A MULTIMODAL VIDEO COPY DETECTION APPROACH WITH SEQUENTIAL PYRAMID MATCHING*

Yonghong Tian[1], Menglin Jiang[1], Luntian Mou[2], Xiaoyu Fang[1], Tiejun Huang[1]

[1] National Engineering Lab for Video Technology, Peking University
[2] Key Lab of Intel. Inf. Proc., ICT, Chinese Academy of Sciences
[1] {yhTian, mlJiang, xyFang, tjHuang}@pku.edu.cn, [2] ltMou@jdl.ac.cn

ABSTRACT

Content-based video copy detection over large corpus with complex transformations is important but challenging. It is not surprising that most existing methods fall short of either sufficient robustness to detect severely deformed copies or high accuracy to localize copy segments. In this paper, we propose a video copy detection approach which exploits complementary audio-visual features and sequential pyramid matching (SPM). Several independent detectors first match visual key frames or audio clips using individual features, and then aggregate the frame level results into video level results with SPM, which calculates video similarities by sequence matching at multiple granularities. Finally, detection results from basic detectors are fused and further filtered to generate the final result. Excellent performance evaluated on TRECVid 2010 copy detection task demonstrates the effectiveness of our approach.

Index Terms—copy detection, sequential pyramid matching, feature fusion

1. INTRODUCTION

Content-based video copy detection addresses the issue that automatically determines whether a query video contains a copy from a given database of reference videos and if so from where the copy comes. Here the term "copy" means a video segment derived from another video usually by visual and/or audio transformations. Nowadays, copy detection has shown great value in many video applications such as copyright control, illegal content monitoring, and so on.

However, copy detection is pretty challenging due to the following factors. First, one certain kind of feature is robust only to several kinds of modifications. The invariant features proposed in the literature include an augmented local visual feature of SIFT [1], a global visual feature based on spatio-temporal distribution of intensities [2], and an audio feature as the combination of MFCC and RASTA-PLP [3]. Second, for frame-based methods without proper temporal voting mechanism, copies are not likely to be accurately detected and precisely located. For this reason, a spatio-temporal post-filtering mechanism is presented in [1] to keep only the frame matches that are consistent with a spatio-temporal model. A 2-D Hough transform is applied to the audio frame matches to localize the copy segment [3]. Last but not least, compact feature representation and efficient index are required for a practical copy detection system. Toward this end, bag-of-words (BoW) representation and inverted index are often used [1, 3, 4].

Therefore, we propose a copy detection approach with multimodal feature fusion and sequential pyramid matching (SPM), which is shown in Figure 1. Complementary audio-visual features are employed to achieve the goal of total robustness to various transformations through later result fusion. And SPM is adopted to aggregate frame level results into video level results.

The remainder of this paper is organized as follows. Sec. 2 describes the proposed approach. Sec. 3 presents the experimental results. And sec. 4 concludes this paper.

2. PROPOSED APPROACH

This section presents the modules of our copy detection approach, namely preprocessing, basic detectors, SPM as a component of each detector, and fusion & verification.

2.1. Preprocessing

Visual key frames are obtained by uniform sampling at a rate of 3 frames per second. Audio frames are obtained by dividing the audio signal into segments of 60ms with a 40ms overlap between consecutive frames, and 4-second-long audio clips are constructed by every 198 audio frames with a 3.8 seconds overlap between adjacent clips. Additionally, Hough transform is employed to detect the Picture-in-Picture transformation, and queries asserted as non-copies will be flipped and matched again to deal with potential flipping transformation.

2.2. Basic detectors

Four detectors are constructed respectively upon two local visual features, one global visual feature and an audio feature. Each detector is briefly described as follows,

* This work is partially supported by grants from the Chinese National Natural Science Foundation under contract No. 90820003 and No. 60973055, and the CADAL project.

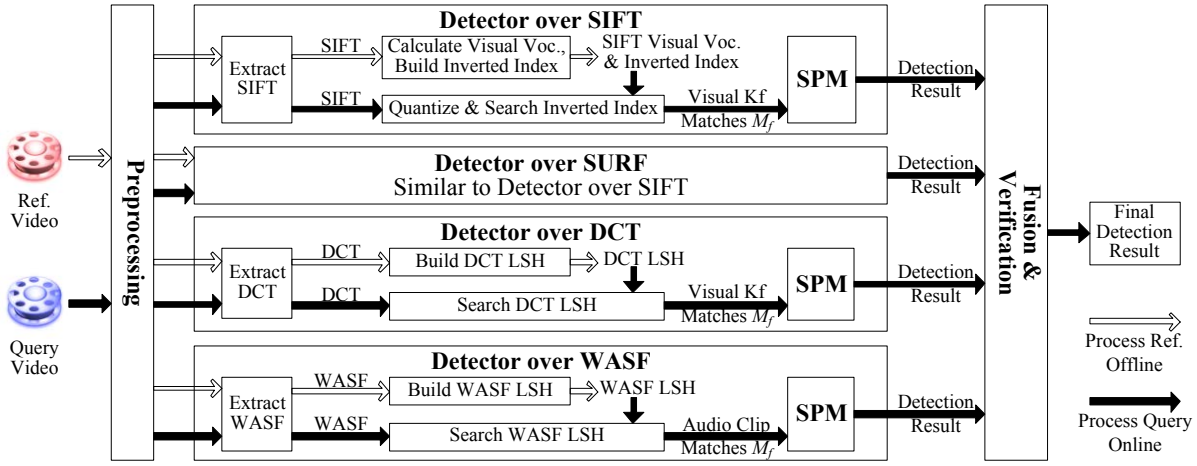

Figure 1. Overview of our video copy detection approach

leaving SPM to be presented in the next subsection.

Detectors over local visual features: two similar detectors employ the BoW framework in [4] for SIFT [5] and SURF [6] respectively. Take the detector over SIFT for example. During feature extraction, a refinement proposed in [7] is utilized to keep the most stable features. K-means algorithm is conducted on a random subset of references' features to calculate a visual vocabulary, and all the features are quantized as visual words. Position, orientation and scale of SIFT features are also used so that only features belonging to the same word with similar position, orientation and scale are regarded as matches. All these information are quantized and stored in an inverted index along with reference videos' SIFT visual words to accelerate feature matching process.

Detector over global visual feature: we propose a global visual feature based on the relationship between the discrete cosine transform (DCT) coefficients of adjacent image blocks. In particular, a key frame is firstly normalized to 64×64 pixels and divided into 64 blocks with the size of 8×8 pixels. Then a 2-D DCT is applied over each block to obtain a coefficient matrix with the same size. After that, energies of the first four subbands of each coefficient matrix (i.e. the top left four diagonals of the matrix) are computed by summing up the absolute values of corresponding DCT coefficients. Finally, a 256-bit DCT feature D_{256} can be obtained by computing relative magnitudes of the energies:

$$d_{i,j}=\begin{cases}1, if\ e_{i,j}\ge e_{i,(j+1)\%64}\\ 0, otherwise\end{cases}\quad 0\le i\le 3, 0\le j\le 63 \qquad (1)$$

$$D_{256}=<d_{0,0},\cdots,d_{0,63},\cdots,d_{3,0},\cdots,d_{3,63}> \qquad (2)$$

where $e_{i,j}$ is the energy of the *i*-th subband of the *j*-th block. Hamming distance is used as the distance metric for DCT feature and all the reference videos' DCT features are indexed by locality sensitive hashing (LSH) [8].

Detector over audio feature: Weighted ASF (WASF) [9] is used as audio feature. In brief, a 14-D feature is first extracted from each 60ms audio frame. Then, each audio clip's 198 14-D features are assembled and reduced to a 126-D WASF feature. Euclidean Distance is adopted to measure the dissimilarity between two WASF features, and all the reference videos' features are indexed by LSH for efficient feature matching.

Given a query video, each detector picks up the top K_1 ($K_1=20$) similar reference key frames (audio clips) for each query key frame (audio clip), resulting in a collection M_f which contains a series of frame level matches m_f:

$$m_f=<q,t_q,r,t_r,s_f> \qquad (3)$$

where q and r identify the query and reference videos, t_q and t_r are timestamps of the query and reference key frames (audio clips), and s_f is the similarity of the key frame (audio clip) pair. Since s_f values computed through different features are not consistent, histogram equalization is applied in each detector to make these scores more evenly distributed and comparable. Distribution of s_f values for each feature is learned on the training data set.

2.3. Sequential Pyramid Matching

Given the frame matches M_f, copies are detected through the following three steps. First, a 2-D Hough transform like [3] is conducted on M_f to vote in K_2 hypotheses $<r,\delta t>$ ($K_2=10$), where $\delta t=t_q-t_r$ specifies the temporal offset between query and reference video. Second, for each hypothesis, the begin and end of copy are identified by picking up the first and last matches m_f in M_f that accord with this hypothesis. Finally, SPM is performed on each potential video match to calculate its similarity, getting:

$$m_v=<q,t_{q,b},t_{q,e},r,t_{r,b},t_{r,e},s_v> \qquad (4)$$

which means the sequence $[t_{q,b},t_{q,e}]$ of query q is likely to be a copy from the sequence $[t_{r,b},t_{r,e}]$ of reference r with a similarity s_v. Only if s_v is above a threshold T_1, will m_v be

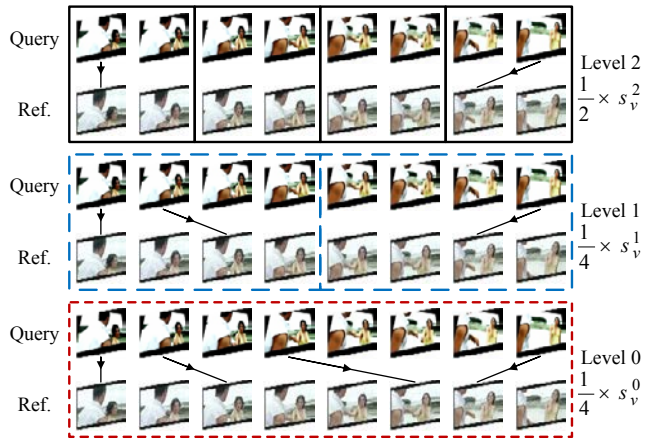

Figure 2. Toy example for a L=2 SPM

accepted as a video match. When several m_v for query q exceed T_1, only the one with the highest s_v is reserved.

Now we'll detail SPM. Intuitively speaking, all the frame matches between q and r should accord with the same δt to preserve the temporal characteristic of videos and eliminate potential false positives. But in practice such restriction may be counterproductive since strictly aligned frame matches are so few that many true positives could be dropped. To obtain a good tradeoff, inspired by spatial pyramid matching [10] which conducts pyramid match kernel [11] in 2-D image space, we adapt the kernel to 1-D video temporal space, resulting in the SPM which works by partitioning videos into increasingly finer segments and computing video similarities at each resolution (c.f. Figure 2). Besides, the SPM algorithm only needs a set of frame level matches as input, thus it is suitable for all kinds of audio-visual features and computationally efficient.

Specifically, SPM performs a series of sequence matching at level $0,\ldots,L$ (in practice $L=3$), such that the sequence $[t_{q,b},t_{q,e}]$ (along with $[t_{r,b},t_{r,e}]$) at level ℓ is divided into $D=2^\ell$ segments, namely $ts_{q,1},\cdots,ts_{q,D}$ ($ts_{r,1},\cdots,ts_{r,D}$), where key frames within corresponding segments can be matched across two sequences, i.e. the video similarity at level ℓ is given by the following formula:

$$s_{v,i}^\ell = sum\{s_f \mid < q,t_q,r,t_r,s_f > \in M_f, t_q \in ts_{q,i}, t_r \in ts_{r,i}\} \quad (5)$$

$$s_v^\ell = \frac{1}{n_f}\sum_{i=1}^{D} s_{v,i}^\ell \quad (6)$$

where n_f denotes the number of key frames (audio clips) in $[t_{q,b},t_{q,e}]$, so that s_v^ℓ is normalized to eliminate the influence of sequence length. The weight of level ℓ is set to 2^{-L} for $\ell=0$, and $2^{\ell-L-1}$ for $\ell=1,\ldots,L$, reflecting the penalization for matches in coarser levels. The final s_v is calculated by accumulating the weighted similarities from multiple levels:

$$s_v = \kappa^L = 2^{-L}s_v^0 + \sum_{\ell=1}^{L} 2^{\ell-L-1}s_v^\ell \quad (7)$$

2.4. Fusion and verification

A result level fusion is utilized to fuse the detection results from different detectors. Besides, considering that the BoW representation inevitably causes decrease in feature's discriminability, a verification module is added to calculate the similarities of certain video matches again with original (vectorial) SIFT and SURF features. More specifically, if a query is asserted as a copy by any two detectors, i.e. there're two tuples like (8) and (9) satisfying (10), it is confirmed as a copy represented by (11):

$$\bar{m}_v = < q,\bar{t}_{q,b},\bar{t}_{q,e},r,\bar{t}_{r,b},\bar{t}_{r,e},\bar{s}_v > \quad (8)$$

$$\hat{m}_v = < q,\hat{t}_{q,b},\hat{t}_{q,e},r,\hat{t}_{r,b},\hat{t}_{r,e},\hat{s}_v > \quad (9)$$

$$[\bar{t}_{q,b},\bar{t}_{q,e}]\bigcap[\hat{t}_{q,b},\hat{t}_{q,e}] \neq \varphi, [\bar{t}_{r,b},\bar{t}_{r,e}]\bigcap[\hat{t}_{r,b},\hat{t}_{r,e}] \neq \varphi \quad (10)$$

$$\begin{aligned} m_v = < q, &\max(\bar{t}_{q,b},\hat{t}_{q,b}), \min(\bar{t}_{q,e},\hat{t}_{q,e}), \\ r, &\max(\bar{t}_{r,b},\hat{t}_{r,b}), \min(\bar{t}_{r,e},\hat{t}_{r,e}), \max(\bar{s}_v,\hat{s}_v) > \end{aligned} \quad (11)$$

Query asserted as a copy by only one detector is passed to the verification module. Only if the new calculated similarity for the video match is above a threshold T_2, will it be accepted as a copy.

3. EXPERIMENTS

Experiments are conducted over the TRECVid 2010 CCD task [12]. The task contains a 420-hour-long reference database composed of videos collected from the internet and 10,976 query videos[1] which are averagely 70 seconds long. It adopts 8 visual transformations and 7 audio transformations, combining into 56 mixed transformations, which cover most practical video modifications. We test four runs, the first pair "balanced.perseus" & "nofa.perseus" follows the exact scenario presented above, while the second pair "balanced.kraken" & "nofa.kraken" omits the verification module and instead uses higher threshold T_1 in SPM to prevent false positives. Official evaluation results are summarized below.

NDCR: Normalized Detection Cost Rate synthesizes the cost for false negatives and false positives, measuring a system's detection effectiveness. Our system achieves excellent NDCR: among all the 56 transformations, it gets 39 best (lowest) "Actual NDCR" and 51 best "Optimal NDCR" for BALANCED profile, and it gets 52 best "Actual NDCR" and 50 best "Optimal NDCR" for NOFA profile. Figure 3 exhibits the details about "Optimal NDCR" for BALANCED profile, note that we achieve perfect results (NDCR=0) for 20 transformations.

The NDCR performance demonstrates that with preprocessing, the combination of multimodal features is

[1] 11,256 query videos were used at first, 280 of which were dropped by NIST later.

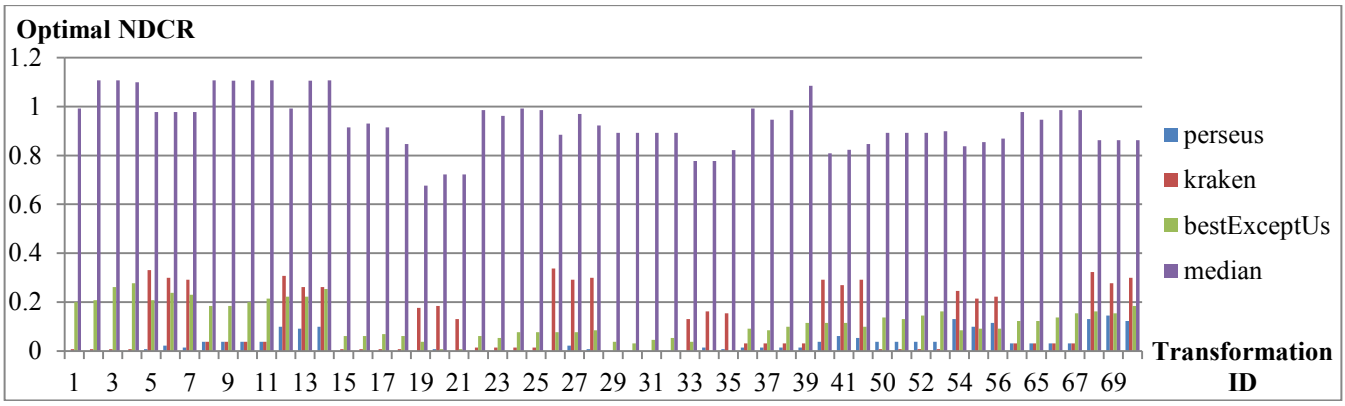

Figure 3. Optimal NDCR for BALANCED profile. The "bestExceptUs" columns present the best NDCR obtained by all the other participants, and the "median" columns present the median NDCR of all the participants (including our results)

largely robust to all kinds of transformations. Lower NDCRs achieved by "perseus" than those of "kraken" over most transformations imply the effectiveness of verification module. Furthermore, average NDCRs obtained by the DCT detector with single level sequence matching and SPM respectively are listed in Table 1, illustrating that SPM significantly outperforms single level sequence matching. On the one hand, result at single level $\ell = 0$ is unsatisfactory since the malposed frame matches are included in the video similarity calculation, thus leading to many false positives. On the other hand, sequence matching at $\ell = 4$ misses some short copies because strictly aligned frame matches are too few. In comparison, SPM obtains a better tradeoff through the multi-granularity strategy.

Table 1. DCT detector's average Optimal NDCR for BALANCED profile

L	Single Level	SPM
0 (1 ts)	0.415	
1 (2 ts)	0.346	0.312
2 (4 ts)	0.294	0.241
3 (8 ts)	0.251	**0.179**
4 (16 ts)	0.263	0.180

Mean F1: F1 evaluates the accuracy of copy localization (only for true positives). Our system achieves competitive F1 performance: for both profiles and all the transformations, the F1 values are all around 0.9 with minor deviation. Table 2 exhibits the average "Optimal Mean F1" for BALANCED profile over 56 transformations. The difference between our F1 and the best ones may be attributed to the "overcautious" strategy for copy extent computation expressed by (11) in the fusion module.

Table 2. Average Optimal Mean F1 for BALANCED profile

perseus	kraken	bestExceptUs	median
0.889	0.892	0.968	0.794

4. CONCLUSION

We have proposed a multimodal video copy detection approach with sequential pyramid matching to address the challenging issues posed by detecting video copies over large corpus with complex transformations. Official evaluation results prove that our approach is effective in both copy detection and localization. Further endeavors will be devoted to optimizing fusion strategy for better localization accuracy.

REFERENCES

[1] M. Douze, H. Jégou, and C. Schmid, "An Image-Based Approach to Video Copy Detection With Spatio-Temporal Post-Filtering", *IEEE TMM*, Vol. 12, No. 4, pp. 257-266, June 2010.
[2] C. Kim, and B. Vasudev, "Spatiotemporal Sequence Matching for Efficient Video Copy Detection", *IEEE TCSVT*, Vol. 15, No. 1, pp. 127-132, January 2005.
[3] Y. Liu, W. Zhao, C. Ngo, C. Xu, and H. Lu, "Coherent Bag-of Audio Words Model For Efficient Large-Scale Video Copy Detection", *ACM CIVR'10*, pp. 89-96, July 5-7, 2010.
[4] J. Sivic, and A. Zisserman, "Video Google: A Text Retrieval Approach to Object Matching in Videos", *IEEE ICCV'03*, pp. 1470-1477, October 13-16, 2003.
[5] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints", *IJCV*, Vol. 60, No. 2, pp. 91-110, 2004.
[6] H. Bay, T. Tuytelaars, and L. V. Gool, "SURF: Speeded Up Robust Features", *ECCV'06*, Vol. 3951, pp. 404-417, May 2006.
[7] S. Zhang, Q. Huang, G. Hua, S. Jiang, W. Gao, and Q. Tian, "Building Contextual Visual Vocabulary for Large-scale Image Applications", *ACM MM'10*, pp. 501-510, October 2010.
[8] A. Gionis, P. Indyk, and R. Motwani, "Similarity Search in High Dimensions via Hashing", *VLDB'99*, pp. 518-529, 1999.
[9] J. Chen, and T. Huang, "A Robust Feature Extraction Algorithm for Audio Fingerprinting", *PCM'08*, pp. 887-890, December 9-13, 2008.
[10] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories", *CVPR'06*, Vol. 2, pp. 2169-2178, June 17-22, 2006.
[11] K. Grauman, and T. Darrell, "The Pyramid Match Kernel: Discriminative Classification with Sets of Image Features", *IEEE ICCV'05*, pp. 1458-1465, October 17-21, 2005.
[12] A. F. Smeaton, P. Over, and W. Kraaij, "Evaluation Campaigns and TRECVid", *ACM MIR'06*, pp. 321-330, October 26-27, 2006.