PicSOM Experiments in TRECVID 2008

Markus Koskela, Mats Sjöberg, Ville Viitaniemi, Jorma Laaksonen
Adaptive Informatics Research Centre, Department of Information and Computer Science
Helsinki University of Technology (TKK), Finland

Abstract

Our experiments in TRECVID 2008 include participation in the high-level feature extraction, automatic search, video summarization, and video copy detection tasks, using a common system framework.

In the high-level feature extraction task, we extended our last year's experiments, which were based on SOM-based semantic concept modeling followed by a post-processing stage utilizing the concepts' temporal and inter-concept co-occurrences. We also studied the effects of a more comprehensive feature selection and the inclusion of audio features and face detection. We submitted the following six runs:

- `A_PicSOM_1_6`: Visual features, baseline feature selection
- `A_PicSOM_2_2`: Visual features, baseline feature selection, temporal context
- `A_PicSOM_3_5`: Visual features, extended feature selection
- `A_PicSOM_4_4`: All features, extended feature selection
- `A_PicSOM_5_3`: All features, extended feature selection, PRF
- `A_PicSOM_6_1`: All features, extended feature selection, temporal context

The results show that a more comprehensive feature selection can be useful, and that the temporal and inter-concept co-occurrence analysis has the potential to improve the performance if good concept-wise post-processors can be chosen. The use of audio features and face detection resulted in minor improvements.

In the search task, we again concentrated on the fully-automatic runs. We combined ASR/MT text search and concept-based retrieval. If none of the concept models could be matched with the query, we used content-based retrieval based on the video and image examples instead. We also experimented with topic-wise feature selection and the addition of face detection and motion-based features. We submitted the following six fully-automatic runs:

- `F_A_1_PicSOM_1_6`: Required text search baseline
- `F_A_1_PicSOM_2_5`: Alternative visual baseline, only examples
- `F_A_1_PicSOM_3_4`: Alternative visual baseline, examples or concepts
- `F_A_2_PicSOM_4_3`: Text search + visual examples or concepts
- `F_A_2_PicSOM_5_2`: Text search + visual examples or concepts with feature selection
- `F_A_2_PicSOM_6_1`: Text search + visual examples or concepts with feature selection + additional features

The results show that the combination of concept-based retrieval and text search performed better than any of the single modalities in the baseline runs. Concept-based feature selection and additional features, however, degraded the average results.

In BBC rushes summarization, we submitted one run which extended our last year's approach consisting of initial shot boundary detection followed by shot content analysis and similarity assessment and pruning. We included new detectors for frames containing clapper boards, three different motion detectors, and a speech detector. The results of our summarization run are quite close to the median in the fraction of ground-truth inclusions found and in redundancy, with somewhat shorter average duration than the median. Our run's performance was above the median on the amount of junk present and on tempo/rhythm.

For video copy detection, we submitted some preliminary experiments based on our algorithm for shot similarity determination in video summarization. We used only the video modality.

I. Introduction

In this paper, we describe our experiments for the TRECVID 2008 [1] evaluations. This year we participated in the high-level feature extraction, automatic search, video summarization, and video copy detection pilot tasks. The basic system and methodology used in these experiments remains similar to our previous participations since 2005.

In the high-level feature extraction task, we studied the effects of a more comprehensive feature selection and the inclusion of audio features and face detection. For our automatic search runs, we combined concept-based retrieval with text search and experimented with topic-wise feature selection and the addition of face detection and motion-based features. In video summarization, we included new detectors for clapper boards, motion, and speech.

The rest of this notebook paper is organized as follows. The PicSOM system framework and the used visual and textual content descriptors are briefly described in Section II. Our experiments for the high-level feature extraction and fully automatic search tasks are described in Sections III and IV, respectively. Our approach and results in the video summarization task are described in Section V. The video copy detection experiments are discussed in Section VI and conclusions are presented in Section VII.

II. Indexing Video with PicSOM

The PicSOM system [2] is a general framework for research on content-based indexing and retrieval of visual objects. For video material, the indexing is based on a multimodal hierarchy for each video shot, which is in the standard setup

considered as the main or parent object. The associated keyframes, the audio track, and ASR/MT text are linked as children of the parent object. For a more detailed description of the basic setup, see [3], [4].

We extracted in total 15 video and 17 still image (keyframe) features from the Sound and Vision (S&V) material. From the BBC rushes, we extracted three image features. For the high-level feature extraction and search tasks, the keyframes were extracted from the video shots in the master shot reference [5] using a heuristic algorithm (see [6]). For video summarization and video copy detection, we sampled the BBC rushes and the S&V videos, respectively, at one keyframe per second.

Separate Self-Organizing Maps (SOMs) were trained for each of the video and image features. The size of the used SOMs was 64×64 map units in the video copy detection task and 256×256 map units in all other tasks.

All used features are briefly described in Sections II-A to II-D.

A. Image features

For the keyframes, we extract a large set of different features. The extracted features include five MPEG-7 descriptors implemented in the XM[1] reference software, our own implementations of four of the MPEG-7 descriptors, and seven other image features (*Average Color*, *Color Moments*, *Texture Neighborhood*, *Edge Histogram*, *Edge Co-occurrence*, *Edge Fourier*, and *Interest Points*). See [6] for details of these features.

B. Video features

On the video shot level, we used temporal versions of both our own implementations of the four MPEG-7 descriptors and the six non-standard still image features described above. See [6] for details.

In addition, we extracted a motion feature *KLT Histogram* based on tracked feature points using a public domain implementation[2] of the Kanade-Lucas-Tomasi feature tracker [7]. For each frame, each feature point is classified either as missed, static, or moving. The moving feature points are then mapped into eight principal directions similarly as in [8]. In addition, we use two relative directions, one toward the center of the frame and one away from it, for detection of zoom-in and zoom-out. This results in a twelve bin motion histogram, which is used both as a statistical motion feature and a basis for detectors of different types of motion (camera motion, camera zoom, object motion) present.

C. Audio features

As audio features we used two different implementations of the popular mel-scaled cepstral coefficients feature (MFCC), which is the discrete cosine transform (DCT) applied to the logarithm of the mel-scaled filter bank energies. The first implementation *MFCC1* takes 12 coefficients and these are organized as a vector. Finally the total power of the signal is appended to the vector giving a feature vector of length 13. The second implementation *MFCC2* is by the Muvis group at Tampere University of Technology [9] and produces a 24 dimensional vector.

D. Text features

The Dutch automatic speech recognition (ASR) output [10] was machine-translated (MT) to English. We used only the English documents on the shot level. The ASR/MT output was indexed using the Apache Lucene[3] text search engine. The Snowball stemmer included in Lucene was used with its included default stop word list.

E. Additional detectors

For face detection, we used the Viola-Jones [11] face detector implementation in OpenCV[4] combined with a simple skin color detector [12] based on the keyframes. We detect shots with either one face, two faces, and three or more faces in them. The face detector was utilized in high-level feature extraction, automatic search, and video summarization.

In addition, we utilized a set of specific frame and shot level content detectors for various purposes in the video summarization task. These are described in Section V-A.

III. High-level Feature Extraction

In our experiments this year in the high-level feature extraction task, we studied the effect of more exhaustive feature selection for each of the modeled concepts. In addition, we incorporated a similar temporal and inter-concept co-occurrence analysis step as in last year's experiments. Finally, we included optionally two audio features, face detection, and pseudo relevance feedback in the experiments.

Based on the last year's results, we did not include any text search results in the experiments as the visual-only baseline performed considerably better. Also, we always used a separate validation set for the selection of the algorithm used in post-processing based on the temporal and inter-concept co-occurrences, since including a validation set resulted in slightly better performance in last year's experiments. This post-processing analysis is described in detail in [13].

The basic method for detecting semantic concepts is based on modeling probability densities of the concepts using kernel-based estimation of discrete class densities over the (256×256 map unit) SOM grids. See [3], [14] for more details. In previous experiments, it has been observed that using the SOM-based approach is efficient and highly scalable to large ontologies, but does not quite reach the level of computationally more complex discriminative methods such as SVMs.

In all runs, we used a triangle-shaped kernel whose size was fixed to 8 units. Last year we observed that the modeling performance is not particularly sensitive to the size of the kernel. While the results can be slightly improved by using the optimal size parameter for each concept separately, it is

[1] *http://www.lis.ei.tum.de/research/bv/topics/mmdb/e_mpeg7.html*
[2] *http://www.ces.clemson.edu/~stb/klt/*
[3] *http://lucene.apache.org/*
[4] *http://opencv.willowgarage.com/*

TABLE I
AN OVERVIEW OF THE RUNS IN THE HIGH-LEVEL FEATURE EXTRACTION TASK. SEE TEXT FOR DETAILS.

#	run id	feat. sel. sfs	feat. sel. all	tem-poral	addit. feat	PRF	MIAP
1	A_PicSOM_1_6	•					0.0499
2	A_PicSOM_2_2	•		•			0.0567
3	A_PicSOM_3_5		•				0.0561
4	A_PicSOM_4_4		•		•		0.0566
5	A_PicSOM_5_3		•		•	•	0.0567
6	A_PicSOM_6_1		•	•	•		0.0568
2a	additional run	•		oracle			0.0627
6a	additional run		•	oracle	•		0.0730

difficult to obtain these optimal values for the test data using cross-validation in the development set.

The concepts were detected using the same procedure based on the concept-wise ground-truth annotations gathered by the organized collaborative annotation effort [15]. All our submitted runs were of type A.

A. Overview of the submitted and additional runs

Table I gives an overview of the high-level feature extraction runs. The columns refer to the type of feature selection used, whether the temporal and inter-concept co-occurrence analysis was applied, the presence or absence of two additional features (two audio features and face detection output), and whether pseudo relevance feedback was used. The rightmost column lists the corresponding mean inferred average precision (MIAP) [16] values.

The first run can be considered as a baseline in which 26 visual features (all image and video features except the motion feature) are used and selected with standard greedy feature selection for each concept separately. Run 3 is also based on the visual features but contains an extended feature selection scheme described in Section III-B below.

In runs 4 and 5, the pool of available features is expanded with two audio features and the face detectors. The extended feature selection scheme was used in both runs. Run 5 also contains a pseudo relevance feedback step in which the 20 initially retrieved best-scoring shots are marked as additional relevant objects.

Runs 2 and 6 include the post-processing step based on the temporal and inter-concept co-occurrences of the concepts, with runs 1 and 4, respectively, as the starting points for the analysis. For each concept, the used post-processing algorithm was selected from a set of 18 possible algorithms using a separate validation set [13]. Afterwards, we examined the algorithm selection and the performance of the different post-processors. In runs 2a and 6a, we repeat the submitted runs 2 and 6 but select the optimal (oracle) post-processing algorithm.

B. Feature selection and weighting

In our previous TRECVID experiments, the set of features (and associated SOM indices) has been selected for each concept separately using greedy sequential forward selection as the feature selection scheme (see e.g. [3]). This year, we studied the effects of using more exhaustive feature selection algorithms. As the total number of potential features, i.e. 26 or 28, is rather high, a full exhaustive search of all combinations becomes computationally too costly. Therefore, we apply three alternative feature selection algorithms: sequential forward search (sfs), sequential backward search (sbs), and exhaustive search on a feature subset.

The sfs and sbs search types use the complete set of 26 visual features. Sfs starts with an empty set and in each step adds the feature that increases the average precision the most in two-fold cross validation in the development set. The process stops when adding a new feature would decrease the result. As a result on average 4.9 features were selected per concept. Sbs conversely starts with the full set and proceeds to remove features one by one according to the performance. Naturally, this resulted in more features selected, on average 12 features per concept. Furthermore, we also performed an exhaustive search, which checks all possible combinations of a smaller subset of 12 features. These features were selected as those that were most often selected by the sfs algorithm. The exhaustive search resulted in 4.4 features per concept being selected.

Finally we picked the feature set from the three algorithms' results that worked best for each concept. In this way sfs contributed 3 feature sets, sbs 9 sets and exhaustive search 8 sets, resulting in a total of 7.9 or 8.3 features per concept without and with the additional features, respectively. The most frequently selected features were *Interest Points*, the temporal versions of *Color Layout* and *Color Moments*, and both the still image and temporal versions of the *Edge Histogram* feature.

The face detection outputs were included as external features into the feature fusion stage. However, as the inherent feature weighting mechanism of the system is not able to automatically weight such externally provided features, the corresponding concept-wise weights for the face detection features were optimized separately using the development set. In the end, this resulted in a non-zero weight only for the concept 007: *Two people*.

C. Results

Figure 1 illustrates the mean inferred average precision (MIAP) [16] values of our runs in the high-level feature extraction task. The highest MIAP score of our submitted runs was 0.0568 obtained with run 6, with runs 2, 4, and 5 having almost equal performance. The median and maximum over all 161 type A submissions were 0.0477 and 0.167, respectively. In addition, the concept-wise IAP values for all our submitted runs along with the median and maximum values over all submissions are illustrated in Figure 2.

First of all, an improvement of 12% resulting from the more exhaustive feature selection scheme can be observed by comparing runs 1 and 3.

Next, the temporal and inter-concept co-occurrences analysis notably improved the results when using only the sfs feature selection (run 2 compared to run 1), but had little

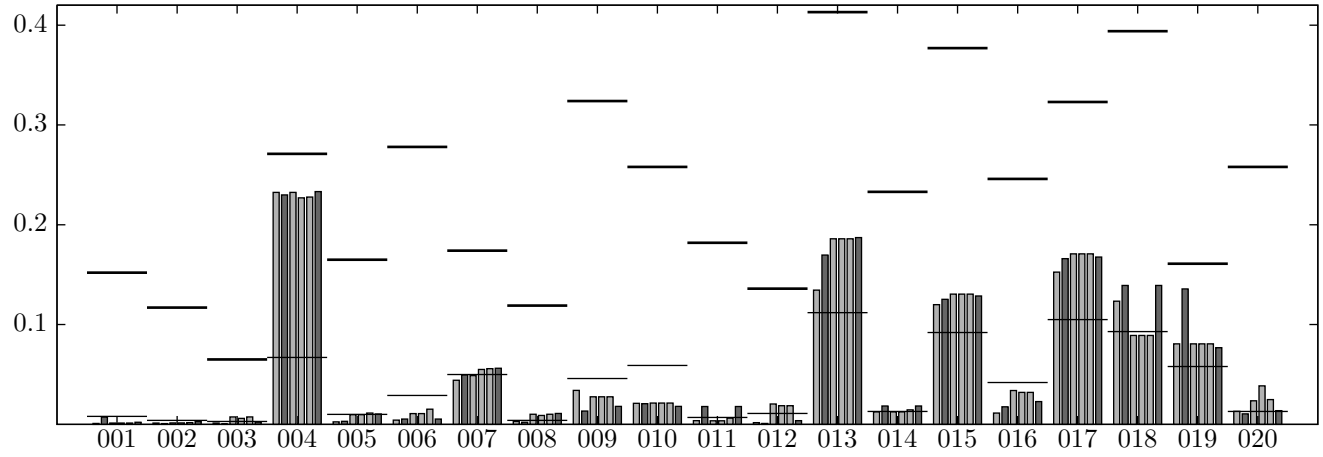

Fig. 2. The concept-wise IAP results of our submitted runs for each evaluated concept. The runs including the temporal co-occurrence analysis are drawn as darker bars. The median and maximum values over all submissions are illustrated as horizontal lines.

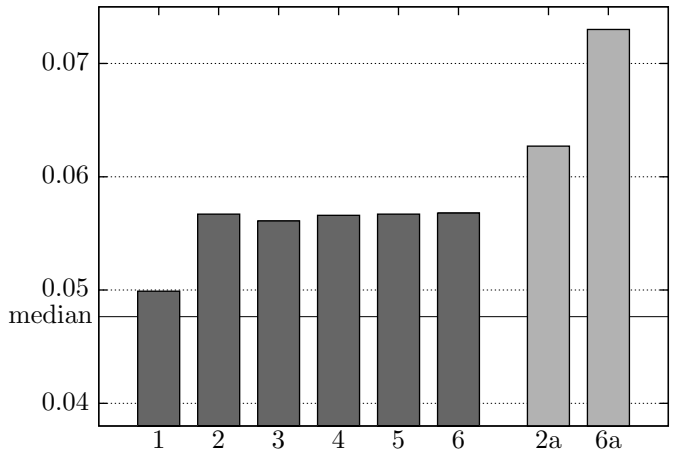

Fig. 1. MIAP values for our runs in the high-level feature extraction task; the submitted runs shown as darker bars. The median of all submitted type A runs in also shown for comparison.

effect when using the more exhaustive feature selection (run 6 compared to run 4). The latter result can be attributed to the known problem of selecting a good post-processor for each concept. This is illustrated by the relatively large differences in MIAP values between the submitted runs 2 and 6 and the additional runs 2a and 6a where the optimal post-processors are used.

Of the additional features included in run 4, the *MFCC1* audio feature was selected for six concepts, the *MFCC2* feature once, and the face detection feature once. However, the additional features noticeably improved the retrieval results only with the concept 020: *Singing* where the two audio features were employed and 007: *Two people* where face detection was used.

In run 5, we also experimented with the inclusion of pseudo relevance feedback, but its effect was rather negligible with all concepts.

IV. Automatic Search

For the search task, we submitted six automatic runs summarized in Table II. All runs were trained only on common TRECVID development data, thus qualifying them as type A runs. The retrieval technique combines content-based retrieval based on the image and video SOM indices with external text-based search and semantic concept models [3], [4].

Runs 1–3 constitute the baseline runs using only either text-based search (run 1) or the visual features (runs 2 and 3). Run 2 is based solely on content-based search with the provided image and video examples whereas run 3 replaces the visual examples with semantic concepts whenever one or more of the concept models can be mapped to the query. Overall, of the 48 topics defined for the automatic search task, 37 had at least one mapped concept model. For the remaining 11 topics[5], the visual examples are utilized and thus runs 2 and 3 are identical for these topics. This secondary use of visual examples is shown with a "○" in Table II.

Runs 4–6 combine both modalities. In run 4, the features used with the concept models are pre-selected whereas in runs 5 and 6, the features used with the concept models are selected based on the concept-wise feature selection results. Run 6 includes also additional features based on face detection and motion (see Section II-B).

For completeness, we also show the remaining combinations of the three search types as additional runs 7–9. Run 7 combines the text search (run 1) and example-based retrieval (run 2), run 8 consists of content-based search using both the visual examples and the semantic concepts, and run 9 combines all three types of search cues.

A. Text search

For text-based search, the topic-wise English queries were analyzed using the Stanford part-of-speech tagger[6] [17]. The

[5]topics 222, 228, 233, 240, 242, 243, 252, 262, 264, 266, and 268
[6]*http://nlp.stanford.edu/software/tagger.shtml*

TABLE II
AN OVERVIEW OF THE SEARCH TASK RUNS. SEE TEXT FOR DETAILS.

#	run id	text	visual	con-cepts	feat. sel.	addit. feat.	MIAP
1	F_A_1_PicSOM_1_6	•					0.0085
2	F_A_1_PicSOM_2_5		•				0.0116
3	F_A_1_PicSOM_3_4		○	•			0.0208
4	F_A_2_PicSOM_4_3	•	○	•			0.0228
5	F_A_2_PicSOM_5_2	•	○	•	•		0.0199
6	F_A_2_PicSOM_6_1	•	○	•	•	•	0.0189
7	additional run	•	•				0.0110
8	additional run		•	•			0.0222
9	additional run	•	•	•			0.0187

nouns and verbs of each query were used as the text search queries, expanded with synonyms using the WordNet [18] package included in the Lucene search engine.

The ASR/MT documents were used on the shot level. The shot-level retrieval results were spread to the temporally neighboring shots using a triangular kernel of five shots in width.

B. Feature selection for visual examples

The selection of features for content-based retrieval based on the visual examples only is always somewhat problematic since the number of examples is usually quite small. In previous years, we have typically used a small fixed set of features based on their frequency of appearance among the sets of selected features in the high-level feature extraction runs. This year we tried a new method which calculates a score for each feature SOM depending on how well the examples are grouped into tight clusters.

For each SOM, we calculate the pair-wise distances between the best-matching units (BMUs) of the topic examples in question. The distances are passed through a Gaussian kernel function which gives a high score only for short distances and scores approaching zero for long distances. The final suitability score for the SOM is the average over all distances, giving an indication of how many short pair-wise distances were present in the BMU distribution on the SOM. This favors SOMs trained on features that describe some shared property of the topic examples, i.e. the examples end up into one or a few tight clusters.

C. Feature selection for concept models

The sets of features used with the semantic concept models were also selected for each topic in runs 5 and 6, whereas in runs 3–4 the concept models used a fixed set of five features. The most often selected features in the high-level feature extraction task (Section III-B) were used as the fixed feature set. The concept-wise feature selection in runs 5–6 was based on aggregating the lists of concept-wise features of all used concept models for each topic.

D. Semantic concept matching

The search topics were matched with the semantic concepts to facilitate concept-model-based retrieval in runs 3–6. As the concept ontology, we used the 20 high-level features from this year's high-level feature extraction task and 33 of last year's high-level features. The concepts defined in the last year's experiments were trained using the 2007 training data. In addition, keyframe-based face detection was used as an additional feature in run 6. The matching of semantic concepts to queries was based on a lexical analysis of the topic-wise textual descriptions.

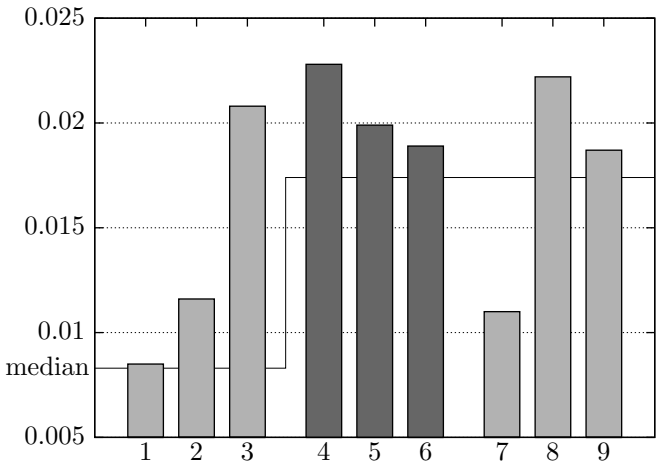

Fig. 3. MIAP values for our runs in the automatic search task. The baseline runs (1–3) and the additional runs (7–9) are shown as lighter bars. The medians of all type A baseline runs and all type A runs are also shown for comparison.

E. Results

The MIAP scores for our search runs are listed in Table II and illustrated in Figure 3. The runs 1–3 are of the required baseline type, among which the maximum and median over all submissions were 0.0365 and 0.0083, respectively. Of our baselines, the ASR/MT text search (run 1) had the worst average performance, and the concept-based retrieval (run 3) performed best. These experiments confirm our earlier results that on average using semantic concepts for automatic search works better than using the visual examples for content-based retrieval, at least when using such a small number of examples that are available in the TRECVID search topics. For this reason, we preferred the semantic concepts in all subsequent non-baseline runs.

All our runs used only the common TRECVID development data, thus qualifying them as type A runs. Over all type A automatic search submissions, the maximum and median values were 0.0669 and 0.0174, respectively. The text search and concept-based retrieval were combined in run 4, resulting in the best overall performance among our search runs.

In run 5, the use of concept-wise feature selection for the concept models degraded the results. This may be due to the aggregation of the lists of concept-wise features for all concepts. Better results might possibly be obtained by using only the concept-wise selected features for each concept. Likewise, the addition of face detection and motion features in run 6 degraded the average results.

The additional runs 7–9 complete our experiments with the remaining combinations of the three search types. The most striking observation from these runs is the relatively low MIAP score of run 9, in which all three types of search cues are used. In that run the inclusion of the text search degrades the results, while in run 4 the text search improves performance over the concept-based retrieval.

V. Video Summarization

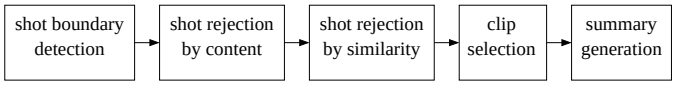

Fig. 4. An overview of the video summarization algorithm.

We participated in the BBC rushes summarization task [19] using an approach consisting of initial shot boundary detection followed by shot content analysis and similarity assessment and pruning. These stages are shown in Figure 4. The basic method is similar to our last year's summarization method, which is described in more detail in [20].

This year we included new detectors for frames containing clapper boards, three different motion detectors, and a speech detector. We also replaced our SOM-based shot boundary detection algorithm [21] with a simple thresholding of the proportion of successfully tracked feature points (Section II-B). A shot boundary corresponds to the case when the proportion of tracked feature points that are lost is greater than a heuristically set threshold. This simple approach works reasonably well with the BBC rushes material as the shot boundaries are almost always cuts.

A. Content detectors

We detect certain types of low and high-level concepts from the rushes material using specialized detectors. To prevent "junk concepts" from appearing in the summaries they are detected and removed. *Color bar test screens*, *black frames*, *white frames* and *clapper boards* are detected using the same SOM-based algorithm as was used for the high-level features in Section III. Small sets of hand-picked frames for each concept were used as positive examples for training the detectors.

The junk concept detection results in a score value for each video frame. These frame-wise scores are then summed in groups of 25 frames to produce a score for each second of video. If this sum exceeds a certain threshold, the whole second was marked as belonging to the concept. The value of the threshold varies from concept to concept and was determined by subjective judgement of the results in the development set. Furthermore, the results were smoothed so that if the previous and next seconds agreed on the detection result, then the central second would have the same result. For the clapper board detectors we used a further heuristic which removed too short or too long sequences.

Based on thresholding the *KLT Histogram* feature, we detect three types of motion: *camera motion*, *camera zoom*, and *object motion*. As in previous tasks, we use the OpenCV face detector for *face detection*.

A speech/non-speech classifier provided by the Speech Group in our department at TKK[7] was used for detection of spoken content. The speech/non-speech detector is a hidden Markov model (HMM) classifier where speech and non-speech sounds are modelled as single states with 24 Gaussian components. Insertion penalties are used in decoding to exclude short speech or non-speech segments. Audio is represented by the 13-dimensional MFCC feature used in the high-level feature detection and search tasks. The audio features are used with their first and second order differentials and treated with cepstral mean subtraction (CMS) and maximum likelihood linear transformation (MLLT). The classifier was trained for a general setting, using training data of over 5 hours of television news data from the Finnish Broadcasting Company (YLE).

After shot boundary detection, we prune the list of shots by using the content detectors. Shots that contain detected faces, speech, or object motion receive positive weight whereas frames detected as color bars or black/white frames lead to negative weight. The shots that score lower than a heuristically set threshold are rejected from further processing. See [20] for details.

These concept detectors are also used in the clip selection stage.

B. Shot similarity pruning

We determine the novelty of a shot based on the shots' visual contents using 256×256-sized SOMs trained with three image features: *Color Layout*, *Edge Co-occurrence*, and *Edge Histogram*. Unlike last year, we use common SOMs trained with the development data for all test videos, instead of training a separate SOM for the frames of each test video. This considerably reduces the computational requirements of the summarization algorithm.

In brief, the low-pass filtered BMU trajectory of the frames within the shot constitutes the shot's signature. We use triangular kernels whose width is set to eight map units. In this task, we ignore the temporal element and construct the shots' signatures by averaging over the frame-wise distributions. The shots are thus modeled as single discrete class densities over the SOM grids. The similarity between two shots is measured using Euclidean distance between the shot signatures.

The most similar shots are removed until both the 2% time limit and an empirically set dissimilarity threshold are reached. This additional threshold is used to reduce the number of shots below the allowed limit of 2% if the remaining shots are still deemed as too similar to each other.

C. Clip selection

A single representative clip of fixed length of one second is selected from each remaining shot to the video summary. The selection is based on scoring the frames within the shot and finding the highest-scoring frame. This frame is then set as the center frame of the one-second clip selected to the summary.

[7] *http://www.cis.hut.fi/projects/speech/*

TABLE III
AN OVERVIEW OF SELECTED SUMMARIZATION RESULTS. THE VALUES ARE AVERAGES OVER ALL TEST VIDEOS (EXCEPT FOR MAX. DURATION).

	Ours	Min	Median	Max	Baseline
Duration (DU)	24.0	13.6	28.1	*31.8*	31.3
Total time (TT)	36.6	22.6	41.4	59.6	59.6
Inclusion (IN)	0.45	0.07	0.45	0.83	0.83
Amount of junk (JU)	3.29	2.52	3.11	3.64	2.66
Redundancy (RE)	3.39	2.02	3.37	3.99	2.02
Tempo/rhythm (TE)	3.05	1.44	2.80	3.38	1.44

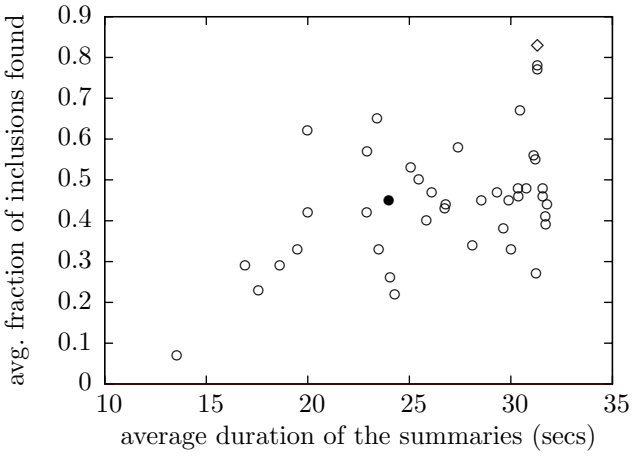

Fig. 5. The average fraction of inclusions found vs. average duration of all submitted summaries; our run shown as "●", baseline run as "◇".

Initially, we favor frames near the center of the shot using linear weighting. The score of frames containing detected faces, speech, object motion, or camera motion is increased using heuristic weights. Correspondingly, scores for any frames containing clapper boards or detected as colorbars, black frames, or white frames are reduced.

The representative clips are then played at normal speed and combined using temporal ordering and fade-outs and fade-ins from black, with the audio track not included.

D. Results

An overview of our summarization results is given in Table III. The shown measures are from the standard measures provided by NIST and described in [19]. The two topmost results (DU and TT) are measures of time (in seconds), IN lists the fraction of ground-truth inclusions found in the summaries, and the three remaining results are from the assessor questionnaire with the range of 1–5 (5 being the desired value in all cases). In can be observed that overall our summarization algorithm obtained results quite close to the median values. The fraction of ground-truth inclusions found in our summaries (45% on average) was equal to the median of all runs, with somewhat shorter times than the median values on average duration and on total time spent on judging the summaries. Of the subjective measures, our run performed better than the median on the amount of junk present and on tempo/rhythm, while the redundancy score was slightly over the median.

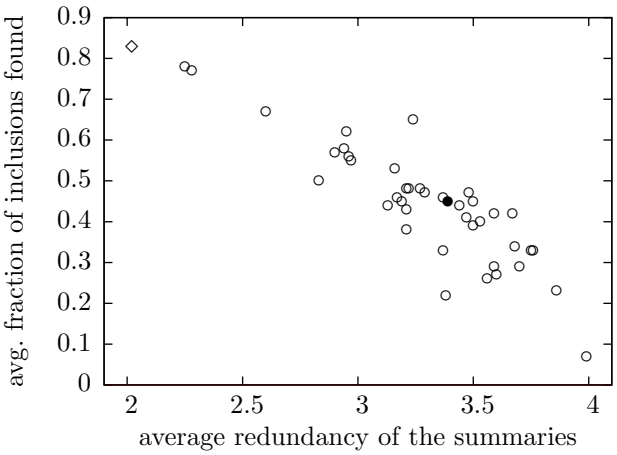

Fig. 6. The average fraction of inclusions found vs. average redundancy of all submitted summaries; our run shown as "●", baseline run as "◇".

It can be assumed that the amount of inclusions included depends on the duration of the generated summaries and, on the other hand, the amount of redundancy present in the summaries. To illustrate these relations, Figures 5 and 6 show plots of the average fractions of inclusions found over the average values of durations and redundancy, respectively, for all submissions. Our submission is highlighted as a filled bullet and the baseline run provided by CMU is shown with a diamond shape. The figures show a clear relation between these measures as the summaries with high fractions of inclusions found tend to have high time expenditure values, as was to be expected. Similarly, the lack of redundancy correlates with lower values of found ground-truth inclusions.

E. Discussion

Overall, our summarization algorithm showed rather average performance on both the objective and subjective measures. The algorithm pipeline shown in Figure 4 is rather generic and it can be assumed that increased performance can be obtained within the framework by improving the individual processing stages. There are a number of ways to improve the summarization algorithm. The used motion detectors were rather crude, and the motion feature could also be used for object segmentation. The playback speed in the summary clips can be varied. As the baseline runs and a number of other submissions to the summarization task have shown, fast-forwarding can be useful. It is also possible to include the temporal element in the shot similarity pruning stage and compare the signatures of individual frames instead of an aggregation of them. In fact, this is already applied in the copy detection task (Section VI-A).

While the computational requirements were not a main concern in these experiments, we obtained a considerable speed-up in the algorithm. The time for overall summary generation was 2 hours 44 minutes (corresponds to 6.2 times real time). In last year's experiments the average time was was 6 hours 17 minutes. The majority of the computational effort was again spent on feature extraction.

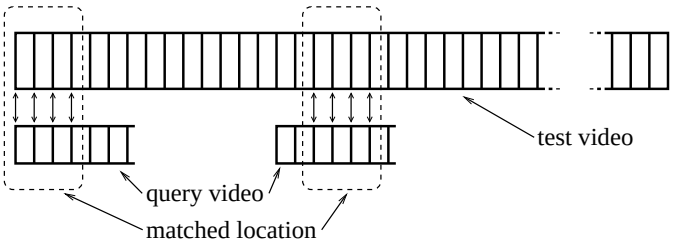

Fig. 7. Matching a query video to a an original video from the test collection in video copy detection.

VI. Video Copy Detection

For the video copy detection pilot task we submitted tentative runs based on our algorithm used for shot similarity determination in the summarization task (c.f. Section V-B). Our runs used only the video modality.

A. Method

Both the test video T and the query video Q are first sampled at one keyframe per second. Let us assume that the test video has L_T and the query video L_Q keyframes and that $L_Q \leq L_T$.

The L_Q keyframes of the query video are aligned to every $L_T - L_Q + 1$ possible positions in the test video so that each keyframe has an aligned pair. The similarity of the query video to the aligned position $p_T \in \{0, 1, \ldots, L_T - L_Q\}$ in the test video is then measured using the low-pass filtered BMU signatures S of the keyframes. We use a triangular kernel of four map units in width. Unlike in summarization, we compare the aligned keyframes individually.

Due to the setup in the copy detection task where the actual copied clip may only be a part of the query video, we compare $l \leq L_Q$ successive keyframes at a time and consider only the best match. Again, there are $L_Q - l + 1$ possible positions p_Q for the matched location. The similarity of the query video to the test video at position p_T is thus

$$D(T, Q; p_T) = \arg\min_{p_Q} \frac{1}{l} \sum_{i=0}^{l-1} d(S_T(i+p_Q+p_T), S_Q(i+p_Q))$$

Figure 7 shows an example where the query video has been aligned to two positions, $p_T = 0$ and $p_T = 14$, to the test video. For the purpose of illustration, we use $L_Q = 7$ and $l = 4$ in the figure.

In the actual experiments, we used a concatenated signature consisting of 64×64-sized SOMs trained with *Color Layout* and *Edge Histogram* features and Euclidean distance metric normalized to $d \in [0, 1]$. In these pilot experiments, we used a fixed overlap of $l = 20$ frames and considered only the best match between a query and test video, i.e. $D(T, Q) = \arg\min_{p_T} D(T, Q; p_T)$. A detection was assumed if $1 - D(T, Q) > \tau$, where τ is an empirically set threshold.

B. Results

Due to the preliminary nature of the experiments and the lack of training data, it was to be expected that the copy detection results are modest. We submitted three runs which differ only on the value of τ ($\tau = 0.1$, 0.3, and 0.5). Of the ten transformations, our runs had the smallest value of minimal normalized Detection Cost Rate (min DCR=0.906) for transformation 3 (insertion of pattern).

VII. Conclusions

All the experiments reported in this paper were performed using a common framework for content-based indexing and retrieval of visual objects. The basic functionality of the framework has remained the same as in previous years' experiments and we have introduced new additions to the system each year.

In high-level feature extraction, the used approach shows relatively good performance but fails to reach the level of the state-of-the-art methods based on SVMs. The efficiency and scalability of our approach, however, makes it feasible to construct models for large sets of concepts such as the LSCOM [22] ontology without high computational requirements.

The temporal and inter-concept co-occurrences can be used to improve the results in a post-processing algorithm. The selection of a suitable post-processor is, however, a difficult problem requiring further attention.

In automatic search, the results validate our earlier observations that semantic concept detectors can be a considerable asset in automatic video retrieval. However, it often turns out that the selection of various parameters for each high-level feature or search topic separately using the development set is fairly inefficient and parameters optimized based on the whole set of concepts or topics tend to generalize better to the test set. Furthermore, the inclusion of additional features such as audio features, face detection, or motion features had little effect in the overall performance in the current setup.

Acknowledgments

This work was supported by the funding of Academy of Finland for the *Finnish Centre of Excellence in Adaptive Informatics Research*. We also wish to thank the Muvis group, and especially Kiranyaz Mustafa Serkan at Tampere University of Technology [9] for supplying audio features that were used in our experiments. For the speech/non-speech classifier we thank the Speech group at Helsinki University of Technology, and especially Ulpu Remes and Kalle J. Palomäki.

References

[1] Alan F. Smeaton, Paul Over, and Wessel Kraaij. Evaluation campaigns and TRECVid. In *MIR '06: Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval*, pages 321–330, New York, NY, USA, 2006. ACM Press.

[2] Jorma Laaksonen, Markus Koskela, and Erkki Oja. PicSOM—Self-organizing image retrieval with MPEG-7 content descriptions. *IEEE Transactions on Neural Networks, Special Issue on Intelligent Multimedia Processing*, 13(4):841–853, July 2002.

[3] Markus Koskela, Jorma Laaksonen, Mats Sjöberg, and Hannes Muurinen. PicSOM experiments in TRECVID 2005. In *Proceedings of the TRECVID 2005 Workshop*, pages 262–270, Gaithersburg, MD, USA, November 2005. Available online at http://www-nlpir.nist.gov/projects/tvpubs/tv.pubs.org.html.

[4] Mats Sjöberg, Hannes Muurinen, Jorma Laaksonen, and Markus Koskela. PicSOM experiments in TRECVID 2006. In *Proceedings of the TRECVID 2006 Workshop*, Gaithersburg, MD, USA, November 2006. Available online at http://www-nlpir.nist.gov/projects/tvpubs/tv.pubs.org.html.

[5] C. Petersohn. Fraunhofer HHI at TRECVID 2004: Shot boundary detection system. In *TREC Video Retrieval Evaluation Online Proceedings*. TRECVID, 2004.

[6] Markus Koskela, Mats Sjöberg, Ville Viitaniemi, Jorma Laaksonen, and Philip Prentis. PicSOM experiments in TRECVID 2007. In *Proceedings of the TRECVID 2007 Workshop*, Gaithersburg, MD, USA, November 2007. Available online at http://www-nlpir.nist.gov/projects/tvpubs/tv.pubs.org.html.

[7] Carlo Tomasi and Takeo Kanade. Detection and tracking of point features. Technical Report CMU-CS-91-132, Carnegie Mellon University, April 1991.

[8] Chitra Dorai and Vikrant Kobla. Perceived visual motion descriptors from MPEG-2 for content-based HDTV annotation and retrieval. In *Proceedings of IEEE 3rd Workshop on Multimedia Signal Processing*, pages 147–152, Copenhagen, Denmark, September 1999.

[9] S. Kiranyaz and M. Gabbouj. Generic content-based audio indexing and retrieval framework. *Vision, Image and Signal Processing, IEE Proceedings*, 153(3):285–297, June 2006.

[10] Marijn Huijbregts, Roeland Ordelman, and Franciska de Jong. Annotation of heterogeneous multimedia content using automatic speech recognition. In *Proceedings of the second international conference on Semantics And digital Media Technologies (SAMT)*, Lecture Notes in Computer Science, Berlin, December 2007. Springer Verlag.

[11] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'01)*, 2001.

[12] D. Chai and K. N. Ngan. Face segmentation using skin-color map in videophone applications. *IEEE Transactions on Circuits and Systems for Video Technology*, 9(4):551–564, June 1999.

[13] Ville Viitaniemi, Mats Sjöberg, Markus Koskela, and Jorma Laaksonen. Exploiting temporal and inter-concept co-occurrence structure to detect high-level features in broadcast videos. In *Proceedings of 9th International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS 2008)*, pages 12–15, Klagenfurt, Austria, May 2008.

[14] Markus Koskela and Jorma Laaksonen. Semantic concept detection from news videos with self-organizing maps. In Ilias Maglogiannis, Kostas Karpouzis, and Max Bramer, editors, *Proceedings of 3rd IFIP Conference on Artificial Intelligence Applications and Innovations*, pages 591–599, Athens, Greece, June 2006. IFIP, Springer.

[15] Stéphane Ayache and Georges Quénot. Video corpus annotation using active learning. In *Proceedings of 30th European Conference on Information Retrieval (ECIR'08)*, pages 187–198, Glasgow, UK, March-April 2008.

[16] Emine Yilmaz and Javed A. Aslam. Estimating average precision with incomplete and imperfect judgments. In *Proceedings of 15th International Conference on Information and Knowledge Management (CIKM 2006)*, Arlington, VA, USA, November 2006.

[17] Kristina Toutanova and Christopher D. Manning. Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. In *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC 2000)*, pages 63–70, Hong Kong, October 2000.

[18] Christiane Fellbaum, editor. *WordNet: An Electronic Lexical Database*. The MIT Press, 1998.

[19] Paul Over, Alan F. Smeaton, and George Awad. The TRECVid 2008 BBC rushes summarization evaluation. In *TVS '08: Proceedings of the International Workshop on TRECVID Video Summarization*, pages 1–20, Vancouver, British Columbia, Canada, 2008. ACM.

[20] Markus Koskela, Mats Sjöberg, Jorma Laaksonen, Ville Viitaniemi, and Hannes Muurinen. Rushes summarization with self-organizing maps. In *Proceedings of the TRECVID Workshop on Video Summarization (TVS'07)*, pages 45–49, Augsburg, Germany, September 2007. ACM Press.

[21] Hannes Muurinen and Jorma Laaksonen. Video segmentation and shot boundary detection using self-organizing maps. In *Proceedings of 15th Scandinavian Conference on Image Analysis (SCIA 2007)*, pages 770–779, Aalborg, Denmark, June 2007.

[22] DTO Challenge Workshop on Large Scale Concept Ontology for Multimedia. LSCOM lexicon definitions and annotations version 1.0. Technical Report #217-2006-3, Columbia University, March 2006.