Proc. 9th International Workshop on Content-Based Multimedia Indexing (CBMI 2011), Madrid, Spain, June 2011, pp. 85-90.

High-level event detection in video exploiting discriminant concepts

Nikolaos Gkalelis [1,2], Vasileios Mezaris [1], Ioannis Kompatsiaris [1]
[1] *Informatics and Telematics Institute / CERTH, Thermi 57001, Greece*
[2] *Electrical and Electronic Engineering Dept., Imperial College London, SW7 2AZ, UK*
{gkalelis, bmezaris, ikom}@iti.gr

Abstract

In this paper a new approach to video event detection is presented, combining visual concept detection scores with a new dimensionality reduction technique. Specifically, a video is first decomposed to a sequence of shots, and trained visual concept detectors are used to represent video content with model vector sequences. Subsequently, an improved subclass discriminant analysis method is used to derive a concept subspace for detecting and recognizing high-level events. In this space, the median Hausdorff distance is used to implicitly align and compare event videos of different lengths, and the nearest neighbor rule is used for recognizing the event depicted in the video. Evaluation results obtained by our participation in the Multimedia Event Detection Task of the TRECVID 2010 competition verify the effectiveness of the proposed approach for event detection and recognition in large scale video collections.

1. Introduction

The event-based indexing of video is a research direction that has started to receive particular attention, following studies in neuroscience which showed that humans remember real life using past experience structured in events [6].

During the past few years significant research has been devoted to the detection and recognition of events in several application areas. In [15], a Bag-of-Words (BoW) algorithm is combined with a multilevel sub-clip pyramid method to represent a video clip in the temporal domain; the earth mover's distance (EMD) is then applied for recognizing events defined in the TRECVID 2005 challenge dataset. In [14] motion relativity and visual relatedness is exploited and relative motion histograms of BoWs are computed to represent video clips and, similarly to [15], the EMD and Support Vector Machines (SVMs) are used to recognize video events in the TRECVID 2005 challenge dataset. In [3], knowledge embedded into ontologies and concept detectors based on SVMs are used for recognizing events in the domains of broadcast news and surveillance. In [16], three types of features, namely, spatiotemporal interest points, SIFT features, and a bag of MFCC audio words, are used to train SVM-based classifiers for recognizing the three events of the TRECVID 2010 Multimedia Event Detection Task (TRECVID 2010 MED). In [7], a wide range of static and dynamic features are extracted, such as SIFT and GIST features, 13 different visual descriptors on 8 granularities, histograms of gradients and flow, MFCC audio features, and other. These features are used for training 272 SVM-based semantic detectors and, thus, represent videos with model vector sequences. Subsequently, hierarchical HMMs are applied to recognize the TRECVID 2010 MED events.

In this paper, we detect and recognize an event by the temporal evolution of specific visual concept patterns. Specifically, we propose the use of a model vector-based approach, where visual concept detectors are used to automatically describe a video sequence in a concept space. The use of concept detectors trained on pre-existing datasets (e.g., MediaMill [12], TRECVID Semantic Indexing (SIN) Task [10]) for forming the model vectors can significantly reduce the training time, as no additional time is introduced for training event and/or concept detectors specifically for the dataset of interest. Additionally, a novel discriminant analysis (DA) technique is invoked for identifying the semantic concepts that best describe the event, thus defining a discriminant concept subspace for each event. This method extends the recently proposed subclass discriminant analysis (SDA) [18], to further improve recognition accuracy and degree of dimensionality reduction. In the resulting discriminant subspace, the nearest neighbor classifier (NN) along with the median Hausdorff distance are used to recognize an event. The proposed approach has been evaluated using the dataset provided by the TRECVID 2010 MED competition for recognizing three high-level events, namely, "batting a run in", "making a cake" and "assembling a shelter". The obtained results demonstrate the validity of this approach for representing and recognizing events in a large-scale video corpus.

The rest of the paper is structured as follows: In section 2 the event detection method is described in detail, while the experiments and results on the TRECVID 2010 MED dataset are presented in section 3. Finally, conclusions are drawn in section 4.

2. Proposed method

Let $\mathcal{X} = \{(\mathbf{X}_p, y_p), p = 1, \ldots, L\}$ be an annotated database of L video sequences belonging to one of $C-1$ different event classes $\{\mathcal{X}_1, \ldots, \mathcal{X}_{C-1}\}$ or to $\mathcal{X}_C$, which denotes the "rest of the world" class. $\mathbf{X}_p$ is the p-th video in the database and $y_p \in [1, C]$ is its event class label. Using this formulation, the task of event detection in video can be stated as follows: given the training database $\mathcal{X}$, compute a decision function $f(\mathbf{X})$ that takes as input an unlabelled video $\mathbf{X}$ and assigns it to one of the classes $\{\mathcal{X}_1, \ldots, \mathcal{X}_C\}$.

2.1. Video representation

At the video preprocessing stage, automatic analysis techniques [13] are applied to each video for segmenting it to shots, and trained concept detectors are used for associating each shot with a model vector [9, 11]. Let this set of trained concept detectors be denoted as $\mathcal{G} = \{(d_\kappa(), h_\kappa),\ \kappa = 1, \ldots, F\}$, where $d_\kappa()$ is the κ-th concept detector functional and h_κ is the respective concept label. Then, the q-th shot of the p-th video is associated with the model vector $\mathbf{x}_{p,q} = [x_{p,q,1}, \ldots, x_{p,q,K}]^T,\ \mathbf{x}_{p,q} \in \mathbb{R}^F$, where $x_{p,q,\kappa}$, computed using the response of concept detector $d_\kappa()$, is a number in the range $[0, 1]$ expressing the degree of confidence (DoC) that the κ-th concept is depicted in the respective shot. Therefore, the p-th video in the database is expressed as $\mathbf{X}_p = [\mathbf{x}_{p,1}, \ldots, \mathbf{x}_{p,l_p}], \mathbf{X}_p \in \mathbb{R}^{F \times l_p}$, where l_p is its length (in terms of shots), and consequently, the total number of model vectors (or equivalently shots) in the database will be $N = \sum_{p=1}^{L} l_p$.

2.2. Discriminant analysis

A large number of concepts may not be relevant with certain events in the database. Hence, the high dimensional model vector video representation is both noisy and expensive to use for event recognition. Instead of using it directly, we adopt a DA method [4] to derive from it a lower dimensional subspace; this process identifies the concepts that are implicitly relevant to each event, and, thus, facilitates event recognition. This is achieved using the set of the N labelled model vectors $\{(\mathbf{x}_{p,q}, y_p), p = 1, \ldots, L,\ q = 1, \ldots, l_p\}$ to learn a transformation matrix $\mathbf{W} \in \mathbb{R}^{F \times D}$, where $D << F$, that projects a model vector $\mathbf{x}_{p,q}$ to a lower-dimensional representation $\mathbf{z}_{p,q} \in \mathbb{R}^D$

$$\mathbf{z}_{p,q} = \mathbf{W}^T \mathbf{x}_{p,q} \ . \tag{1}$$

In this work, one of the three DA methods described in the next paragraphs is used for computing the transformation matrix $\mathbf{W}$ and thus for deriving the low-dimensional representations of the video sequences.

2.2.1. Linear discriminant analysis

LDA seeks directions efficient for class separation by maximizing the following objective function [4]

$$J_{lda}(\mathbf{W}) = \frac{\mathrm{tr}(\mathbf{W}^T \mathbf{S}_b \mathbf{W})}{\mathrm{tr}(\mathbf{W}^T \mathbf{S}_w \mathbf{W})} \ , \tag{2}$$

where $\mathbf{S}_b$ and $\mathbf{S}_w$ are the between- and within-class scatter matrices respectively

$$\mathbf{S}_b = \sum_{i=1}^{C} \pi_i (\boldsymbol{\mu}_i - \boldsymbol{\mu})(\boldsymbol{\mu}_i - \boldsymbol{\mu})^T \ , \tag{3}$$

$$\mathbf{S}_w = \frac{1}{N} \sum_{i=1}^{C} \sum_{\mathbf{x}_{p,q} \in \mathcal{X}_i} (\mathbf{x}_{p,q} - \boldsymbol{\mu}_i)(\mathbf{x}_{p,q} - \boldsymbol{\mu}_i)^T \ , \tag{4}$$

$\boldsymbol{\mu} = \frac{1}{N} \sum_{\mathbf{x}_{p,q} \in \mathcal{X}} \mathbf{x}_{p,q}$ is the total sample mean, and $\pi_i = \frac{N_i}{N},\ \boldsymbol{\mu}_i = \frac{1}{N_i} \sum_{\mathbf{x}_{p,q} \in \mathcal{X}_i} \mathbf{x}_{p,q},\ N_i$, are the prior, sample mean and the number of model vectors of the i-th class respectively. This criterion is equivalent to the following generalized eigenvalue problem

$$\mathbf{S}_b \mathbf{W} = \mathbf{S}_w \mathbf{W} \boldsymbol{\Lambda} \ , \tag{5}$$

where $\boldsymbol{\Lambda}$ is the diagonal matrix containing the generalized eigenvalues. In case that the number of the training model vectors N is adequately larger than their dimensionality F, the optimal transformation matrix $\mathbf{W}$ is formed by the generalized eigenvectors that correspond to the largest eigenvalues of $\mathbf{S}_w^{-1} \mathbf{S}_b$. The rank of $\mathbf{S}_b$ is at most $C-1$, therefore the $C-1$ eigenvectors can be used to formulate the optimal matrix W.

2.2.2. Subclass discriminant analysis

A fundamental assumption of LDA is that class distributions are homoscedastic, which is rarely true in practice. A more realistic strategy is to assume that there exists a subclass homoscedastic partition of the data, $\{\mathcal{X}_{1,1}, \ldots, \mathcal{X}_{C,H_C}\}$, where $\mathcal{X}_{i,j}$ denotes the j-th subclass of the i-th class, H_i is the number of subclasses in the i-th class, and H is the total number of subclasses ($H = \sum_{i=1}^{C} H_i$). Upon this assumption, subclass discriminant analysis (SDA) [18] exploits a clustering algorithm to derive a subclass partition of the data and form the following objective function

$$J_{sda}(\mathbf{W}) = \frac{\mathrm{tr}(\mathbf{W}^T \mathbf{S}_{bsb} \mathbf{W})}{\mathrm{tr}(\mathbf{W}^T \boldsymbol{\Sigma}_{\mathrm{X}} \mathbf{W})} \ , \tag{6}$$

where $\mathbf{\Sigma}_{\mathrm{X}}$ is the sample covariance matrix and $\mathbf{S}_{bsb}$ is the between-subclass scatter matrix measuring the scatter between subclasses of different classes

$$\begin{aligned}\mathbf{\Sigma}_{\mathrm{X}} &= \frac{1}{N}\sum_{i=1}^{C}\sum_{\mathbf{x}_{p,q}\in\mathcal{X}}(\mathbf{x}_{p,q}-\boldsymbol{\mu})(\mathbf{x}_{p,q}-\boldsymbol{\mu})^T \ , \quad (7)\\ \mathbf{S}_{bsb} &= \sum_{i=1}^{C-1}\sum_{j=1}^{H_i}\sum_{k=i+1}^{C}\sum_{l=1}^{H_k}\pi_{i,j}\pi_{k,l}(\boldsymbol{\mu}_{i,j}-\boldsymbol{\mu}_{k,l})\\ &\quad\cdot(\boldsymbol{\mu}_{i,j}-\boldsymbol{\mu}_{k,l})^T \ , \quad (8)\end{aligned}$$

and $\pi_{i,j}, \boldsymbol{\mu}_{i,j}$ are the prior and sample mean of the $\mathcal{X}_{i,j}$ subclass, respectively. The optimization of (6) is done using an iterative procedure, where at the r-th iteration a nearest neighbor based (NN-based) clustering algorithm is used to provide a new subclass partition of the data $\{\mathcal{X}_{1,1}^{(r)},\ldots,\mathcal{X}_{C,H_C}^{(r)}\}$. At each iteration the number of the subclasses referring to the i-th class is increased by one, $H_i^{(r)} = H_i^{(r-1)} + 1$, and, therefore, the total number of subclasses is increased by C, i.e., $H^{(r)} = \sum_{i=1}^{C} H_i^{(r)} = H^{(r-1)} + C$. Each subclass partition is evaluated using either a leave-one-out-cross-validation based (LOOCV-based) criterion, or the DA stability criterion described in [18]. Finally, the best subclass partition is chosen as the one that optimizes the respective criterion.

2.2.3. Improved subclass discriminant analysis

One drawback of SDA is that at each iteration of the algorithm the number of total subclasses is increased by C. Here we propose an improved SDA (ISDA) algorithm, where at each iteration only one additional subclass is introduced, i.e., $H^{(r)} = H^{(r-1)} + 1$. In every iteration, the repartitioning of the classes is done using an effective peak picking-based algorithm so that the efficiency of the algorithm is not sacrificed. Peak picking has been often used in several areas of digital signal processing such as radar target detection [17] and other.

A NN-based algorithm is used to sort the model vectors corresponding to the training set $\mathcal{X}$ and provide a structure in the form $\{\mathbf{x}_1^1,\ldots,\mathbf{x}_{N_1}^1,\ldots,\mathbf{x}_\nu^i,\ldots,\mathbf{x}_1^C,\ldots,\mathbf{x}_{N_C}^C\}$, where $\mathbf{x}_\nu^i$ denotes the model vector placed in the ν-th position of the array regarding the i-th class. Next, the distance between neighboring vectors is taken to provide an array of distances

$$\mathbf{d} = [d_1^1,\ldots,d_{N_1}^1,\ldots,d_\nu^i,\ldots,d_1^C,\ldots,d_{N_C-1}^C]^T \ , \quad (9)$$

where, $d_\nu^i = \|\mathbf{x}_\nu^i - \mathbf{x}_{\nu+1}^i\|_2$ is the Euclidean distance between the model vectors at the ν-th and $(\nu+1)$-th position of the array regarding the i-th class. Conceiving this last array as an one dimensional finite discrete signal, a peak picking algorithm $g(\mathbf{d},h)$ can be used to first mark all the local maxima in the signal and then return the positions of the h largest peaks in the array, $\{r_1,\ldots,r_h\} = g(\mathbf{d},h)$, where $r_1,\ldots,r_h$ are the positions of the largest local maxima in the array. In this work, we applied a very simple peak picking algorithm using the Matlab function `findpeaks` [1]. The positions of these maxima are used as landmarks for partitioning the classes to subclasses. The advantage of using this procedure is that sorting and distance computation is done only once, prior to the optimization stage. Therefore, the repartitioning of the classes at each iteration of the optimization can be performed very efficiently, resulting in a fast training procedure, as opposed to using a clustering method for each optimization step.

2.3. Event detection

Usually, instances of the same high level events differ significantly in their visual representation, duration and temporal shifts. We use a variant of the median Hausdorff distance that can successfully handle such changes between videos corresponding to instances of the same event. The oriented Hausdorff distance between two videos represented as model vector sequences $\mathbf{Z}_p$ and $\mathbf{Z}_t$ in the discriminant concept subspace is given by

$$D_H(\mathbf{Z}_t,\mathbf{Z}_p) = \operatorname*{median}_q(\min_s \parallel \mathbf{z}_{t,s} - \mathbf{z}_{p,q} \parallel) \ . \quad (10)$$

To provide a symmetric measure the following distance measure is used

$$d_H(\mathbf{Z}_t,\mathbf{Z}_p) = D_H(\mathbf{Z}_t,\mathbf{Z}_p) + D_H(\mathbf{Z}_p,\mathbf{Z}_t) \ . \quad (11)$$

Finally, this measure is combined with the nearest neighbor rule to classify an unlabelled video $\mathbf{Z}_t$

$$f(\mathbf{Z}_t) = \operatorname*{argmin}_{p\in[1,\ldots,L]} \left(d_H(\mathbf{Z}_t,\mathbf{Z}_p) \right) \ , \quad (12)$$

where L is the number of video sequences in training set $\mathcal{X}$.

3. Experiments

3.1. Dataset and evaluation

Experimentation with the presented approach was performed as part of our participation in the TRECVID 2010 MED competition. In this competition, a collection of Internet video clips is provided, containing three target events, namely, “batting a run in” (BR), “making a cake” (MC), “assembling a shelter” (AS) and a very large number of clips depicting other uninteresting events. In the training portion of the dataset, the clips depicting other uninteresting events are further subdivided to “negative instances” of each target event (i.e., a few clips that appear in the surface

Table 1. TRECVID 2010 MED video collection.

	Target events			*Uninteresting events*			
	BR	*MC*	*AS*	*NBR*	*NMC*	*NAS*	*OT*
Train. set	50	48	48	4	12	3	1581
Eval. set	47	46	47	-	-	-	1602

to be related to one of the target events, but fail to meet the exact event definition - denoted here as "negative instances of batting a run in" (NBR), "negative instances of making a cake" (NMC) and "negative instances of assembling a shelter" (NAS)), and other clips (OT). The latter are clips that do not belong to one of the target events (BR, MC, AS) and for which additionally there is no information on whether they appear in the surface to be related to one of them (as for NBR, NMC, NAS) or not. Three indicative shot keyframes, one for each of the three defined target events, are shown in Figure 1.

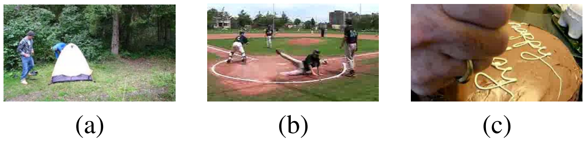
(a) (b) (c)

Figure 1. Example shot keyframes for the three target events: (a) Assembling a shelter, (b) Batting a run in, (c) Making a cake.

The video collection is divided into two data sets, an annotated training dataset consisting of 1746 clips (~ 56 hours of total duration), and an evaluation set consisting of 1742 clips (~ 59 hours). The contents of the training and evaluation datasets regarding the number of video clips for each event are shown in Table 1.

The primary evaluation metric of the TRECVID 2010 MED competition is the actual normalized detection cost (NDC). Moreover, the evaluation of each system is performed for each event independently. NDC is defined as the weighted linear combination of the missed detection (MD) and false alarm (FA) probabilities of the detection algorithm. The NDC for the i-th event is computed by

$$\mathrm{NDC}^i = C^i_{MD} P^i_{MD} P^i_T + C^i_{FA} P^i_{FA} (1 - P^i_T) \ , \qquad (13)$$

where, $P^i_{MD} = \frac{N^i_{MD}}{N^i_T}$, $P^i_{FA} = \frac{N^i_{FA}}{N^i_T}$ are the MD and FA probabilities, P^i_T is the a priori rate of event instances, C^i_{MD}, C^i_{FA} are the costs of MD and FA respectively, and N^i_T, N^i_{MD}, N^i_{FA} are the numbers of video clips, missed detections and false alarms regarding the i-th event, respectively. The values of the parameters P^i_T, C^i_{MD} and C^i_{FA} were provided in advance by the TRECVID organizers. Given these parameters, NDC essentially expresses the degree to which the considered event detection method approximates the desired compromise between missed detections and false alarms.

3.2. Experimental setup

Seven different implementations (runs) of our event detection approach were experimentally evaluated. A common feature extraction procedure for all runs was selected to provide comparable results between them. In particular, the algorithm described in [13] is used for the temporal segmentation of the video clips to shots and a SIFT-based Bag-of-Words (BoW) procedure described in [5, 2] is applied for training one SVM classifier $d_\kappa()$ for each concept h_κ. From our TRECVID 2008 experiments it has been shown that the employed concept detection method [5] ranks close to the median, hence, it generates moderately accurate concept detectors compared to the current state-of-the-art. The 101 MediaMill Challenge [12] concepts and the 130 TRECVID 2010 SIN Task concepts, along with the respective annotated datasets, were used for training the 231 concept detectors. That is, the dataset used for training the 231 visual concept detectors is completely different from the TRECVID 2010 MED one, on which the event detection method was evaluated. As described in section 2.1, each shot whose relation to the defined events needs to be examined can then be represented with a model vector comprising the responses of the 231 trained concept detectors, and subsequently the sequence of model vectors is used for representing the entire corresponding video.

The model vector sequences derived for the TRECVID 2010 MED training dataset that are described above were used for optimizing the parameters of the employed DA algorithm (e.g., computation of the projection matrix, etc.), setting the number of classes C equal to 7 (i.e., treating the "negative instance" classes NBR, NMC, NAS, as well as OT, as separate events for the purpose of training). The optimization process was guided by NDC (section 3.1); that is, the NDC was the quantity to be minimized during the training procedure.

During testing, the same procedure is followed to represent each video clip with the respective model vector sequence. These sequences are further projected in the discriminant subspace using the corresponding optimized subspace algorithm, and are classified in this space using the nearest neighbor classifier (NN) along with a variant of the median Hausdorff distance, depending on the run. Clips assigned during testing to one of the "negative instance" classes are considered to belong to class OT, for the purpose of evaluation.

The seven runs submitted to TRECVID 2010 MED are briefly described in the following: 1) IN: This is the baseline run, the Hausdorff distance and the NN rule are directly used in the 231-dimensional space for comparing the model vector sequences. 2) LDA: This run additionally exploits LDA (section 2.2.1) to project the model vectors in a discriminant subspace. 3) SDA: In this run the original SDA algorithm is used (section 2.2.2), instead of LDA. 4) ISDA1: This run uses the method described in section 2.2.3 in place of LDA or SDA, used in the previous two runs. 5) ISDA2: In this run, ISDA is firstly used to produce a confidence score for each event and for each video in the training set, and these scores are subsequently used to build a Gaussian distribution confidence score model for each event. During testing, the confidence scores produced using ISDA are weighted with the probability that these scores belong to the respective event (using the event probability models), and the maximum score is considered to indicate the underlying event. 6) ISDA3: This run is similar to run 4, however, here, the detection of an event in the discriminant subspace is done using a windowing version of the Hausdorff distance, as we explain in the following. The similarity of the test video $\mathbf{Z}_t$ with the p-th video in the database $\mathbf{Z}_p$ is evaluated, using a sliding window of length $\min\{l_t, l_p\}$ to produce $m_p = |l_t - l_p| + 1$ Hausdorff distance values denoted as $d_p^{i_p}$, $i_p = 1, \ldots, m_p$. This is done with all the videos in the database and the test video receives the label of the video in the database that yields the smaller distance $d_p^{i_p}$, i.e., the following rule is applied $\operatorname{argmin}(d_p^{i_p})$, $p = 1, \ldots, L$, $i_p = 1, \ldots, m_p$. 7) ISDA4: This run combines the procedures followed in runs 5 and 6. That is, model vectors are projected using ISDA, the windowing version of the Hausdorff distance is used to produce a confidence value for each event, and confidence values are weighted with weights resulting from the probability event models constructed during training.

3.3. Experimental results

The evaluation results for the detection of the three target events are presented in Table 2, and the respective detection error tradeoff (DET) curve [8] regarding the event "batting a run in" is shown in Figure 2. The results are given in terms of the True Positives (TP), False Alarms (FA), actual NDC, and dimensionality of the model vectors in the discriminant subspace (D). Best detection results are indicated by lower values of the actual NDC for each event separately.

From the analysis of the obtained results and the comparison with the results submitted to TRECVID 2010 MED by other institutions (including [16], which had the best performance for each of the 3 events), we conclude the following: a) Using the proposed approach very good recognition results are achieved for the event "batting a run in" ($NDC = 0.6213$) and moderately good results for the other two events. In particular, for the event "batting a run in" the achieved performance is the best among all submissions that use only visual information (as opposed to submissions that exploit a combination of visual and audio information). The good performance regarding the event "batting a run in" can be explained by the fact that several of our concept detectors correspond to concepts that are relevant to this event, such as "grass", "running", etc. On the other hand, for the other two events, although our set of trained concept detectors does not include directly relevant concepts, a moderately good performance is still achieved. b) ISDA methods provided better detection rates than SDA for the two of the three events, i.e., for the events "batting a run in" ($NDC = 0.6213$) and "making a cake" ($NDC = 0.9840$). In addition, ISDA provided a better degree of dimensionality reduction ($D_{ISDA} = 42$ while $D_{SDA} = 90$). Consequently, we can say that ISDA, compared to SDA, indeed contributes to better performance both in recognition accuracy (NDC) and system speedup. c) The SDA and the different ISDA methods performed in most cases better than the conventional LDA as well as than the method that directly classifies the test model vector sequences in the input space (IN).

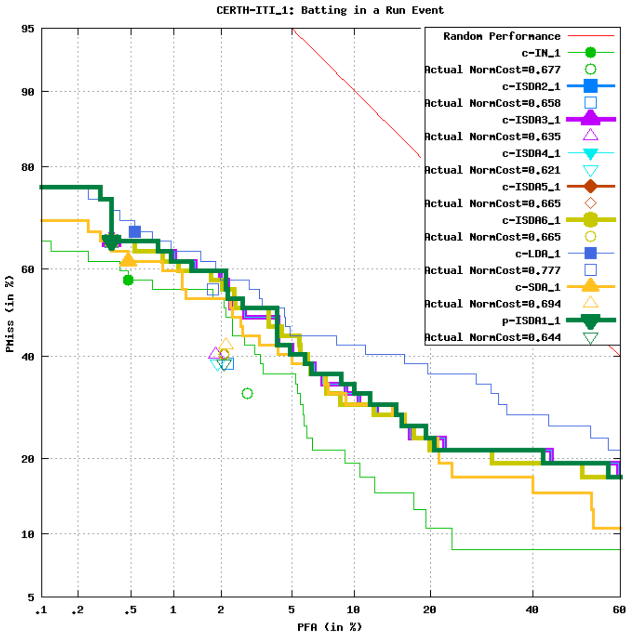

Figure 2. DET curves for "batting a run in".

4. Conclusions

In this paper an efficient model vector-based approach for high-level event detection and recognition is presented, that exploits a set of pre-trained concept detectors to de-

Table 2. Evaluation results for the three target events of TRECVID 2010 MED.

	Batting a run in				Assembling a shelter				Making a cake			
Method	*TP*	*FA*	*NDC*	*D*	*TP*	*FA*	*NDC*	*D*	*TP*	*FA*	*NDC*	*D*
IN	32	48	0.6766	231	13	52	1.1044	231	12	36	1.0127	231
LDA	21	30	0.7766	6	6	24	1.0482	6	6	43	1.1925	6
SDA	27	36	0.6936	90	11	28	**0.9692**	90	5	15	0.9979	90
ISDA1	29	35	0.6436	42	9	38	1.0871	42	6	15	**0.9840**	42
ISDA2	29	37	0.6584	42	10	41	1.0877	42	11	33	1.0117	42
ISDA3	28	31	0.6351	42	11	45	1.0958	42	5	21	1.05	42
ISDA4	29	32	**0.6213**	42	11	49	1.1255	42	8	59	1.2691	42

scribe an event in terms of its related event concepts. In this way, the use of an elaborate and computationally-costly training method for learning the events from an event-annotated dataset is avoided. Moreover, a novel dimensionality reduction algorithm is used to extract the concepts that best discriminate events. Experimental results on the TRECVID 2010 MED competition dataset have demonstrated the effectiveness of this approach.

5. Acknowledgment

This work was supported by the European Commission under contract FP7-248984 GLOCAL.

References

[1] *MATLAB, Users Guide*. The MathWorks, Inc., 1994-2001.

[2] A. Moumtzidou, A. Dimou, P. King, S. Vrochidis, A. Angeletou, V. Mezaris, S. Nikolopoulos, I. Kompatsiaris, L. Makris. ITI-CERTH participation to TRECVID 2009 HLFE and search. In *Proc. TRECVID 2009 Workshop*, pages 665–668. Gaithersburg, MD, USA, 2009.

[3] L. Ballan, M. Bertini, and G. Serra. Video annotation and retrieval using ontologies and rule learning. *IEEE Multimedia*, 17(4):80–88, Oct. 2010.

[4] K. Fukunaga. *Introduction to statistical pattern recognition (2nd ed.)*. Academic Press Professional, Inc., San Diego, CA, USA, 1990.

[5] J. Molina, V. Mezaris, P. Villegas et. al. MESH participation to TRECVID2008 HLFE. In *Proc. TRECVID 2008 Workshop*, Gaithersburg, MD, USA, November 2008.

[6] J. Zacks, T. Braver, M. Sheridan et. al. Human brain activity time-locked to perceptual event boundaries. *Nature Neuroscience*, 4(6):651–655, June 2001.

[7] M. Hill, G. Hua, A. Natsev et. al. IBM research TRECVID 2010 video copy detection and multimedia event detection system. In *Proc. TRECVID 2010 Workshop*, Gaithersburg, MD, USA, Nov. 2010.

[8] A. Martin, G. Doddington, T. Kamm, M. Ordowski, and M. Przybocki. The DET curve in assessment of detection task performance. In *Proc. Eurospeech '97*, pages 1895–1898, Rhodes, Greece, Sept. 1997.

[9] V. Mezaris, P. Sidiropoulos, A. Dimou, and I. Kompatsiaris. On the use of visual soft semantics for video temporal decomposition to scenes. In *Proc. Forth IEEE Int. Conf. on Semantic Computing (ICSC 2010)*, pages 141–148, Pittsburgh, PA, USA, Sept. 2010.

[10] A. F. Smeaton, P. Over, and W. Kraaij. High-Level Feature Detection from Video in TRECVid: a 5-Year Retrospective of Achievements. In A. Divakaran, editor, *Multimedia Content Analysis, Theory and Applications*, pages 151–174. Springer Verlag, Berlin, 2009.

[11] J. Smith, M. Naphade, and A. Natsev. Multimedia semantic indexing using model vectors. In *Proc. IEEE Int. Conf. on Multimedia and Expo (ICME '03)*, pages 445–448, Baltimore, MD, USA, July 2003.

[12] C. Snoek, M. Worring, J. van Gemert, J.-M. Geusebroek, and A. Smeulders. The challenge problem for automated detection of 101 semantic concepts in multimedia. In *Proc. ACM Multimedia*, pages 421–430, Santa Barbara, USA, October 2006.

[13] E. Tsamoura, V. Mezaris, and I. Kompatsiaris. Gradual transition detection using color coherence and other criteria in a video shot meta-segmentation framework. In *Proc. IEEE Int. Conf. on Image Processing (ICIP), MIR Workshop*, pages 45–48. San Diego, CA, USA, Oct. 2008.

[14] F. Wang, Y.-G. Jiang, and C.-W. Ngo. Video event detection using motion relativity and visual relatedness. In *Proc. ACM Multimedia*, pages 239–248, Vancouver, BC, Canada, Oct. 2008.

[15] D. Xu and S.-F. Chang. Video Event Recognition Using Kernel Methods with Multilevel Temporal Alignment. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30(11):1985–1997, Nov. 2008.

[16] Y. Jiang, X. Zeng, G. Ye et. al. Columbia-UCF TRECVID 2010 multimedia event detection: Combining multiple modalities, contextual concepts, and temporal matching. In *Proc. TRECVID 2010 Workshop*. Gaithersburg, MD, USA, 2010.

[17] A. Yildirim, M. Efe, and A. K. Ozdemir. An alternative model for target position estimation in radar processors. *IEEE Signal Process. Lett.*, 14(8):549–552, 2007.

[18] M. Zhu and A. Martinez. Subclass discriminant analysis. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(8):1274–1286, Aug. 2006.