TRECVid 2011 Experiments at Dublin City University

David Scott, Jinlin Guo, Colum Foley, Frank Hopfgartner,
Cathal Gurrin and Alan F. Smeaton
School of Computing
Dublin City University, Glasnevin, Dublin 9, Ireland
{dscott,jguo,fhopfgartner,cgurrin, asmeaton}@computing.dcu.ie

Abstract

This year the iAd-DCU team participated in three of the assigned TRECVid 2011 tasks; Semantic Indexing (SIN), Interactive Known-Item Search (KIS) and Multimedia Event Detection (MED). For the SIN task we presented three full runs using global features, local features and fusion of global, local features and relationships between concepts respectively. The evaluation results show that local features achieve better performance, with marginal gains found when introducing global features and relationships between concepts. With regard to our KIS submission, similar to our 2010 KIS experiments, we have implemented an iPad interface to a KIS video search tool. The aim of this year's experimentation was to evaluate different display methodologies for KIS interaction. For this work, we integrate a clustering element for keyframes, which operates over MPEG-7 features using k-means clustering. In addition, we employ concept detection, not simply for search, but as a means of choosing most representative keyframes for ranked items. For our experiments we compare the baseline non-clustering system to a clustering system on a topic by topic basis. Finally, for the first time this year the iAd group at DCU has been involved in the MED Task. Two techniques are compared, employing low-level features directly and using concepts as intermediate representations. Evaluation results show promising initial results when performing event detection using concepts as intermediate representations.

1 Introduction

The CDVP has participated in TRECVid for almost every year since the first 'video track' in 2001 (most recently [2, 10, 6, 3]). This year the iAd team as part of the CDVP at Dublin City University participated in SIN, KIS and MED tasks. We are first-time participants in both the MED and SIN tasks. We have, in 2010, previously participated in the KIS experimentation.

With regard to SIN, our aim was to compare the SIN performance of global features and local features, testing if global features complementing the local features and verifying if the introduction of relationships between concepts actually boosted performance. To this end, we designed three runs:

- **Run 3**: Using global features, including three MPEG-7 features and grid-based SURF histogram.
- **Run 2**: Average fusion of 3 SVM classification results based on three different-size visual vocabularies in the Bag-of-Visual Word (BoVW) model and using Local feature OpponentSIFT.
- **Run 1**: Introduction of global features in Run 2, while considering simple relationships between concepts.

Our KIS system this year again comprises of an iPad interface communicating with a remote server where the search engine and usable content are hosted. In our experiments this year we have two distinct systems. The first system is our baseline system from last year, the interface displays a (scrollable) ranked list of results, each represented by a single keyframe and some metadata, similar to typical online video search systems. The keyframe is selected by use of concepts or query dependent text on the ASR data. Our second system is similar to the baseline in that it

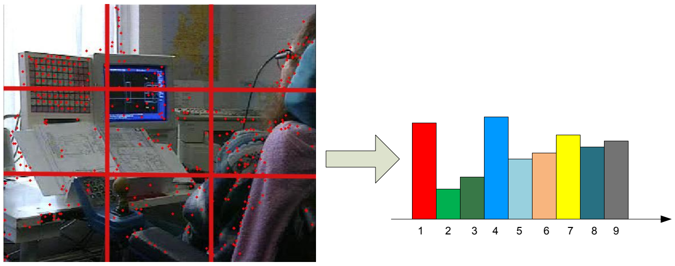


Figure 1: Global SURF extraction

uses the same video search technique, however, instead of a single keyframe per result, in this case, each result is actually a cluster of videos with similar visual features and a scrollable list of clusters is presented. Through our experiments in the KIS task in TRECVid 2010, we discovered very little difference between novice and expert users with respect to searching on our iPad interface. To this end we ran our baseline system with only novice users but ran our more complex modified display system with both experts and novices to see if we could identify any significant difference in performance.

Finally, for a third task, we took part in MED. The automatic detection of events in user-generated videos (UGVs) is a big challenge task for video analysis and processing. In comparison to edited and professionally produced content, such as broadcast sports video and news video, etc., UGVs are captured with various levels of production and in uncontrolled conditions. Therefore, it's more difficult to analyze and discover the spatial and temporal patterns related to events in the UGVs. In the MED task, we compare two methods for event detection in UGVs. One is performing event detection using low-level features directly and the other is using a series of middle-level semantic concepts to bridge the gap between low-level features and events;

- **Run P**: Two-stage SVM classification for event detection, using the low-level features and prediction probabilities for a series of pre-defined concepts as representations respectively. This is the primary run.
- **Run C**: SVM classification directly using BoVW feature representation from two low-level features: OpponentSIFT and Spatial-Temporal Interest Points (STIP). This is the contrastive run.

2 Semantic Indexing

Our SIN runs consist of three components as follows: 1) extraction of visual features, including global MPEG-7 descriptors and a grid-based SURF histogram; 2) generation of BoVW representation and 3) concept classification and result fusion.

2.1 Visual Feature Extraction

Only a single keyframe is selected to represent each shot. Next, global features and local features are extracted from this keyframe.

For global features, we extract three MPEG-7 color and texture descriptors and a grid-based SURF keypoint histogram. Three MPEG-7 descriptors are Color Layout, Edge Histogram and Scalable Color [8]. Since an object can appear in any part of a keyframe, we adopt a histogram by grouping the interest points into regions. Given a keyframe and a set of keypoints, a 3×3 grid is defined. A 9-bin histogram which is a count of the SURF keypoints that occur in each square is created (see Fig. 1).

The global feature representation for a keyframe is obtained by concatenating the four feature vectors mentioned above, in total 165 dimensions.

As shown in [9], the OpponentSIFT feature is a good performing single feature for concept detection and therefore has been chosen as the local feature for concept detection. The OpponentSIFT feature is an extension of the Scale-Invariant-Feature Transform (SIFT) feature to the opponent color space. More details can be found in [9].

2.2 Generation of BoVW Representation

After extracting the OpponentSIFT descriptors in the keyframes, images can be represented by sets of feature descriptors, but the sets vary in cardinality and lack meaningful ordering, which creates difficulties for learning methods (e.g. classifiers) that require feature vectors of fixed dimensions as input. To address this problem, the popular BoVW model is used to construct the representations for keyframes.

In the BoVW mode, three visual vocabularies (VVs) with sizes of 512, 1024 and 4096 are constructed by Hierarchal K-means, since in our internal experiments, visual vocabularies (VVs) with these three sizes achieve better performance with higher probability than other-size VVs. Furthermore, to reduce the computational cost, we sample the training set and cluster 2,000,000 OpponentSIFT features.

Then, a projection process follows, in which each image will be projected to the VV, that is, represented by the frequency distribution of the VWs contained in them. Here, we adopt the soft-weighting scheme proposed in [5].

2.3 Concept Classification and Result Fusion

Once images are represented by visual features, we can perform concept detection by using supervised classifiers trained from labeled images. In our experiments, we adopt a Support Vector Machine (SVM) for concept detection, since it has been proved to be a solid choice, and indeed, it has become the default choice in most concept detection schemes. The χ^2 RBF kernel, which has been shown to produce good performance [5], is used for SVM. The SVM classification is implemented using LIBSVM [1] with probability output. The parameters C and γ are optimized using grid search and five-fold cross validation. Moreover, all the features are normalized before concatenation.

With the representation with the global features, SVM classifiers are trained for each concept. Probability predictions from the SVM classifiers form Run 3. In Run 2, three BoVW representations using different-size VVs are used to train three SVM classifiers for each concept over the development set, and the average fusion of probability predictions from the three SVM classifiers forms our submission Run 2. In Run 3, we use the features by concatenating the 165-dimension global feature to the representation from BoVW model with each VV size, and common average is also used to fused the probability predictions from the three SVM classifiers.

However, in Run 3, we further consider the co-occurrence relationship between concepts after getting the confidences by average fusion. It's based on such an intuition that if a series of objects (or scenes, events) co-appear with high probability (high-related concepts), and one object appears, then the other object should also appear with high probability, and vice versa. And therefore, if one object is detected to appear in one image with high confidence, but its high-related concepts are considered to not appear (or appear with low probability), then we may think the concepts are wrongly detected.

Let P_{t,c_m} be the detection confidence that image t contains concept c_m. Let

$$I = \{c|P\left(c|c_m\right) \geq 0.7\} - \{c_m\}$$

$P\left(c|c_m\right)$ is calculated in the training set and N is the number of elements in I.

The rules for adjusting the detection performance are as follows:

- if $N > 0$, compute $R = \frac{1}{N}\sum_{c\subset I} P_{t,c}$, if $|R - P\left(c|c_m\right)| \leq 0.2$, then not adjust the detection confidence P_{t,c_m}. Else, adjust the detection confidence for image t containing concept c_m as $0.5 * P_{t,c_m}$
- else, not adjust the detection confidence P_{t,c_m}.

2.4 Results

In Table 1, we list the Mean infAPs of our runs and the best result reported by official TRECVid submissions. In the three runs we submitted, Run 2 achieves much better performance than Run 3, which is consistent with the finding that local features perform better for concept detection. After the introduction of global features and relationships between concepts, the performance improves by 6.5%. However, comparing with the best result released by TRECVid evaluation, there is a huge gap. We speculate that using single local feature without dense sampling is not enough to model the intra-concept-class variability and inter-concept-class separability.

Table 1: Mean infAPs of our runs and best results provided by official TRECVid submissions

Run 3	Run 2	Run 1	TRECVid Best
0.01	0.046	0.049	0.173

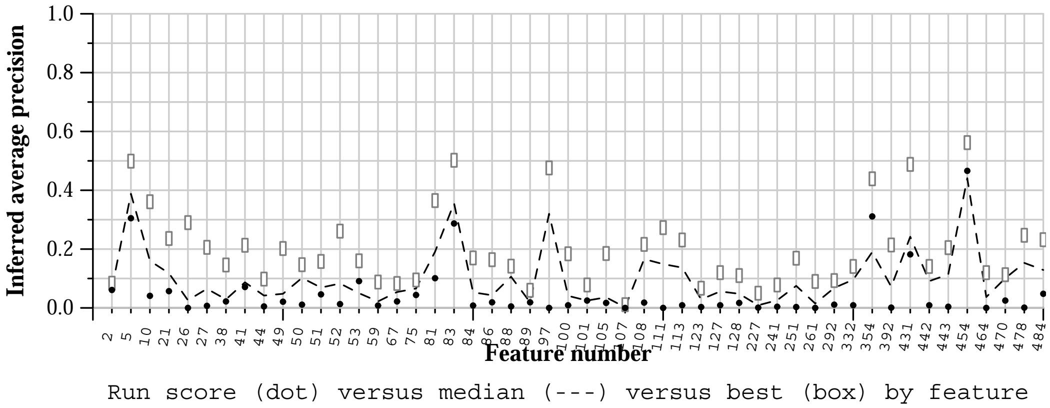


Figure 2: Comparison of our best run (Run 3) with median and best results

In Fig. 2, we show the comparison of our best run (Run 3) with median and best results for 50 concepts released by official TRECVid submissions. Overall, most of our concepts achieve worse performance than median. Interestingly, our infAPs fluctuate consistently as the median and best change. It confirms that the number of training sample affects our results for these concepts.

3 Interactive Known-Item Search

For our experiments in the KIS task of 2011, we devised two systems for comparative evaluation; one which modeled our previous year's TRECVid system, in that the unit of retrieval was the entire video, and a new system for 2011 which incorporated visual clustering techniques to present search results as a ranked list of video clusters. Both systems accepted text and the selection of visual concepts as the query mechanism. Both operated on a tablet PC (the iPad) with the aim of providing the user with a simple interface to a complex back-end video search tool. Most of the search functionality and video content processing for indexing and presentation (outlined below) take place on a server. In the following sections we will explain our methods for:

- Search Engine and Index
- Keyframe Selection
- Clustering Method

3.1 Search Engine and Index

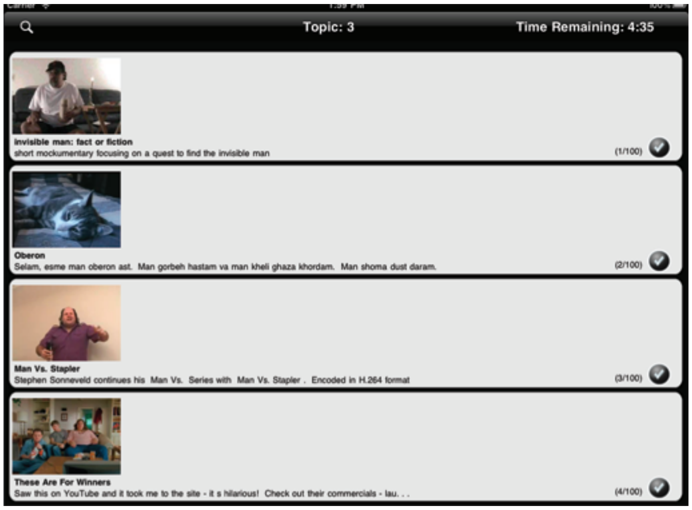
For our underlying search engine, we employed the Lucene build of Solr and created a search index over extracted meta-data provided within the dataset, indexing the title, description and keywords tags. We had considered employing the output of an Automatic Speech Recognition tool provided by the LIMSI and Vecsys Research [4], however initial experimentation on the training topics have shown that while the ASR did increase recall, it actually decreased the average rank of the known items, so it was not included in this year's experimental systems.

Our search engine index was hosted by our search middle-ware developed for TRECVid 2010, which used a modular .NET web service to communicate with the interface and the back-end data repository. Feedback from our 2010 experiments with novice users suggested to us that all query-time complexity should be removed from the system, hence we hide the similarity metrics by incorporating them with the search through clustering techniques which will be discussed later.

3.2 Keyframe selection

Accurate keyframe selection is especially influential given that our experiment is heavily focused on the video ranked result representation. We employ two types of keyframe selection criteria, firstly the 'most average keyframe' is chosen using the MPEG-7 [8] descriptors, Edge, Color Layout and Scalable Color, and secondly we employ a query-biased keyframe selection approach when the user has entered visual concepts to identify query-appropriate keyframes. For cases when a single visual concept is included, the top-ranking keyframe (for that concept) is chosen; in the case where more than one concept is selected evidence from all concepts are fused to identify the top-ranked frame.

3.3 Clustering Method



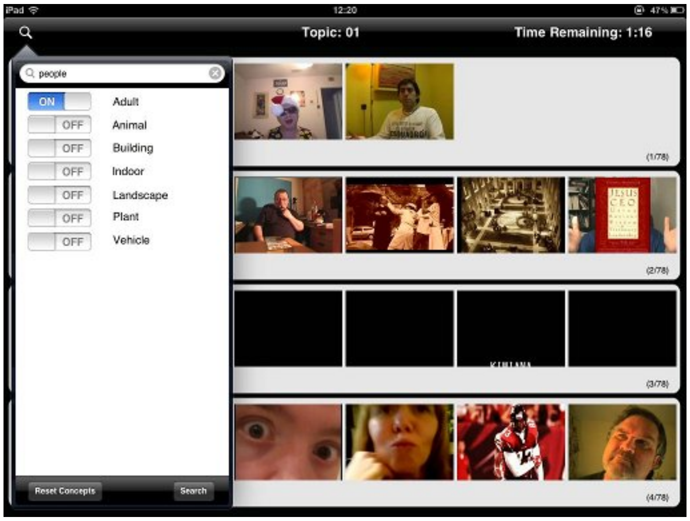
Invisible man: fact or fiction
short mockumentary focusing on a quest to find the invisible man (1/100)

Oberon
Selam, esme man oberon ast. Man gorbeh hastam va man kheli ghaza khordam. Man shoma dust daram. (2/100)

Man Vs. Stapler
Stephen Sonneveld continues his Man Vs. Series with Man Vs. Stapler . Encoded in H.264 format (3/100)

These Are For Winners
Saw this on YouTube and it took me to the site - it s hilarious! Check out their commercials - lau. . . (4/100)

Figure 3: A screenshot showing the baseline system where each line represents a video in the ranked list



people

ON Adult
OFF Animal
OFF Building
OFF Indoor
OFF Landscape
OFF Plant
OFF Vehicle

Figure 4: A screenshot showing the clustering system where each line represents a video in the ranked list

Our two systems for comparison in 2011 were a single keyframe per video (WWW style) baseline system, as shown in Fig. 3 and a result clustering system, as seen in Fig. 4. The result clustering system allows for users to view items which exhibit similar features, those of the MPEG-7 descriptors, and have them presented side-by-side. This allows the users to view visually similar content clustered together, and reduces the overhead of scrolling/browsing through the whole ranked list. We found that in most tasks, users found the known item faster on the clustering system, rather than the baseline approach.

3.4 Experiments

We had six users for this experiment, four novice users and two expert users. The four novice users were students from a business school with English was their second language. The novice users of both systems had never seen either system before and had limited experience of using a tablet PC device, one user had a galaxy tab but used it infrequently. Overall user satisfaction was positive based on a post-experiment user evaluation.

In DCU we also ran an experiment with our expert group, two volunteers not directly involved in the system but with experience in video search who each executed half of the topics. They ran their experiments on the clustering system, a baseline for the expert users was deemed unnecessary due to last years evidence that both novice users and experts performed similarly on the simple iPad system.

The clustering system outperformed the baseline with regard to Mean Elapsed Time with the novice users taking an average of 2.66 minutes per topic for the clustering system and experts taking 3.022 minutes for the experts clustering; the novices using the baseline system took an average of 3.324 minutes per topic. In our baseline experiments the novice users found a total of 12 topics out of the 25, our novices using clustering found one more topic at 13 and our expert users found one more again to give us a total of 14 out of 25 found.

4 Multimedia Event Detection

In the MED task, we submitted two runs, which aim to compare the effectiveness of two techniques, one is using low-level features directly, while the other considers the semantic concepts as intermediate representations. Further, we collect a new development set, consisting of the training samples and videos from YouTube.

4.1 Generation of Feature Descriptions

In our experiments, we only use visual features: OpponentSIFT and STIP feature. Extraction of OpponentSIFT descriptors has been mentioned in Section 2.1. Since processing all MED video frames will be computationally very expensive, we sample one frame every four seconds. Then we aggregate all the descriptors extracted from the same video clip and project them to pre-defined VVs with the size of 512, 1024 and 4094 respectively.

The STIP feature effectively captures space-time volumes where the image values have significant local variations in both space and time. We use Laptev's method [7] (Software available at [1]) to compute locations and descriptors for STIPs in video. The detector is based on an extension of Harris operator to space-time as described in [7]. Their code does not contain scale selection; instead interest points are detected at multiple spatial and temporal scales. HOG (Histograms of Oriented Gradients; 72 dimensions) and HOF (Histograms of Optical Flow; 90 dimensions) descriptors are computed for the 3D video patches in the neighborhood of the detected STIPs. We use concatenated HOGHOF feature (162 dimensions) as the final descriptor for each STIP. Similar to OpponentSIFT feature, we also generate three BoVW representations for each video clip using VV with the size of 512, 1024 and 4094 respectively.

In these steps, all these VVs generated by Hierarchal K-means using feature descriptors extracted only from the positive video samples in the training set. Moreover soft weight scheme proposed in [5] is used.

4.2 Semantic Concept Detection

In the UGVs, it's difficult to capture the spatial and temporal patterns related to events. However, events always co-occur with one or several (moving) objects in a certain scene. This event-scene-object-action dependency provides a feasible method for detecting event using event-related

[1] http://www.irisa.fr/vista/Equipe/People/Laptev/download.html

Table 2: Concepts selected for event detection

Object	streamers, road, vehicle, tire, lug wrench, animal, hand, snow, sink, bread, plates, jack, buildings, walls, stairs, sewing machine, fabric, balloons, birthday cake, screwdriver, table
Scene	indoor, outdoor with trees or grass visible, outdoor with cityscape, crowd, street, waterfront
Human Action	pushing, digging, slicing, spreading condiments on bread, jumping, clapping, rolling, unscrewing screws, bending over, dancing, eating, kneeling, laughing, playing games, running, turning lugwrench, unscrewing bolts, walking, sewing, singing, holding objects, cutting, pressing

concepts. Base on the observation of positive training samples, we define 50 concepts, including scene, object and human action.

Training a great number of concept detectors needs a lot of annotation work. Therefore, we use the detectors trained in Section 2 for these scene concepts and object concepts overlapped in the SIN task, but only the ones trained by using VV with the size of 1024. Each video is divided into multiple 6-sec clips, and we extract one keyframe every two seconds, and the annotation is performed at the frame-level. we then extract the OpponentSIFT features from these frame images and STIP features in these sub-clips. The two features are further processed to generate BoVW representations, which is the same to 4.1, but only use VV with 1024 visual words. With the annotated training data, we train SVM classifiers for detecting the concepts. For the rest of the scene concepts and object concepts, we use the BoVW representation of the OpponentSIFT feature, while for the human action concepts, we use the STIP features. We use the same methods as described in Section 4.1 to construct the BoVW representations, but only adopt the VV with the size of 1024. Then the χ^2 kernel is used to train the classifier for each concept.

4.3 Event Detection

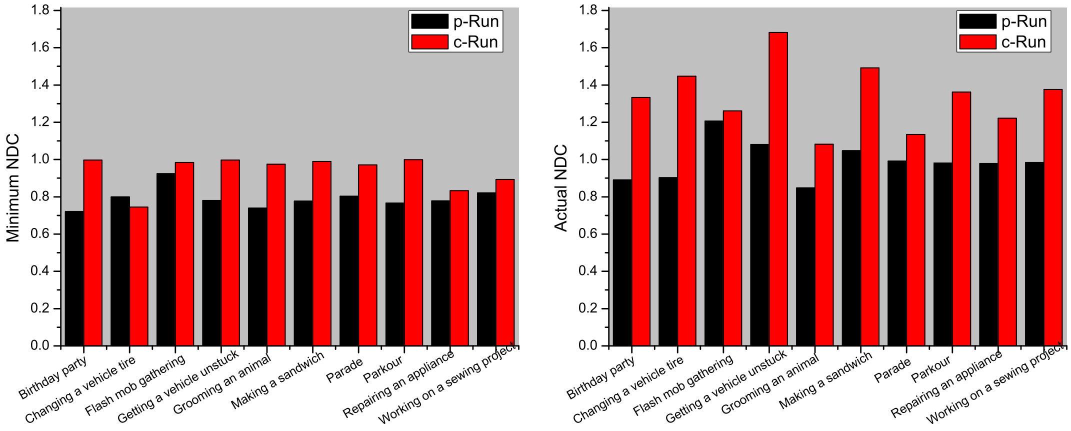
Contrastive Run. For each feature and each VV, we train SVM classifiers using χ^2 kernel for each of the events over new development sets. Due to the variations of videos in length, the clip-level features are normalized before classification by SVM. The average fusion of probability predictions from (in total) six SVM classifiers forms our results for the contrastive run.

Fusion of Concepts. After performing the concept detection using the trained SVM classifiers, we can get three 50-dimension vectors for each video clip by using maximum, average and minimum fusion of probability predictions from each concept classifier on the frames (or the sub-video clips with the length of six seconds) extracted in this video. Then, we further train three clip-level SVM classifiers using χ^2 kernel for each events. The average fusion of probability predictions from the three SVM classifiers forms the results for the primary run.

4.4 Results

Here, we mainly compare the results of the primary run and contrastive run. Fig. 5a shows the performance of our two submissions by the measurement of Minimum Normalized Detection Cost (MinimumNDC). The MinimumNDC is computed based on the best threshold of the detection scores, reflecting the best possible detection performance a system can reach[2]. The mean MinimumNDCs for two runs are listed in Table 3. From the MinimumNDCs and mean MinimumNDCs, we can see that using concept as intermediate representations outperforms the method using low-level feature directly for each event, even though the performance of the concept detection is far less than perfect. In addition to MinimumNDC, Actual NDC (ActualNDC) is also computed based on our provided detection threshold value as shown in Fig. 5b. The ActualNDCs also show that primary run achieves better performance. Therefore, we conclude that performing event detection, using concept as intermediate representations does appear to be effective.

[2] The detailed evaluation framework and description of the metrics can be found at http://www.nist.gov/itl/iad/mig/med11.cfm



(a) MinimumNDC

(b) ActualNDC

Figure 5: MinimumNDC and ActualNDC for each event. Note lower Actual NDC means better performance.

Table 3: Mean MinimumNDCs for two runs

p-Run	0.7909
c-Run	0.9384

5 Conclusions

This year, our team participated in three tasks: SIN, KIS and MED. For SIN task, we submitted three runs, respectively using global features, local features and fusion of global and local features with introduction of simple relationships between concepts. Evaluation results show that local features achieve better performance, and it gains marginally when introductions of global feature and relationships between concepts. In total, there are big gaps between our runs and the best submissions. In the KIS task, we developed a new system based on a visual clustering technique and compared it with a baseline system from last year. Our experiments suggest that the clustering system has potential to assist users in KIS. For the MED task, we try two different techniques, the contrastive run use the low-level features directly for event detection, while the primary run adopts a series of selected concepts as intermediate representations. Results show that it's more effective to perform event detection using concepts as intermediate representation.

Acknowledgements

We would like to thank Espen Andersen and his students in BI, Oslo for participating in our experiments. The research was funded by iAD - information Access Disruptions, a centre for research-based innovation with CRI number: 174867, funded in part by the Norwegian Research Council.

References

[1] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[2] C. Foley, J. Guo, D. Scott, P. Ferguson, P. Wilkins, K. McCusker, E. S. Diaz, C. Gurrin, A. F. Smeaton, X. G. i Niero, F. Marques, K. McGuinness, and N. E. O'Connor. Trecvid 2010 experiments at dublin city university. In *TRECVid 2010 - Text REtrieval Conference TRECVid Workshop*, 2010.

[3] C. Foley, C. Gurrin, G. Jones, H. Lee, S. M. Givney, N. O'Connor, S. Sav, A. F. Smeaton, and P. Wilkins. Trecvid 2005 experiments at dublin city university. In *TRECVid 2005 - Text REtrieval Conference TRECVID Workshop*, MD, USA, 2005. National Institute of Standards and Technology.

[4] J.-L. Gauvain, L. Lamel, and G. Adda. The limsi broadcast news transcription system. *Speech Commun.*, 37(1-2):89–108, 2002.

[5] Y.-G. Jiang, J. Y. 0003, C.-W. Ngo, and A. G. Hauptmann. Representations of keypoint-based semantic concept detection: A comprehensive study. *IEEE Transactions on Multimedia*, 12(1):42–53, 2010.

[6] M. Koskela, P. Wilkins, T. Adamek, A. F. Smeaton, and N. O'Connor. Trecvid 2006 experiments at dublin city university. In *TRECVid 2006 - Text REtrieval Conference TRECVid Workshop*, 2006.

[7] I. Laptev. On space-time interest points. *International Journal of Computer Vision*, 64(2-3):107–123, 2005.

[8] MPEG. MPEG-7 Overview(Version 10). *http://mpeg.chiariglione.org/standards/mpeg-7/mpeg-7.htm.*

[9] K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek. Evaluating color descriptors for object and scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1582–1596, 2010.

[10] P. Wilkins, T. Adamek, G. Jones, N. O'Connor, and A. F. Smeaton. Trecvid 2007 experiments at dublin city university. In *TRECVid 2007 - Text REtrieval Conference TRECVid Workshop*, 2007.