order cumulant of R-D frames, which is a global feature and is robust to common modifications, is used to calculate the hash sequence.

Video tomography is an image obtained by projecting a certain horizontal or vertical line in each video frame into one image, as described in [17]. Video tomography has been primarily explored for summarization and camera work detection in movies. Video tomography transforms a video sequence composed of a 3D data set (a time sequence of 2D frames) into a 2D image. A tomography image is obtained by taking a fixed line from each of the frames in a clip and then arranging them from top to bottom or left to right. For example, let $V = \{f_1, f_2, \ldots, f_n\}$ denote a video with n frames and take the brightness component of each frame f_k as an image with size $w \times h$. The brightness of the pixel (i, j) is denoted by $f_k(i, j)$, where i and j are the coordinates of the row and column, respectively. The video tomography of V is an image $\mathbf{I}$ of size $n \times w$ such that

$$\mathbf{I}(:, k) = [\mathrm{diag}(f_k),\ \mathrm{idiag}(f_k)]^{\mathrm{T}}, \quad 1 \leqslant k \leqslant n, \tag{6}$$

where $\mathrm{diag}(f_k)$ and $\mathrm{idiag}(f_k)$ are diagonal elements in two directions, as shown in Figure 2.

Determining the feature selected by the video tomography image is important for video copy detection. A direct selection is concerned with the pixels of the image, but these pixel are fragile and are altered after modifications. Hence, a more robust feature should be applied. This study uses the fourth-order cumulant as the feature of the video tomography image.

Cumulant is a quantity in statistics that measures deviation from Gaussian and is often used for evaluating the non-Gaussianity of a signal. The r-order cumulant of a zero-mean stationary process X is C_{rX}, which is generally called high-order cumulant when $r \geqslant 3$.

Supposing zero-mean random variables X and Y are independent and Y is Gaussian, according to the properties of high-order cumulant, we have

$$C_{r(X+Y)} = C_{rX} + C_{rY} = C_{rX}, \quad r \geqslant 3. \tag{7}$$

The Gaussian noise can be separated by high-order cumulant. Content-preserving modifications, such as lossy compression, scaling, and low-pass filtering, can be modeled as Gaussian noises added to the content of a video [18] as they do not change the perceptual content of the video. Therefore, the content-preserving modifications can be separated when the higher-order cumulant is taken as the feature of video content. A robust feature can be constructed from a video tomography image by using a high-order cumulant.

In this study, the fourth-order cumulant is used as the feature of the video tomograph image defined

as follows:

$$\begin{aligned} C_{4X}(k,l,m) = {} & E\{X(n)X(n+k)X(n+l)X(n+m)\} \\ & -C_{2X}(k)C_{2X}(l-m) - C_{2X}(l)C_{2X}(k-m) - C_{2X}(m)C_{2X}(k-l), \end{aligned} \tag{8}$$

where $E(\cdot)$ is a mathematical expectation and C_{2X} is the second-order cumulant denoted by $C_{2X}(k) = E\{X(n)X(n+k)\}$.

Given that fourth-order cumulant are three-dimensional, they are only calculated along a line by setting l and m to zero in (8) to obtain a compact hash.

The fourth-order cumulant of the luminance component in a video tomography image is calculated and is considered as the feature of the tomography image. However, this cumulant contains excessive cumulant coefficients with some redundancy. Therefore, DCT is applied and then the largest M coefficients that contain most of the energy are selected to represent the image. Ref. [19] provided a theoretic framework for analyzing binary hash-based content identification systems and showed that a 256-bit hash sequence can perform well if the number of videos in a database does not exceed 2^{30}. Therefore, M is set to 256, and the first M DCT coefficients of the fourth-order cumulant are denoted by $\boldsymbol{c} = \{c_k\}_{1\times 256}$. The vector $\boldsymbol{c}$ is used to generate the hash sequence and a matching tag that can reduce the matching range. Let μ_c and σ_c be the mean value and the variance of $\boldsymbol{c}$, respectively. In this study, the matching tag is denoted by $T_q = \mu_c + b \times \sigma_c$, where $b = 0.6$.

3.3.2 *Hash computation*

A robust hash function is the key issue of the proposed scheme. In the proposed algorithm, a random vector whose entries are uniformly distributed random variables in [0,1] is generated first, and then the mean of the vector is subtracted from each element of the vector. Then, the vector is defined as $\boldsymbol{p} = \{p_m\}_{1\times 256}$ and is taken as a key.

The hash sequence $\boldsymbol{h} = \{h_m\}_{1\times 256}$ with a threshold θ is generated as follows:

$$h_m = \begin{cases} 1, & \text{if } |c_m \cdot p_m| \geqslant \theta, \\ 0, & \text{if } |c_m \cdot p_m| < \theta, \end{cases} \quad 1 \leqslant m \leqslant 256, \tag{9}$$

where c_m is the DCT coefficient of the cumulant. For each video, a threshold θ is calculated as follows:

$$\theta = \text{median}(|c_m \cdot p_m|), \quad 1 \leqslant m \leqslant 256, \tag{10}$$

where median$(\cdot)$ is a function for calculating the median of a vector. The hash sequence $\boldsymbol{h}$ and the matching tag constitute the video signature $\boldsymbol{s} = \{T_q, \boldsymbol{h}\}$ that is used to identify the video. For convenience, the hash sequence of the corresponding video is $\boldsymbol{s}$, even though T_q is not binary.

3.4 Video matching

The process of matching consists of two steps. First, the matching tag T_q is used to scan all matching tags of the videos in the database, and then the hash sequences whose tag values are between $T_q - \xi$ and $T_q + \xi$ ($\xi = 0.2\,T_q$) are selected to form a group that is significantly smaller than the original database.

Then, a finer matching is used to compare the query hash sequences and the sequences in the selected group. The video is identified when the bit error rate (BER) is below a given threshold α. BER is defined as

$$\text{BER} = d/l, \tag{11}$$

where d is the number of different bits between the query and original hash sequences, and l is the length of the hash sequence. For convenience, the BER can be taken as the metric since the hash sequences are binary.

The median-based quantization used in the generation of hash sequence guarantees that the bits "1" and "0" each represents a half of each hash sequence, respectively. The number of different values between

Table 1 BER statistics

Modify type	BER with original (%)			BER with different videos (%)		
	Proposed method	DCT-based	TIRI-based	Proposed method	DCT-based	TIRI-based
Noises	0.41	0.48	0.76	48	51	49
Brightness	0.31	0.1	0.6	49	50	49
Media filtering	3.3	4.12	3.76	49	48	51
Frame rotation	0.01	0.01	0.43	52	49	48
Frame drop	0.52	6.79	1.61	49	50	51
Frame shift	0.19	0.29	3.75	53	51	49
Scaling	1.13	1.53	1.42	53	49	52
Compression	5.6	5.98	5.62	49	48	49

two video hash sequences is approximately half of the length of sequence in finer matching. On the other hand, for two similar videos, the threshold α directly determines the false positive rate P_f, which is the probability that two video sequences are incorrectly declared to be equal. A smaller α indicates a smaller probability P_f. On the other hand, a small value of α will negatively affect the false negative probability P_n, which is the probability that two signals are equal but unidentified. The selection of threshold can be seen in [19,20]. In the current study, the threshold is set to 0.16.

4 Experimental results and analysis

4.1 Experimental design

The test videos downloaded from http://www.open-video.org/ were used to evaluate the performance of the proposed video hashing. The performance of the proposed video hashing is compared with those of the methods in [6,7,11,12], and the results are summarized in Section 2. The following two types of experiments are designed:

1) Evaluating the robustness and discrimination. Two hundred different videos with some modifications over the original sequences are used to evaluate the proposed scheme. The results are compared with the methods in [6] (DCT-Based), [7] (TIRI-Based), and [11]. The mean results are shown in Figure 3 and Table 1.

2) Evaluating the performance on precision and recall. The queries are constructed using copies of videos in the database (some videos that were not in the database were also selected). The task of the system is to evaluate the rate of correct returns to fingerprint queries. The precision rate P_{r} and recall rate R_{e} are defined as follows:

$$P_{\mathrm{r}} = \frac{N_{\mathrm{tp}}}{N_{\mathrm{mp}}} \times 100\%, \tag{12}$$

$$R_{\mathrm{e}} = \frac{N_{\mathrm{tp}}}{N_{\mathrm{ep}}} \times 100\%, \tag{13}$$

where N_{tp}, N_{mp}, and N_{ep} represent the number of true positives, matched video clips and total video copies, respectively. Figure 4 shows a comparison of the proposed system with those in [6,7,12] in terms of precision and recall rates.

4.2 Experimental results

Figure 3 shows the robustness of the proposed method under different modifications, wherein the proposed method was found to be more robust than the method in [11].

In Table 1, the matching performances of the original hashes and the hashes extracted from manipulated videos are given for the proposed method and the methods in [6,7]. The BER is the rate of the

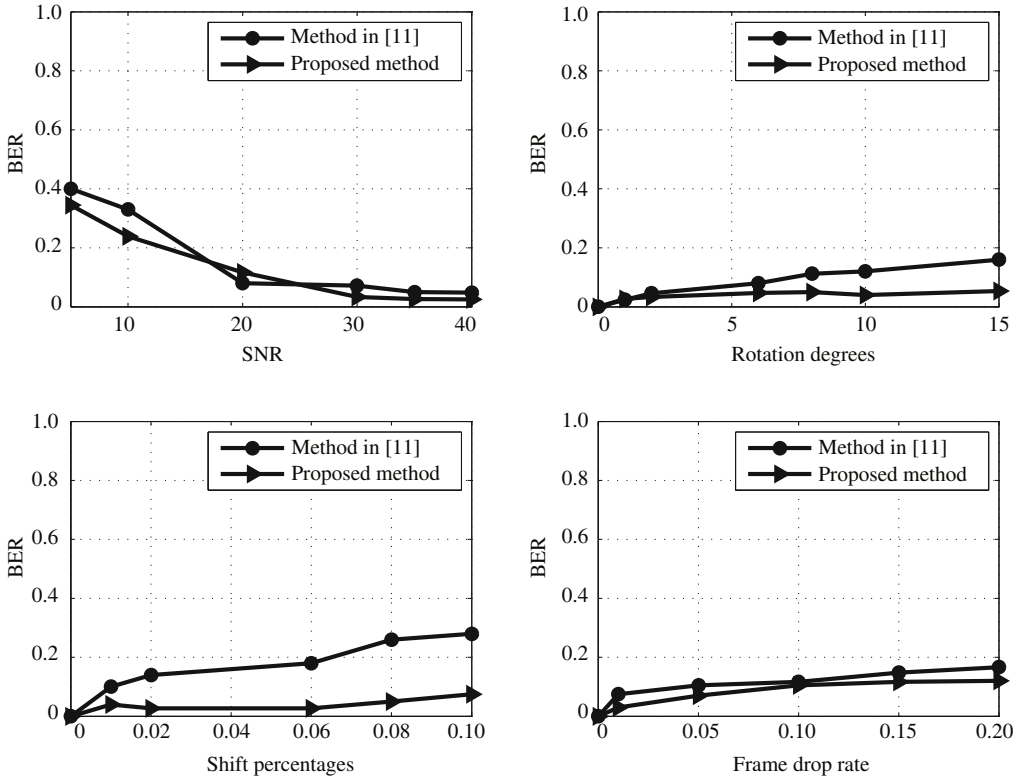

Figure 3 Average BER under the modifications of random noises (top-left), frame rotation (top-right), frame shifting (bottom-left), and frame dropping (bottom-right).

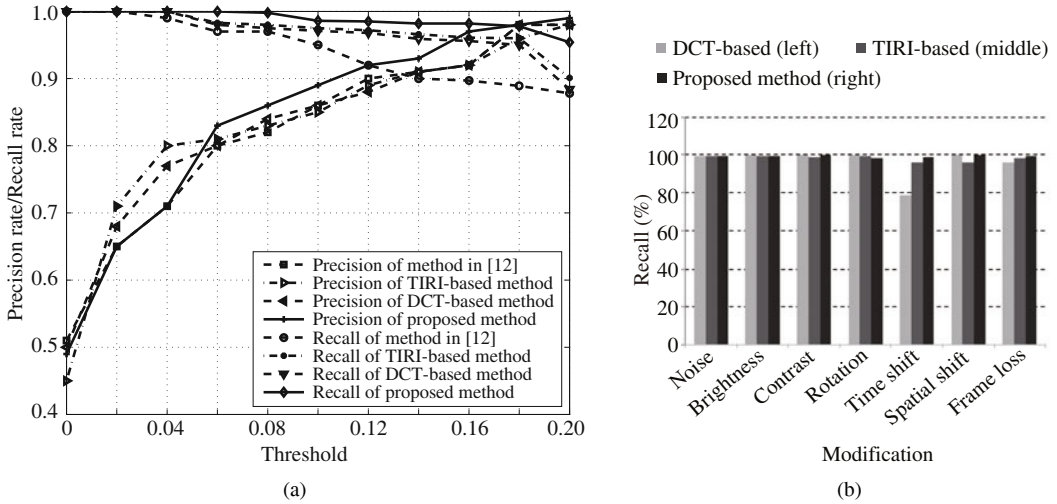

Figure 4 The performances of the methods on precision and recall. (a) The mean precision and recall rates of the proposed method compared with methods in [6,7,12]; (b) the histograms of recall of the proposed method compared with methods in [6] and [7] under different attacks.

mismatching bits. The proposed method is below the threshold, and the BER has been improved for almost all the video operations. In particular, the BER of the proposed method is 6% lower than that of the method in [6] under the condition of frame dropping, indicating the better performance of the proposed method under temporal modifications. The last three columns of Table 1 show the discriminability. The BERs of the three methods with different videos in content are approximately 50% , indicating that the proposed method has the same good discrimination as the other two methods. The process of the determination of the BER for discrimination is described in detail in [6].

Figure 4 shows the performances of four algorithms, including DCT-based [6], TIRI-basd [7], method in [12], and the proposed method. The selection of the threshold α is very important for a video hashing algorithm. An appropriate α can keep both the recall rate and precision rate sufficiently and high simultaneously. In this study, an α of 0.16 was chosen as appropriate, as shown in Figure 4(a). Figure 4(a) shows the mean performances of precision and recall according to different thresholds, wherein the proposed method was shown to outperform the methods in [6,7,12] in terms of both precision and recall. Figure 4(b) shows the performance of recall under different attacks (threshold of 0.16), wherein all the other three algorithms exhibit good performance on the common attacks, but the proposed method performs better on some modifications, such as time shift and frame loss, because it generates hashes from the tomography image, which is insensitive to temporal information. Based on the view of average under the proposed modification, the average recall rate of proposed algorithm is 99.3%.

5 Conclusions and future research

This study presented a scheme of robust video hashing for video copy detection. In the proposed scheme, the R-D frames are first selected based on a graph model and then projected into a video tomography image. The hash values are generated from the cumulant coefficients of video tomography image. Experimental results show that the proposed video hashing has good robustness and discriminability.

A preliminary study on graph model for video was conducted. However, further research is necessary. As one of the important issues in video hashing, the definition of weights of edges in the graph model is denoted in this study by using geometrical information. Besides geometrical information, the visual perception of video should also be considered. In future research, we will conduct an extensive study on the definition of weights based on visual perception.

In addition, the efficiency of indexing in a very large video dataset should be improved. As part of our future research, we will conduct a detailed analytical study on the indexing of such scheme.

Acknowledgements

This work was supported partially by National Basic Research Program of China (973 Program) (Grant Nos. 2009CB320905, 2010CB735906), National Natural Science Foundation of China (Grant Nos. 61101162, 61001-180), and Cultivation Fund of the Key Scientific and Technical Innovation Project (Grant No. 708059).

References

1 Cox I, Kilian J, Leighton F, et al. Secure spread spectrum watermarking for multimedia. IEEE Trans Image Process, 1997, 6: 1673–1687
2 Hartung F, Kutter M. Multimedia watermarking techniques. Proc IEEE, 1999, 87: 1079–1107
3 Joly A, Buisson O, Frelicot C. Content-based copy retrieval using distortion-based probabilistic similarity search. IEEE Trans Multimedia, 2007, 9: 293–306
4 Lee S, Yoo C D. Video fingerprinting based on centroids of gradient orientations. In: Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, Toulouse, 2006. 401–404
5 Zhou X B, Schmucker M, Christopher L. Perceptual hashing of video content based on differential block similarity. In: Proceedings of the International Conference on Computational Intelligence and Security, Xi'an, 2005. 80–85
6 Coskun B, Sankur B, Memon N. Spatio-temporal transform based video hashing. IEEE Trans Multimedia, 2006, 8: 1190–1208
7 Esmaeili M M, Fatourechi M, Ward R K. A robust and fast video copy detection system using content-based fingerprinting. IEEE Trans Inf Forensic Secur, 2011, 6: 213–216
8 Law T J, Chen L, Joly A, et al. Video copy detection: a comparative study. In: Proceedings of the 6th ACM International Conference on Image and Video Retrieval, Amsterdam, 2007. 371–378
9 Xiang S J, Yang J Q, Huang J W. Perceptual video hashing robust against geometric distortions. Sci China Inf Sci, 2012, 55: 1520–1527
10 Li W, Preneel B. From image hashing to video hashing. In: Proceedings of the 16th International Multimedia Modeling Conference, Chongqing, 2010. 662–668

11 Nie X S, Liu J, Sun J D. Robust video hashing for identification based on MDS. In: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP), Dallas, 2010. 1834–1837
12 Nie X S, Liu J, Sun J D, et al. Robust video hashing based on double-layer embedding. IEEE Signal Process Lett, 2011, 18: 307–310
13 Wei Z K, Zhao Y, Zhu C, et al. Frame fusion for video copy detection. IEEE Trans Circuits Syst Video Technol, 2011, 21: 15–28
14 Douze M, Jegou H, Schmid C. An image-based approach to video copy detection with spatio-temporal post-filtering. IEEE Trans Multimedia, 2010, 12: 257–266
15 Sun J D, Wang J, Zhang J, et al. Video hashing algorithm with weighted matching based on visual saliency. IEEE Signal Process Lett, 2012, 19: 328–331
16 Wang J, Sun J D, Liu J, et al. A visual saliency based video hashing algorithm. In: Proceedings of the IEEE International Conference on Image Processing (ICIP), Orlando, 2012. 645–648
17 Akutsu A, Tonomura Y. Video tomography: An efficient method for camerawork extraction and motion analysis. In: Proceedings of the ACM International Conference on Multimedia. New York: ACM, 1994. 349–356
18 Yu L J, Schmucker M, Busch C, et al. Cumulant-based image fingerprints. In: Proceedings of SPIE—Security, Steganography, and Watermarking of Multimedia Contents, San Jose, 2005. 68–75
19 Varna A L, Swaminathan A, Wu M. A decision theoretic framework for analyzing binary hash-based content identification systems. In: Proceedings of the ACM Conference on Computer and Communications Security. New York: ACM, 2008. 67–76
20 Oostveen J C, Kalker T, Haitsma J. Visual hashing of digital video: Applications and techniques. In: Proceedings of SPIE—Storage and Retrieval for Image and Video Databases, San Diego, 2001. 121–131