Spatio-temporal Features for Efficient Video Copy Detection

Ruijuan Hu[1], Bing Li[2], Weiming Hu[2], and Jinfeng Yang[1]

[1] College of Aviation Automation, Civil Aviation University of China, Tianjin, China
rjhu0525@163.com, jfyang@cauc.edu.cn
[2] Institute of Automation, Chinese Academy of Sciences, Beijing, China
{bli,wmhu}@nlpr.ia.ac.cn

Abstract. Content-Based Video Copy Detection (CBVCD) aims at detecting whether or not a query video is a copy or part of a reference video from database. In this paper, we present a CBVCD system based on spatio-temporal features that can competitively deal with large database in terms of both performance and efficiency. Instead of selecting keyframes or uniformly sampling from original videos and then extracting global or local visual features for frames, we first divide a video into segments with fixed length and then extract 3D spatio-temporal features for the whole segment. After that, we perform similarity search comparing all the reference segments with query segments and apply a copy verifying to decide the final copy detection result. The experimental results on the TRECVID 2011 video copy detection dataset show that the proposed system is effective and efficient.

Keywords: Content-Based video copy detection, spatio-temporal features, similarity search, copy verifying.

1 Introduction

The goal of Content-Based Video Copy Detection (CBVCD) is to locate video fragments within a query video that are copies of reference videos. It is essential for many applications, for example, illegal content monitoring, copyright control, tracking the source and so on.

1.1 Related Work

In CBVCD, the copied videos are usually subject to various tolerated transformations (TTs) such as camcording, picture-in-picture(PIP), strong re-encoding, frame dropping, cropping, stretching, contrast changing, etc [1]. Some of these transformations are intrinsic to the video creation process, others are introduced intentionally for specific use. The transformation applied to a video can be one of the TTs mentioned above or combination of some of them. Besides, a query video can also be compiled in three modes: 1) only keep the reference video segment; 2) only keep the non-reference video segment; 3) inserting the reference

C. Sun et al. (Eds.): IScIDE 2013, LNCS 8261, pp. 120–127, 2013.

video segment into the non-reference video segment at a random offset. In addition, in any video task, the dataset is always much larger than image task. So the cost of computation and the detection efficiency must be considered at the same time. All the aforementioned problems make the task more challenging.

Most existing CBVCD systems are based on visual cues, which can be roughly divided into two categories: frame-based and video-based. Frame-based methods typically extract 2D interest points on selected keyframes or uniformly sampled frames of the videos and then use local descriptors to represent them [2]. These descriptors indicate spatial information significantly, while neglect temporal information. In order to be more discriminative for video task, temporal information is introduced via post-processing. Douze et al. [3] report their system which depends on bag-of-features combined with Hamming embedding of the frames. Then they determine the time shift using 1D Hough voting algorithm, and a 2D affine transformation is estimated between temporally consistent frame matches. In [4], R.Cameron first extracts SURF features [5] for the frames and then creates temporal signature by sorting the SURF feature counts in each region along the time-line. Although those frame-based algorithms are spatio-temporal to some extent and have achieved siginificant result, they have some obvious limitations. One is that they largely depend on the selection of frames. For uniformly sampled frames, there is no guarantee that the same frames will be selected both in reference and query videos unless for the assumption that the scene changes slowly so adjacent frames are similar. And the data is usually large. For keyframes, though the number of frames is much less, the system is highly lied on robustness of the shot-boundary detection. Moreover, in these frame-based method,the spatial and temporal information is not processed at the same time, it is hard to guarantee the correspondence of spatio-temporal information. For video-based approaches, trajectories are proposed by means of tracking 2D interest points throughout the video sequence. Law et al. [6] use 2D Harris detector and Kanade-Lucas-Tomasi (KLT) feature tracking for CBVCD. Although the local descriptor is enhanced with temporal information by using trajectories and have achieved promissing result, the redundancy of the local descriptor is reduced. Moreover, it adds additional computations due to the need for tracking interest points over the whole video frames.

1.2 Our Work

Considering the limitations discussed above, we propose an alternative video copy detection system based on spatio-temporal features [7]. Fig. 1 gives an overview of our system. Having preprocessed videos, we first cut them into segments (a set of consecutive frames) and then extract spatio-temporal interest points from segments instead of spatial keypoints from frames. Extended scale-invariant feature transform (SIFT) features combined with PCA algorithm are extracted to represent these segments [8]. By comparing segments similarity and copy verifying, we can obtain final detection result.

The most important improvement of our method is that we directly use 3D interest points. Different from an image $I(x, y)$, here we must operate interest

point detector on a stack of images denoted by $I(x, y, t)$, making localization-proceed not only along the spatial dimensions x and y but also the tempoal dimension t . The third dimension sufficiently represents the sequence of frames, which is the fundamental difference from images. In this regard, our spatio-temporal features based system has several advantages:

•The interest points are detected not only spatially but also over time, which makes them more discriminative as well as better localized within the segments.

•The resulting set of descriptors contains information from the whole segment, rather than focusing on a few selected frames.

•We extract features both spatially and tempoally at the same time, so there is no post-processing for adding the temporal cues.

•The PCA-SIFT descriptor, which is more discriminative and has lower demension than SIFT, makes our system much more efficient.

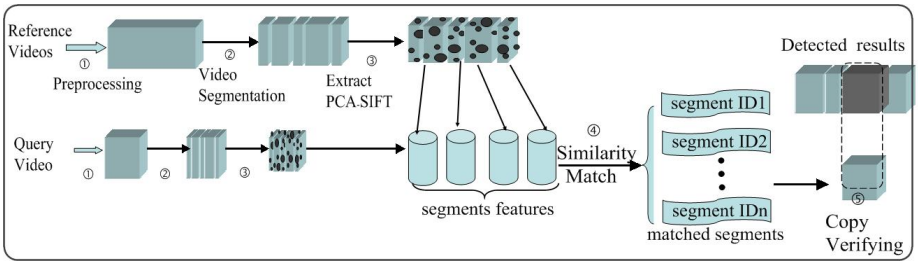

Fig. 1. An overview of the proposed system

2 Proposed System

In this section, we introduce our system based on spatio-temporal features in detail. For convenience, let R be the set of reference videos, let Q be the set of query videos. As shown in Fig. 1, the proposed system involves five parts in total: (1) Preprocessing. This step is to process input videos to diminish the effect of TTs. This creates a new set of reference videos R' and query videos Q'; (2) Video Segmentation. This part partitions videos into short segments with fixed length; (3) Spatio-Temporal Feature Extration. This step detects spatio-temporal interest points and represents them with PCA-SIFT; (4) Segment Similarity Match. It performs k-nearest neighbor method (KNN) for each query segment and returns most similar reference segments; (5) Copy Verifying. This part compares all the candidates for each query segment, and returns the final detection result.

2.1 Preprocessing

For better detection results, we preprocess input videos with two procedures: (1) skip frames that contain little information (2) diminish transformations effect.

For skipping frames, this task detects black frames by computing the variance between the intensity of frame pixels, and those frames whose values are under

a threshood will be skipped. We also skip exceptional frames, which is an outlier compared with its former frame f_{i-1} and the next f_{i+1} [9].

To diminish TTs effect, we focus on camcording and PIP that are very difficult to detect in CBVCD. The method is to detect persistent strong lines using Hough lines, which are consistent over the whole video, thus referring to the boundary of camcording or the window of PIP. If the edge lines are not vertical or horizontal, we use them to form a wrapping quadrilateral, the biggest one is seen as camcording boundary, then a new query is created by mapping the detected quadrilateral to the video corners. If most of the detected lines are vertical or horizontal, we remove short edge lines according to the size of PIP window (one third to half of the original video size) and merge others into a regular rectangle, we then build two new query videos, one is the foreground, another is the background. As we cannot guarantee the preprocessing to be completely correct, we process both the prepocessed and original query video with this system and determine the final result by choosing the more similar one.

2.2 Video Segmentation

In order to extract spatio-temporal features, we need partition every video into short segments. Here we do not divide a video into shots based on boundary because this will result in too few segments and largely depend on the efficiency of the boundary detection algorithm. In our system, each video is partitioned into segments with fixed length of 25 frames (less than a second, as the fps is 30 frames/s). The reason is that if the frames within a segment are too few, content of these frames are almost identical, the detected interest points in latter step are too sparse. And that if the frames are too many, the length of extracted features in a single segment is very high, which is a great challenge for efficiency and storage. To make a balance choice, we choose 25 in a segment to be discriminative enough as well as limited computation costs.

2.3 Spatio-Temporal Feature Extration

After obtaining video segments, this section is to extract the spatio-temporal features. Different from most methods that extract spatial descriptors and add temporal information in the post-processing step, we directly extract features containing both spatial and temporal information, known as spatio-temporal features that are widely used in behavior recognition. The general idea of feature extraction is similar to the spatial case. First, We need a response function and find the interest points where the function reaches its local maxima. This step considers both the spatial and temporal information since the response function has two parameters σ and τ , corresponding roughly to the spatial and temporal scale of the detector. Then, at each interest point, a cuboid is extracted. To further represent the cuboid, in this paper we use the flattened gradient as the descriptor, which is essentially a generalization of the PCA-SIFT descriptor.

Interest Points Detection. The most widely used spatio-temporal interest point operator is proposed by Laptev and Lindeberg [10] that extends the 2D scale-invariant Harris-Laplace corner detector into the spatio-temporal domain. The basic idea is to find a spatial corner in an image region whose velocity vector is reversing direction. This Harris detector has proved to be efficient in many applications. Nevertheless, for some videos in our dataset, the true spatio-temporal corners are quite rare, greatly affecting the detection result. So, here we use an alternative detector proposed by P.Dollar in [7] whose feature set is more dense than the Harris detector. First, we define the response function as

$$R = (I * g * h_{ev})^2 + (I * g * h_{od})^2. \tag{1}$$

where $g(x, y; \sigma)$ is the 2D Gaussian smothing kernel, applied only along the spatial dimensions and h_{ev} and h_{od} are a quadrature pair of 1D Gabor filters applied temporally, defined as $h_{ev}(t; \tau, \omega) = -\cos(2\pi t\omega)e^{-t^2/\tau^2}, h_{ev}(t; \tau, \omega) = -\sin(2\pi t\omega)e^{-t^2/\tau^2}$. Usually, let $\omega = 4/\tau$, so R simply correspond to σ and τ.

As h_{ev} and h_{od} are periodic, variations in local image intensities that contain frequency components will evoke the strongest response. This property is very important in behavior recognition, like people waving or bird flapping its wings. In our dataset, periodic contents are not so common. But this response function is still available because it also responds strongly to spatio-temporal corners. Areas undergoing drastic changes along temporal dimension or with spatially distinguishing features can induce strong response. Due to this ability, it is good at detecting 3D interest points, where R reaches local maxima.

Cuboids and Descriptor. At each interest point, a cuboid is extracted. This cuboid contains most of the volume of data that contribute to the response function. Given a large number of cuboids in our video dataset, we use a descriptor to represent each cuboid which can be computed once off-line. This descriptor is required to be discriminative and invariant to most of the transformations. The simplest way is to create a vector of flattened cuboid values by computing the gradient or Lucas-Kanade optical flow [11] of that cuboid. As the optical flow is more often used to extract motion information, which is not common in our videos, we focus more on the gradient. It has been proved that SIFT is effective in various video retrieving and near duplicate image detection task. So in this work, we adopt an extension of Lowes SIFT [12]. A cubiod is divided into regions and the extended SIFT is created by sampling the magnitudes and orientations of 3 axis-aligned gradient in that cuboid around the interest point. Then smoothed local orientation histograms are built which capture the important aspects of that cuboid, creating a high-dimension features. Then we use PCA to reduce the dimensi-onality of these descriptors. This idea is from Yan Ke [8], known as PCA-SIFT. The whole procedure can be summarized in the following steps: (1) given a segmented video, extract all the cuboids of segments set (2) compute descriptors for each cuboid (3) create an eigenspace by computing the covariance matrix of these vectors, and the top m eigenvectors are used as the projection matrix for PCA-SIFT (4) project all the descriptor vectors

using the eigenspace and result in new descriptor. This effectively linearly-project high-dimension vectors onto a low-dimensional feature space.

2.4 Segment Similarity Match

This task is to compute distance between two descriptors to determine whether the two vectors belong to the same cuboid in different segments. Distance between the descriptors can be calculated by using Euclidean. Then we perform KNN to retrieve the most similar reference segments for every query segment and obtain the k closest reference segments. Note that if the input data is large, we need to build an effective index to improve the seach efficiency, such as vocabulary tree combined with inverted file [13].

2.5 Copy Verifying

The objective of the last step is to compare the candidate segments for each query segment and determine the final detection result. We opt to use an aggregate votes algorithm. In order to improve the detection accuracy, the votes $S_f(v)$ for any reference video $v(v \in R')$ are combined with a weighted value

$$S_f(v) = \sum_{i=1}^{m} \sum_{j=1}^{n} \omega_i^j S(s_i, r_i^j). \tag{2}$$

where ω_i^j is the weighted value and $\omega_i^j \in (0, 1]$, $S(s_i, r_i^j)$ is the similarity score of query segment s_i and reference segment r_i^j , r_i^j is a segment of v . Obviously, we can use distance value to replace this similarity score, but to be simple, we normalize it as $S(s_i, r_i^j) = 1$. This task is calculated as follows:

(1) For query segment , add rank imformation $rank_i^j$ to the similarity list.

(2) Compute the conresponding w_i, as $w_i = 1 - (rank_i^j - 1) * (1/k)$.

(3) Compute votes for all reference videos and aggregate the votes, then locate the maximun value and if the maximum is larger than a threshold, then the copy detection result is , and if it is less than the threshold, there is no copy.

3 Experiments

In this section, we evaluate our system on TRECVID 2011 dataset [1]. This dataset contains more than 12,000 reference videos and over 10,000 query videos which are created with TTs. Our system is tested on a subset of TRECVID 2011 dataset and perform two experiments to show the results.

3.1 Effectiveness of Preprocessing

To assess the impact of preprocessing on features match, a simple but typical image similarity detection experiment is performed and the results are listed in Table 1. All the images used in this experiment are frames of videos from TRECVID 2011 dataset, containing about 60,000 reference images and 4000

query images. We extract SIFT features for each frame and bag-of-words is used combined with inverted files [13]. In the last step we compare distance using Euclidean distance and return 30 most similar frames for each query frame.

We can clearly find that camcording and PIP are difficult (only 61% or so) to be correctly detected compared with other TTs which can achieve an accuracy of nearly 84%. However, even the preprocessing is not perfect enough, it indeed helps to improve the feature matches, making it an essential step in our system.

Table 1. Accuracy of preprocessing, image similarity detection

	Preprocessing	Image similarity detection (no preprocessing)	Image similarity detection (with preprocessing)
camcording	78.6%	61.9%	77.4%
PIP	84.7%	60.6%	80.7%
T3,T4,T5,T6,T8,T10	–	83.3%	–

3.2 Effectiveness of Spatio-temporal Feature-Based System

In this experiment, after preprocessing the input videos, we segment the videos with a fixed length of 25 frames. After extracting the SIFT, we set the PCA coefficient number $m = 200$. In the Segment Similarity Match step, we set$k = 20$ meaning we select the top 20 closest segments for each query segment. To make a comparison we also perform an frame-based video copy detection system (with preprocessed) with SIFT extraction and to speed up the procedure we apply vocabulary tree combined with inverted file [13]. In order to evaluate our systems performance, we measure recall, precision and F1. For evaluating the computation cost, we average the time of query segments (25 frames).

Table 2. Results of video copy detection

	precision	recall	F1	Average computation
Proposed method	79.1%	88.3%	83.4%	39.3237s
Frame-based	60.5%	75.3%	67.1%	44.6943s

We can find from Table 2 that the proposed method has better result than typical frame-based method. As most of computation are finished off-line, and the features are much less than frame features, the process time is appropriate.

4 Conclusion

In this work, we propose an effective video copy detection system, which extracts spatio-temporal features instead of using spatial features and adding temporal information in a latter step. We abandon selecting frames (keyframes extracting by boundary-detection or uniformly sampling) but divide videos into segments with fixed length after preprocessing. Then a spatio-temporal interest point

detector is presented. Similar to the image case, we extract cuboid for each interest point containing most of the volume of data that contributed to the response function at that detected points. Then an extension of SIFT for cuboid representation is introduced. In order to further reduce the high-dimension of SIFT, we apply a PCA method. In the last step, we compare the similarity among segments using Euclidean distance, and determine the final detection result with a weighted-aggregate vote strategy. Experimental results show the effectiveness of our system. Moreover, we may fuse audio information to improve the detection.

Acknowledgement. This work is partly supported by NSFC (Grant No. 60935002), the National 863 High-Tech R&D Program of China (Grant No. 2012AA012504), the Natural Science Foundation of Beijing (Grant No. 4121003), and The Project Supported by Guangdong Natural Science Foundation (Grant No. S2012020011081).

References

1. Guidelines for the TRECVID 2011CD task Evaluation(OL) (2011), `http://www-nlpir.nist.gov/projects/tv2011/tv2011.html`
2. Kompatsiaris, Y., Merialdo, B., Lian, S.: TV Content Analysis: Techniques and Application. CRC Press, Taylor&Francis Group, Boca Raton, FL (2012)
3. Douze, M., Jegou, H., Schmid, C.: An Image-Based Approach to Video Copy Detection with Spatio-Temporal Post-Filtering. IEEE Transactions on Multimedia 12, 257–266 (2010)
4. Harvey, R.C., Hefeeda, M.: Spatio-Temporal Video Copy Detection. In: 20th ACM International Conference on Multimedia, pp. 35–46 (2012)
5. Bay, H., Tuytelaars, T., Van Gool, L.: SURF: Speeded Up Robust Features. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006, Part I. LNCS, vol. 3951, pp. 404–417. Springer, Heidelberg (2006)
6. Law, J., Buisson, O., Gouet, V., Boujemaa, N.: Robust Voting Algorithm Based on Labels of Behavior for Video Copy Detection. In: 14th ACM International Conference on Multimedia, pp. 835–844 (2006)
7. Dollar, P., Rabaud, V., Cottrell, G., Belongie, S.: Behavior Recognition via Sparse Spatio-Temporal Features. In: IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance, pp. 65–72 (2005)
8. Ke, Y., Sukthankar, R.: PCA-SIFT: A More Distinctive Representation for Local Image Descriptors. In: Proceedings of 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), vol. 2, pp. 506–513 (2004)
9. Barrios, J.M., Bustos, B.: Competitive content-based video copy detection using global descriptors. Multimedia Tools and Applications 62, 75–110 (2013)
10. Laptev, I., Lindeberg, T.: Space-time interest points. In: 9th IEEE International Conference on Computer Vision, pp. 432–439. IEEE Press, New York (2003)
11. Lucas, B.D., Kanade, T.: An iterative image registration technique with an application to stereo vision. IJCAI, 674–679 (1981)
12. Lowe, D.G.: Distinctive Image Features from Scale-Invariant Keypoints. International Journal of Computer Vision 60(2), 91–100 (2004)
13. Nister, D., Stewenius, H.: Scalable Recognition with a Vocabulary Tree Cover. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 2161–2168 (2006)