SRI-Sarnoff AURORA System at TRECVID 2013
Multimedia Event Detection and Recounting

Jingen Liu†, Hui Cheng†, Omar Javed†, Qian Yu†, Ishani Chakraborty†, Weiyu Zhang†, Ajay Divakaran†, Harpreet S. Sawhney†, James Allan❋, R. Manmatha❋, John Foley❋, Mubarak Shah♠, Afshin Dehghan♠, Michael Witbrock♡, Jon Curtis♡, Gerald Friedland✧

† SRI-International Sarnoff, Vision Technologies Lab, 201 Washington Road, Princeton NJ 08540
❋ University of Massachusetts-Amherst
♠ University of Central Florida,
♡ Cycorp Inc, ✧ ICSI – Berkeley

Abstract

In Multimedia Event Detection 2013 evaluation, SRI Aurora team participated in EK100, EK10, and EK0 tasks with full system evaluation. We submitted 15 runs for both pre-specified events (PS-Events) and ad-hoc events (AH-Events). The majority of them achieved satisfactory results. In particular, thanks to the well-designed concept features, our EK10 system works consistently much better for both PS-Events and AH-Events. By creating the concept language model from the web source, we build our EK0 system to perform event detection without training examples. This system achieved promising results on PS-Events. In MER task, we developed an approach to provide a breakdown of the evidences of why the MED decision has been made by exploring the SVM-based event detector. Furthermore, we designed evidence specific verification and detection to reduce uncertainty and improve key evidence discovery.

1 Introduction

In TRECVID 2013[20], Multimedia Event Detection (MED) aims at detecting complex events, such as "birthday party", "wedding ceremony", "parkour" and so on. One of the common characteristic of these events is that the event videos usually cover a great diversity of visual contents including various objects, atomic human actions, physical scene, and audio information. To capture all aspects of an event, we develop various low-level static and dynamic visual features and audio features. Given sufficient training examples, such as the EK100 systems which provide 100 positive examples from each event for training, the event models built on low-level features preform reasonably well. In Section 2, we brief all the low-level features and their variations employed in our Aurora system. When the number of positive training examples decreases, however, the low-level features based event models get less generalization [7]. To make up this shortage, we also introduce a large number of concepts from which various high-level features are developed. It has been verified that high-level concept features bear better generalization capability, especially when the number of training examples is small [7]. In Section 3, we focus on describing the concepts used in Aurora system. To relieve the human manual labor in concept annotation, we introduce a semi-automatic annotation strategy by which we are capable of dramatically increasing the number of concepts. In addition, various high-level concept features have been developed in Aurora system. The evaluation results demonstrate that our EK10 systems work relatively better due to the well-designed concept features. In Section 4, we describe the process to build our EK0 system using concept features. Our EK0 system performs MED without event training examples. Multimedia Event Recounting

(MER) enables the disclosure of the evidences which result in the MED decision in AURORA system. This procedure is introduced in Section 5. Finally, in section 7, we discuss briefly the geometric and greedy fusion strategy adopted in our system, and then go through the MED13 and MER13 evaluation results.

2 Low-Level Visual and Audio Features

We developed a variety of low-level features to capture various aspects of an event, such as scene, object, action, and so on. There features are extracted either from sample frames (static features), or spatio-temporal windows of frames (i.e., XYT-volumes, dynamic features) of a video. All low-level features are quantized into visual-words/audio-words, which are used to model an event as a Bag of Words (BOW). We treat this BOW as an average feature pooling over the whole frame. However, a specific event typically has its own Region of Interests that produce most informative evidence of this event. Hence, we also employ a new strategy for spatial pooling of the low-level features, which result in an event model capturing spatial information.

2.1 Static Visual Features

Static features are computed from sampled frames (i.e., one sample every second). They are assumed to provide object or scene appearance information of an event. Following static features are extracted:

A. SIFT [2]: SIFT feature is a widely used feature descriptor for image matching and classification. The 128 dimensional SIFT descriptor is rotation invariant, which captures the local texture structure of an image. We extracted two types of SIFT features: sparse SIFT (S-SIFT) and dense SIFT (D-SIFT). S-SIFT is computed around an interest point detected by corner detector, and D-SIFT is computed for dense sampled image patches. The former one is used to describe informative patches of an object, while the latter is good to capture local patch distribution over a scene.

B. colorSIFT [3] : This feature is an extension of SIFT. Instead of computing SIFT based on intensity gradient, colorSIFT detects interest points and create descriptors on color gradients. It actually contains 3 128 dimensional vector with first one from intensity gradient and the other two from color gradient. As a result, it is able to capture both intensity and color information.

C. Transformed Color Histogram [4]: It is a normalized color histogram as describe in [4].

2.2 Dynamic Motion Features

Dynamic features are computed from detected XYT-volumes of a video. These XYT-volumes are sampled by detecting spatio-temporal interesting points or 2D corner point trajectories. They are supposed to capture the motion information of a video. But with the design of various descriptors, they are able to capture the appearance information too. The following dynamic features have been extracted.

A. STIP [5]: The Space-Time Interest Points (STIP) detects 3D interest points in the spatio-temporal domain, which is the extension of 2D Harris corner detector. It assumes the detected points have the most intensive motions in a video. STIP generate a descriptor on the intensity gradient of frames (HOG) and on the optical flow space (HOF). The final descriptor encodes both HOG and HOF feature description.

B. Dense Trajectory Feature (DTF) [6]: Rather than detecting interest point in XYT space, DTF detects 2D corner points and tracks them in a short time period. The 2D corners are usually associated with objects in a video. By analyzing the velocity or shape of trajectories, we are able to select trajectories with strong enough motions to represent the characteristics of a video. The corners are tracked by KLT track-

ing. From these trajectories, various features/descriptors can be extracted, such as shape, velocity. The AURORA adopts two types of descriptors: HOG (histogram of orientated gradient) and MBH (Motion Boundary Histogram). HOG captures the static appearance information along the trajectory, while MBH captures the motion information along the trajectory.

C. MoSIFT [11]: Motion SIFT (MoSIFT) extends the 2D SIFT descriptor to the temporal dimension. Unlike SIFT, it combines both local appearance and motion information to detect interest points. The motion information is obtained by computing optical flow.

2.3 Audio Features

A. MFCC Feature: The audio is PCM-formatted with a sample rate of 16kHz. The extracted acoustic features, using HTK[1], are the typical Mel-Frequency Cepstral Coefficients (MFCCs) C0-C19, with delta and double deltas, for a total of 60 dimensions. Each feature frame is computing using a 25 ms window, with 10 ms frame shifts. Short-time Gaussian feature warping using a three-second window is used, and temporal regions containing identical frames are removed.

B. CMU Audio Features: We also adopt another two types of AUD feature (UC, Bauds), as described in [13] and [14].

2.4 Spatial Pooling over Low-Level Visual Features

This success of Bag of Features is due to the fact that the statistics information of the bag of local features in terms of histogram of visual-words captures the major cues of events to some extent. However, one obvious disadvantage of BoF is that it ignores the spatial or temporal distribution of the features, which might be discriminative for some events. For example, the motion features usually concentrate on the central regions of a video for "sewing project", while "flash a mop" produces motion from the entire frame. Therefore, we employ the Fixed-Pattern Based Spatial Pooling: the basic idea is similar to Spatial Pyramid Match, which constructs a pyramid structure for two images, and the matches happening to the fine level will contribute more to the final match score. Instead of having a strict pyramid structure, we predefine 12 Region of Interests (ROI) including the full frame to pool features. The event training strategy for the pooling features is different to SPM. Please refer to [15] for more details.

3 High-Level Concept Features

One of the challenges for event recognition is to bridge the semantic gap between low-level features and high-level events. Concepts are directly connected to the Event Kit Descriptions. Thanks to the semantic meaning of concepts, the concept-based event representation (CBER) [7] potentially has better generalization capability, which is significantly important for event recognition, especially when only a few training examples are available (EK10 task) or even without any training examples (EK0 task). In addition, CBER offers a natural schema for multimedia event recounting. In Aurora system, we develop three types of concepts: visual concepts, audio concepts, and ASR/OCR text.

3.1 Visual Concepts

As we see, an event usually contains various objects, scenes and atomic human actions. To precisely describe the event, we need define the corresponding concepts. Our visual concepts include object, scene and action concepts. The former two are usually defined over a still image, while the latter is defined over a spatial-temporal video volume.

A. Action Concepts:

Actions are typically atomic and localized motion and appearance patterns, which are strongly associated with some specific event. Our action concepts cover general actions such as "person walking", "person running", "person climbing", as well as event specific actions such as "standing on top of bike", and "running next to a dog". Other than the 185 action concepts developed in MED12, we further collected about 110 concepts and annotated on consumer videos by human annotators. The new concepts are most likely not relevant to the PS-events when we define them.

However, the human manual annotation is very laborious. To release annotators from the time-consuming job, we have been exploring a novel strategy to achieve semi-automatic concept annotation (SAA). A regular process of concept annotation over consumer videos starts with downloading relevant videos of one concept using search queries and then annotators start to annotate the starting and ending period as the positive clips. During this procedure, we noticed that, given a specific well-defined concept, the major parts of the majority of the collected videos are relevant. Having this observation, we developed the SAA system to automatically select relevant video clips for a given concept using PageRank technique. The assumption of our approach is that the majority of the videos are relevant to the concept. A result example of the semi-automatic concept annotation is shown in figure 1 for the concept "Blowing off candles". From the snapshot, we can tell about more than 95% video clips are correctly annotated as "blowing off candles".

Figure 1: The automatically selected video clips for concept "Blowing off candles"

In addition to the self-defined action concepts (i.e., both manual annotated concepts and SAA concepts), we also employ the third-party action concepts such as UCF101 and HMDB 51 dataset, which define a lot of sport actions and daily life actions.

We employ well-established techniques to build our concept detectors. In particular dynamic features (i.e., STIP [5] and Dense Trajectory Based features [6]), and the bag-of-word representations [12] defined over codebooks of these features are used to represent action concepts. Binary SVM classifiers with Histogram Intersection kernel are used for concept classification.

B. Object and Scene Concepts:

The object and scene concepts are covered by TRECVID SIN concepts and UMass Pseudo annotations. TRECVID SIN task defines about 500 concepts which include objects such as TV screen, car, building, and scenes such as mountain, beach, street, office, and so on. Other than the use of CMU SIN 12 features, we also developed our own SIN concept features based DenseSIFT and STIP low-level features. Please refer to our SIN task report for the details [16].

Pseudo-annotation takes the advantage of object-based concepts while annotating each video frame individually. It is a semantically richer representation, because it is constructed on top of the densely sample SIFT features (i.e. low-level features). In pseudo annotation, the idea is to annotate video frames with k highest responsive object concepts among a pool of them. The distribution of highly responsive concepts enables us to rank videos for a given query. Here highly responsive concepts means that a number of concepts where they have the highest probability for a given video.

3.2 Audio Concepts

The audio concepts are either taken from CMU or annotated by our team. The Neural Network -based audio concept classification system employs the Parallel Neural Network Trainer TNet [17] technology from Brno University. It has a basic architecture which consists of two hidden layers with 1,000 neurons each and sigmoid activation functions. For the training phase a stochastic gradient descent optimizing cross-entropy loss function was used. The learning rate was updated using the "newbob" algorithm: It's kept fixed at LR=0.002 as long as the single epoch increment in cross-validation frames accuracy is higher than 0.5%. For the subsequent epochs, the learning rate is being halved until the cross-validation increment of the accuracy is inferior to the stopping threshold 0.1%. The NN weights and biases are randomly initialized and updates were performed per blocks of 1024 frames. Short-time Gaussian feature warping using a three-second window is used, and temporal regions containing identical frames are removed.

3.3 ASR/OCR Text Information

We adopted an information retrieval based approach retrieve the videos based on OCR/ASR. The event kit is used to automatically construct the query. All fields in the event kit are used for ASR query while the audio field is dropped in the OCR query. An index is created for ONR/ASR outputs of video clips using the Galago engine. A sequential dependence model is used for retrieval both OCR and ASR. The model takes both ordered and unordered phrases into account. Terms are weighted based on event kit fields. The weighting is set manually. In order to fuse OCR/ASR results with low-level and high level features, an expected-precision is computed. Since many videos do not have OCR/ASR data, a video-level fusion is carried out; where a low OCR/ASR retrieval score does not affect the feature based retrieval score, while a very high OCR/ASR retrieval score significantly increases the final score.

3.4 Concept Based Event Representation (CBER)

Given a video x, a concept detector φ_i can return a confidence value c_i. In practice, however, it is not wise to feed a long length video into a detector and get a single detection confidence for the entire video, because concept detectors are trained on single frames or short video segments. Our method uses the atomic concept detectors as filters that are applied to a given XYT segment of a video clip to capture the similarity of content to the given concept. So as a first step towards representing a video clip with concepts, each concept detector is applied to each XYT window in a video to obtain an K*W matrix C of scores, where $C_{ij} \propto p(c_i|w_j)$. Each C_{ij} is the detection confidence of concept i applied to window j.

Given the raw detection scores of concepts over the full video, the event depicted in the clip can be represented using a number of features derived from C_{ij}. One option is to select the maximum detection score C_i^{max} over all sliding windows as the detection confidence of concept detector φ_i. As a result, we are able to obtain a K-dimensional vector C^{max} to represent a video. Meanwhile, we have embedded a video into the concept space defined above. What is more, based on the K-dimensional semantic space, we also explore the following four representations:

MAX pooling: for each concept detector, only the maximum detecting score over all sliding windows is pooled to show the probability of concept given a video.

Max-Avg-Std (MAS): Other than the maximum detecting score, we believe other information of the concept distribution over a video, such as average and standard deviation, is also discriminative for an event. Hence, for each concept detector, the maximum, average, and standard deviation values over all sliding windows are selected to form MAS feature.

Bag of Concepts (BOC): Akin to the bag of words descriptors used for visual word like features, a bag of concepts features measures the frequency of occurrence of each concept over the whole video clip. To compute this histogram feature, the SVM output is binarized to represent the presence or absence of each concept in each window.

Co-occurrence Matrix (CoMat): A histogram of pairwise co-occurrences can be used to represent the pairwise presence of concepts.

Max Outer Product (MaxCoMat): Since concepts represent semantic content in a video, the max value of each concept across the whole video represents the confidence in the presence of a concept in a video. The outer product of the vector of max values of each of the concepts represents both the strength of the presence of each concept (diagonal values) as well as the strength of co-occurrence of pairwise concepts (off-diagonal values).

4 Zero-shot Learning for EK0 Task

EK0 task is to conduct event detection without any training examples. The only information available is the event kit which provides the description of the target events. We developed a system which leverages the open knowledge source such as Wikipedia to bridge the gap between the event kit and the CBER models and available OCR/ASR text. As a result, our system is able to achieve good performance using the sequential dependence model [18] given only OCR/ASR information and concept detection results. This model assumes dependencies between neighboring words without modifying order and achieves substantial gains in common text collections. In this section, we briefly describe the major steps to build our system.

4.1 Expanding Textual Descriptions

Since the event kit is the only input in the EK0 task, the specific text used is the key to our performance, one of our focuses is to improve the textual descriptions of the events. We replaced the name of the event with a short query. Then, we automatically removed common phrases based upon the Lemur 418 English-word stop-list. Using the name and short description fields, we ran these queries against Wikipedia, adding a field of pseudo-relevance feedback terms.

4.2 Concept Language Model and Selecting Concepts

One challenge for EK0 task is to determine which model to be used for a particular query, i.e., the target event. The query will contain only textual features, so we need a process to select image and video features. Given a query like "Birthday Party", it is useful to know that the detector focused on "blowing out candles on a cake" is more relevant than "person raking leaves," given the textual description of this event.

To select good detectors, we took an information retrieval approach, and considered that we want to retrieve, or rank all the detectors given a query. To achieve this ranking, we built up language models or documents for each concept through searching the web corpus. The top results from these searches were used as the language model for that particular action concept or video feature.

4.3 Event Query and Fusion Methodology

Having the concept language models, we are able to select the top few relevant concepts for an event query. For each methodology which is used to create the concept language model, we keep the concepts in separate partitions. Using these "concept types" separately allow us to learn the liability of each concept set with respect to our training data and the task. With the selected concepts in each partition, we are able to rank all the videos with respect to the matching concepts, and produce a handful of concept ranked lists.

Given the results returned by different partitions, we need to fuse them, as well as the returns by OCR and ASR source. As Galago is used for all our searching needs, the returned scores are rank-safe approximations of log probabilities, which are further normalized by Max/Min approach. Given the multiple ranked lists, our system extracts summary and per-list features. The summary features include the number of matches above a given rank as well as the sum of reciprocal ranks and the sum of reciprocal scores. Each list also includes reciprocal rank and score features. A linear model of these features is trained using the top 3000 results from each list using the RankLib package [19], and then RankLib is used to generate our final ranked list for test and evaluation purposes.

5 Multimedia Event Recounting

Event recounting is to describe the spatial and temporal details of why the event detection decision has been made. Multimedia event is typically a complex activity occurring at a specific place and time. On the other hand, a video may contain a lot of other irrelevant information as well. The recounting captures key observations regarding the scene, people, objects, and activities pertaining to the event occurrence. Such recounting provides user a semantic description that is useful to perform further analyses. As the concept features that we are using by definition contain semantic information, we can directly use the concept features to perform recounting.

As our event classification is based on Support Vector Machines (SVMs), we present an approach to perform the recounting in the context of SVMs. Given the feature vector $x \in R^n$ where n is the feature dimension, the SVM decision function $h(x)$ can be represented as follows,

$$h(x) = \sum_{l=1}^{m} \alpha_l K(x, x_l) + b \quad (1),$$

where x_l is one of *m* support vectors, ie. $l = 1, \ldots, m$. $K(x, x_l)$ is the kernel value between x and x_l. α_l is the signed weight of x_l and b is the bias. If the kernel functions have the following form, $K(x, z) = \sum_{i=1}^{n} f(x_i, z_i)$, where f is the function and x_i and z_i are the ith feature value of x and z. For example, intersection kernel satisfies such a form where $f_{INT} = \min(x, z)$. Linear kernel also follows this form. Now the decision function can be rewritten as follows,

$$h(x) = \sum_{i=1}^{n} \sum_{l=1}^{m} \alpha_l f(x_i, z_i^l) + b,$$

where $z_i^l$is the ith feature value of lth support vector. Supporse $h_i(x) = \sum_{l=1}^{m} \alpha_l f(x_i, z_i^l)$, we can decompose the decision value of $h(x)$ as

$$h(x) = \sum_{i=1}^{n} h_i(x) + b,$$

where $h_i(x)$ encodes how encodes how much ith feature contributes towards the final decision value. For our event recounting application, as each feature has semantic information, we are able to retrieve the important evidences by sorting $h_i(x)$. We have shown our recounting approach in the context of SVM classifiers. In fact, the approach can be applied to any additive classifiers as in Eq 1, which covers a wide spectrum of classification approaches.

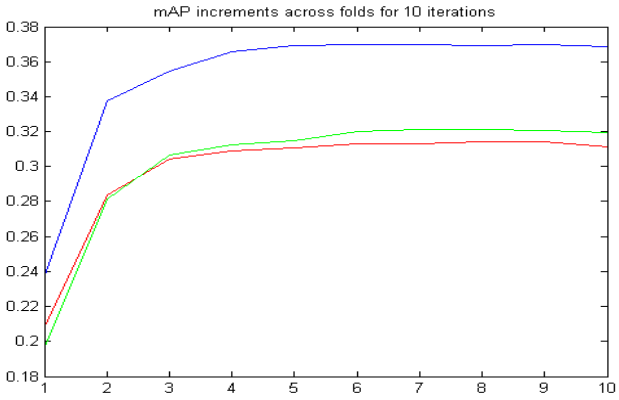

Figure 2. Plot of mAP changes with greedy added features during the fusion process of three folds created for AH-Events evaluation. Each curve corresponds to one fold.

6 Experiments

6.1 Training/Testing Methodology

We adopt the Support Vector Machine (SVM) as our basic classifiers and use intersection kernel for all histogram-based features and RBF (Radial Basis Function) kernel for concept-based features. Other SVM parameters are default values. We apply L1 normalization to histogram-based features. Event videos are used as positive samples and all non-event videos are used as negative samples to train a binary classifier for each event independently. Each classifier outputs a probability of detection as a score. LibSVM [10] is used as the SVM solver.

Standard Training/Testing Evaluation Folds: We follow the MED13 evaluation plan, and use the exact positive and negative videos specified in the evaluation package to training our event models. All training process adopts the same 5K background videos as the negatives.

6.2 Fusion Approach

Our basic fusion strategy is geometric mean based fusion. Our system produces several dozens of low-level and high-level features, but not all of them are reasonable work well. For the PS-Events, we use almost all features which perform reasonably well on the MEDTest dataset. For AH-Events, to select a subset of classifiers, we optimize the mean average precision (mAP) across all events. The idea is to search for classifiers which capture complementary characteristics of the data and hence give an optimal mAP. Our algorithm is based on recursively searching for the next best classifier. We start with one feature classifier (which individually provides maximum mAP) and greedily combine a new feature classifier that maximizes the average mAP score across all events. This process generates a classifier path. As we can see from Figure 2, which plots the mAP increments for three folds of EK100 evaluation for AH-Events,

we observe that, the mAP first increases for 6-10 features, then plateaus out and then decreases as more irrelevant features are added to the ensemble. We use the plateau as a threshold to threshold the feature paths, and then use the union of feature paths across all folds as our classifier ensemble. Finally, the classifiers in the ensemble are fused using geometric mean, which is equal to the nth root of the product of the scores, where n is equal to the number of classifiers.

In the greedy fusion process, we observed that most good features are selected. In particular, comparing

mAP		FullSys	ASRSys	AudioSys	OCRSys	VisualSys
PS-EVENTS	EK100	24.70%	3.00%	0.80%	3.70%	22.50%
	EK10	13.70%	3.00%	0.90%	3.70%	12.40%
	EK0	7.00%	3.00%	0.20%	3.70%	6.50%
AH-EVENTS	EK100	24.20%	3.90%	9.60%	4.30%	20.40%
	EK10	14.40%	3.90%	5.40%	4.30%	10.20%
	EK0	1.40%	3.90%	0.20%	4.30%	0.60%

Table 2. MED 13 evaluation results for both PS-Events and AH-Events in term of mean Average Precision.

Accuracy	ObsTextScore	PRRT
73.26%	158.00%	148.95%

Table 3. Multimedia Event Recouting (MER 13) evaluation results.

the selected feature list of EK100 and EK10, we found that EK10 fusion selects more concept features generated from various concept datasets. This scenario validate our conjecture that high-level concept features enable better recognition when fewer training examples available.

6.3 MED13 Results and Discussion

All the computations reported in this notebook were performed on the SRI-Sarnoff AURORA system. This system comprises of a number of servers with web interfaces for managing experiments run over a distributed computational pipeline, annotating training data and just browsing the datasets. The computational pipeline currently consists of about 350 AMD Opteron nodes with 5GB RAM per node as well as a number of nVidia Tesla M2050 GPUs and is based on HTConder which is designed for handling the dependency between different tasks.

In MED13 [20], we submitted 5 runs (i.e., Full System, ASR System, Audio System, OCR System, and Visual System) for each of the three training modes (i.e., EK100, EK10 and EK0) for both PS-Events and AH-Events. The mean Average Precision (mAP) of each run is listed in Table 2. As comparing to other teams, most of our runs achieved satisfactory results. Relatively speaking, our EK10 system works particularly better for both PS-Events and AH-Events.

Our MER system obtains 73.26% accuracy. More details for MER shown in Table 3. As comparing to other runs in MER13, we did very well. This is due to our MER system is built on the Concept Based Event Representation [7].

7 Acknowledgement

This work has been supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior National Business Center contract number D11-PC20066. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes not with-standing any copy-

right annotation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DOI/NBC, or the U.S. Government.

References

1. S. Young, "The HTK hidden markov model toolkit: Design and philosophy", Entropic Cambridge Research Laboratory, Ltd, vol. 2, pp. 2–44, 1994.
2. D. Lowe, Distinctive image features from scale invariant key-points. IJCV, pp. 91-110. 2004
3. K. E. Sande, T. Gevers, C. G. Snoek, Evaluating color descriptors for object and scene recognition. TPAMI, 2010.
4. G.J. Burghouts and J.M. Geusebroek, Performance Evaluation of Local Color Invariants, CVIU, vol. 113, pp. 48-62, 2009.
5. I. Laptev and T. Lindeberg. Space-time interest points. ICCV, pages 432 – 439, 2003.
6. H. Wang, A. Klser, C. Schmid, and C. L. Liu. Action recognition by dense trajectories. CVPR, 2011.
7. J. Liu, Y. Qian, et al., Video event recognition using concept attributes, WACV, 2013.
8. Y. Qian, J. Liu, et al., Multimedia event recounting with concept base representation, ACM MM 2012.
9. A. Tamrakar, S. Ali, Q. Yu, J. Liu, O. Javed, A. Divakaran, H. Cheng, and H. Sawhney, Evaluation of low-level features and their combinations for complex event detection in open source videos, CVPR 2012.
10. C.-C Chang and C.-J. Lin LIBSVM : a library for support vector machines. ACM T-IST, pp. 1-27.2011
11. M. Chen and A. Hauptmann MoSIFT: Reocgnizing Human Actions in Surveillance Videos. CMU-CS-09-161, 2009.
12. J. Liu, J. Luo, and M. Shah, Recognizing realistic human actions from videos "in the wild". CVPR 2009.
13. S. Chaudhuri, B. Raj. Unsupervised Structure Discovery for Semantic Analysis of Audio, Neural Information Processing Systems (NIPS), 2012
14. S. Chaudhuri, B. Raj., Unsupervised Hierarchical Structure Induction For Deeper Semantic Analysis of Audio. IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP) 2013
15. H. Cheng, J. Liu, O. Javed, et al., SRI-AURORA System at TRECVID 2012, Multimedia Event Detection and Recounting, TRECVID 2012 Evaluation Workshop, 2012.
16. UCF SIN Task report
17. K. Vesely, L. Burget, and F. Grezl, Parallel Training of Neural Networks for Speech Recognition, in Proceeding of Interspeech, 2010
18. Metzler, Donald, and W. Bruce Croft. A Markov random field model for term dependencies. In Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 472-479. ACM, 2005.
19. https://sourceforge.net/p/lemur/wiki/RankLib/
20. P. Over, G. Awad, M. Michel, J. Fiscus, G. Sanders, W. Kraaij, A.F. Smeaton, and G. Quénot, TRECVID 2013 -- An Overview of the Goals, Tasks, Data, Evaluation Mechanisms and Metrics, Proceedings of RECVID 2013, NIST, USA