PKU-IDM @ TRECVid 2011 CBCD: Content-Based Copy Detection with Cascade of Multimodal Features and Temporal Pyramid Matching *

Menglin Jiang, Shu Fang, Yonghong Tian+, Tiejun Huang, Wen Gao
National Engineering Laboratory for Video Technology, School of EE & CS, Peking University
+ Corresponding author: Phn: +86-10-62758116, E-mail: yhtian@pku.edu.cn

Abstract

Content-based copy detection (CBCD) is drawing increasing attention from both academia and industry as an alternative technology to watermarking for video identification and copyright protection. In this paper, we present a comprehensive method for detecting copies subjected to complicated transformations in a large video corpus. Basically, two core techniques are employed by our method. One is multimodal feature representation organized in a cascade architecture, which exploits the complementary characteristics of audio features, global and local visual features to keep robust to a wide range of transformations and meanwhile preserves efficiency as far as possible. The other is Temporal Pyramid Matching (TPM), which fuses frame-level similarity search results into sequence-level matching results. We have submitted two runs, i.e. "PKU-IDM.m.balanced.cascade" & "PKU-IDM.m.nofa.cascade". Official results demonstrate that the proposed approach achieved excellent NDCR and competitive Mean F1 at the cost of median Processing Time.

1. Introduction

Along with the exponential growth of digital videos and the development of video delivering techniques, content-based copy detection (CBCD) 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, query videos often suffer from severe quality decrease and even change in content, which makes it difficult to extract largely-invariant features from a copy and its original reference. Actually it's almost intractable to find a universal feature that keeps robust to all the transformations. Second, for frame-based copy detection methods without proper temporal fusing mechanism, copies are difficult to be accurately detected and precisely localized. Last but not least, compact feature representation and efficient indexing are also required for building a practical copy detection system for large, continuously expanding reference databases.

To address these challenges, we propose a copy detection approach with a cascade of multimodal features and Temporal Pyramid Matching (TPM), which is shown in Figure 1. Complementary audio-visual features are employed to achieve total robustness to various transformations and are organized in cascade architecture to improve efficiency. TPM is adopted to aggregate frame level results into video level results. Note that the improved version of SPM in [1] is renamed TPM in this article to avoid confusion. Furthermore, inverted indexing and locality sensitive hashing (LSH) are utilized to accelerate similarity search.

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

2. The Proposed Approach

This section presents the modules of our copy detection approach, namely preprocessing, basic detectors, TPM as a component of each detector, and the cascade architecture.

2.1. Preprocessing

During preprocessing, reference/query videos are first split into video and audio components. Then, 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. Visual key frames where intensity of each pixel is below a predefined threshold are dropped as black frames. Finally, additional preprocessing is dedicated to handle the Picture-in-Picture (PiP) and Flip transformations. Hough transform that detects two pairs of parallel lines is employed to detect and localize the inserted foreground videos. For those queries with PiP transformation, our system will process the foreground and the original key frames respectively. Also those queries asserted as non-copies will be flipped and matched again to deal with

* 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.

potential flip transformation.

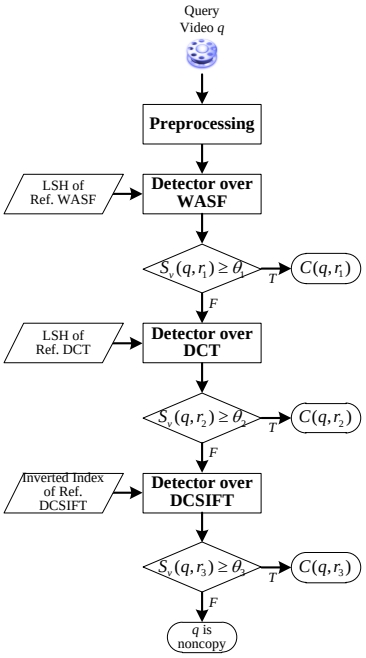

Figure 1. Overview of our video copy detection approach

2.2. Basic detectors

To keep robustness to diverse complicated transformations, we propose to exploit complementary multimodal features for representing a video. Here we put our special emphasis on the "complementary" characteristics of these features, owing to one of our basic beliefs that none of any single feature can work well for all transformations. The multimodal features used in our current implementation are a local visual feature of Dense Color SIFT (DCSIFT) [2], a global visual feature based on DCT and an audio feature named WASF [3]. Each detector is briefly described as follows, leaving TPM to be presented in the next subsection.

Detectors over Local Visual Feature: A dense color version of SIFT descriptor [4] is employed to cope with spatial content-altering transformations such as V1-Camcording, V2-Picture-in-Picture, V3-Pattern Insertion and V8-Postproduction. DCSIFT differs from SIFT in that there is no keypoint detection and localization. Instead, regular grids with overlapping (i.e. dense sampling) are used for descriptor construction. And grids with single color values are discarded. Then, SIFT descriptors are computed at points on a regular grid with spacing M pixels, here $M = 21, 33, 45$. At each grid point, SIFT descriptors are computed over circular support patches with radii $r = 10, 16, 22$ pixels. For each LAB component, the patch is divided into $3 \times 3 = 9$ subpatches and an 8-bin orientation histogram is calculated in each subpatch. Consequently, each keypoint is represented by a $3 \times 9 \times 8 = 216$ dimensional SIFT descriptors.

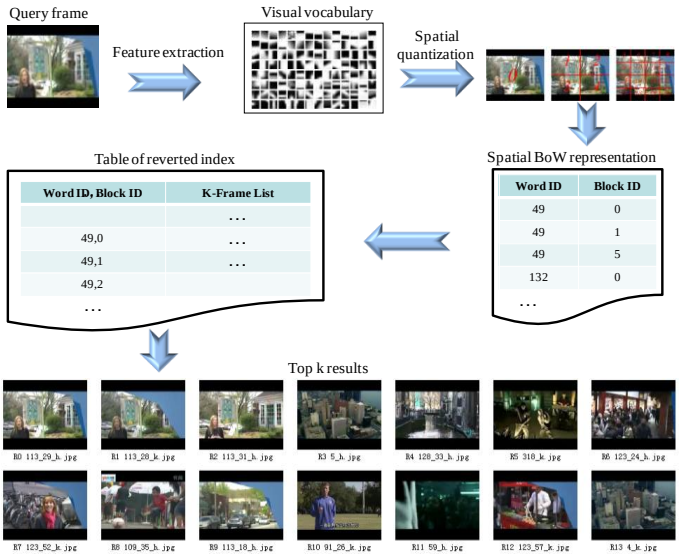

Figure 2. Keyframe retrieval using the inverted index of DCSIFT visual words and spatial information

Furthermore, the Bag of Words (BoW) framework proposed by Sivic and Zisserman [5] is applied in converting each feature vector into a visual word. During offline process, it first extracts DCSIFT features from all the reference videos' key frames. After that, K-means algorithm ($K = 800$) is implemented on a random subset (10M) of the features to calculate a visual vocabulary. Then all the reference features are quantized as visual words and stored in an inverted index. Since BoW representation might lead to loss of discriminability of descriptors, position of each keypoint is taken into account so that only keypoints mapped to the same visual word and with roughly the same position will be regarded as matches. In particular, the spatial region of a keyframe is divided into 1×1, 2×2 and 4×4 multi-granularity cells, thus the position of each keypoint is quantized into three integers (0-20) indexing the cells. Accordingly, such quantized information is integrated within the inverted index. During the online query process, DCSIFT BoW along with the additional position information is obtained from each query keyframe through the same feature extraction and quantization method. By searching the inverted index, reference keyframes that have similar appearance and spatial layout can be found efficiently. Figure 2 illustrates the keyframe retrieval process using the inverted index of DCSIFT visual words and spatial information.

Detector over Global Visual Feature: inspired by [6], we propose a global image feature based on the relationship between the discrete cosine transform (DCT) coefficients of adjacent image blocks. It has been shown that the DCT feature is robust to content-preserving transformations such as V4-Reencoding, V5-Change of Gamma and V6-Decrease of Quality. DCT also works well on several complex transformations such as V2-Picture-in-Picture with the help of preprocessing. In particular, a key frame is firstly normalized to 64×64 pixels and converted to YUV color space, keeping the Y channel only. Then the Y-channel image is divided into 64 blocks (numbered from 0 to 63) with the size of 8×8 pixels, and 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 block (c.f. Figure 3) are computed by summing up the absolute values of DCT coefficients belonging to each subband. Finally, a 256-bit DCT feature D_{256} can be obtained by computing the relative magnitudes of the energies:

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

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

where $e_{i,j}$ is the energy of the *i*-th band of the *j*-th image block. Hamming Distance is used as the distance metric. To speed up feature matching, all the reference videos' DCT features are indexed by Locality Sensitive Hashing (LSH) [7].

Subband 0	0	1	5	6	14	15	27	28
Subband 1	2	4	7	13	16	26	29	42
Subband 2	3	8	12	17	25	30	41	43
Subband 3	9	11	18	24	31	40	44	53
	10	19	23	32	39	45	52	54
	20	22	33	38	46	51	55	60
	21	34	37	47	50	56	59	61
	35	36	48	49	57	58	62	63

Figure 3. Illustration of DCT subband indexing

Detector over Audio Feature: Weighted Audio Spectrum Flatness (WASF) proposed by Chen and Huang [3] is used here to address audio transformations such as A1-mp3 Compression. It extends the MPEG-7 descriptor - Audio Spectrum Flatness (ASF) by introducing Human Auditory System (HAS) functions to weight audio data. In brief, a 14-D single WASF feature is first extracted from each 60ms audio frame. Then, each audio clip's 198 single WASF features are assembled and reduced to a 72-D integrated WASF feature. Euclidean Distance is adopted to measure the dissimilarity between two 72-D integrated WASF features, and all the reference videos' integrated WASF features are stored in LSH for efficient feature matching.

Similarity Equalization for Frame Level Retrieval: Given a query video, a 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 :

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

Where q and r identifies the query and reference video, 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 computed through different features are not consistent, histogram equalization is applied in each detector to make these scores more evenly distributed and comparable:

$$bin = \lfloor s_f \times 1000 \rfloor \tag{4}$$

$$s_f = \min\left\{1.0, \sum_{i=0}^{bin} p_i\right\} \tag{5}$$

Here, the range of similarity score $[0,1]$ is divided into 1000 bins, p_i is the frequency of the *i*-th bin, which is measured on a training data set.

2.3. Temporal Pyramid Matching

Inspired by spatial pyramid matching [8] which conducts pyramid match kernel [9] in 2-D image space, we adapt the kernel to 1-D video temporal space, leading to the concept of Temporal Pyramid Matching. Although the frames of two matched video sequences should have consistent timestamps, a certain extent of freedom is also required at video matching due to the existence of various transformations, especially temporal transformations. That is, the timestamps of two matched frames are allowed to have a moderate deviation. Therefore, TPM is proposed to partition videos into increasingly finer temporal segments and compute video similarities over each granularity (see Figure 4 for an example). The details of TPM are described as follows.

Given the candidate frame matches M_f, a 2-D Hough transform like Liu et al. [10] is first conducted on M_f to vote in K_2 ($K_2 = 10$) hypotheses $< r, \delta t >$, where $\delta t = t_q - t_r$ specifies the temporal offset between a query video and a reference video. Then, for each hypothesis, the extent of copy in the query video and the reference video, denoted by $[t_{q,b}, t_{q,e}]$ and $[t_{r,b}, t_{r,e}]$, are identified by picking up the first and the last matches m_f in M_f that accord with the hypothesis. After that, the two subsequences of $[t_{q,b}, t_{q,e}]$ and $[t_{r,b}, t_{r,e}]$ at level ℓ are uniformly divided into $D = 2^\ell$ segments respectively, namely $ts_{q,1}, \cdots, ts_{q,D}$ and $ts_{r,1}, \cdots, ts_{r,D}$, and similarity scores of frame matches within two corresponding segments across the two subsequences are accumulated to reach at the similarity of the two segments (6). And the similarity of the two subsequences at this level is obtained by averaging the pairwise segment similarity values (7).

$$s_{v,i}^{\ell} = sum\{s_f \mid < q, t_q, r, t_r, s_f > \in M_f, t_q \in ts_{q,i}, t_r \in ts_{r,i}\} \tag{6}$$

$$s_v^{\ell} = \frac{1}{n_f} \sum_{i=1}^{D} s_{v,i}^{\ell} \tag{7}$$

where n_f is the number of keyframes in $[t_{q,b}, t_{q,e}]$ used 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 $\ell = 1, \ldots, L$ (in practice $L = 3$) to penalize matches in coarser levels. Finally, the video similarity score s_v is calculated by accumulating the weighted similarities from multiple levels:

$$s_v = \kappa^L = 2^{-L} s_v^0 + \sum_{\ell=1}^{L} 2^{\ell - L - 1} s_v^{\ell} \tag{8}$$

Only if s_v is greater than or equal to a threshold T, will q be accepted as a copy. In the case that several candidate video matches meet this constraint of similarity threshold, only the one with the highest similarity score is retained. Formally, a video-level match can be expressed as follows:

$$m_v = < q, t_{q,b}, t_{q,e}, r, t_{r,b}, t_{r,e}, s_v > \tag{9}$$

which means the subsequence $[t_{q,b}, t_{q,e}]$ of a query video q is a copy originated from the subsequence $[t_{r,b}, t_{r,e}]$ of a reference video r with a similarity score of s_v. Since TPM only needs a set of frame-level matches as its input, it is suitable for various visual/audio features and computationally efficient.

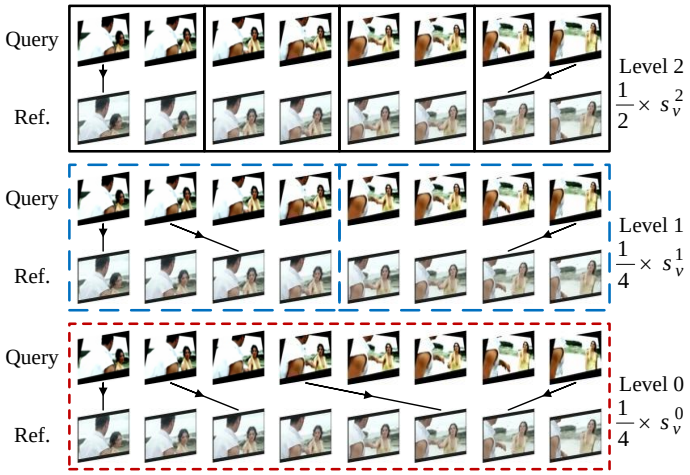

Figure 4. Toy example for a *L*=2 TPM

2.4. Cascade Architecture

After constructing three complementary audio-visual detectors which could produce individual detection results, it is still a question how to integrate them in an organism and generate a final result. In our approach proposed last year, results of basic detectors (using SIFT detector and SURF detector instead of DCSIFT detector) are first obtained and then fused into final result. Through this strategy, excellent NDCR and comparable Mean F1 are achieved at the cost of high Processing Time. This year, to achieve high efficiency, cascade architecture is proposed to combine three basic detectors discussed above. Under such architecture, a query video is first processed by the most efficient WASF detector. A positive detection result (i.e. the query contains a copy clip) leads to immediate acceptance while a negative result triggers the evaluation of the second DCT detector. Only if the query is asserted as a non-copy again by the DCT detector, will it be passed to the last DCSIFT detector. Through this strategy, most copy queries are processed only by the first two efficient detectors, thus save a major part of processing time.

3. Experimental Results

NDCR: Our system achieves excellent NDCR performance. For BALANCED profile, our system gets 34 top 1 among 56 "Actual NDCR" and 31 top 1 among 56 "Optimal NDCR"; for NOFA profile, it gets 31 top 1 among 56 "Actual NDCR" and 14 top 1 among 56 "Optimal NDCR". The detailed analysis on Actual NDCR for BALANCED profile is shown in Figure 5. Figures on the other three NDCRs are similar and not listed due to space limitation.

As to our NDCR for each transformation, results indicate that NDCRs for "simple" transformations are relatively better (lower) than those for "complex" transformations, which accords with people's intuitive sense. For instance, our NDCRs for video transformation V5 merged with audio transformations A1~A4 are all below 0.02 while the NDCRs for video transformation V10 merged with audio transformation A5~A7 are all above 0.10, as is shown in Figure 5.

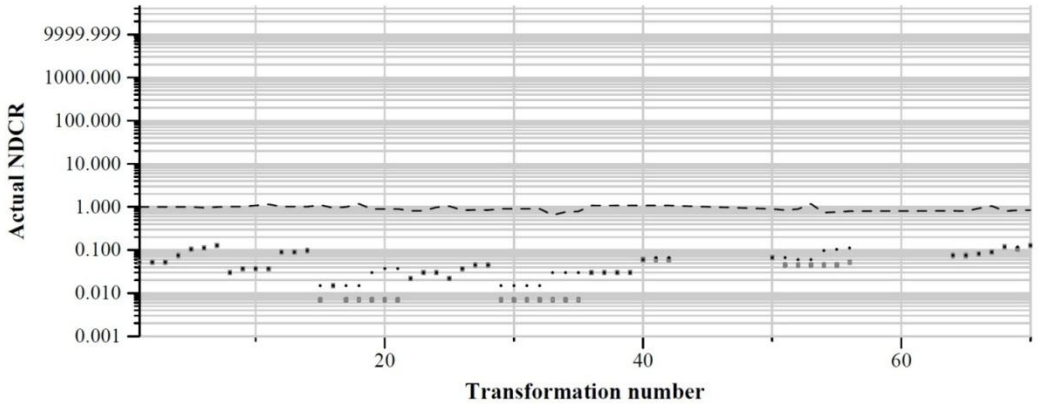

Figure 5. Actual NDCR for BALANCED profile. The dots presents our results, the boxes and dashed line present the best results and median results among all the participants respectively

Mean F1: Our system achieves competitive F1 performance. For both BALANCED and NOFA profiles and all the transformations, our F1 measures are all around 0.95 with little deviation. Take Actual Mean F1 for BALANCED profile as an example, which is shown in Figure 6, we have got 1 top 1 and the other 55 F1 are extremely close to the best ones. Besides, our F1 measures for different transformations are at the same level even though the NDCRs vary. This demonstrates that once the correct reference video is found, our TPM strategy generally localizes the copy position precisely.

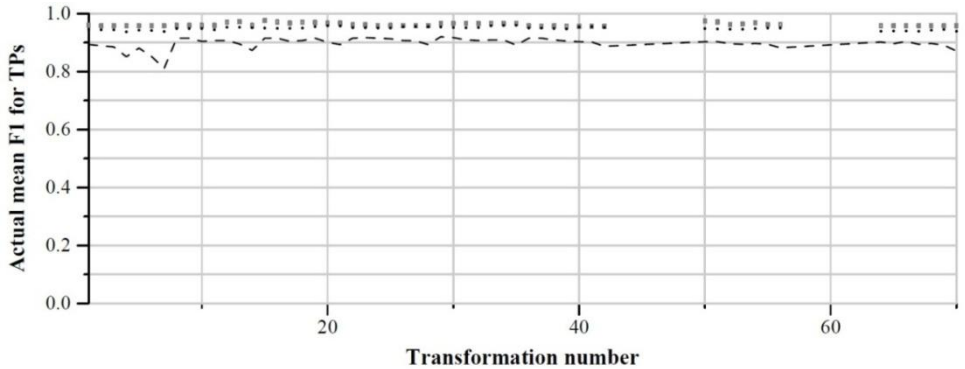

Figure 6. Actual Mean F1 for BALANCED profile

Mean Processing Time: Most of our Processing Times are shorter than the median ones of all the participants, few are longer, as is shown in Figure 7. Attention should be paid to the observation that it takes shorter time for our system to process queries with simple transformations than those with complex transformations. This is contributed by the adoption of cascade architecture and has great advantage in practical applications. Also it is worth to mention that our system is configurable, and when using only WASF and DCT detectors, it could obtain a slightly less excellent result with a small fraction of current processing time.

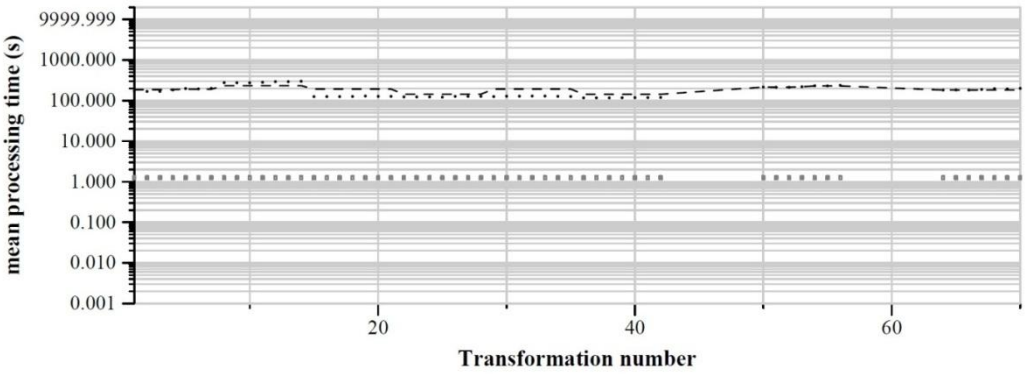

Figure 7. Mean Proc. Time for BALANCED profile

4. Conclusion

Official evaluation results show that our system outperforms other systems at most transformations in terms of NDCR and Mean F1. It demonstrates the effectiveness of the adopted strategies: multi-feature representation, Temporal Pyramid Matching and cascade architecture. Further endeavors will be devoted to introducing machine learning algorithm into the process of parameter optimization in the cascade architecture.

Reference

[1] Y. H. Tian, M. L. Jiang, L. T. Mou, X. Y. Fang, and T. J. Huang, "A Multimodal Video Copy Detection Approach with Sequential Pyramid Matching", IEEE ICIP'11, Brussels, Belgium, September 11-14, 2011.
[2] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints", IJCV, Vol. 60, No. 2, pp. 91-110, 2004.
[3] J. Chen, and T. Huang, "A Robust Feature Extraction Algorithm for Audio Fingerprinting", PCM'08, pp. 887-890, December 9-13, 2008.
[4] A. Bosch, A. Zisserman, and X. Muoz, "Scene Classification Using a Hybrid Generative/Discriminative Ap-proach", IEEE TPAMI, Vol. 30, No. 4, pp. 712-727.
[5] 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.
[6] C. Lin, and S. Chang, "A Robust Image Authentication Method Distinguishing JPEG Compression from Malicious Manipulation", IEEE TCSVT, Vol. 11, No. 2, pp. 153-168, February 2001.
[7] A. Gionis, P. Indyk, and R. Motwani, "Similarity Search in High Dimensions via Hashing", VLDB'99, Edinburgh, Scotland, pp. 518-529, 1999.
[8] 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.
[9] K. Grauman, and T. Darrell, "The Pyramid Match Kernel: Discriminative Classification with Sets of Image Fea-tures", IEEE ICCV'05, pp. 1458-1465, October 17-21, 2005.
[10] 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.