PicSOM Experiments in TRECVID 2005

Markus Koskela, Jorma Laaksonen, Mats Sjöberg, Hannes Muurinen

Neural Networks Research Centre, Helsinki University of Technology
P.O.Box 5400, FI-02015 TKK, FINLAND

ABSTRACT

Our experiments in TRECVID 2005 include participation in the high-level feature extraction and search tasks. In the high-level feature extraction task, we applied a method of representing semantic concepts as class models on a set of parallel Self-Organizing Maps (SOMs). We submitted one run, `A_PicSOM_1`, in which we applied a feature selection scheme for each concept separately. The results showed that the SOM-based class models can be used for representing semantic concepts on multimodal feature indices and that the proposed method is suitable for detecting video shots with specific semantic content.

In the search task, we submitted a total of seven runs (three automatic, three manual, and one interactive run). Our main motivation was to study the utilization of parallel multimodal features and class models compared to using only text-based queries. The overall settings for the runs were as follows:

- `F_A_1_SOM-F1_7`: a baseline automatic run using only ASR/MT output
- `F_A_2_SOM-F2_3`: an automatic run using ASR/MT output, multimodal features, and class models
- `F_A_2_SOM-F3_5`: an automatic run using multimodal features and class models
- `M_A_1_SOM-M1_6`: a baseline manual run using only ASR/MT output
- `M_A_2_SOM-M2_4`: a manual run using ASR/MT output and multimodal features
- `M_A_2_SOM-M3_2`: a manual run using ASR/MT output, multimodal features, and class models
- `I_A_2_SOM-I_1`: an interactive run

Both in the automatic and manual experiments, we observed that the proposed method is able to combine the text query, multimodal features and class models successfully. In both cases, the overall best results are obtained using all three information sources with the MAP value being nearly double when compared to text-only search. Our small-scale interactive search experiments were performed with our prototype retrieval interface supporting only relevance feedback -based retrieval. Still, the experiments demonstrate that the proposed method can also be used in an interactive setting, where the search is guided with iterative feedback from the user.

I. INTRODUCTION

In this paper, we describe our experiments with the PicSOM system in TRECVID 2005. We participated in the high-level feature extraction and automatic, manual, and interactive search tasks. As this is our first participation in TRECVID, the first objective this year was to implement all the necessary functionality into our PicSOM system, which has not been previously used for this type of video retrieval, but mainly for still images.

One motivation for the performed experiments was to test our existing method for indexing multimodal hierarchical objects and relevance propagation for digital video. Earlier we have applied a similar approach to indexing and retrieval of web pages [1]. In the high-level feature extraction task, we applied our method of representing *semantic concepts* as *class models* on a set of parallel feature indices. The class models were used for image group annotation in [2], whereas in the TRECVID context we are interested in shots of the test collection that have the highest likelihood of being relevant to the given concept. Also, instead of using a fixed set of features, we apply a manual feature selection scheme separately for each concept. In the three types of search tasks, we wanted to experiment with combining a text query with a set of multimodal features and both positive and negative class models in video retrieval.

The rest of the paper is organized as follows. The extension of the PicSOM system for video retrieval and the used multimodal features are briefly described in Section II. In Section III we discuss extending the use of multiple SOM indices from representing online queries into modeling semantic concepts. Our experiments for the high-level feature extraction and search tasks are described in Sections IV and V, respectively, and conclusions are presented in Section VI.

II. Indexing Video with PicSOM

The PicSOM system [3] is a general framework for research on content-based indexing and retrieval of visual objects. The system is based on using several complementary Self-Organizing Maps (SOMs) [4], each trained with separate feature data. The SOM defines an elastic, topology-preserving grid of points that is fitted to the input space. The distribution of the data vectors over the map forms a two-dimensional discrete probability density. As a result, the different SOMs impose different similarity relations on the objects. The task of the retrieval system then becomes to select, weight and combine these similarity relations so that their composite would approximate the human notion of similarity in the current retrieval task as closely as possible. The parallel SOMs can also be augmented with other types of additional information and different indices. In this application, a such source of information is the ASR/MT text output, for which the inverted file provides an effective indexing structure.

Ordinary retrieval usage of the PicSOM system is based on relevance feedback: the user determines the relevance of all returned objects and marks the ones she considers relevant to the current task, the others are deemed non-relevant. The SOM units on all maps are awarded positive and negative scores for every relevant and non-relevant object mapped in them, respectively. The system remembers all responses the user has given since the query was started in these sparse value fields.

Due to the topology preservation property of the SOM, we are also motivated to spread this relevance information to the neighboring map units on the SOM grids. Spreading of the response values can be performed by convolving the sparse value fields with a tapered kernel function. This results in polarization of the entire map surface in areas of positive and negative cumulative relevance.

By locating a given database object in all SOM indices, we get its relevance scores with respect to the different features. Then, as the response values of the parallel indices are mutually comparable, we can determine a global ordering and the overall best candidate objects using simple unweighted linear combination.

A. Indexing hierarchical objects

An extension of the PicSOM system to support general multi-part and multimodal objects having a natural hierarchy was proposed in [1]. Such object hierarchies can be found e.g. in web pages, e-mail and MMS messages, and also digital video. The multi-part hierarchy for video shots used for indexing the TRECVID 2005 collection is illustrated in Fig. 1. The video shot itself is considered as the main or parent object. The keyframes (one or more) associated with the shot, the audio track, and ASR/MT text are linked as children of the parent object. This hierarchy could also be extended further, e.g. the image objects could have image segments as subobjects, the original video is the video shot's parent, etc. All object modalities may have one or more SOMs or other feature indices, and thus all objects in the hierarchy may have links to a set of associated feature indices.

In this setting, the relevance of each object in the tree structure can be considered as a property of not only the object itself, but to some extent also of the other objects in the same structure. With this approach, denoted as *relevance sharing*, any relevance assessment or existing annotation can be propagated from the original object to its parent, children and siblings, depending on the application [1]. For example, if an e-mail message is considered relevant in a certain query, its attachments will also get increased relevance values. As a result of this *relevance propagation*, any e-mail message with similar attachments will then later get a share of that relevance.

In the case of video shot retrieval, both the object of retrieval and the target of the relevance assessments is the video object[1]. Therefore, the relevance assessments are first propagated from the parent, i.e. video shot, object to the children objects. The relevant and non-relevant objects in the hierarchy are then mapped to the corresponding SOMs and kernel smoothing is performed. Finally, when determining the best-scoring video shots, the relevance scores are propagated from the subobject indices to their parent objects.

B. Multimodal features

In indexing the video shots of the TRECVID 2005 collection, we used in total four video features, six still image features, and one audio feature. A separate 256×256-sized SOM was trained for each of these eleven features. For the ASR/MT output, we used two alternative conceptwise text features based on an inverted file in the high-level feature extraction task. All these features are briefly described below.

[1]with the exception of the example images in the search topics of TRECVID 2005

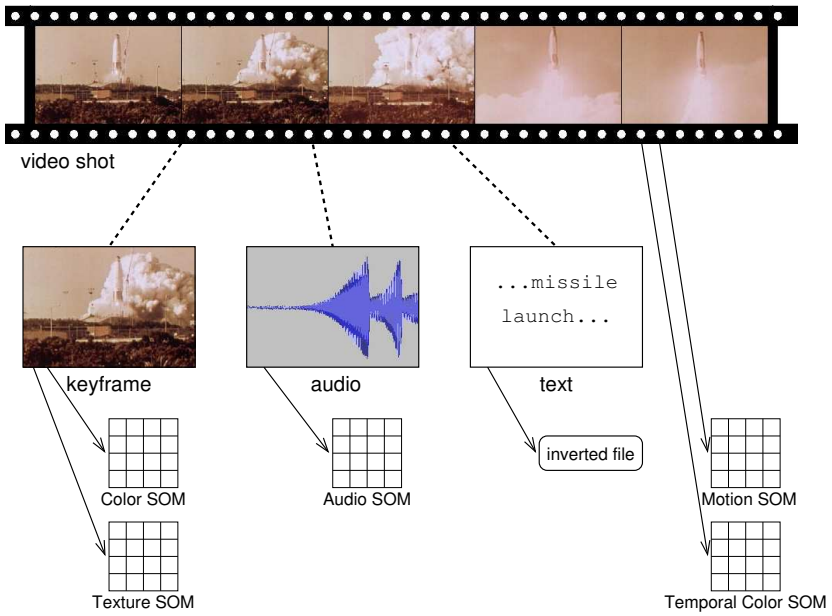

Fig. 1. The hierarchy of video and multimodal SOMs.

1) Video features: On the video shot level, we used the MPEG-7 [5] *Motion Activity* descriptor (MA) and temporal versions of three still image features. The temporal image features are calculated as follows. The video clip is first divided into five non-overlapping parts with equal lengths. The resulting video clips are called slices. All the frames of the five slices are then extracted, and each frame is divided into five separate zones: the upper, the lower, the left hand side, the right hand side and the central zone. A feature vector is calculated separately for each zone, and then the zone feature vectors are concatenated to form a vector depicting the whole frame. All the frame feature vectors of a video slice are then averaged to form the feature vector for the slice. Finally the feature vectors of the five slices are concatenated to form the feature vector of the video clip. Several different video features can be calculated using this method by varying the feature that is calculated for the zones of the frames. *Average Color* (AC), *Color Moments* (CM) and *Texture Neighborhood* (TN) features were the three zone features that were used.

The Average Color feature vector is a three element vector that contains the average RGB values of all the pixels within the zone. The Color Moments feature is calculated by separating the HSV color channels from the zone. Then the values of the color channels are treated as probability distributions, and the first three moments (mean, variance and skewness) are calculated for each distribution. The feature vector contains the three moment values for the three color channels.

The Texture Neighborhood feature is calculated from the Y (luminance) component of the YIQ color representation of the zone pixels. The 8-neighborhood of each inner pixel is examined, and a probability estimate is calculated for the probabilities that the neighbor pixel in each surrounding relative position is brighter than the central pixel. The feature vector contains these eight probability estimates.

2) Image features: For the keyframe indices we used a set of six standard MPEG-7 [5] descriptors, viz. *Color Layout* (CL), *Color Structure* (CS), *Dominant Color* (DC), *Scalable Color* (SC), *Edge Histogram* (EH), and *Homogeneous Texture* (HT). The descriptors were extracted globally from every keyframe in the collection, i.e. no segmentation or zoning was used. In addition, these descriptors were extracted from the example images of the search topics. An illustration of a SOM of keyframes trained with the Edge Histogram descriptor is shown in Figure 2.

3) Audio features: The Mel-scaled cepstral coefficient, or shortly *Mel Cepstrum* (CE) is the discrete cosine transform (DCT) applied to the logarithm of the mel-scaled filter bank energies. The number of coefficients taken is 12, and these are organized as vector. Finally the total power of the signal is appended to the vector giving a

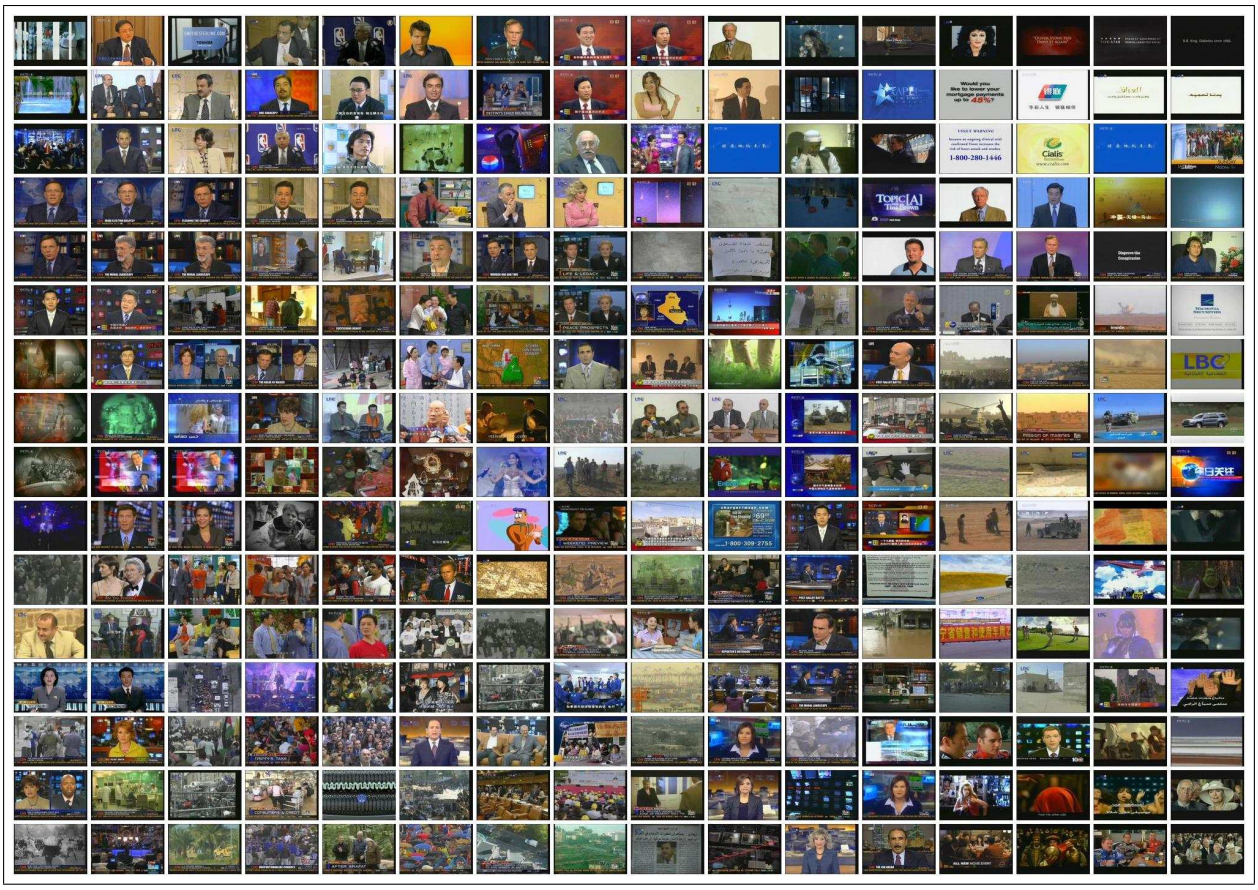

Fig. 2. An example keyframe SOM of 16×16 map units trained with MPEG-7 *Edge Histogram* descriptor. In the experiments, SOMs of size 256×256 map units were used.

feature vector of length 13.

4) Text features: Unlike the other features, an inverted file instead of a SOM index was used for the ASR/MT output. The extension of the PicSOM system for using such indices in parallel with the SOMs was presented in [6].

For the high-level feature extraction task, the text features were constructed by gathering concept-dependent lists of most informative terms. Let us denote the number of shots in the development set associated with concept c as N_c and assume that of these shots, $n_{c,t}$ contain the term t in the ASR/MT output. After preprocessing and stemming, the following measure is applied for term t regarding the concept c:

$$S_c(t) = \frac{n_{c,t}}{N_c} - \frac{n_{all,t}}{N_{all}} \ .$$

For every concept, we record the 10 and 100 most informative terms and use them as alternative text features.

III. Modeling Semantic Concepts

In addition to the relevant and non-relevant object sets during online queries, the sparse value fields can also be constructed with any other object subsets, such as groups of objects with semantically similar content. Such modeling of mid-level semantic concepts can be a very useful step in supporting high-level querying on visual data. As in retrieval, the sign of the impulses depends on the relevance of the concept: positive impulses are used for relevant concepts, negative impulses for non-relevant concepts. The kernel smoothing step is again useful to spread the concept information and also to ease visual inspection of large SOMs (see Figure 3 (right) for an example). Areas occupied by objects of the concept in question are shown with gray shades. In Figure 3 (left), it is visualized how the original very-high-dimensional pattern space is first projected to feature space, the vectors of which are then used in training a SOM.

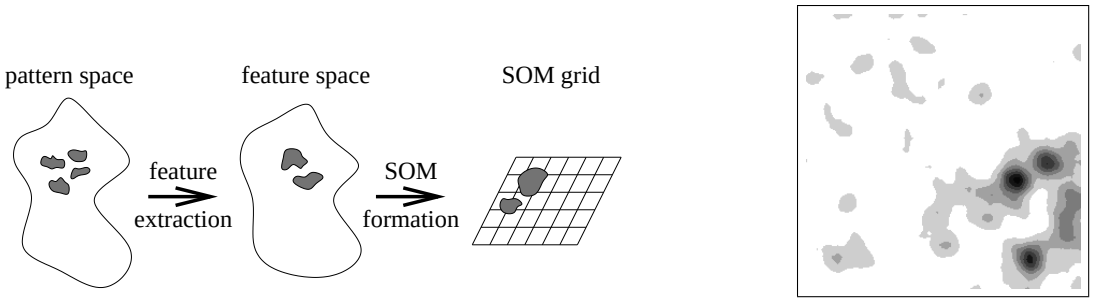

Fig. 3. Left: Stages in creating a class model from the very-high-dimensional pattern space through the high-dimensional feature space to the two-dimensional SOM grid. Right: An example class model (concept *explosion/fire* on the MPEG-7 Color Layout SOM).

These class-conditional distributions or class models can be considered as estimates of the true distributions of the semantic concepts in question, not on the original feature spaces, but on the discrete two-dimensional grids defined by the used SOMs. Thereby, instead of modeling probability densities in the high-dimensional feature spaces, we are essentially performing kernel-based estimation of discrete class densities over the SOM grid. Depending on the variance of the kernel function, these kernels will overlap and weight vectors close to each other will partially share each other's probability mass.

As an example, the most representative objects of a given semantic concept can be obtained by locating the SOM units, and the objects mapped to these units, that have highest responses on the estimated class distribution. Combining the responses of multiple features can be performed similarly as in the retrieval stage, after which we obtain the overall most representative objects of a specific concept regarding all the used features. Taking this approach with new data we end up with the concept detection method used in our experiments in the high-level feature extraction task.

Multiple class models can be linearly combined in an application where multiple concepts are relevant or non-relevant simultaneously. This is the case in the search task where the topic of the search may warrant using one or more models to represent either positive or negative concepts. Auxiliary class models could also be utilized in concept detection, especially negative ones, as a high response on a contradictory concept can be helpful in discarding false positives.

IV. High-level Feature Extraction Experiments

For the high-level feature extraction task, a method for estimating the joint distribution of video shot representations and semantic concepts is required. For this purpose, we utilize the existing common annotations for the development set and construct class models for the semantic concepts in the list of high-level features to be detected, as described in Section III. In these experiments, we do not use any specific detectors or concept-specific processing, so all ten concepts are detected using the same procedure based only on the ground-truth annotation for each concept. The other concepts defined in the LSCOM-lite ontology were not utilized, nor any other data. We submitted one run, `A_PicSOM_1`, for this task.

A. Feature selection

The set of used features was selected for each concept separately. For this purpose, we applied a SFS-type feature selection scheme, in which we begin with an empty set of features and compute a criterion value for each of the potential features. If adding the feature with the highest value improves the overall result, the feature is used in the task and the process is continued. Otherwise we stop the selection process. As the optimization criterion we used the average precision at 2000 returned items with two-fold cross validation on the development set.

The eleven features with SOM indices described in Section II-B along with the two concept-dependent text features were always included as potential features. The text features were alternative to each other, so only one of them could be selected. The conceptwise sets of selected features are listed in Table I (the feature abbreviations are listed in Section II-B). As can be seen, the selection process typically resulted in 4–7 parallel features. The *prisoner*

TABLE I

FEATURES USED IN THE HIGH-LEVEL FEATURE EXTRACTION TASK FOR EACH CONCEPT. IN ADDITION, THE ASTERISKS DENOTE FEATURES USED IN THE SEARCH TASK DY DEFAULT.

high-level feature	video				image						audio	text	
	MA*	AC*	CM*	TN*	CL*	CS	DC	SC	EH*	HT*	CE	10	100
38: walking/running	×	×							×				
39: explosion/fire			×		×					×			×
40: maps		×	×			×		×	×	×			×
41: flag-us	×		×	×					×			×	
42: building		×	×				×		×	×			×
43: waterscape/waterfront	×	×	×	×	×				×			×	
44: mountain	×	×		×					×	×	×		
45: prisoner										×			
46: sports	×	×	×		×				×			×	
47: car	×	×	×	×	×				×	×	×		×

concept was a notable exception as adding any second feature, including the text features, beside Homogeneous Texture resulted in performance degradation.

B. Results

The results of the run `A_PicSOM_1` in the high-level feature extraction task are shown in Figure 4. The MAP score of our was 0.196, compared to the median of 0.141 and best run of 0.336. From the conceptwise results it can be seen that the performance on concepts *maps* (40), *waterscape/waterfront* (43), and *mountain* (44) is relatively good whereas on other concepts the results are rather close to the median.

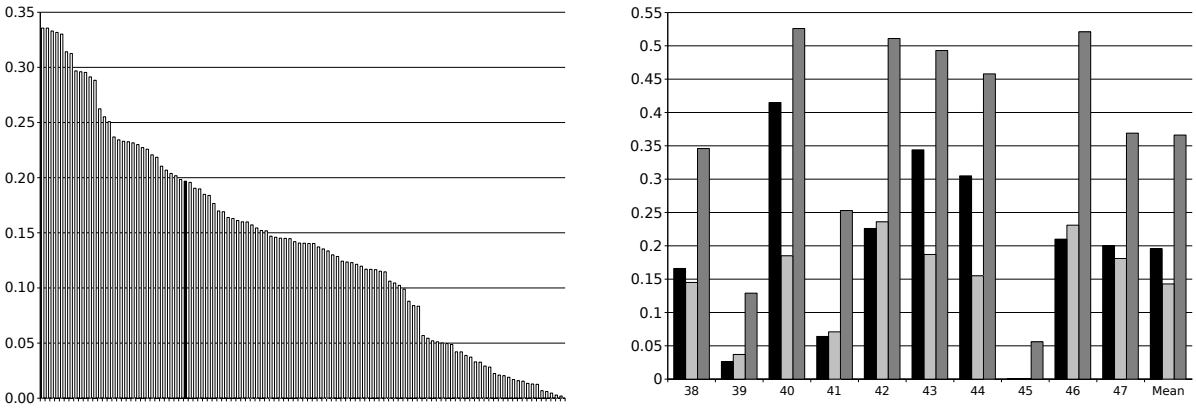

Fig. 4. Left: MAP values for all runs submitted to the high-level feature extraction task, our run highlighted. Right: Conceptwise average precision values for our run (black) compared to median (light gray) and maximum (dark gray) values over all runs.

V. SEARCH EXPERIMENTS

For the search task, we submitted three automatic, three manual, and one interactive run. The run ids for the runs are listed in Table II. The main motivation for the experiments was to test our approach in combining a text query with multiple multimodal features and both positive and negative class models in video shot retrieval.

Before the search experiments, the used features were extracted from the provided example videos and images for the search topics. The audio track was also extracted from the videos and all matching keyframes for the videos were gathered from the collection of common keyframes of the development set. After feature extraction, the best-matching map unit for each example object was located on every SOM of the corresponding modality in use and the objects were mapped to them.

TABLE II

AN OVERVIEW OF PERFORMED RUNS IN THE SEARCH TASK. THE MEDIAN AND MAXIMUM VALUES OF MAP ARE CALCULATED FROM THE TYPE A RUNS ONLY.

run id	type	text query	features	concepts	MAP	median	max
F_A_1_SOM-F1_7	automatic	automatic modification	-	-	0.0473	0.0385	0.0674
F_A_2_SOM-F2_3	automatic	automatic modification	default set	string matching	0.0831	0.0476	0.119
F_A_2_SOM-F3_5	automatic	-	default set	string matching	0.0651	0.0476	0.119
M_A_1_SOM-M1_6	manual	manually modified	-	-	0.0568	0.0519	0.0815
M_A_2_SOM-M2_4	manual	manually modified	default set	-	0.0890	0.0760	0.169
M_A_2_SOM-M3_2	manual	manually modified	default set	manually selected	0.109	0.0760	0.169
I_A_2_SOM-I_1	interactive	user modifiable	user selectable	user selectable	0.158*	0.256	0.408

A default set of multimodal features was gathered based on feature-wise performance in the feature selection process of the high-level feature extraction task, as listed in Table I. Therefore, all four video features (Motion Activity, Average Color, Color Moments, Texture Neighborhood) and three image features (Color Layout, Edge Histogram, and Homogeneous Texture) were used by default in the search experiments. Only the interactive search mode allowed changing this set. As the text features in the high-level feature extraction task were concept-specific, they were not eligible for the search task.

Instead of the concept-specific text features, a vector space was generated using the terms in the ASR/MT output and text-based queries were used in all but one experiment of automatic search. In order to balance between the longer ASR/MT text segments for the Arabic and Chinese shots and the shorter ASR text segments for the English shots, we always concatenated five successive English ASR text segments together for elongating them. In preprocessing, the text excerpts and query expressions were stemmed using the Porter stemming algorithm [7] and the SMART stop list [8] for common terms was applied. The topic description was used as a default text query after applying the stop list and stemming, although in all subtasks the default query was modified according to the type of the subtask as described below. The query terms were weighted using inverse document frequency.

Table II gives an overview of the experiments performed in the search task. The runs F_A_1_SOM-F1_7 and M_A_1_SOM-M1_6 constitute the required ASR/MT baseline runs for automatic and manual modes. The results of the search experiments are discussed in Section V-D.

A. Text query processing and concept matching for automatic search

The ASR/MT output of non-English videos included additional information, such as if a certain proper name was a person, location or organization. Of these we used the person and location information to create an index of "known" proper names and if they were persons or locations. Furthermore, discriminative words were picked up from the LSCOM-lite ontology descriptions to create a word–concept index. For example the word "minister" would map to the concept *government_leader*. This information was used in processing the text queries in the automatic search experiments before being used in the retrieval.

Initially proper names were identified in the text query by recognizing single or consecutive words with a capitalized first letter. These proper names were then compared with the index of known proper names using the Levenshtein distance. If the index name with the shortest distance was sufficiently close to the query name then the query name was deemed to be a misspelled version of the index name. The tolerance was dependent on the length of the query name, so that for short names a shorter Levenshtein distance was needed for acceptance. The identified misspelled words were corrected and the query string was cleaned, i.e. lowercased, dots and commas removed, and unnecessary text such as the preceding "Find shots of" discarded.

Additionally, the word–concept index was used to identify words that might indicate useful class models. The presence of negative words, like a preceding "not" word would negate the class model. Finally if a person's name was identified previously, the class models *face* and *person* were added automatically. In addition, in order to reduce the number of studio shots retrieved, the class model *studio* was always used as a negative concept.

Table III shows the transformations done to a fictitious query string "Find shots of Omar Karammi, the former prime minister of Lebannon". The first row in the table shows the original string, and the second row the

identifications found by the system. The identification WORD-CONCEPT signifies a word found in the word–concept index. The third row shows the actions or transformations performed, CORRECT means correcting a spelling error. The fourth row shows the class models added, the sign before the class name identifies a positive or negative class model. The last row shows the final processed text, capital letters, dots and commas removed.

TABLE III
TEXT QUERY EXAMPLE IN AUTOMATIC SEARCH.

original text	*Find shots of*	*Omar Karammi,*	*the former prime*	*minister*	*of*	*Lebannon*
identification		PERSON		WORD-CONCEPT		LOCATION
actions	DELETE	CORRECT(Omar Karami)				CORRECT(Lebanon)
classes		+face, +person		+government_leader		
processed text		*omar karami*	*the former prime*	*minister*	*of*	*lebanon*

B. Settings for manual search

In the manual search experiments, our motivation was to study the utilization and usefulness of incorporating the multimodal features and class models to the retrieval compared to using only text-based queries. The manual search runs share the properties of the automatic runs, the only difference being that instead of using automatic methods, text query modification and concept selection are performed manually.

The selection of the manually-modified text queries and positive and negative class models was not performed systematically, but chosen based on a small number of reference queries on the development set. For each topic, the exact text queries and the class models used in manual search as positive and negative are listed in Table IV. The negative class model *studio* was used for all topics except 0163 and 0172.

C. Interactive search experiment

Our interactive search experiment for this year's TRECVID can be considered preliminary as the user interface was not specifically designed for video shot browsing and retrieval. The used user interface was a slightly modified version of the basic PicSOM user interface designed for prototyping relevance feedback based retrieval of images.

Each query with the system began with an initial screen which contained all modifiable parameters for the search session. The initial screen contained the description of the current search topic, the external example images and videos, a text query box, and the lists of available multimodal features and class models. All parameters could be changed from the default values, which were set as follows: The textual description of the search topic was used as the default text query after removing the preceding "Find shots of". All example images and videos were marked as relevant and the default set of multimodal features (shown in Table I with asterisks) was selected. All class models were turned off by default, but could be selected either as positive or negative models. After the user had made the initial selections, the system proceeded to the ordinary round-based retrieval operation.

The system was set to always return 25 best-scoring shots. On each round, the query continues as the user assesses the returned shots and marks the ones that she considers relevant. The remaining ones are regarded as non-relevant. All previously found relevant objects are shown below to facilitate their subsequent removal from the set of relevant objects. The user interface also supports the return to the previous query round or back to the initial screen, where it was possible to make any changes to the parameters and start a new search. By clicking on a thumbnail of a video shot, the system displayed the actual video shot and all keyframes associated with the shot in a pop-up window. The ASR/MT text associated with a given shot was displayed in an overlay window when the user moved the cursor over the corresponding thumbnail. The user interface is displayed in Figure 5.

The interactive experiment was performed by five researchers of our laboratory, four of which are involved in different research topics with the PicSOM system. They did not have any direct contact with the TRECVID 2005 test data prior to the experiment. The arrangement was due to time constraints and the unpolished state of our current user interface for interactive video search. The search sessions were limited to 15 minutes, during which time, on average, the search was started from the initial screen 2.7 times and a total of 16 rounds of retrieval were performed. In the results submitted to NIST, the union of all relevant shots in the end of a query were returned as

TABLE IV

TEXT QUERIES AND CLASS MODELS USED IN MANUAL SEARCH.

topic	text query	positive class models	negative class models
0149	condoleezza rice	government_leader	studio
0150	iyad allawi	government_leader	studio
0151	omar karami	government_leader	studio
0152	hu jintao	government_leader	studio
0153	tony blair	government_leader	studio
0154	mahmoud abbas abu mazen	government_leader	studio
0155	iraq baghdad		weather, studio
0156	tennis		studio
0157	shaking hands	meeting	studio
0158	helicopter	sky	studio
0159	george w. bush	government_leader, walking/running	studio
0160	fire smoke	explosion/fire	studio
0161	banners signs	people_marching	studio
0162	enter leave	walking/running, urban	studio
0163	meeting	meeting, office	
0164	boat ship	boat/ship, waterscape/waterfront	studio
0165	basketball		studio
0166	palm trees	vegetation	studio
0167	plane take off	airplane, sky	studio
0168	road car	road, car	studio
0169	tank military	military, desert	studio
0170	building	building, urban	studio
0171	soccer football goal		studio
0172	office	office	

the result of the query without extending the result set with any other shots. The 24 interactive searches resulted in a total of 829 shots marked as relevant, 689 (83%) of which were then judged relevant by the NIST assessors.

In order to be more comparable with other submitted interactive runs, the result sets were later augmented to the allowed size of 1000 shots. The augmentation was performed as a virtual additional query round in which all the relevant-marked shots during the original search are used as positive examples. The same class model selections as in the manual search experiments were used (Table IV). The virtual query round was then set to return the number of shots that were missing from the allowed number of 1000 shots.

D. Results

The MAP scores for all our search runs are listed in Table II. For comparison, the median and maximum values of MAP calculated from all corresponding type A runs are also displayed. For the ASR/MT baseline runs (`F_A_1_SOM-F1_7` and `M_A_1_SOM-M1_6`), the median and maximum values of MAP are calculated from the baseline runs.

Overall, the results indicate that video search performance of the PicSOM system can be improved by augmenting a text query with automatically extracted multimodal features and suitable semantic class models. This improvement can be observed both in automatic and manual search experiments. Perhaps surprisingly, using only the multimodal features and class models resulted in a higher MAP value than using only the text query in automatic search. This result naturally depends fully on the selection of topics and there is substantial topicwise variation. Still, the combined run clearly outperforms both partial runs.

In the manual search experiments, we begin with the text query and first add the multimodal features, followed by the addition of manually selected positive and negative class models. As can be seen in Table II, both additions increase the resulting MAP value.

The results of our interactive search experiment were clearly below the median of all results. This was quite expected as the user interface was not designed or optimized for this kind of experiments, and we did not augment the result lists to the allowed 1000 shots. After augmenting the result lists as described in Section V-C, the MAP value of our interactive run increased from 0.158 to 0.198.

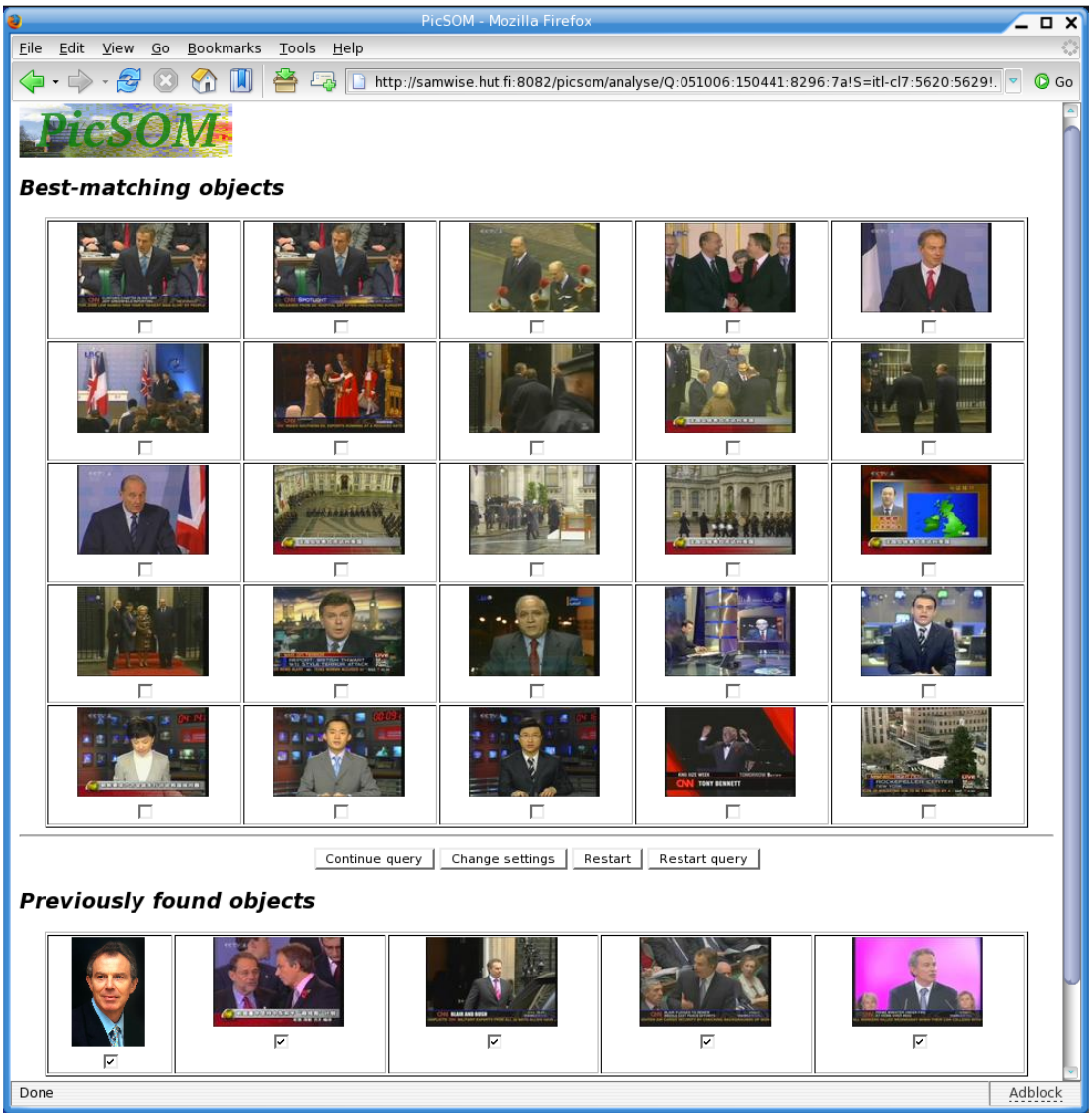

Fig. 5. A screen capture of interactive retrieval with PicSOM.

VI. Conclusions

As a first time participant in TRECVID, our research group faced a lot of system development and other non-recurring work in order to be able to run the TRECVID 2005 experiments. Therefore, we had limited time to study the effects of different setups and parameter values on the overall performance. Furthermore, as low-level video processing is not within our area of expertise, we decided to participate only in high-level feature extraction and search tasks.

For indexing video shots, we have adapted a recent extension of the PicSOM system to support general hierarchical multimodal objects. The video clip, audio track, associated keyframes and text data are all indexed separately and the relevance assessments are propagated intrinsically. The common annotation of the development set is utilized by building SOM class models for the available semantic concepts. The results of the experiments indicate that these class models can be successfully used for representing semantic concepts together with textual features. In the search experiments, we showed that the PicSOM system is able to merge different cues of the semantic content of video shots without an explicit fusion stage. This can be seen from the automatic and manual runs, as in both cases the best results are obtained when combining the text query with both multimodal features and class models.

ACKNOWLEDGEMENTS

This work was supported by the Academy of Finland in the projects *Neural methods in information retrieval based on automatic content analysis and relevance feedback* and *New information processing principles*, the latter being part of the Finnish Centre of Excellence Programme.

REFERENCES

[1] Mats Sjöberg and Jorma Laaksonen. Content-based retrieval of web pages and other hierarchical objects with Self-Organizing Maps. In *Proceedings of 15th International Conference on Artificial Neural Networks (ICANN 2005)*, pages 841–846, Warsaw, Poland, September 2005.

[2] Markus Koskela and Jorma Laaksonen. Semantic annotation of image groups with Self-Organizing Maps. In *Proceedings of 4th International Conference on Image and Video Retrieval (CIVR 2005)*, pages 518–527, Singapore, July 2005.

[3] 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.

[4] Teuvo Kohonen. *Self-Organizing Maps*, volume 30 of *Springer Series in Information Sciences*. Springer-Verlag, third edition, 2001.

[5] ISO/IEC. Information technology - Multimedia content description interface - Part 3: Visual, 2002. 15938-3:2002(E).

[6] Markus Koskela, Jorma Laaksonen, and Erkki Oja. Use of image subset features in image retrieval with self-organizing maps. In *Proceedings of 3rd International Conference on Image and Video Retrieval (CIVR 2004)*, pages 508–516, Dublin, Ireland, July 2004.

[7] Martin Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.

[8] Gerard Salton, editor. *The SMART retrieval system: Experiments in automatic document processing*. Prentice-Hall, 1971.