HAL
open science

The AXES submissions at TrecVid 2013

Robin Aly, Relja Arandjelovic, Ken Chatfield, Matthijs Douze, Basura Fernando, Zaid Harchaoui, Kevin Mcguiness, Noël O'Connor, Dan Oneata, Omkar Parkhi, et al.

To cite this version:

Robin Aly, Relja Arandjelovic, Ken Chatfield, Matthijs Douze, Basura Fernando, et al.. The AXES submissions at TrecVid 2013. 2013. hal-00904404v1

HAL Id: hal-00904404

https://inria.hal.science/hal-00904404v1

Submitted on 19 Nov 2013 (v1), last revised 26 Mar 2014 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

The AXES submissions at TrecVid 2013

Robin Aly[1], Relja Arandjelović[3], Ken Chatfield[3], Matthijs Douze[6], Basura Fernando[4], Zaid Harchaoui[6], Kevin McGuinness[2], Noel E. O'Conner[2], Dan Oneata[6], Omkar M. Parkhi[3], Danila Potapov[6], Jérôme Revaud[6], Cordelia Schmid[6], Jochen Schwenninger[5], David Scott[2], Tinne Tuytelaars[4], Jakob Verbeek[6], Heng Wang[6], Andrew Zisserman[3]

[1]University of Twente [2]Dublin City University [3]Oxford University
[4]KU Leuven [5]Fraunhofer Sankt Augustin [6]INRIA Grenoble

***Abstract*— The AXES project participated in the interactive instance search task (INS), the semantic indexing task (SIN) the multimedia event recounting task (MER), and the multimedia event detection task (MED) for TRECVid 2013. Our interactive INS focused this year on using classifiers trained at query time with positive examples collected from external search engines. Participants in our INS experiments were carried out by students and researchers at Dublin City University. Our best INS runs performed on par with the top ranked INS runs in terms of P@10 and P@30, and around the median in terms of mAP.**

For SIN, MED and MER, we use systems based on state-of-the-art local low-level descriptors for motion, image, and sound, as well as high-level features to capture speech and text and the visual and audio stream respectively. The low-level descriptors were aggregated by means of Fisher vectors into high-dimensional video-level signatures, the high-level features are aggregated into bag-of-word histograms. Using these features we train linear classifiers, and use early and late-fusion to combine the different features. Our MED system achieved the best score of all submitted runs in the main track, as well as in the ad-hoc track.

This paper describes in detail our INS, MER, and MED systems and the results and findings of our experiments.

I. Introduction

This paper describes the third participation of the EU Project AXES at TRECVid . The AXES project aims to connect users and content from large multimedia archives by means of technology. The project partners involved in this year's participation (with references to earlier participations) were: 1) Dublin City University (CLARITY: Center for Sensor Web Technologies) [13], [37]; 2) University of Twente [2], [3]; 3) Oxford University [34]; 4) KU Leuven [24]; 5) Fraunhofer; and 6) INRIA Lear [15], [6].

Since AXES is about bringing users, technology, and content together, we conducted interactive user experiments in the instance search (INS). Additionally, because users often search for events in multimedia content, we participated in the multimedia event detection task (MED) and multimedia event recounting (MER) tasks. The goal of the MER task is to provide the users with a better understanding of why certain videos were returned for a MED event query. To this end the system returns a short-list of video fragments that are most relevant to the event of interest, and a description of why these fragments are relevant. We addressed this task by applying the MED system over small video fragments of several lengths in a sliding-window fashion, and report the highest scoring fragments. To generate the description of the fragments, we reported the feature source that contributed most to the classification score, and report the highest scoring SIN concepts in the fragment.

For MED we developed an approach based on three low-level descriptors: SIFT [19] and color [8] features for static visual appearance, camera-stabilized dense trajectory features [36] for motion, and MFCC for audio. We used two high-level features: optical character recognition (OCR) with MSER regions [20] to capture written words, and automatic speech recognition (ASR) to capture spoken words. The local low-level audio-visual descriptors were aggregated into high dimensional video-level signatures using Fisher vectors [29]. We use late-fusion to combine classifiers trained over the individual feature channels. The system used for SIN is similar, but does not use OCR and ASR.

This paper is structured as follows: Section II describes the methods and the system we developed for this year's INS participation, including the system architecture and the user interface. Sections III, IV, and V describe our participation in the MED, SIN, and MER tasks respectively. Section VI describes the experiments and discusses the results and findings. Section VII summarizes this paper.

II. INS Participation

In this section we describe the system we developed for this year's INS participation. We used a service oriented architecture for this year's TRECVid participation, see [21] for details. The central component of the system is our LIMAS service that merges search results from several retrieval services that each produce a confidence score for each shot whether it is relevant. The scores are then fused (combined) according to a single confidence score, producing a list of retrieval units (videos or shots). This list is then sent back to the user interface. In the following, we first describe the individual retrieval services, the fusion scheme we used, and the employed user interface.

A. Metadata

We stored the available text for each shot in a text index. One of the indexes featured the text extracted from the closed-captions, a more reliable source than ASR which was used in previous years. At query time, the standard Lucene retrieval function was used to calculate a confidence retrieval score for each retrieval unit if the query contained any text terms. We used Lucene version 3.1.2 [33] in our experiments.

B. On the fly instance retrieval

The aim of the instance retrieval system is to quickly retrieve key-frames which contain queried general classes of objects (e.g. all *cars* in a dataset, or all examples of *gothic architecture*). The query is specified by entering a text term which is used to train a model for the query on-the-fly.

Like last year, our system is based on the on the fly training of a discriminative classifier, and so in addition to the feature vectors for the dataset itself, features for negative and positive training data related to the target query are required. The negative training data is also sourced during the offline stage, and is fixed for all queries. Features are computed for $\sim 1,000$ images downloaded from Google Image search using the publicly available API and the search term 'things' and 'photos'.

The features for the positive training data are computed on the fly after the user has made a query, and again are sourced from Google Image search, which is used to translate the user's textual query into a set of images. We use the top-ranking ~ 200 images from a search for the query term entered by the user. Features are extracted from these images in the same way, and a linear SVM is trained against the pool of negative training features computed during the offline stage. The output of the classifier is a w vector of the same dimensions as the features, and the dot product between this and all features in the target dataset is then taken to provide an output score for each image. Finally, this score is used to rank the images in descending order of relevance to the entered query. As with last year, the system follows closely the details given in [7], with the difference this year being that we used VLAD [16] instead of BoVW encoding.

C. On the fly Face Retrieval

The aim of the face retrieval system is to retrieve key-frames based on the faces they contain. Given a query, a discriminative classifier is learnt using images containing faces downloaded from Google image search for that query.

To achieve real time performance, it is essential to perform as much of the processing in advance. In the offline processing faces are detected in every frame of every video and faces of same person are linked together within a shot to form face tracks. At the same time, nine facial features such as eyes, nose, mouth etc. are located within every face detection using pictorial structure based method [12], [11]. These features provide landmarks for computing facial descriptors (feature vectors). The whole process of representing faces in the videos by tracks results in substantial reduction in data to be processed. On the KIS dataset, tracking and filtering results in reduction in the granularity of the problem from 2.9 Million face detections to 17,390 face tracks.

Negative training images needed for training of the classifiers are taken from publicly available dataset [14]. These images are kept the same for all queries. The face detector, facial feature detector and appearance descriptor described above is applied to each of the negative images to produce feature vectors.

The online processing part consists of two steps collecting positive training images of faces from Google and training and ranking using a classifier. Once the features for positive training examples are computed, a linear SVM is trained, and used to assign scores to tracks in the corpus.

The resultant face search system can be used for searching both for specific people as well as those with specific (facial) attributes such as gender, facial hair, eyewear, etc. For details of the method refer to [25].

D. On the fly Logo / Place Retrieval

The aim of the logo / place recognition system is to quickly retrieve key-frames which contain queried specific logos or places based on their visual appearance. A method based on early fusion which is reranked based on geometric verification created the classifiers which are learnt using images sourced from Google.

The early fusion system architecture is identical to the one described in [4], which is based on the standard specific object retrieval approach by Philbin et al. [28] with some recent improvements which are discussed next. RootSIFT [5] descriptors are extracted from affine-Hessian interest points [26], [1] and quantized into 1M visual words using approximate k-means. Given a single query, the system ranks images based on the term frequency inverse document frequency (tf-idf) score [32]. The ranking is computed efficiently through the use of an inverted index. Spatial reranking is performed on the top 200 tf-idf results using an affine transformation [28].

In this on-the-fly system, given a text query of a logo or place, example images are retrieved by textual Google image search using the publicly available API. A visual query set is constructed from the top 8 retrieved Google images. To retrieve from the corpus, a visual query is issued for each image in the query set independently and retrieved ranked lists are combined by scoring each image by the maximum of the individual scores obtained from each query. This is the MQ-Max method from [4], where further details are given.

For the version with early fusion, we first build a query specific model of the object or place. To this end, we mine local-bag-of words around the keypoints detected in the query images, resulting in a more powerful mid-level representation tuned towards the object or place we want to retrieve. Using this query specific model we construct a new histogram representation on the fly for each database image and retrieve images using a tf-idf based retrieval approach, using an inverted file system. This is again followed by spatial verification.

E. Score Fusion

Unlike last years interface, our interface this year places more emphasis on unimodal search, and so feature fusion was less important. The reason for this shift was based on feedback from users in previous years – it was more difficult for them to understand and accept results from fused sources than it was for a unimodal search.

Where fusion was needed, we chose a relatively simple algorithm to fuse the scores from the retrieval services. We first

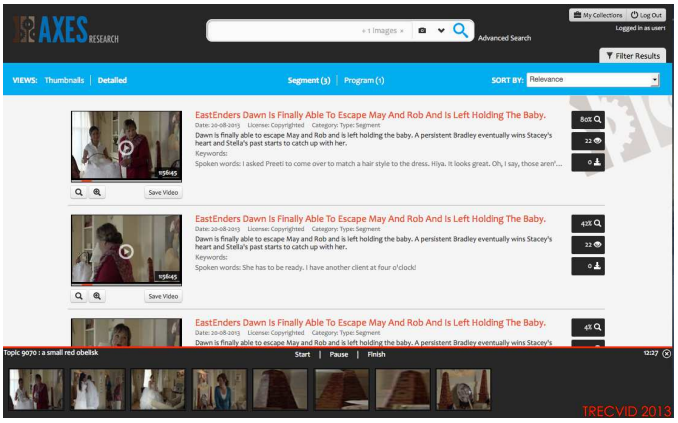


Fig. 1: The AXES research user interface showing a detailed view of the search results.

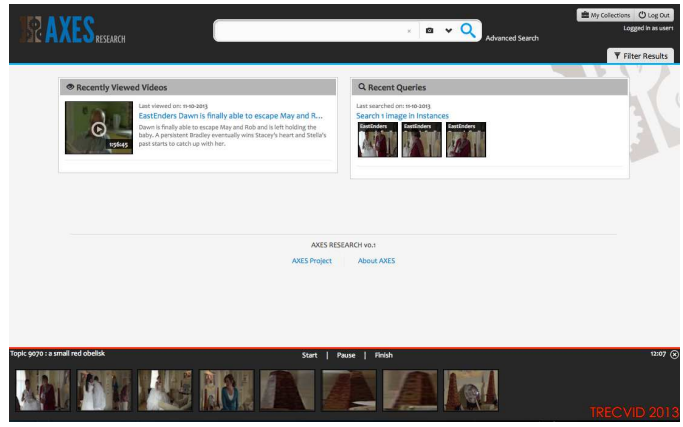


Fig. 2: The AXES user interface showing the home view.

normalized the scores of each component to the interval $[0, 1]$ by dividing them through the maximum score and then fused them using a linear combination as follows (see also [30]):

$$score = \sum_{i=1}^{n} score_i \tag{1}$$

where $score$ is the final score, and $score_i$ is the confidence score of the ith retrieval service.

F. User Interface

The user interface used for the INS task was based on a version of the AXES Research search system interface that was developed by the AXES consortium based on professional user requirements and feedback from TRECVid 2011 and 2012. Figure 1 shows a screenshot of the AXES research user interface. As with our 2012 TRECVid interface, the AXES research interface is a browser-based user interface targeted at traditional desktop-based interaction. The client-side interface uses HTML5, CSS3, and Javascript, and AJAX to communicate asynchronously with the server side.

The interface is composed of three panels: the *search* panel, the *retrieved results* panel and the *TRECVid* panel.

- The search panel allows user to formulate text-based, concept-based, or image-based queries. It supports the on the fly concept selection and visual similarity search.
- The retrieved results panel shows the results of a query in various ways. It also facilitates the home view, where users can look back at historical searches without the need to reformulate.
- The TRECVid panel features information on the current TRECVid topic along with features which facilitate the experiment.

1) Search Panel: Text-based queries can be entered via the search panel at the top of the interface. If the user selects the metadata or spoken words options, the relevance score will be calculated based on textual metadata (author, title, short description) or audio transcripts generated using the closed caption data. The visual search enables the on the fly visual concept classification, which uses images from an external source to build a visual model of the specified text.

Multiple search types can be combined using the advanced search option, in which case the results are compiled by fusing the output of the selected search components.

Similarity search is based upon retrieving a set of results, the user can add any thumbnails to the similarity search panel to use them as query images. Images from external websites and search engines, as well as images from the users local machine can also be added to the similarity search by clicking on the add external image button on the bottom right of the similarity search area. Clicking on this button displayed a selection overlay that allows the user to upload local images or specify external images by pasting in the URL for the image. Images can be removed from the similarity search panel simply by clicking the ‘x’ in the top right corner of the thumbnail.

2) Retrieved Result Panel: The retrieved results area of the the interface displays all videos retrieved that match the user’s query. The results area allows the user to view the result list using two different views: thumbnails and detailed. In the thumbnail view, each retrieved video is represented as a single thumbnail. User can click any thumbnail to quickly preview the entire content of corresponding video in a popup overlay. If the retrieved result is a segment from the video, then the preview overlay will automatically jump to the relevant location in the video. The advantage of this panel is that it provides a global overview of a large number of retrieved videos on a single screen; the disadvantage is the lack of detailed information on the videos.

In the detailed view (Figure 1), each row contains one retrieved video with more detailed information than what is presented in compact panel. Each video is displayed as a thumbnail with associated metadata, and the thumbnail may be clicked to start a preview playback. The metadata and matching information is located beside the thumbnails. A coloured segment location bar is also shown in this view. It describes the temporal location of the retrieved video segment with respect to the overall video. The length of the grey bar indicates video duration, while the length of orange bar describes the duration of video segment and the position where it is located. The duration of segment is displayed textually over the segment location bar. Below it, there are three buttons: query by keyframe, add to similarity and save video. The first

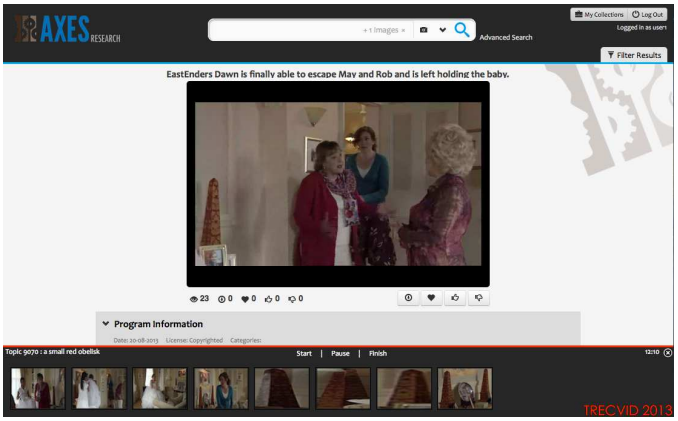


Fig. 3: The asset view showing detailed information about a single video.

button issues a similarity search based on the current video keyframe and only that frame. The second button adds the keyframe to the similarity search section for future searches. Finally videos are saved by clicking the 'Save Video' button. The saved videos are stored under the 'My Collection' link located on the search panel. Clicking on title shows a detailed *asset view* (Figure 3) for the selected video.

3) TRECVid Panel: The TRECVid panel as seen at the bottom of Figure 3 gives the user an overview of the current topic for the INS task. It features a topic number, description, exemplar images, timer and experiment control buttons. The example images feature four images which are provided as query images which when clicked perform a similarity search, they can also be added to a future search or visualized in larger forms based on mouse-over buttons. There are a further four images which feature the extracted masks of the object instance in question which can be used in a similar manner. There is a timer which starts upon the users first query which can be paused if the user needs a break for any reason. Finally, there is a finish button which the user can use to end the current topic if they do not wish to use the full fifteen minutes.

III. Multimedia Event Detection

In this section we describe the AXES submission to the multimedia event detection (MED) task. In Section III-A we describe the low-level features we used, and how we encode them to obtain a video-level signature. Section III-B describes the high-level OCR and ASR features that that capture written and spoken words. Details on classifier training and feature fusion are provided in III-C. Figure 4 gives a schematic illustration of the processing stages.

A. Low-level audio-visual features

In this section we describe the different low-level audio-visual features we used, and the common feature aggregation using Fisher vectors.

1) Audio: For the audio channel we down-sample the original audio track to 16 kHz with 16 bit resolution and then compute Mel-frequency cepstral coefficients (MFCC) with a window size of 25 ms and a step-size of 10 ms, keeping the

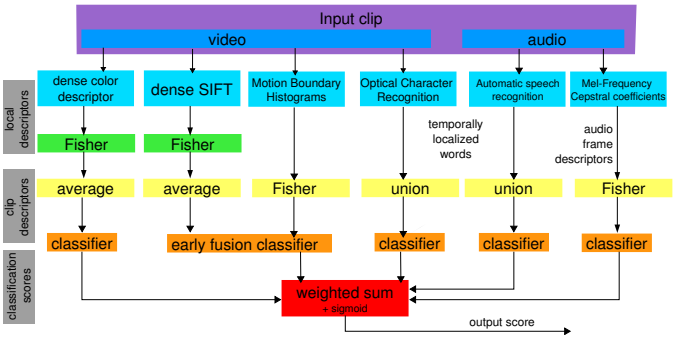


Fig. 4: Schematic overview of the processing of a video clip for MED.

first 12 coefficients of the final cosine transformation plus the energy of the signal. We enhance the MFCCs with their first and second order derivatives.

2) Image: The visual content is described by static appearance and motion features. For static visual appearances we use SIFT [19] and color features. We use the local color descriptor of [8], which consists in computing the mean and standard deviation on each of the three RGB channels (six values), for each of the 16 cells in a 4×4 grid over the local image patch, yielding a 96 dimensional descriptor. We compute both descriptors on one frame out of 60, using a regular dense multi-scale sampling grid.

We added spatial information to the descriptors using the spatial Fisher vector method of [17]. Compared to the traditional spatial pyramid method [18], the descriptor is much more compact while performing similarly for Fisher vector feature encoding.

The use of color descriptors and spatial Fisher vectors is new as compared to our submission to MED 2012.

3) Motion: Motion information is captured using recent camera-stabilized dense trajectory features [36], which obtain state-of-the-art results for human action recognition. This method tracks local image features, initialized on a dense multi-scale sampling grid, on a short time-scale of 15 frames. Stabilization w.r.t. camera motion is obtained by estimating the dominant motion between two successive frames, and correcting the optical flow calculations using the estimated motion. Besides obtaining more accurate optical flow estimates, the motion stabilization also allows us to identify background feature trajectories that are static once normalized for the dominant motion. Such background features are then removed. Figure 5 illustrates the effect of camera motion stabilization.

Several features are computed along the retained trajectories: HOG, HOF, and MBH. These features are similar to SIFT, but computed over a spatio-temporal volume along the feature tracks. While the HOG [9] descriptor encodes the spatial intensity gradient orientation distribution, the HOF and MBH descriptors are based on the optical flow field. The HOF descriptor encodes the orientation distribution of the flow; correction for camera motion is therefore important for this descriptor. The MBH descriptor, instead, is based on the horizontal and vertical gradients of the flow field, and encodes the orientation distribution of these gradients. Just like in SIFT,

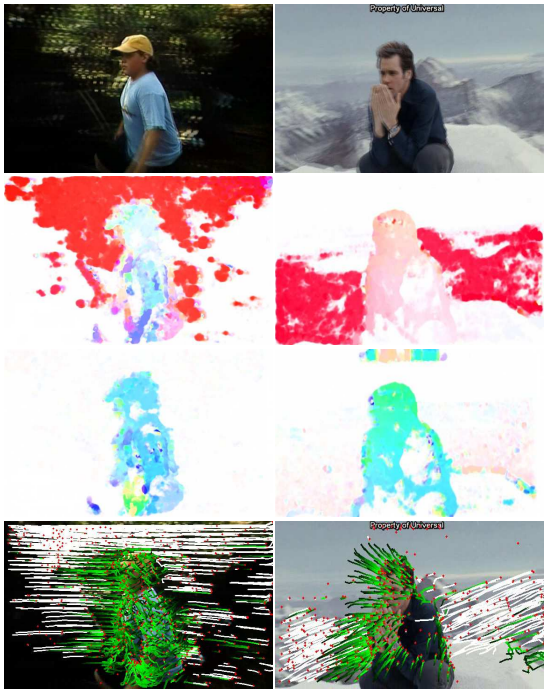


Fig. 5: First row: overlaid images of two consecutive frames; second row: optical flow between the two frames; third row: optical flow automatically enhanced after removing camera motion; last row: trajectories removed due to camera motion in white.

these descriptors are computed in several regular cells along each trajectory, and then concatenated.

The use of camera stabilization for the motion features is new as compared to our submission to MED 2012, where we used only MBH features computed over non-stabilized flow and did not remove static trajectories [35].

4) Frame-level and video-level representation: Once the local low-level audio and visual features are extracted, we use them to construct a signature to characterize the video. For this feature encoding step we proceed in the same manner for all three low-level features by using a Fisher Vector (FV) representation [29]. This is an extension of the bag-of-visual-words (BoV) representation. The Fisher vector (FV) records, for each quantization cell, not only the number of assigned descriptors as in BoV, but also the mean and variance of the assigned descriptors along each dimension. Therefore, a smaller number of quantization cells can be used than for BoV. This leads to a signature with a dimension of $K(2D+1)$ for K quantization cells and D dimensional descriptors. Since the assignment of local descriptors to quantization cells is the main computational cost, the FV signature is faster to compute. Instead of using a k-means clustering, a mixture of Gaussian clustering is used in the FV representation. Local descriptors are then assigned not only to a single quantization cell, but in a weighted manner to multiple clusters using the posterior component probability given the descriptor. We apply the power and ℓ_2 normalization to the FV, as introduced in [27] for image classification. In a recent evaluation study [23], we have shown that these normalizations also lead to significantly better performance for action and event recognition in videos.

For the image-based features, SIFT and color, we compute the normalized Fisher vectors per frame, and then average and re-normalize these to obtain the video-level representation. This approach performed somewhat better in initial experiments than directly aggregating all local descriptors for the video in the Fisher vector. For the MFCC and motion-based features this approach is not used, since these features are not localized in a single frame.

B. High-level OCR and ASR features

We use automatic speech recognition (ASR) and optical character recognition (OCR) to extract high-level features from the videos.

1) OCR: For OCR we used the same system as in our 2012 MED submission. In each video frame (sampling rate of 5Hz), MSER [20] regions are extracted from the luminance channel. Regions that do not have a suitable aspect ratio or weak gradients on their boundary are eliminated. Remaining ones are grouped into text lines, which are further segmented into words. Then, each region is expressed in term of a HOG-based descriptor [10], and a RBF kernel SVM classifier (trained on standard Windows fonts) predicts the probability of each character. Those probabilities are combined using an English language model based on 4-grams over letters to yield the final OCR results at the word level.

2) ASR: The ASR feature is the output of a large-vocabulary continuous speech recognition system. The underlying acoustic models are trained on approximately 100h of American English broadcast data which was manually transcribed. The language model include online news and newswire articles as well as patents. The vocabulary uses the most frequent 130k words and provides multiple pronunciations. Decoding is performed by the Julius recognition engine (`http://julius.sourceforge.jp/en_index.php`) with optimized parameters on automatically generated segments with model-based speech activity detection.

3) Video-level representation: For both the OCR and ASR output, a sparse bag-of-word descriptor is formed for each video. It is composed of the unique words that were detected either in the speech (for ASR) or in the text (for OCR). In this descriptor, the words are weighted by their tf-idf score, and then the vector is ℓ_2 normalized.

C. Classifier training and feature fusion strategies

We used linear support vector machine (SVM) classifiers, which permit efficient training and testing. Early and late fusion techniques are used to combine the different low-level and high-level features.

1) Early fusion: In order to combine the different trajectory-based features and the SIFT features we consider an early fusion strategy, which consists in concatenating the FVs extracted from the different features. A relative scaling of the features is determined using cross-validation in combination with a local search over a multi-dimensional grid of the feature weighting coefficients. The same cross-validation procedure

also optimizes the hyper-parameters of the classifiers for all features such as the regularization strength, and the weight to balance errors in positive and negative examples.

2) Late fusion: We include a late-fusion stage in which we combine the early-fusion system based on trajectory and SIFT features with the color, audio, ASR and OCR features. The late fusion consists in finding a linear combination of the classifier scores computed from the various sources. The late fusion training is done using scores that obtained in a cross-validated manner, i.e. the individual features are used to learn classifiers on 75% of the data and evaluated on the remaining 25%, which is repeated 30 times with random train and test sets. In this manner we obtain a number of test scores for each video and for each channel, and we use all of those to learn the late-fusion weights with a logistic discriminant classifier.

IV. Semantic Indexing

For the SIN task we used a setup that is similar to the one used for MED. The main difference is that we use a smaller set of features, namely only the non-stabilized MBH features of [35], SIFT features [19], and color features [8].

a) Features: For MBH, we used the non-stabilized version of [35], with standard parameters: sampling stride of 5 pixel, track length of 15 frames. We re-scale the videos to be between 100 and 200 pixels wide, and sub-sampled them temporally by skipping one frame out of two. We compress the MBH feature to $D = 64$ dimensions by PCA, and then use a GMM with $K = 256$ components to aggregate them into one Fisher vector per shot.

We extracted color and SIFT features densely across five scales, on one frame in sixty in every shot. The local features are then aggregated into a Fisher vector, using PCA and GMM parameters $D = 32$ and $K = 256$.

b) Training: We train a linear binary SVM for each feature and concept. In order to deal with the strongly varying number of positives and negatives per concept, we have used all the positives and at most 100,000 negatives per concept. For the SVM, we cross-validated per concept both the regularization parameter and the weight of the positive *vs*. the negative samples using three validation folds, where the train/test size ratio was 75%–25% for each of the folds.

We combine the features with late fusion. First, we split the training data into 15 folds, with a train/test size ration of 80%–20% for each fold. For each training fold we train a binary linear SVM using the previously learned hyper-parameters (regularization and class weighting). We then score the fold test data and aggregate the test samples and their scores together. For each test sample we keep only the median score. Finally, we obtain the late fusion weights by training a logistic regression on the test scores.

V. Multimedia Event Recounting

The goal of the MER task is to provide the users with a better understanding of why certain videos were returned for a MED event query. To this end the system returns a short list of video fragments that are most relevant to the event of interest, and a description of why these fragments are relevant.

This was our first participation to the MER task, and we opted for a relatively simple approach which consists in selecting the highest scoring video fragments, and using template sentences to describe them. We did not use any external training data except the data provided for the MED and SIN tasks.

A. Descriptors and classifiers

We used the exact same descriptors and classifiers as for MED, see Section III. The only differences are:

- for frame-level descriptors (SIFT and color), we extracted descriptors with a higher temporal density (1 out of 30 frames).
- for the motion features, we had to re-compute the descriptors because they were not stored during the MED computation (only the per-video Fisher Vectors were)

B. Selection of video fragments

We applied the MED system over small video fragments of several lengths in a sliding-window fashion, and report the highest scoring fragments. The selection proceeds as follows:

1) select videos from the MED test set whose score exceed the classification threshold for a given MED category;
2) in each selected video clip, extract all snippets for 2, 4 and 8 seconds long with at a temporal spacing of 2 seconds. Each of these snippets is scored with the MED classifiers, and they are ranked accordingly;
3) select 5 snippets in a greedy way: take the highest-scoring snippet that
   - does not overlap with an already selected snippet (non-maximum suppression)
   - does not have the same dominant channel as already selected snippet. The dominant channel is the one that contributes most to the MED score.

   This rule enforces some variety in the returned snippets.

In this manner, we obtain at most five snippets corresponding to the SIFT+motion features, color, MFCC, OCR and ASR features respectively.

C. Textual description

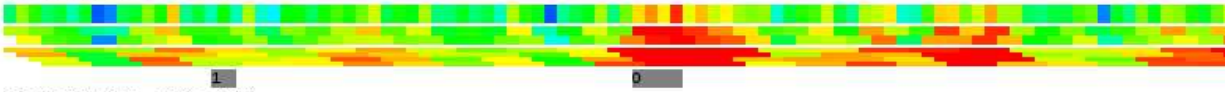
The description is generated using several template sentences. The sentence indicates which feature contributed most to its score, *e.g*. when MFCC or OCR features contribute most to the score we use sentences like: "The most relevant information for this snippet is audio.", or "The spotted words 'happy' and 'birthday' provide evidence for the event 'Birthday Party'." For each of the snippets we apply the SIN concept detectors to identify objects and other scene properties. We report the highest scoring SIN concepts in the description of the snippet, but exclude some concepts that were observed to perform poorly (*e.g*. the "Yasser Arafat" detector), or to be of little descriptive use (*e.g*. "Primate"). These are reported in a second sentence, such as "The snippet seems to contain: 'room', 'dining room', 'girl' and 'boy'.". In Figure 6 we show the interface we use to visualize the MER results, see the caption for a detailed description.

Video HVC410561

Class E013 Parkour

Combination of scores

combined score 15.105



channel 1: dan_audio_256
channel 2: hof_stab_h3,hog_stab_h3,mbhx_stab_h3,mbhy_stab_h3,sift_w_mu_sigma_sfv
channel 3: OCR (linear_SVM_sssl2_6AF9298FF1364ADE)
channel 4: ASR (linear_SVM_sssl2_4B49979D65C66AC5)
channel 5: color_1024_d32_sfv

clip no	t0 (s)	length (s)	scores					combination	attributes	description
			1	2	3	4	5			
0	100	8	2.345	4.377	0.362	0.608	3.450	6.295	Text (0.88) Primate (0.74) Building (0.70) Man_Made_Thing (0.67) Vegetation (0.65) Body_Parts (0.65) Yasser_Arafat (0.63)	The most relevant information for this snippet is visual. The snippet seems to contain: "text".
1	33	4	3.136	1.986	0.362	0.608	-0.637	3.095	Cats (0.99) Primate (0.96) Yasser_Arafat (0.92) Attached_Body_Parts (0.86) Snow (0.84) Surprise (0.81) Infants (0.78)	The most relevant information for this snippet is audio. The snippet seems to contain: "cats", "snow" and "surprise".

Fig. 6: The interface showing the MER results on an example video for the Parkour category. Below the video player is a visualization that displays the snippet scores in a color coding (blue = low, red = high). The table in the bottom of the interface gives information on the top-scoring snippets: the starting point, duration, scores for different feature channels, detected SIN classes, and a short description of why the snippet was highly ranked. The description indicates the feature that contributed most to the score, and spotted words if OCR was the strongest feature, and detected SIN classes that were not banned. The user can click on the snippet number to start a video playback of the selected snippet.

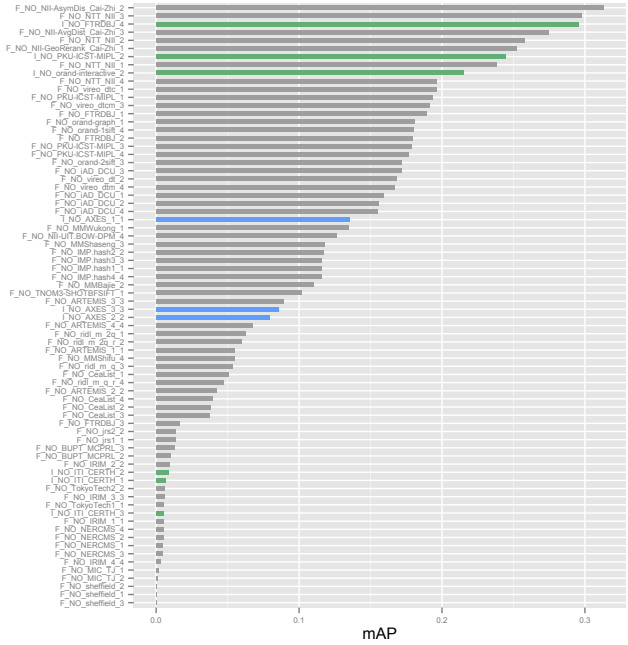


Fig. 7: Mean average precision over all submitted INS runs. AXES runs are shown in blue and other interactive runs in green.

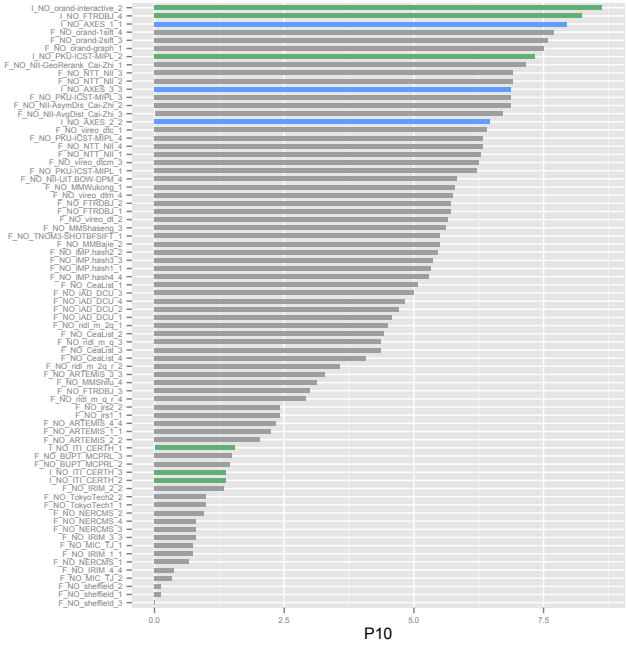


Fig. 8: Number of correct results out of the first 10 results returned for all submitted INS runs. AXES runs are shown in blue and other interactive runs in green.

VI. Experiments

A. Instance Search

The instance search experiments were carried out at Dublin City University in August 2013. A total of 12 people participated in the experiments. Participants were primarily research assistants, students, and post doctoral researchers. Each participant was assigned four or six topics and had 15 minutes to complete each topic. Participants were briefed on the purpose of the experiment the day it was run, and shown how to operate the user interface. They were also given two sample topics and some time to familiarize themselves with the interface before the experiment.

We submitted three runs of our system for evaluation.

1) AXES_1_1: Mainly post-doctoral researchers;
2) AXES_2_2: Mainly PhD students;
3) AXES_3_3: Mainly research assistants;

Figure 7 shows mean average precision for all submitted INS runs, with AXES runs shown in blue and other interactive runs shown in green. From the mAP score we see that our best system, AXES 1, only achieved median performance compared with other participants in the INS task.

Figure 8 shows the number of correct results out of the first 10 results returned for all submitted INS runs. From this we see that the system are comparable to other interactive systems within this criteria

Figure 9 shows the number of correct results out of the first 100 results returned for all submitted INS runs. From here we begin to see the decline in rank of all systems, we believe this is due to the maximum number of returned results for a topic being 296.

Figure 10 shows a more detailed plot of the proportion of relevant videos found by the experiment participants in each of four runs. Each bar in this plot represents the performance of a single user on a single topic. There was, in general, more relevant videos per topic than in TRECVid 2012. There was also clearly quite a few very difficult topics, particularly 9071, 9084, and 9087, in which our users were able to find very few relevant examples.

Figure 11 shows the relative proportions of relevant and non-relevant videos saved by each participant by topic. In comparison with TRECVid 2012, the proportion of non-relevant videos that users believed were relevant (saved) has significantly reduced, indicating that there was less ambiguity in the topics this year.

B. Multimedia Event Detection

The setup of the MED 2013 evaluation is the similar to that of 2012. The same 20 pre-specified event categories were used as in 2012, as well as the same test set which contains 98,000 videos for a total of 4,000 hours. A set of 10 novel ad-hoc categories were introduced this year.

1) Preliminary experiments: During the development of our system, we used a subset of the TRECVID MED 2011 (events E006-E015) dataset to quickly validate our methods on the individual channels. It also allowed us to tune the parameters of the low-level descriptors. From the scores in Table I we can draw a series of comparisons[1]:

- #1 *vs.* #2 and #3 *vs.* #16: the spatial Fisher improves

[1]The standard deviation associated each result is about 0.4 mAP points, because of the random cross-validation folds.

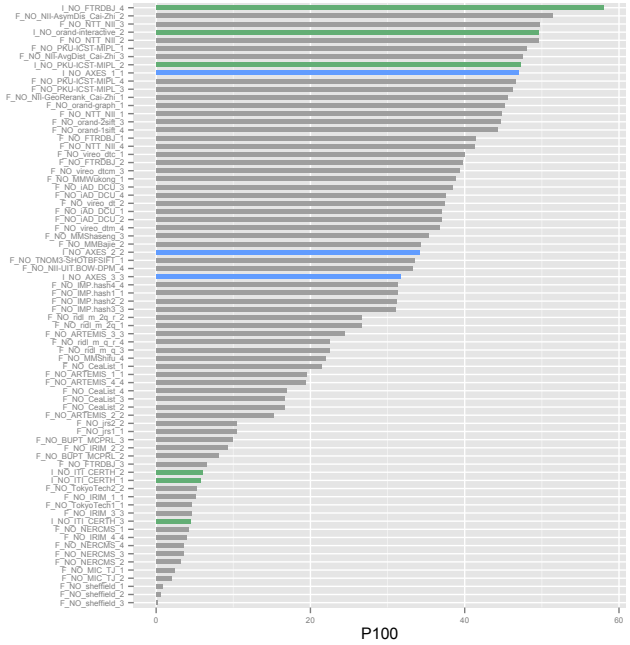


Fig. 9: Number of correct results out of the first 100 results returned for all submitted INS runs. AXES runs are shown in blue and other interactive runs in green.

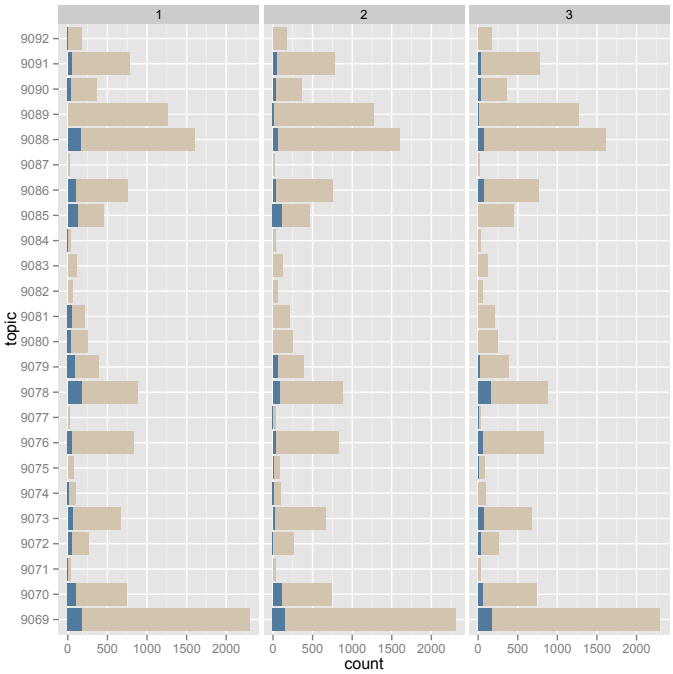


Fig. 10: Comparison of the number of relevant videos with the number of saved (returned) videos for each of the three AXES runs. The number of saved videos are shown as dark blue bars; the total number of relevant videos are shown as light brown bars.

results by 2 points (#1 is the SIFT descriptor we used in Trecvid 2012)

- #4*vs*. #5 and #12*vs*. #13: sometimes, increasing the size of the Fisher vectors (K set to 1024 instead of 256) *decreases* performance. This was a surprise.
- #3 *vs*. #15: for color, increasing the vocabulary size seems to help significantly, even when the PCA size on the local descriptors is decreased to 32,
- #6 *vs*. #7: we experimented with the self-similar descriptor, combined with a Fisher aggregation. Its individual performance is reasonable, but in combination with SIFT, it decreases. It is also expensive to compute, so we do not use it.
- #2 *vs*. #8: changing the power normalization from 0.5 to 0.2 increases the performace slightly (within standard deviation)
- #10 *vs*. #11: the HOG+HOF combination is more effective than MBH, combining all three further improves results in #12
- #12 *vs*. #19: the SIFT descriptors are complementary to the motion features, despite the fact that the HOG feature is similar to SIFT.
- #17*vs*. #18: optical character recognition is more reliable than speech recognition on this task.

2) Evaluation on MED 2011: To compare the state of our system in 2011, 2012 and 2013, we evaluated it using the TRECVID MED 2011 data set, and present results for the 10 event categories that were also used in MED 2012. For each category between 100 and 300 training videos are available, while the null class contains 9600 videos. The test set consists of 32,000 videos totaling 1,000 hours of video. We report both the NDC error measure for the optimal threshold (lower is better), as well as mAP (higher is better).

Besides our own results from 2011, 2012, and this year, we also included the results of the best system that entered in the 2011 edition [22] in Table II, as well as results of our current system using only the motion and SIFT features. The evaluation shows a clear progression of our results over the years. Our current system outperforms our earlier results, and those of [22]. It is interesting so see that using only motion and SIFT alone our current system performs only 2.1 mAP points worse than our full 2012 system. Among the other features, the MFCC audio features are the most complementary to the motion+SIFT features.

3) Description of the submission: Below, we summarize implementation details and parameter settings for each low-level feature used in our MED submission. Compared to the classification step, the feature extraction is by far the most expensive operation. For feature extraction, the computation of the various local descriptors the most expensive part. Feature encoding with FVs is less expensive, and is done in-memory, so that for each video we only store the FVs to disk. The dimension of each descriptor, and the computational costs are summarized in Table III.

- **Motion features:** For the dense trajectory features[2] we used the same settings for each of the MBH, HOG and HOF descriptors. The vocabulary size was 256, PCA was used to reduce the dimension of the local descriptors by a

[2]The implementation is available at: http://lear.inrialpes.fr/people/wang/dense_trajectories

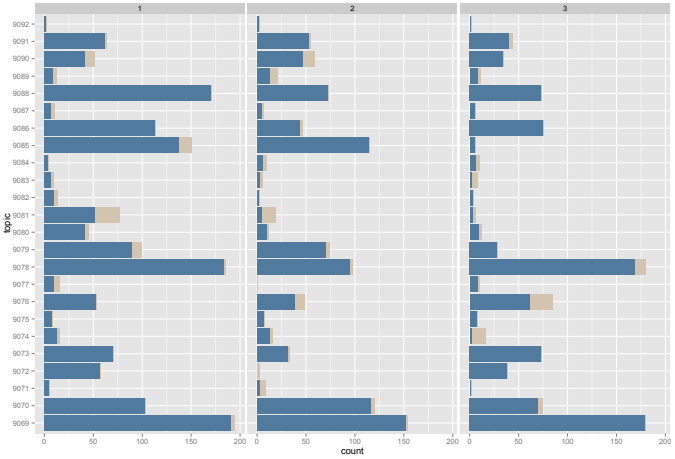


Fig. 11: Plot showing the relative proportions of relevant and non-relevant videos saved by each participant by topic. The dark blue bars represent the number of relevant videos; the light brown bars represent the number of non-relevant videos.

#	Channel & parameters	mAP
1	SIFT, K=256	54.2
2	#1 with spatial Fisher [17]	56.2
3	Color, K=1024, PCA 32	46.8
4	Audio, K=256	33.7
5	Audio, K=1024	32.8
6	Self-similar descriptor [31]	40.7
7	#1 + #6	53.7
8	#2, power 0.2	56.6
9	#2 + #3	56.7
10	MBHx + MBHy, stabilized, K=256	58.9
11	HOF + HOG, stabilized, K=256	64.3
12	#10 + #11	65.1
13	#12, K=1024	64.3
14	MBH, K=256, not stabilized (2012 version)	54.2
15	Color, K=256, PCA 64	42.8
16	#3, spatial Fisher	48.6
17	ASR	11.8
18	OCR	16.6
19	#12 + #1	67.6

TABLE I: Results on our internal validation set of 5908 train + 6338 test videos for several low-level descriptors. We measure the mAP over events E006 to E015. Several components with "+" are combined with early fusion. The K parameter is the number of components of Fisher vectors.

factor 2; the original feature dimensions are 108 for HOF, 96 for the others. Before feature extraction the videos were re-sized spatially to be at most 480 pixels wide, and sub-sampled temporally by a factor 2.

- **SIFT:** We computed SIFT descriptors on a 4x4 pixel dense grid and 5 scales. The 128 D SIFTs were reduced to 64 D by PCA and aggregated into a Fisher Vector with 256 components[3], and combined with spatial information [17].

[3]Code for SIFT and Fisher vector computation is available at: http://lear.inrialpes.fr/src/inria_fisher

	DCR	mAP
Best TV MED 2011 result	0.437	
LEAR 2011 submission	0.642	
AXES 2012	0.411	44.5
AXES 2013	0.379	52.6
Components		
Motion + SIFT (#19)		46.4
Color (#16)		27.7
Audio (#4)		18.2
ASR (#17)		8.2
OCR (#18)		10.8

TABLE II: Results on the MED2011 dataset, for which the ground-truth was shared by NIST. The numbers starting with "#" refer to Table I.

modality	feature	encoding	dim.	slowdown
Motion	MBH+HOG+HOF	FV	50688	10
Image	SIFT	FV	34559	2
Image	Color	FV	72703	10
Audio	MFCC	FV	20223	0.05
Image	OCR	BoW	110k (sparse)	1.5
Audio	ASR	BoW	110k (sparse)	3
Total	—	—	178173	27

TABLE III: Descriptor dimension and processing time as a slowdown factor with respect to real video time. The total dimension gives an order of magnitude of the storage needed to describe the video (excluding the variable-sized sparse BoW vectors from ASR and OCR).

- **Color:** Color descriptors were computed on the same grid as SIFT, they were reduced by PCA to 32 D and aggregated in a 1024 component Fisher vector, including spatial information.
- **MFCC:** The 13 dimensional MFCC is concatenated with its first and second derivatives, resulting in $D = 39$. The MFCCs are then aggregated in Fisher vectors of 256 components.

4) Results: The main results consist two mAP values, one across all pre-specified events, and one across all ad-hoc events. Table IV shows that both on the pre-specified and ad-hoc events we obtained the best results among all participants. We did not submit runs for the 10Ex and 0Ex versions of the challenge.

Interestingly, many participants have used similar descriptors: according to the text in the submissions, MBH + Fisher was used by at least CMU, Genie, MediaMill, NII and Sesame. However, there are probably large differences in implementations, since the results differ significantly. It is useful to invest a considerable amount of engineering in fine-tuning the stages of the method.

We submitted the per-channel runs too late for the pre-specified events, so we do not have evaluation results for them. However, Table V compares the per-channel results on the Ad-hoc submission. It shows that only our visual channel is state-of-the-art, but that our other features can be further improved. In particular our high-level ASR and OCR features

MED pre-specified		MED ad-hoc	
group	mAP	group	mAP
AXES (1/15)	34.6	AXES (1/14)	36.6
BBNVISER (2/15)	33.0	CMU (2/14)	36.3
median	24.7	median	23.3

TABLE IV: MED results for PROGAll with 100 training examples (100Ex).

Group	Full system	ASR	Audio	OCR	Visual
AXES	**36.6**	1.0	12.4	1.1	**29.4**
BBNVISER	32.2	**8.0**	15.1	**5.3**	23.4
CMU	36.3	5.7	**16.1**	3.7	28.4
Genie	20.2	4.3	10.1	—	16.9
IBM-Columbia	2.8	—	0.2	—	2.8
MediaMill	25.3	—	5.6	—	23.8
NII	24.9	—	8.8	—	19.9
ORAND	3.8	—	—	—	3.8
PicSOM	0.6	—	0.1	—	0.6
SRIAURORA	24.2	3.9	9.6	4.3	20.4
Sesame	25.7	3.9	5.6	0.2	23.2
VisQMUL	0.2	—	0.2	—	0.2

TABLE V: Per-channel results on the MED ad-hoc categories for 100 training examples per class (100Ex), as well as the full system performances. Best result in each column indicated in bold.

are relatively far from the best entries.

C. MER Results

The Multimedia Event Recounting results were evaluated on three measures:

- Accuracy: the proportion of correctly labeled clips, as assessed by the judge (in percent, higher is better)
- PPRT: fraction of the video time the judges spent to evaluate the result (lower is better, > 100 % is useless, as the judge did not gain time using the recounting system).
- precision of the observation text (OTS): how well the text describes the video (subjective score, between 0 and 4, higher is better).

We present a selection of the results for the MER task in Table VI. In the left part of the table, we consider the accuracy and the PRRT. Our accuracy result of 54.2% are ranked in the 8-th position of the 10 participants, but only slightly below to the median result of 56.0%. The PRRT shows that our snippets (39.2%) tend to be somewhat shorter than the median length of 46.2%. Note that the best entry in terms of accuracy (73.3%) is obtained when judges spend a time that is 149% of the video length.

In the right-hand part of Table VI we consider the OTS score. Our template sentence descriptions are ranked in the 8-th position with a score of 1.4, again this is not far from the median value of 1.7. One of the issues with our textual descriptions is that we simply report the best scoring SIN concepts, whereas it may be more useful to report concepts that are both high scoring and correlated with the category of interest. A second limiting factor might be our use of the SIN

group	Acc.	PRRT	group	OTS
SRIAURORA (1/10)	73.3	149.0	Sesame (1/10)	2.5
AXES (8/10)	54.2	39.2	AXES (8/10)	1.4
median	56.0	46.2	median	1.7

TABLE VI: MER results. Left: accuracy and PRRT, systems ranked for accuracy, the median is given for accuracy and PRRT separately. Right: OTS rating of description text. See text for details.

concept vocabulary rather than other larger and more diverse ones.

In our submission we did not tune the size of the shortlists to be scored to optimize the precision/accuracy of the results. It is unclear at this moment whether this might have impacted the MER performance of our system.

D. Semantic Indexing Results

SIN was evaluated in terms of mAP across the 346 concepts. We obtained an mAP value of 25.1%. This is the 7-th best result across the 26 participants, and well above the median result of 15.4% mAP, but also significantly below the best result of 32.1% mAP. This can be considered a relatively good result, given that we used only two channels, with few descriptors (MBH without stabilization, SIFT and color descriptors on sparsely sampled descriptors).

VII. Summary

This paper described the AXES participation in the interactive INS task, the MER task and the MED task for TRECVid 2013. Our interactive INS used an interface which focused on using classifiers trained at query time with positive examples collected from external search engines. We had twelve participants from our research carry out the interactive experiments. Our system performed similarly to the other best-performing interactive systems with respect to P@10 and P@30. The recall of our system after this point, however, decreased. We believe this was due to two reasons: first, our users did not use the full amount of time available for each task, and second, we had an error in our processing pipeline which meant that shots were duplicated, effectively reducing the number of saved shots per user by half.

Our MED, MER and SIN systems use state-of-the-art low-level features for describing audio, static and dynamic visual content, as well as high-level features capturing written and spoken words. The low-level features are aggregated into high-dimensional video-level signatures by means of Fisher vectors, the high-level features are aggregated into bag-of-word histograms. Our MED system achieved the best results among all submissions both in the main track, and in the ad-hoc track. Our MER and SIN results were less satisfying. For SIN our results ranked 7th among the 26 participants; our results can probably be improved by using more features than in our current system. For MER our results ranked 8th among 10 submissions; our textual descriptions based on template sentences and high-scoring SIN concepts leave room for improvement.

ACKNOWLEDGEMENTS

We would also like to acknowledge everyone that participated in the experiments, both the media professionals and the visiting students. This work was funded by the EU FP7 Project AXES ICT-269980 and the QUAERO project supported by OSEO. Furthermore, we are grateful to the UK EPSRC and ERC grant VisRec no. 228180 for financial support.

REFERENCES

[1] http://cmp.felk.cvut.cz/∼perdom1/code/index.html.

[2] R. Aly, C. Hauff, W. Heeren, D. Hiemstra, F. de Jong, R. Ordelman, T. Verschoor, and A. de Vries. The lowlands team at TRECVid 2007. In *Proceedings of the 7th TRECVid Workshop*, Geithesburg, U.S., February 2007. NIST.

[3] R. Aly, D. Hiemstra, A. P. de Vries, and H. Rode. The lowlands team at TRECVid 2008. In *Proceedings of the 8th TRECVid Workshop*, 2008.

[4] R. Arandjelović and A. Zisserman. Multiple queries for large scale specific object retrieval. In *Proceedings of the British Machine Vision Conference*, 2012.

[5] R. Arandjelović and A. Zisserman. Three things everyone should know to improve object retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[6] M. Ayari, J. Delhumeau, M. Douze, H. Jégou, D. Potapov, J. Revaud, C. Schmid, and J. Yuan. INRIA@TRECVID'2011: Copy Detection & Multimedia Event Detection. In *TRECVID*, Gaithersburg, United States, December 2011.

[7] K. Chatfield and A. Zisserman. Visor: Towards on-the-fly large-scale object category retrieval. 2012.

[8] S. Clinchant, G. Csurka, F. Perronnin, and J.-M. Renders. XRCE's participation to ImagEval. In *ImageEval workshop at CVIR*, 2007.

[9] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005.

[10] N. Dalal, B. Triggs, and C. Schmid. Human detection using oriented histograms of flow and appearance. *European Conference on Computer Vision*, 2006.

[11] M. Everingham, J. Sivic, and A. Zisserman. Taking the bite out of automatic naming of characters in TV video. 2009.

[12] P. Felzenszwalb and D. Huttenlocher. Pictorial structures for object recognition. *International Journal of Computer Vision*, 2005.

[13] C. Foley, J. Guo, D. Scott, P. Wilkins, C. Gurrin, A. F. Smeaton, P. Ferguson, K. McCusker, E. S. Diaz, X. Giro-i-Nieto, F. Marques, K. McGuinness, and N. E. O'Connor. TRECVID 2010 Experiments at Dublin City University. In *Proceedings of the 10th TRECVid Workshop*, Gaithersburg, USA, 2010.

[14] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, October 2007.

[15] H. Jégou, M. Douze, G. Gravier, C. Schmid, and P. Gros. INRIA LEAR-TEXMEX: Video copy detection task. In *Proc. of the TRECVid 2010 Workshop*, Gaithersburg, United States, 2010.

[16] H. Jégou, M. Douze, C. Schmid, and P. Pérez. Aggregating local descriptors into a compact image representation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3304–3311. IEEE, 2010.

[17] J. Krapac, J. Verbeek, and F. Jurie. Modeling Spatial Layout with Fisher Vectors for Image Categorization. In *International Conference on Computer Vision*, Barcelona, Spain, November 2011.

[18] S. Lazebnik, C. Schmid, and J. Ponce. Spatial pyramid matching. In A. Leonardis, B. Schiele, S. J. Dickinson, and M. J. Tarr, editors, *Object Categorization: Computer and Human Vision Perspectives*, pages 401–415. Cambridge University Press, November 2009.

[19] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.

[20] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In *Proceedings of the British Machine Vision Conference*, pages 384–393, 2002.

[21] K. McGuinness, R. Aly, S. Chen, M. Frappier, M. Kleppe, H. Lee, R. J. F. Ordelman, R. Arandjelovic, M. Juneja, C. V. Jawahar, A. Vedaldi, J. Schwenninger, S. Tschöpel, D. Schneider, N. E. O'Conner, A. Zisserman, A. Smeaton, and H. Beunders. AXES at TRECVid 2011. In *TREC 2011 Video Retrieval Evaluation Online Proceedings (TRECVid 2011)*,, Geithesburg, U.S., December 2011. NIST.

[22] P. Natarajan, S. Wu, S. Vitaladevuni, X. Zhuang, S. Tsakalidis, U. Park, R. Prasad, and P. Natarajan. Multimodal feature fusion for robust event detection in web videos. In *CVPR*, 2012.

[23] D. Oneata, J. Verbeek, and C. Schmid. Action and event recognition with Fisher vectors on a compact feature set. In *ICCV*, 2013.

[24] M. Osian and L. V. Gool. Video shot characterization. In *Proceedings of the 1th TRECVid Workshop*, Gaithersburg, USA, November 2003.

[25] O. M. Parkhi, A. Vedaldi, and A. Zisserman. On-the-fly specific person retrieval. In *International Workshop on Image Analysis for Multimedia Interactive Services*. IEEE, 2012.

[26] M. Perd'och, O. Chum, and J. Matas. Efficient representation of local geometry for large scale object retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[27] F. Perronnin, J.Sánchez, and T. Mensink. Improving the Fisher kernel for large-scale image classification. In *Proceedings of the European Conference on Computer Vision*, 2010.

[28] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2007.

[29] J. Sánchez, F. Perronnin, T. Mensink, and J. Verbeek. Image classification with the Fisher vector: Theory and practice. *IJCV*, 105(3):222–245, 2013.

[30] J. A. Shaw, E. A. Fox, J. A. Shaw, and E. A. Fox. Combination of multiple searches. In *The Third Text REtrieval Conference (TREC-3)*, pages 243–252, 1994.

[31] E. Shechtman and M. Irani. Matching local self-similarities across images and videos. In *IEEE Conference on Computer Vision and Pattern Recognition 2007 (CVPR'07)*, June 2007.

[32] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *Proceedings of the IEEE International Conf. on Computer Vision (ICCV)*, 2003.

[33] L. C. D. Team. *Lucene 3.2*.

[34] S. Vempati, M. Jain, O. M. Parkhi, C. V. Jawahar, M. Marszalek, A. Vedaldi, and A. Zisserman. Oxford-IIIT TRECVID 2009 - Notebook Paper. In *Proceedings of the 5th TRECVid Workshop*, 2009.

[35] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu. Dense trajectories and motion boundary descriptors for action recognition. *IJCV*, 2013. In press, available online.

[36] H. Wang and C. Schmid. Action recognition with improved trajectories. In *ICCV*, 2013.

[37] P. Wilkins, D. Byrne, G. J. F. Jones, H. Lee, G. Keenan, K. McGuinness, N. E. O'Connor, N. O'Hare, A. F. Smeaton, T. Adamek, R. Troncy, A. Amin, R. Benmokhtar, E. Dumont, B. Huet, B. Merialdo, G. Tolias, E. Spyrou, Y. Avrithis, G. T. Papadopoulous, V. Mezaris, I. Kompatsiaris, R. Mörzinger, P. Schallauer, W. Bailer, K. Chandramouli, E. Izquierdo, L. Goldmann, M. Haller, A. Samour, A. Corbet, T. Sikora, P. Praks, D. Hannah, M. Halvey, F. Hopfgartner, R. Villa, P. Punitha, A. Goyal, and J. M. Jose. K-Space at TRECVid 2008. In *Proceedings of the 8th TRECVid Workshop*, Gaithersburg, USA, 2008.

[38] R. Yan. *Probabilistic Models for Combining Diverse Knowledge Sources in Multimedia Retrieval*. PhD thesis, Carnegie Mellon University, 2006.