PicSOM Experiments in TRECVID 2010

Mats Sjöberg, Markus Koskela, Milen Chechev, Jorma Laaksonen
Adaptive Informatics Research Centre, Department of Information and Computer Science
Aalto University School of Science and Technology, Finland

Abstract

Our experiments in TRECVID 2010 include participation in the semantic indexing and known-item search tasks.

In the semantic indexing task we implemented SVM-based classifiers on five different low-level visual features extracted from the keyframes. In addition to the main keyframes provided by NIST, we also extracted and analysed additional frames from longer shots. The feature-wise classifiers were fused using standard and weighted geometric mean. We submitted the following four runs:

- `PicSOM_geom`: Geometric mean of five features, all keyframes.
- `PicSOM_wgeom`: Weighted geometric mean of five features, all keyframes.
- `PicSOM_2geom-mkf`: Geometric mean of two "best" features, main keyframe only.
- `PicSOM_2geom-max`: Geometric mean of two "best" features, all keyframes.

The runs `2geom-max` and `wgeom` obtained the highest MIAP scores (with essentially the same score, 0.0697 vs. 0.0694). Overall, using more keyframes always improved the results substantially. Our weighting approach improved the result over the standard geometric mean. However, by using only two features in fusion without weighting we achieved a similar result.

In the known-item search task we submitted two automatic and two interactive runs:

- `PicSOM_1`: Text search + concept detectors with distribution
- `PicSOM_2`: Text search + concept detectors with rank
- `PicSOM_3`: Interactive with detail view ("normal")
- `PicSOM_4`: Interactive without detail view ("fast")

Our automatic runs used text search with a single video-level index containing all the ASR text plus the title, description and subjects from the meta data. In addition we used automatic selection of concepts based on matching keywords in the query text. We tried two approaches for combining the concept detector outcomes with the text search results. They both recieved very similar scores (0.264 vs. 0.260 in mean reciprocal rank).

Our interactive runs were with a very simple setup: the results of the `PicSOM_1` automatic run were presented in order in a set of screens through which the user should browse to find the correct result. When a promising video was found, the user could examine a detailed view from which he or she could access the oracle service. We also tried a faster variant of the system where the user could make quicker decisions and use the oracle directly from the overview screen. The fast version received lower user satisfaction score (5.0 vs 6.0) but higher performance (0.455 vs 0.318 in mean reciprocal rank).

I. Introduction

In this notebook paper, we describe our experiments for the TRECVID 2010 evaluation. We participated in the semantic indexing and known-item search tasks. The rest of this paper is organized as follows. Our experiments for the semantic indexing task are described in Section II, and our automatic and interactive submissions to the known-item search task are described in in Sections III and IV, respectively.

II. Semantic indexing

This year we utilised a subset of our last year's high-level feature extraction (HLF) system architecture [1] for the semantic indexing task. The system architecture is based on fusing a large number of supervised detectors trained for each concept, based on different shot-level image and video features. The fused detection scores can then be re-adjusted based on the detector outcomes for temporally neighbouring video shots.

In this year's experiments, we studied the use of multiple keyframes extracted from each shot and using weighted and non-weighted geometric mean in detector fusion. Due to the large number of concepts this year, some functionality of the last year's system was omitted from these experiments, including the use of supervised fusion methods and temporal post-processing of the fusion outcomes. We also extracted significantly fewer low-level features than in last year's experiments.

An identical procedure was used for detecting all the concepts. As the concept-wise ground-truth for the supervised detectors we used the annotations gathered by the organised collaborative annotation effort [2]. All our submitted runs were of type A.

A. Low-level features

One main keyframe was provided for each video shot in the master shot reference. In addition we also extracted additional frames from video shots longer than 2 seconds. For shots whose duration was 2–10 seconds we sampled one frame per second, and for longer shots we sampled a total of 10 frames with equal time intervals. The open-source tool FFmpeg[1] was used to extract the video frames.

We extracted a set of five image features from all video keyframes. First, we extracted ColorSIFT features [3] using

[1] *http://www.ffmpeg.org/*

the opponent color space and spatial pyramids with two different sampling strategies: the Harris-Laplace salient point detector (*ColorSIFT*) and dense sampling (*ColorSIFTds*). We also extracted the standard SIFT features using dense sampling (*SIFTds*). In addition, we extracted two global image features: *Edge Histogram* and *Color Moments*, as they had the best performance in our TRECVID 2009 experiments [1].

The codebooks for the SIFT-based features were generated by first taking a random sample of 100 keyframes and calculating the features for all of their sampled points. The resulting vectors were partitioned into 1000 clusters using k-means. The cluster centroids were then selected as the codebook vectors.

B. SVM-based classifiers

In our system, a number of feature and concept-specific SVM detectors are trained based on the extracted features. We used an adaptation of the C-SVC implementation of LIBSVM [4]. The SVM parameters were selected with an approximate 10-fold cross-validation search procedure that consisted of a heuristic line search to identify a promising parameter region, followed by a grid search in that region. The RBF kernel was used for *Color Moments* and the χ^2 kernel for the other four visual features.

C. Fusion of classifiers

Due to the low number of features and feature-wise detectors, we used geometric mean and weighted geometric mean to fuse the detector outcomes. In addition, as the two best features—*ColorSIFTds* and *SIFTds*—showed consistently better performance than the other features, we submitted also runs where only these two features are included in the fusion with geometric mean.

We also utilised weighted geometric mean

$$\overline{x} = \left(\prod_{i=1}^{N} x_i^{w_i} \right)^{1/\sum_{i=1}^{N} w_i} \tag{1}$$

as a fusion method. Here, $N = 5$ is the number of features. The feature-wise weights w_i were obtained with the *Similarity Cluster* weighting proposed in [5].

D. The submitted and additional runs

This section details our submitted and some additional semantic indexing runs. Table I shows an overview. The columns refer to the used fusion method (geometric mean or weighted geometric mean), the number of features used in the fusion, and the number of keyframes used from each shot. The two rightmost columns list the corresponding mean inferred average precision (MIAP) [6] values; first for the original 30 concepts evaluated during TRECVID 2010, and second including the 20 additional concepts evaluated afterwards. In addition, Figure 1 shows the IAP results of our submitted runs for each of the originally evaluated concepts.

The run `geom` uses geometric mean to fuse the detector outputs of all available five features. The shot-wise probability estimates are obtained from all extracted keyframes as the maximum over the keyframe-wise probabilities. After the evaluation we tried other ways of combining the keyframe-wise scores, such as mean, median, taking the second highest values, smoothing over the three highest values, but we could not improve upon using the maximum value.

In the run `wgeom`, weighted geometric mean is used in the fusion of the five features. Otherwise the run is similar to `geom`.

The run `2geom-mkf` uses geometric mean to fuse the detector outputs of only two features, *ColorSIFTds* and *SIFTds*. This selection was based on the relatively good performance of these features observed with the validation set. In order to study the effect of having multiple keyframes per shot, this run used only the main keyframe of each shot.

The run `2geom-max` is otherwise similar to `2geom-mkf` but uses all extracted keyframes.

In addition, Table I shows some additional non-submitted runs. First, we experimented with using only the main keyframe of each shot with the settings of the runs `geom` and `wgeom`. Second, we show the results of the two best-performing single features, *ColorSIFTds* and *SIFTds*, for comparison with the run `2geom-max`.

E. Conclusions from the semantic indexing task

In our set of experiments in the semantic indexing task there are two apparent results. First, the results in Table I suggest that using multiple shot-wise keyframes generally improves the concept detection accuracy. This is the case in all three experiment setups, and with a clear margin in MIAP scores.

Second, the fusion of a larger set of features with an unsupervised method, such as the (non-weighted) geometric mean used in these experiments, does not improve the accuracy over using only the best-performing features. In last year's HLF extraction experiments [1], we tested supervised fusion with methods such as sequential forward/backward search (SFBS) and multi-fold SFBS, which were in those experiments superior to geometric mean. This year we had a much smaller pool of potential features so we did not use SFBS, but instead applied a weighting scheme to the geometric mean fusion. Using weighted geometric mean did improve the results, but a similar performance level could be reached also by geometric mean of the two best features according to the performance in cross-validation and discarding the remaining three features. For the two best features, fusion was, however, clearly useful: The MIAP values of the individual features (two last rows of Table I), are clearly inferior to results of `2geom-max`.

Overall, from both these and last year's experiments, we can conclude that the concept detection performance can be improved with feature fusion, but the use of less accurate detectors in the fusion requires careful consideration as blindly fusing all available features will likely result in performance degradation.

III. Automatic known-item search

In the known-item search task we submitted two automatic runs and two interactive runs. This section describes our

TABLE I
AN OVERVIEW OF THE SUBMITTED AND SOME ADDITIONAL RUNS IN THE SEMANTIC INDEXING TASK. SEE TEXT FOR DETAILS.

	run id (PicSOM_+) /	fusion		number of		MIAP	
#	additional run info	gm	wgm	features	keyframes	30 concepts	30 + 20
1	geom	●		5	all	0.0625	0.0848
2	wgeom		●	5	all	0.0694	0.0886
3	2geom-mkf	●		2	main	0.0550	0.0780
4	2geom-max	●		2	all	0.0697	0.0884
-	geom-mkf	●		5	main	0.0488	0.0738
-	wgeom-mkf		●	5	main	0.0550	0.0784
-	only ColorSIFTds			1	all	0.0541	0.0664
-	only SIFTds			1	all	0.0573	0.0763

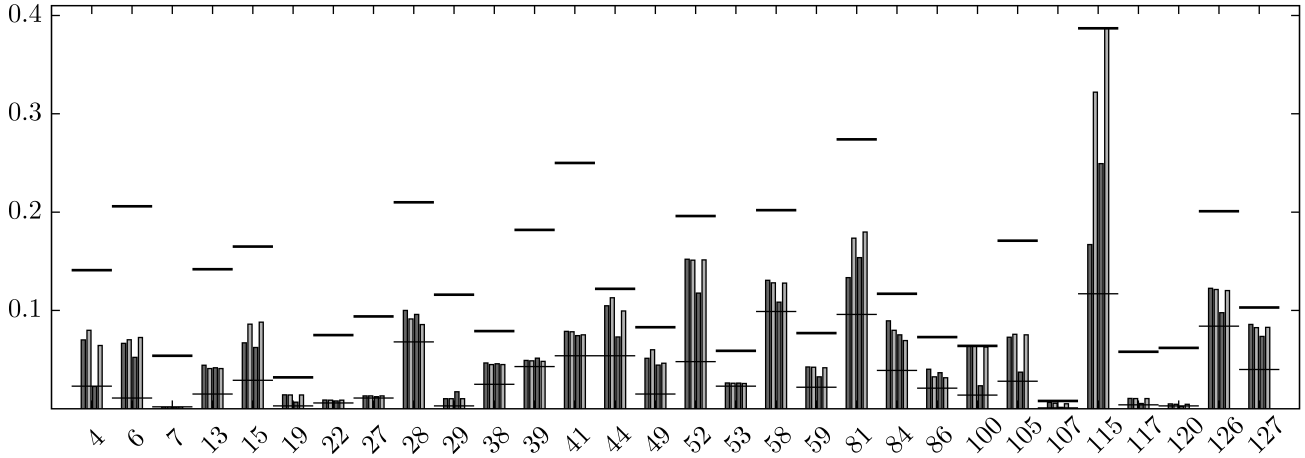


Fig. 1. The concept-wise IAP results of our submitted runs for each evaluated concept. The order of the runs is as in Table I. (i.e. the leftmost bar corresponds to PicSOM_geom, etc.) The median and maximum values over all submissions are illustrated as horizontal lines.

approach for the automatic runs which are used as the base for the interactive runs described in Section IV.

Our automatic search approach is based on combining simple text search with automatically matched semantic concepts using the concept detectors from the semantic indexing task.

A. Text search

For text search we used the Lucene[2] search engine based on the meta data and automatic speech recognition texts of the videos. We tried different ways of creating the search index—with/without Porter stemmer, with/without stop words and with/without WordNet synonyms. All these approaches were compared using the sample queries, and the index with stemmer, without stop words and without synonyms was chosen as it was the one with the best performance.

We also compared using a single search index for all textual data and using two separate indexes: one for meta data and one for the automatic speech recognition output. The approach with the single index had better performance, and was thus chosen for our runs with the test queries.

B. Semantic concept matching

For each semantic concept, a word list was generated by taking the concept name itself as the initial word or words and expanding with WordNet synonyms. These lists were then cleaned up by hand (without knowledge of the particular search topics). For example, words with too broad meaning were removed, e.g. the concept *people marching* was set to be activated for the word "march" but not for "people" appearing in the textual query.

We used a regular expression syntax, where we could match not only a list of alternative words, but could require more complex expressions. For example, the concept *two people* needs the word "two" later followed by "people" or "person". Then, for a particular query a set of matching concepts is found by activating those with matching regular expression.

C. Use of concepts

In experiments using the sample topics we quickly realised that using the concept detectors alone was not going to give very good results. Text search, however, was working much better, and the main challenge turned out to be how to gain any improvement on the text search results by adding information from the concept detectors.

We tried two approaches of combining the text search results with the concept detector scores. The first approach simply takes a weighted mean of the detection scores, the second approach takes the text search results as a baseline, and uses the concept detector outcomes to re-rank them. The two approaches are described in more detail below.

[2] *http://lucene.apache.org/*

1) Weighting of detector outcomes: In our first approach we calculated a weighted geometric mean of the relevant concept detector scores for each query, as in Eq. (1). Here, N is the number of concepts, which were selected as described in Section III-B. We used the Media Mill concept detectors, since those obtained the best results in the semantic indexing task, and also in our experiments with the sample topics. Since we did not have the Media Mill detector scores for the shots but only the rankings, we took the score distributions from our own detectors.

The weights w_i in the weighted geometric mean were selected in an approach analogous to the inverse document frequency (idf) method used in information retrieval:

$$w_i^{\text{idf}} = \log \frac{N_t}{N_i}, \tag{2}$$

where N_i is the number of shots where the concept occurs in the training set, and N_t is the total number of shots.

After the submissions we noticed that some videos contain quite many concepts, e.g. videos that display a series of diverse photographs in rapid sequence will match a large proportion of concepts. Such videos will often turn up in the top search results for many search queries. To alleviate this we included, in an additional run, a factor analogous to the term frequency (tf), which is also common in information retrieval:

$$w_{i,j}^{\text{tf}} = \frac{r_{i,j}}{\sum_k r_{k,j}}, \tag{3}$$

where $r_{i,j}$ is the detector score for the concept indexed i for the video j. In the denominator is the sum over all concept scores for that video. In the classical term frequency definition we have the number of occurrences of words in a document, these are here replaced by the detector outcomes. The total tf-idf weight would then be, for a concept i and video j:

$$w_{i,j} = w_{i,j}^{\text{tf}} w_i^{\text{idf}}. \tag{4}$$

Finally, the weighted geometric means of the concept scores were combined with the Lucene scores by weighted arithmetic mean. The weights were determined by testing using the sample queries: we took 85% of the Lucene score and 15% of the concept detector scores.

2) Re-ranking based on concept detectors: Our second approach was based on using the outcomes of the six concept detection systems with the highest performance in the semantic indexing task to re-rank the text search results.

Each of the concept detection systems give as output a sorted list of shots for each concept. The `trec_eval` rank of the shots starts from 9999 and then continuously decreases by one. We transform these lists of shots to a single list of videos by a heuristic approach which goes through the system specific lists, in the order of best performance, and takes the maximum `trec_eval` rank over its shots in the first list where the video is encountered. The total number of shots from that particular video in all lists is added to that video specific score.

Then, for each query we select the concepts as described in Section III-C and merge the lists of the videos for those concepts. The final score of each video is the sum of the scores for the video in all concept-specific lists. The merged list is gained by sorting the videos in descending order of summed scores. Finally, we merge this list with the text search outcome by calculating a final video score r_j that is in the range $[0, 1]$:

$$r_j = r_{j,\text{text}} + w\frac{1}{s_j^2}, \tag{5}$$

where $r_{j,\text{text}}$ is the text search outcome, s_j is the rank of the video in the video list merged over concepts (described above), and $w = 0.1$ is a weight that was determined experimentally using the sample queries.

D. Results and conclusions from the automatic search task

Our submitted and additional runs in the known-item search task are summarised in Table II together with their mean reciprocal (or inverse) rank scores (MIR). The run `PicSOM_1` uses the method based on weighting the distributions of detector scores (Section III-C1), and `PicSOM_2` the method based on re-ranking based on many different detectors (Section III-C2).

For completeness, the table also shows the interactive search results, which will be explained in the next section. Two additional runs were added for comparative purposes, one using purely text search and one using only concepts (otherwise with the same methodology as in `PicSOM_1`). Furthermore a third additional run is similar to `PicSOM_1` except that it uses tf-idf weighting (4) instead of just idf.

The results in Table II confirm our earlier observation with the sample queries that at least with our current approach one cannot achieve good performance with concepts only. In our experiments with the sample queries we managed to make small improvements upon the baseline text search, however, with the test queries no such improvement was made.

To investigate the possibility that our automatic selection of concepts was making bad choices, we also performed manual concept selection for all sample queries. These were made by a researcher familiar with our system, by both looking at the query texts and visually inspecting the known-item videos. This did not however improve the results, in fact the mean reciprocal rank performance decreased from the automatic selection.

This result is in sharp contrast to our experience in earlier TRECVID competitions. For example, in our runs in TRECVID 2009 [1], with a very similar system, the most important contributor to getting a high retrieval performance in the video search task was the concept-based search. Text search typically gave a small positive overall improvement, but was not a very important factor. This difference might be explained by several factors. First, while it might not be initially apparent, the known-item search is fundamentally different from the video search tasks of earlier years. Previously we were looking for a large set of videos that satisfied the query, now we are looking for a single specific video that is described in the query—a much more difficult task.

TABLE II
An overview of the submitted and additional runs in the known-item search task. See text for details.

			concepts			
#	run id	text search	distr	rank	interactive	MIR
1	`PicSOM_1`	•	•		–	0.260
2	`PicSOM_2`	•		•	–	0.264
3	`PicSOM_3`	•	•		normal	0.318
4	`PicSOM_4`	•	•		fast	0.455
-	text only	•			–	0.266
-	concepts only		•		–	0.003
-	text + conc. (“tf-idf”)	•	•		–	0.265

Second, search topics were previously often written with the intention that some of the available concepts might match them. Now the topics are simply describing a single random video, without any specific attention given to what concepts might be available. Finally, this year the videos contained manually entered meta-data describing the contents of the videos. In our experiments with the sample queries, adding the meta-data to the ASR text improved the performance significantly. On the other hand, even without the meta-data the text search performance was an order of magnitude better than that of the concept-based search.

IV. Interactive known-item search

We also participated in the interactive part of the known-item search task. Our interactive setup was very simple. First an automatic run was performed using the methodology of `PicSOM_1`, i.e. using text search combined linearly with the concept-based search results. The results from the automatic run were presented to the user in order of decreasing scores, in a set of screens of video thumbnail images. The user was able to choose freely the number of rows and columns of thumbnail images to be shown on each screen. Each video thumbnail was created as a 4×4 collage of keyframes sampled evenly over the video. If there were more than 16 shots and thus more than that number of main keyframes in the video, then a random selection of them were ignored when the video thumbnail was created. The keyframes of the first and last shots were, however, always retained. If there were fewer than 16 shots, then also additional keyframes were randomly inserted in the thumbnail.

A. Interactive interface

We had two alternative web-based interfaces: a *normal* and a *fast* version, both depicted in Figure 2. In the normal interface the user first searches through screens in a *search view* (shown left in Figure 2), and then when he or she finds a potential video a *detail view* can be brought up by clicking on that video’s collage thumbnail (shown in the middle of the figure). The detailed view allows the searcher to view or skim the actual video, see a larger version of the collage of keyframes, and also inspect the title, description and subjects of the meta data, if these are present. If the user is reasonably sure that this is the correct video there is a “Check video” button which consults the oracle service provided by DCU. If the video was correct the search ends there, otherwise the searcher can continue by going back to the search view and by proceeding to inspect other videos.

In the fast interface there is only the search view, i.e. no detailed view. The oracle query is done by clicking on the corresponding thumbnail collage of the video. This means that the searcher must, in general, make quicker and less accurate decisions—but may also be able to go through a larger number of results. The fast interface was thus expected to produce more oracle queries than the normal interface. The fast interface was given a red background so that the searchers can easily distinguish which system they were using. This system is depicted on the right in Figure 2.

In both systems the searchers were able to quickly move between screens by keyboard shortcuts, e.g. by pressing space-bar to move to the next screen. At the top of each page the remaining time was displayed to the searcher. When the time reached the allotted 5 minutes, the search would end unsuccessfully.

B. Experimental setup

The interactive runs were performed by six persons, all using both the normal and fast systems. The search topics were split up so that each person did four topics with the normal system and four different ones with the fast system. All topics were covered with both systems, and the order of systems was interchanged so that three searchers started with the normal interface and three with the fast interface.

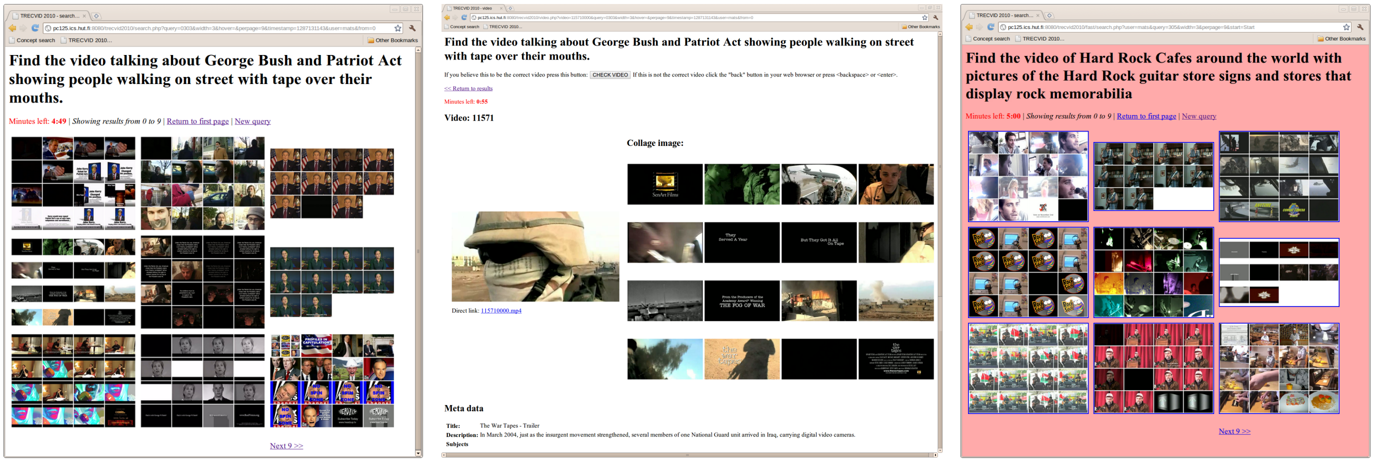
All searchers performed a few “dry runs” using the sample queries to familiarise themselves with the web-based system and adjust parameters to their particular monitor size and personal preferences. At this point it was observed that some searchers preferred to maximise the number of video thumbnails seen per screen, in order to see more at once – while others wanted to reduce the cognitive load by showing only a few per screen, but instead being able to view screens much faster.

After the real interactive runs, each searcher returned their user satisfaction scores in folded notes to ensure anonymity. The final user satisfaction scores were calculated for each system (normal and fast) as the median over the returned numbers.

C. Conclusions from the interactive search task

The MIR scores for the interactive search runs were presented in Table II together with the automatic results. The results are repeated in Table III together with some other statistics particular to the interactive runs. It can be seen that the fast system performed better than the normal system: in the fast system 11 of the 24 queries were successful, while in the normal system only 7 were found. (Percentages are shown in the table.) Interestingly the fast system received a lower user satisfaction score: 5.0, while the normal system got 6.0—even though the performance was better.

When we look at only the successful runs, i.e. where the correct result was found within the allotted 5 minutes, the


TRECVID 2010 - search...
pc125.ics.hut.fi:8080/trecvid2010/search.php?query=0303&width=3&hover=&perpage=9&timestamp=1287131143&user=mats&from=0
Concept search
TRECVID 2010...
Other Bookmarks

Find the video talking about George Bush and Patriot Act showing people walking on street with tape over their mouths.

Minutes left: **4:49** | *Showing results from 0 to 9* | Return to first page | New query

Next 9 >>

TRECVID 2010 - video
pc125.ics.hut.fi:8080/trecvid2010/video.php?video=115710000&query=0303&width=3&hover=&perpage=9&timestamp=1287131143&user=mats&from=0
Concept search
TRECVID 2010...
Other Bookmarks

Find the video talking about George Bush and Patriot Act showing people walking on street with tape over their mouths.

If you believe this to be the correct video press this button: CHECK VIDEO If this is not the correct video click the "back" button in your web browser or press <backspace> or <enter>.

<< Return to results

Minutes left: **0:55**

Video: 11571

Collage image:

SenArt Films
They Served A Year
But They Got It All On Tape
From the Producers of the Academy Award® Winning THE FOG OF WAR
the war tapes

Direct link: 115710000.mp4

Meta data

Title: The War Tapes - Trailer
Description: In March 2004, just as the insurgent movement strengthened, several members of one National Guard unit arrived in Iraq, carrying digital video cameras.
Subjects

TRECVID 2010 - search...
pc125.ics.hut.fi:8080/trecvid2010/fast/search.php?user=mats&query=305&width=3&perpage=9&start=Start
Concept search
TRECVID 2010...
Other Bookmarks

Find the video of Hard Rock Cafes around the world with pictures of the Hard Rock guitar store signs and stores that display rock memorabilia

Minutes left: **5:00** | *Showing results from 0 to 9* | Return to first page | New query

Next 9 >>

Fig. 2. Screen shots of the interactive system: the normal system in its search view (left) and detail view (middle), and the fast system (right).

median time to find the correct video (column 4 in Table III) is quite similar between the two systems, but the average time (column 5) is much longer for the fast system. This is due to the fact that while in the normal system each correct answer was found within the first minute or not at all, in the fast system the correct answer was found later than that on three occasions. In fact, on two occasions even after four minutes.

Our hypothesis that the fast system would enable the searcher to go through a larger number of videos in the allotted time does not seem to be supported. In the failed queries, i.e. were the searcher kept browsing through screens of videos until 5 minutes had passed, the average number of videos seen was the same for both systems – slightly above 500 videos. One might speculate that the browse-through rate is more dependent on the particular query than which of our two systems is in use. Some queries might be easier to quickly visually distinguish than others, e.g. a scene by the beach might have big patches of blue water that can be very quickly detected while scanning through a large set of thumbnail images.

The normal system produced 1.2 oracle checks per query on average, while the fast system produced an average of 5.4 checks. In the fast system the number of checks were however highly individual, a group of users checked on average 10 times per query, while another only around 2.5 times. The frequency of oracle checks might explain the higher performance of the fast system. The lower threshold for consulting the oracle might have found some videos that initially seemed not to match the query. In fact, anecdotal evidence based on discussions with the searchers afterwards indicates this to be true at least for one topic, where the user of the normal system skipped the correct video since not all elements of the query seemed to be present, while the user of the fast system found at least one element and decided to try the oracle.

TABLE III
INTERACTIVE RUNS.

system	MIR	found	median time (s)	average time (s)	avg. oracle checks	user satisf.
normal	0.318	29%	32.4	34.6	1.2	6.0
fast	0.455	46%	40.2	89.2	5.4	5.0

The known-item search task simulates a quite realistic scenario, where a person knows of a video in a large collection and wants to retrieve it. However, even though the oracle service was provided in the interactive task, the scenario in that case is seriously lacking. If a person has previously seen a video, he or she doesn't typically merely know its semantic content, but also might have a visual image of the video in his or her head. This means that if the correct video is shown among others as thumbnails in a grid, as in our interactive system, he or she will most likely immediately recognise it visually. This was not the case in the task setup, since the user only had a textual description and could consult the oracle service if he or she was reasonably confident.

ACKNOWLEDGMENTS

This work was supported by the funding of Academy of Finland for the *Finnish Centre of Excellence in Adaptive Informatics Research*.

REFERENCES

[1] Mats Sjöberg, Ville Viitaniemi, Markus Koskela, and Jorma Laaksonen. PicSOM experiments in TRECVID 2009. In *Proceedings of the TRECVID 2009 Workshop*, Gaithersburg, MD, USA, November 2009.

[2] 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.

[3] 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.

[4] Chih-Chung Chang and Chih-Jen Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at http://www.csie.ntu.edu.tw/˜cjlin/libsvm.

[5] P. Wilkins, T. Adamek, G.J.F. Jones, N. E. O'Connor, and Alan F. Smeaton. TRECVid 2007 experiments at Dublin City University. In *Proceedings of the TRECVID 2007 Workshop*, Gaithersburg, MD, USA, November 2007.

[6] 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.