PKU-NEC @ TRECVid 2011 SED: Sequence-Based Event Detection in Surveillance Video*

Xiaoyu Fang [a], Hongming Zhang [b], Chi Su [a], Teng Xu [a], Feng Wang [b], Shaopeng Tang [b], Ziwei Xia [a], Peixi Peng [a], Guoyi Liu [b], Yaowei Wang [a], Wei Zeng [b], Yonghong Tian [a+]

[a] National Engineering Laboratory for Video Technology, School of EE & CS, Peking University

[b] NEC Laboratories, China

[+] Corresponding author: Phn: +86-10-62758116, E-mail: yhtian@pku.edu.cn

Abstract

In this paper, we describe our system for surveillance event detection task in TRECVid 2011. We focus on pair-wise events (e.g., PeopleMeet, PeopleSplitUp, Embrace) that need to explore the relationship between two active persons, and action-like events (e.g. ObjectPut and Pointing) that need to find the happenings of a person's action. Our team had participated in the TRECVid SED task in 2009 and 2010. This year the new improvements of our system are three-folds. First, we treat object detection and tracking as one problem, and integrate detection and tracking in one unified framework. That is mean "detection by tracking" and "tracking by detection". Also, we fuse multiple trackers to obtain a more accurate tracking result. Experimental results show that our system can achieve a much better precision and recall than our previous systems. Second, we propose sequence learning based method for pair-wise events detection. Visual features are extracted as a cubic feature representation and the discrimination is based on multiple relational and sequence kernels. Experimental results show that our system can detect more correct events with less false alarms. Third, a Markov-model based classifier is employed for action-like event detection. We define some states and learn the transition relation among these states to detect the event. Experimental results show our detectors are feasible and effective. Overall, we have submitted three versions of results, which are obtained by using different human detection, tracking and events detection modules. According to the results in the TRECVid SED formal evaluation, our experimental results are promising.

1. Introduction

This year we chose five events of two classes. One class is pair-wise events (e.g., PeopleMeet, PeopleSplitUp, Embrace) that need to explore the relationship between two active persons, the other is action-like events (e.g. ObjectPut and Pointing) that need to find the happening of a person's action. The diagram of our system is shown in Fig.1.

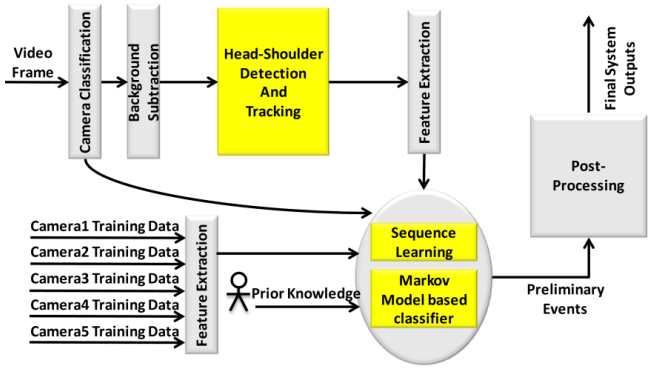

Fig.1 Diagram of our system

Three key improvements are made in the sytem than the 2010 and 2009 systems. First, we treat object detection and tracking as one problem, and integrate detection and tracking in one unified framework. That is mean "detection by tracking" and "tracking by detection". Also, we fuse multiple trackers to obtain a more ac-

* This work was cooperatively done by Peking University and NEC Laboratories, China. This work is partially supported by grants from the Chinese National Natural Science Foundation under contract No. 61035001, No. 60973055, No. 61072095 and No. 61003165, National Basic Research Program of China under contract No. 2009CB320906, and Fok Ying Dong Education Foundation under contract No. 122008. The authors would like to thank Mr. Yingkun Xu for MHT tracking method evaluation when he worked as intern in NEC Labs,China, and thank Dr. Guangyu Zhu for helpful discussion. They also would like to thank Mr. Huang Quan and Ms.Luo Yanlin from NEC Labs, China for large-scale video computing platform.

curate tracking result. Second, As the events videos are inherently sequential data, we propose sequence learning based method for pair-wise events detection. Visual features are extracted as a cubic feature representation. Instead of simply concatenating the features into a vector, we treat them as sequential data to exploit not only the discrete information from individual frames, but also the sequence and correlation information among frames. Therefore, a sequence discriminant learning method based on multiple relational and sequence kernels is employed in our system. Third, a Markov-model based classifier is employed for action-like event detection. We define some states and learn the transition relation among these states to detect the event. Experimental results show our sytem is feasible and effective. According to the results in the TRECVid SED formal evaluation, our experimental results are promising.

The remainder of this paper is organized as follows. In section 2, we describe our head-shoulder detection and tracking approach. In section 3, we present our approach for detecting different events in given surveillance video sequences. Experimental results and analysis are given out in section 4. Finally, we conclude this paper in section 5.

2. Detection and Tracking

2.1 Detection-by-Tracking and Tacking-by-Detection

Pedestrian Detection is an important step in this system. For there are many occlusions in the TRECVid corpus, we apply head-shoulder detection instead of human body detection. Many people in complex scenes will be occluded for a fairly long period. Thus, the human detection in individual frames and data-association of the detection results among several continuous frames are challenging and ambiguous. In [1] and [2], temporal coherency is involved to detection. In our system, we try to exploit temporal coherency by integrate detection and tracking in one unified framework. People-trajectories are extracted from a small number of consecutive frames and from those trajectories build models of the individual people.

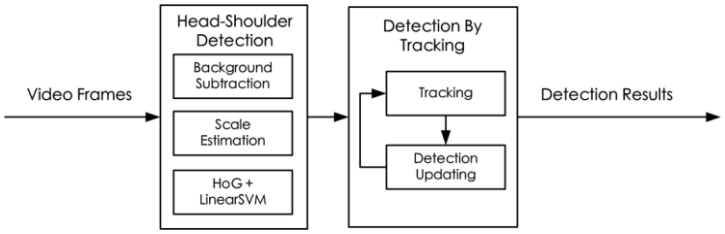

Fig.2 Framework of Detection-by-Tracking

Head-Shoulder Detection

In [3], Dalal and Triggs proved that Histograms of Oriented Gradients are powerful for pedestrian detection. In order to speed up, Zhu et al. [4] combined the cascaded rejection approach with HOG feature. They used AdaBoost to select the best features and constructed the rejection-based cascade.

In our system, we apply a simple and fast method to generate initial detection result. We use HOG feature to represent head-shoulder samples, and apply linear SVM classifier. With the coarse foreground regions extracted from background modeling module, we wipe out candidate regions that do not have enough foreground in them. Moreover, by using statistical data of each camera, we can simply estimate the possible size of person appeared in different positions. Thus, the detection process is more efficient.

In practice, we labeled about 5000 head-shoulders as positive training samples, and collected hundreds of images without head-shoulders as the source to extract negative training samples.

Head-Shoulder Detection Update

The final probability of detection $\mathrm{p}(d_N)$ of current frame N will be predicted or updated with the following equation

$$\mathrm{p}(d_N) = w_1 C(d_N) + w_2 S_f(d_N, d_{N-1}) + w_3 S_l(d_N, d_{N-1}),$$

where w_1, w_2, and w_3 are weights, d_N is the detection in frame N, $C(d_N)$ is confidence of d_N, $S_f(d_N, d_{N-1})$ is the appearance similarity (HOG) of d_N and d_{N-1}, and $S_l(d_N, d_{N-1})$ is the location and scale similarity of d_N and d_{N-1}. $S_l(d_N, d_{N-1})$ is defined by

$$S_l(d_N, d_{N-1}) = p_N\left(\frac{size_N - size_{N-1}}{size_N}\right) \times p_N(|d_N - d_{N-1}|),$$

where $size_N$ is the size of d_N, $p_N\left(\frac{size_N - size_{N-1}}{size_N}\right)$ is the scale similarity of d_N and d_{N-1}, and $p_N(|d_N - d_{N-1}|)$ is the location distance of d_N and d_{N-1}.

We set diffrent weights for different scenes. Head-shoulder detection updating will terminate when the

tracking result change. Then, if the detection results have maximum $\mathrm{p}(d_N)$ and $\mathrm{p}(d_N) > Th$ (Th is the detection threshold) they are appended to the final detection results.

Particle Filter Tracking by Detection

In the TRECVid corpus, target appearance always changes significantly. This year we use a new framework for tracking process as described by Michael D. Breitenstein[5].

Our tracking algorithm is based on estimating the distribution of each target state by a particle filter. We use a constant velocity motion model of each particle [6]. To compute the weight for a particle of the tracker, we estimates the likelihood of each particle. For this purpose, we combine information from different sources, the associated detection score, the preliminary detection results of the detection-by-tracking algorithm metioned in section 2.1, and the classifier outputs.

Considering most of the head-shoulder of pedestrians are small and blurred, we apply the online Multiple Instance Learning algorithm [6] instead of the Online Boosting algorithm in [5]. For each classifier, weak learners are selected using MIL Boost.

2.2 Head-Shoulder Detection Based on Gradient Tree Boosting

We also propose another approach using Gradient Tree Boosting [7] to detect object with high accuracy and fast speed. The essential component of the proposed approach is a cascade Gradient Boosting Tree based object detector, which uses HoG features as object representation. In order to track multiple objects in Trecvid video, we adopt Multiple Hypothesis Tracking (MHT) Method. MHT algorithm was invented by Reid [8] in the context of multi-target tracking, and was improved by Cox and Hingorani[9] by an efficient implementation.
We also propose another approach using Gradient Tree Boosting [8] to detect object with high accuracy and fast speed and adopting Multiple Hypothesis Tracking (MHT) Method.

Head-Shoulder Detection Based on Gradient Tree Boosting

Fig.3 shows the overall architecture of our object detection approach, which contains training stage and detection stage. The essential component of the proposed approach is a cascade Gradient Boosting Tree based object detector, which uses HoG (Histograms of Oriented Gradients) [4] features as object representation. During training stage, a lot of samples of object and negative images are used to select informative features and to train the object detector. The detection stage is the process to locate object instances in any given input image by using the object detector.

Gradient boosting method was invented by Jerome H. Friedman [8] in 1999 and can be used for classification problems by reducing them to regression with a suitable loss function. In our system, we use decision tree as base learner, and cascade gradient boosting as learning framework.

Multiple Hypothesis Tracking Method

In order to track multiple objects in Trecvid video, we adopt Multiple Hypothesis Tracking (MHT) Method. MHT algorithm was invented by Reid [9] in the context of multi-target tracking, and was improved by Cox and Hingorani [10] by an efficient implementation. It uses statistical data association to deal with some tracking issues, such as track initiation, track termination, and track continuation. In our system, head-shoulder detection is incorporated with MHT tracking process to construct one integrated system. For any video, the track results are computed frame by frame. We tested the system on Trecvid dataset. Table 2 shows the evaluation results.

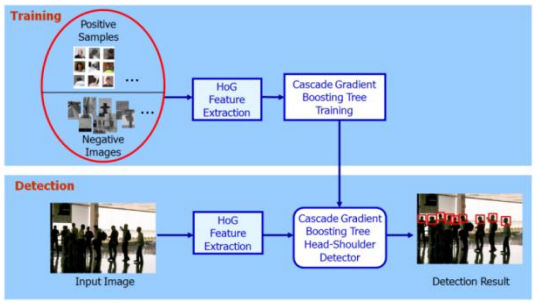

Fig.3 Object detection architecture based on Gradient Tree Boosting

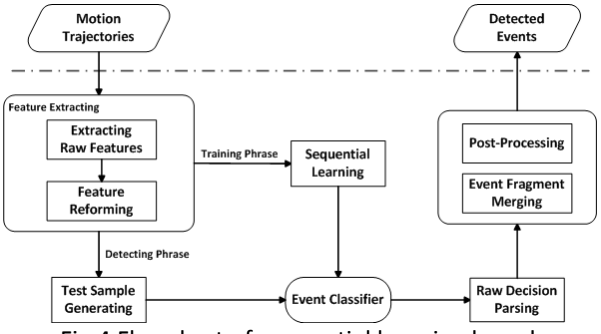

Fig.4 Flowchart of sequential learning based event detection

3. Event detection

3. 1 Pair-wise Event Detection

To detect the pair-wise events in this year's SED task, the interactive events, such as PeopleMeet, PeopleSplitUp, and Embrace, are considered as a time-variant holistic pattern, and spatio-temporal cubic feature and sequence discriminant learning method are introduced to serve the detection task.

The discriminative patterns for these three events in video sequences are inherently time sequential. However, most pervious activity recognition methods did not handle this properly with only modeling the patterns in single frames or simply concatenating them together. In our solution, the event is considered as a whole sequence and described by the spatio-temporal cubic feature. Specifically, we employ Support Vector Machine with dynamic time alignment kernel proposed in [11]. This method handles time series feature with varying length and the learning procedure is based on a maximum margin criterion. With the sequence discriminant learning method, the temporal correlations between different stages of the event are properly considered, and decisions based on integrated event sequences are reliable and semantically reasonable.

As shown in Fig.4, features are extracted based on the motion trajectories generated by human detecting and tracking module mentioned in previous sections. We first segment video sequences into several cubes, and then, according to the locations of every person in a frame, we calculate the mean absolute velocity, acceleration, distance between each pair of people and the angular separation of moving directions in each cube as the raw features. Then the extracted raw features from the same video clips (ground truth event samples for training and test samples for detecting) are transformed to structural sequence feature. Some statistics of raw features are also included into the reformed features to explicitly employ the information of the temporal dependencies over adjacent frames.

With the structural features, an appropriate implementation of SVM with dynamic time alignment kernel [8], is applied to train events classifiers and make decisions. As the raw decision is a sequence of binary decisions for each frame in a testing sample, we need to parse it into a single decision for the testing sample with the strategy like voting. As the detection task is actually transformed to a classification problem by using sliding window method to generate testing samples, the original results would be fragmental. So in the post-processing phrase, we merge the preliminary detections and introduce some prior knowledge based rules to filter out incredible detections. These rules are usually empirical restrictions such as a distance threshold between persons before "PeopleSplitUp" or after "PeopleMeet".

3. 2 Action-like Event Detection

To detect "ObjectPut" and "Pointing", a Markov-model based classifier is employed for action-like event detection. We first define some states and learn the transition relation among these states. Then a state transition model is constructed for each event. Base on the tracking results of objects, we use histogram of optical flow (HOF) for "ObjectPut" and MoSift for "Pointing" to represent their motions, which will cause transition of their states. Therefore, action-like events are recognized by classifying objects' state transition process with their models.

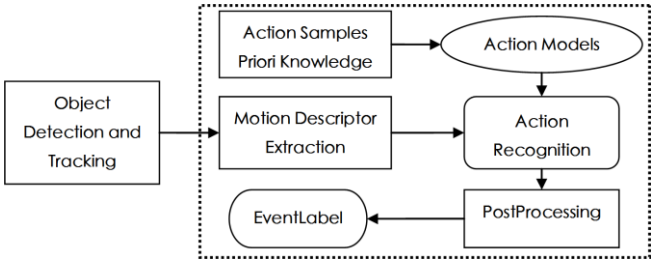

Fig.5 Action-like events detection

4. Experiment and results

Our team submitted three versions of results, which are obtained by using different human detection, tracking and events detection modules.

Table 1 Head-shoulder detection results of this year and last year

Camera1	Recall	Precision	F-score	Camera2	Recall	Precision	F-score
Last Year-SVM	0.511	0.832	0.6331	Last Year-SVM	0.373	0.615	0.4644
Last Year-MPL	0.539	0.796	0.6429	Last Year-MPL	0.560	0.773	0.6495
This Year-GTB	0.553	0.803	0.6550	This Year-GTB	0.356	0.727	0.4780
This Year-SVM	0.557	0.848	0.6724	This Year-SVM	0.372	0.785	0.5048
Camera3				**Camera5**			
Last Year-SVM	0.403	0.713	0.5149	Last Year-SVM	0.265	0.613	0.3700
Last Year-MPL	0.429	0.667	0.5222	Last Year-MPL	0.468	0.757	0.5783
This Year-GTB	0.294	0.801	0.4301	This Year-GTB	0.271	0.732	0.3755
This Year-SVM	0.423	0.756	0.5425	This Year-SVM	0.318	0.775	0.4510

Table 2 Tracking results of this year and last year

Camera1	MOTA	MOTP	Miss	FA	ID Switch
Last Year	0.321	0.591	0.510	0.134	0.035
This Year-MHT	0.368	0.571	0.486	0.134	0.012
This Year-PFT	0.364	0.567	0.472	0.154	0.010
Camera2					
Last Year	-0.135	0.599	0.791	0.317	0.027
This Year-MHT	0.151	0.601	0.680	0.160	0.009
This Year -PFT	0.213	0.607	0.644	0.132	0.011
Camera3					
Last Year	0.022	0.571	0.652	0.293	0.033
This Year-MHT	0.198	0.583	0.680	0.160	0.009
This Year-PFT	0.271	0.591	0.667	0.050	0.010
Camera5					
Last Year	-0.002	0.602	0.537	0.440	0.025
This Year-MHT	0.168	0.591	0.737	0.088	0.008
This Year-PFT	0.170	0.589	0.731	0.089	0.009

Table 1 and 2 show the comparison detection and tracking results between the best outputs of our system this year and those of last year. It can be seen from the tables that detection result is improved greatly in recall with low or no decrease in the precision. Here we introduce Multiple Object Tracking Accuracy (MOTA) and Multiple Object Tracking Precision (MOTP) [8], metrics used in PETS 2009, to evaluate overall performance. These ID switches used in MOTA are calculated from the number of identity mismatches in a frame, from the mapped objects in its preceding frame. The MOTP is calculated from the spatiotemporal overlap between the ground truth tracks and the algorithm's output tracks. Conclusion can be drawn from table 2 that our performance is improved greatly.

According to the results in the TRECVid SED formal evaluation, our experimental results are promising this year, especially for the events PeopleMeet and Embrace. Table 3 shows the comparison results between the best outputs of our system this year and those of last year. It can be seen from the table that our eSur system is greatly improved by detecting more correct events. The number of correctly detected PeopleMeet and Embrace events is two times more than last year. Meanwhile, the false alarms do not rise too much and even dramatically decreased for PeopleMeet. Table 3 also shows results of ObjectPut and Pointing detection, which we participant and submit results for the first time this year. The correctly detected number of ObjectPut and Pointing is more than that of best results of last year, and DCR of our ObjectPut is even lower; and DCR of our Pointing is also comparable with the best of last year.

Table 3 Comparison results between the best outputs of eSur this year and last year

PeopleMeet	#Ref	#Sys	#CorDet	#FA	#Miss	Act.DCR
2010's eSur	449	156	12	144	437	1.02
2011's eSur	449	2382	24	108	425	0.9820
PeopleSplitUp						
2010's eSur	187	167	16	136	171	0.959
2011's eSur	187	2988	4	192	183	1.0416
Embrace						
2010's eSur	175	925	6	71	169	0.989
2011's eSur	175	5234	15	102	160	0.9477
ObjectPut						
2010's Best	621	8	1	7	620	1.001
2011's eSur	621	50	8	41	613	1.0006
Pointing						
2010's Best	1063	113	10	26	1053	0.999
2011's eSur	1063	2113	21	123	1042	1.0206

5. Conclusion

This year we improved our system significantly in head-shoulder detection and tracking where unified framework is employed and event detection where sequence discriminant learning method is used for pair-wise events detection and Markov-Model based classifier is used for the action-like event detection. The promising results of our system this year verify the effectiveness of these improvements. However, we believe there are still large improvement spaces for our system in exploring more effective and descriptive event models.

Reference

[1] Zhipeng Hu, Yaowei Wang, Yonghong Tian, Tiejun Huang, Selective Eigenbackgrounds Method for Background Subtraction in Crowed Scenes. ICIP 2010

[2]M. Andriluka, S. Roth, B. Schiele. People-tracking-by-detection and people-detection-by-tracking. Conference on Computer Vision and Pattern Recognition (CVPR), Page(s): 1–8, 2008.

[3] A. Garcia-Martin, A. Hauptmann, J.M. Martinez: People detection based on appearance and motion models. Advanced Video and Signal-Based Surveillance (AVSS), Page(s): 256 – 260, 2011

[4] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. Conference on Computer Vision and Pattern Recognition (CVPR), 2005.

[5]Qiang Zhu, Mei-Chen Yeh, Kwang-Ting Cheng, ShaiAvidan: Fast Human Detection Using a Cascade of Histograms of Oriented Gradients. CVPR (2) 2006: 1491-1498

[6] Michael D. Breitenstein, Fabian Reichlin, Bastian Leibe, Esther Koller-Meier, Luc Van Gool. Online Multi-Person Tracking-by-Detection from a Single, Uncalibrated Camera. PAMI, 2010.

[7]Yasemin Altun, Loannis Tsochantaridis and Thomas Hofmann. Hidden Markov Support Vector Machines. ICML, 2003.

[8] J. Friedman. Greedy Function Approximation: A Gradient Boosting Machine. Ann. Statist. 29(5), 2001, 1189-1232.

[9] D. Reid, An algorithm for tracking multiple targets, IEEE Transactions on Automatic Control, Volume: 24, Issue: 6, 843 – 854, 1979

[10] I.J. Cox, S.L. Hingorani, An efficient implementation of Reid's multiple hypothesis tracking algorithm and its evaluation for the purpose of visual tracking, IEEE Transactions on Pattern Analysis and Machine Intelligence, Volume: 18, Issue: 2, 138 – 150, 1996

[11] H. Shimodaira, et al, Dynamic Time-Alignment Kernel in Support Vector Machine, Proc. Advances in Neural Information Processing Systems, 14, vol.2, pp.921-928, 2001.

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.

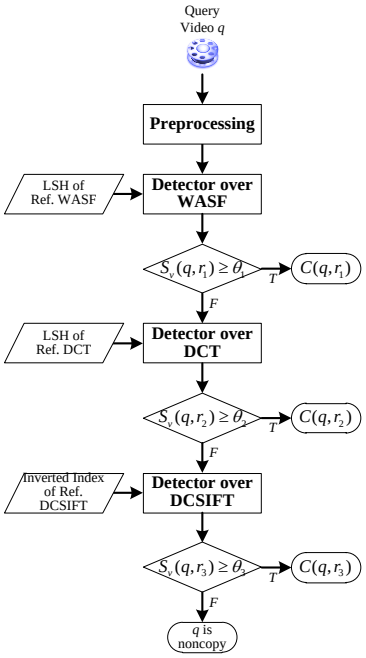

Figure 1. Overview of our video copy detection approach

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.

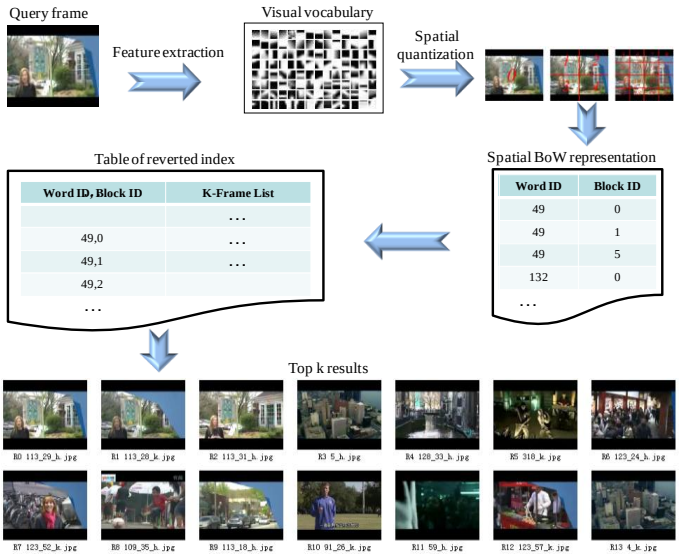

Figure 2. Keyframe retrieval using the inverted index of DCSIFT visual words and spatial information

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.

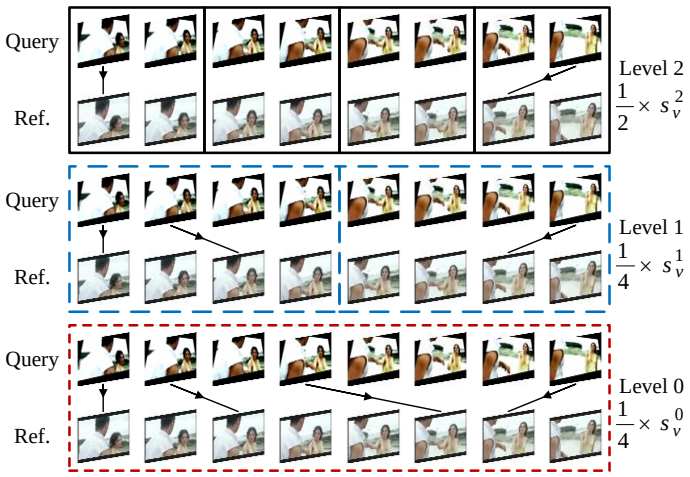

Figure 4. Toy example for a *L*=2 TPM

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.

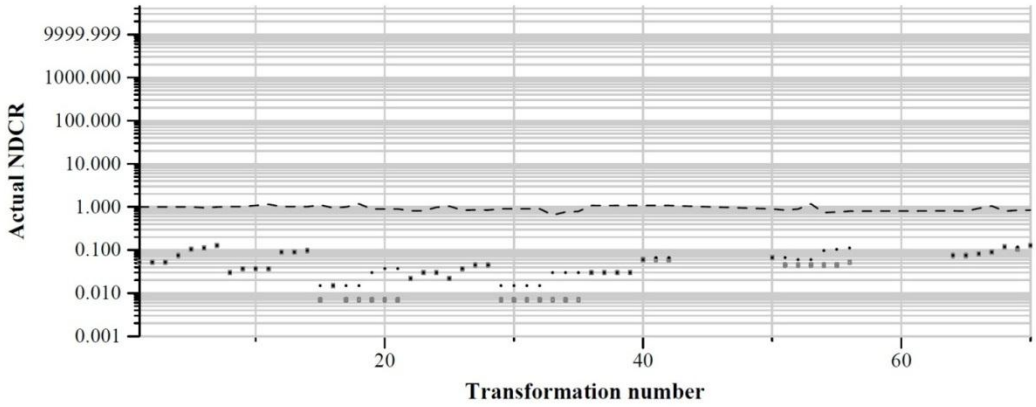

Figure 5. Actual NDCR for BALANCED profile. The dots presents our results, the boxes and dashed line present the best results and median results among all the participants respectively

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.

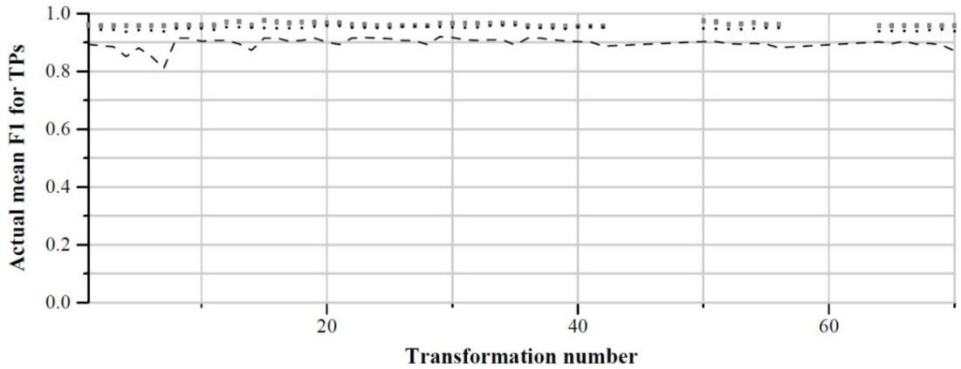

Figure 6. Actual Mean F1 for BALANCED profile

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.

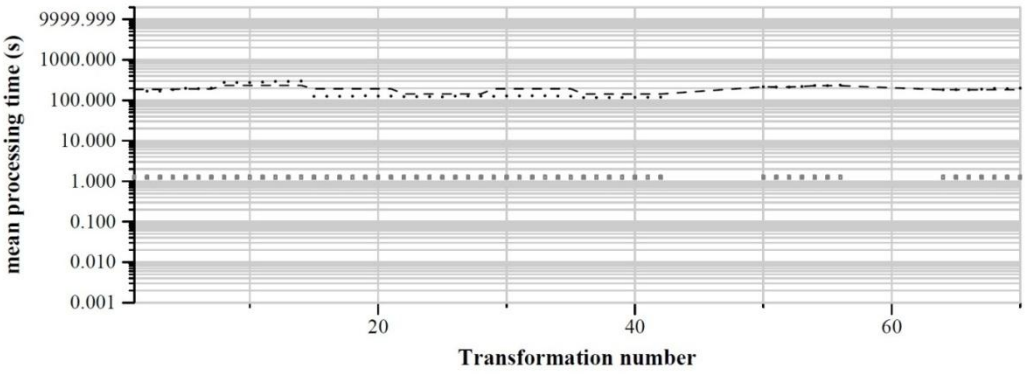

Figure 7. Mean Proc. Time for BALANCED profile

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.