PicSOM Experiments in TRECVID 2011

Mats Sjöberg, Satoru Ishikawa, Markus Koskela, Jorma Laaksonen, Erkki Oja
Department of Information and Computer Science
Aalto University School of Science, Finland

Abstract

Our experiments in TRECVID 2011 include participation in the semantic indexing and known-item search tasks.

In the semantic indexing task we implemented linear and SVM-based classifiers on 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 classifiers were fused using standard and weighted geometric mean. We submitted to the full task the following four runs:

- `PicSOM_1`: Weighted geometric mean of all linear and SVM classifiers.
- `PicSOM_2`: Weighted geometric mean of the SVM classifiers.
- `PicSOM_3`: Geometric mean of the linear classifiers.
- `PicSOM_4`: Geometric mean of all linear and SVM classifiers.

The run `PicSOM_1` obtained the highest MXIAP score of 0.1355.

In the known-item search task we submitted four automatic runs:

- `PicSOM_1`: Text search with OCR and spell check + concept detectors
- `PicSOM_2`: Text search with OCR and spell check
- `PicSOM_3`: Text search with lemmatisation + concept detectors
- `PicSOM_4`: Text search with lemmatisation

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. We also included text detected by OCR and tried lemmatisation in some runs. In addition we used automatic selection of concepts based on matching keywords in the query text. Neither the concept detectors nor the lemmatisation managed to improve over our best run which was `PicSOM_2` with a MIR score of 0.2720.

I. Introduction

In this notebook paper, we describe our experiments for the TRECVID 2011 evaluation. We participated in the semantic indexing and automatic known-item search tasks. Our experiments for the semantic indexing task are described in Section II and our submissions to the automatic known-item search task are described in Section III.

II. Semantic indexing

Similarly as in previous years we utilised our high-level feature extraction (HLF) system architecture [1], [2] for the semantic indexing task. The system is based on fusing a large number of supervised detectors trained for each concept, based on different shot-level image and video features. The detection scores can then optionally be re-adjusted based on the detector outcomes for temporally neighbouring video shots.

In this year's experiments, we studied methods for reducing the computational requirements of high-level semantic feature extraction both by using features that can be extracted rapidly and by experimenting with linear classifiers instead of or in addition to the baseline SVM classifiers. Otherwise our experiments are similar to last year's. In particular, we continue to use multiple keyframes extracted from each shot and apply weighted and non-weighted geometric mean in detector fusion.

An identical procedure was used for detecting each of the concepts. As the concept-wise ground-truth for the supervised detectors we used the annotations gathered by the organised collaborative annotation effort [3]. All our runs were submitted to the full task and are 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 total of 14 image features from all video keyframes. First, we extracted SIFT and ColorSIFT features [4] using the opponent colour space and spatial pyramids with two different sampling strategies: the Harris-Laplace salient point detector (*SIFT* and *ColorSIFT*) and dense sampling (*SIFTds* and *ColorSIFTds*). We employ the bag-of-visual-words (BoV) approach and generate the codebooks 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, and the cluster centroids were used as the codebook vectors.

We used also the CENsus TRansform hISTogram[2] descrip-

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

[2] *http://c2inet.sce.ntu.edu.sg/Jianxin/PACT/PACT.htm*

TABLE I

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

	run id /	fusion		classifiers		MXIAP (submitted)		MXIAP (subsequent)	
#	additional run info	gm	wgm	linear	SVM	all concepts	light task	all concepts	light task
1	PicSOM_1		●	14	3	0.1355	0.0986	0.1398	0.1068
2	PicSOM_2		●	-	3	0.1320	0.1006	0.1363	0.1055
3	PicSOM_3	●		14	-	0.1145	0.0778		
4	PicSOM_4	●		14	3	0.1322	0.0954	0.1352	0.0999
-	three linear classifiers	●		3	-			0.0827	0.0592
-	Centrist+linear 10	●		10	1			0.0884	0.0487
-	Centrist+linear 10, two stage fusion	*	*	10	1			0.1116	0.0635
-	SVM+linear, two stage fusion	*	*	14	3			0.1386	0.1016

tor (*Centrist*) proposed in [5], using the suggested configuration of two levels in the spatial pyramid, reducing the LBP histograms to 40 dimensions with PCA, and including the average value and standard deviation of pixels in each block. The dimensionality of the descriptor thus becomes 1302.

In addition, we extracted nine of the global image features we haved used in our earlier TRECVID experiments. See [6] for details of these features.

B. Linear and kernel SVM classifiers

In our system, a number of feature and concept-specific classifiers are trained based on the extracted features. We trained linear classifiers for all 14 extracted features using the LIBLINEAR[3] [7] library. From earlier experiments we have found that the L^2-regularized logistic regression solver in LIBLINEAR has a good performance with reasonable run time. This solves the following unconstrained optimisation problem for the training vectors $\mathbf{x}_i \in \mathcal{R}^n$ and corresponding labels $y_i \in \{-1, +1\}$:

$$\min_{\mathbf{w}} \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{l} \log(1 + e^{-y_i \mathbf{w}^T \mathbf{x}_i}), \qquad (1)$$

where C is a penalty parameter, and $\mathbf{w}$ the model to be trained (weights of the separating hyperplane of the classifier).

For the SVM classifiers we used an adaptation of the C-SVC implementation of LIBSVM [8]. We trained SVM classifiers for *SIFTds*, *ColorSIFTds*, and *Centrist* features. The RBF kernel was used for *Centrist* as suggested in [5] and the χ^2 kernel for the BoV features.

C. Classifier parameter selection

Eq. 1 contains only a single parameter (C) that needs to be determined for each concept detector, as opposed to the C-SVM which typically needs two parameters (C, and γ for the kernel width). Furthermore, since the linear classifier is much faster we can use many more examples for training and parameter selection. For the linear parameter selection we use an approximate 10-fold cross-validation search procedure that consists of a uniform sampling of the parameter space followed by another more detailed sampling in the best region. The SVM parameters were selected similarly, but starting with a heuristic line search to identify a promising parameter region, followed by a grid search in that region.

In experiments done with the TRECVID 2010 database, the parameter selection (i.e. the most time-consuming part) took on average 45 hours per concept for the SVM classifier. The parameter selection for the linear classifier took on average 40 minutes – a 60-fold improvement!

D. Classifier fusion

Similarly as last year, we used either geometric mean or weighted geometric mean to fuse the classifier outcomes. The latter is calculated as

$$\overline{x} = \left(\prod_{i=1}^{N} x_i^{w_i}\right)^{1/\sum_{i=1}^{N} w_i} , \qquad (2)$$

where N is the number of used features. The feature-wise weights w_i were obtained with the *Similarity Cluster* weighting proposed in [9].

E. 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), and the number of linear and SVM classifiers used in the fusion. The four rightmost columns list the corresponding mean extended inferred average precision (MXIAP) [10] values, both for all 50 concepts evaluated (full task) and for only the 23 concepts in the light task. The first MXIAP columns show our submitted results and the latter columns list results obtained from subsequent experiments. Three of the submitted runs were rerun subsequently due to a small error in the training of the *Centrist* SVM classifiers. After fixing the error, the results of the submitted runs containing this classifier were somewhat improved. In addition, the MXIAP results for some of the individual classifiers are shown in Table II. Furthermore, Figure 1 illustrates the concept-wise XIAP results of our submitted runs. The concept-wise results are shown for the evaluated concepts in the light task.

The run `PicSOM_1` uses weighted geometric mean to fuse the classifier outputs of all available linear and SVM classifiers. The shot-wise probability estimates are obtained from all extracted keyframes as the maximum over the keyframe-wise probabilities. The run `PicSOM_4` is otherwise identical

[3] *http://www.csie.ntu.edu.tw/˜cjlin/liblinear/*

TABLE II
MXIAP SCORES OF SELECTED INDIVIDUAL CLASSIFIERS IN THE SEMANTIC INDEXING TASK.

classifier type	feature	MXIAP	
		all concepts	light task
SVM	ColorSIFTds	0.1233	0.0964
	SIFTds	0.1139	0.0844
	Centrist	0.0939	0.0681
linear	ColorSIFTds	0.0329	0.0227
	ColorSIFT	0.0097	0.0079
	SIFTds	0.0292	0.0216
	SIFT	0.0075	0.0053
	Centrist	0.0289	0.0224
	EdgeFourier	0.0101	0.0048
	ScalableColor	0.0182	0.0094

to PicSOM_1 except geometric mean is used to fuse all the detectors.

In PicSOM_2, only the SVM classifiers are used in the fusion. The run PicSOM_3 uses geometric mean to fuse the outputs of only the linear detectors. The use of geometric mean instead of the weighted variant is based on its observed better performance on earlier experiments.

In addition, Table I shows some additional non-submitted runs. The first additional run shows a fusion of the three best linear classifiers (*ColorSIFTds*, *SIFTds* and *Centrist*). The second run shows a run avoiding the most computationally expensive features, i.e. fusion *Centrist* SVM with linear classifiers of all non-BoV features. We also tried a two-stage fusion approach where the linear and SVM classifiers were fused separately (with geometric respectively weighted geometric mean), and then fused together with geometric mean. The third additional run shows approach for the non-BoV features scenario, and the last additional run uses the full feature set.

The results for individual classifiers listed in Table II show that as solitary detectors the SVM classifiers significantly outperform the linear ones, as was to be expected. Also, the BoV features receive the highest scores, again as anticipated. The results for the *Centrist* feature are somewhat lower than those for the BoV features for both SVM and linear classifiers. Of the other global image features, only two the best-scoring ones, *EdgeFourier* and *ScalableColor*, are included in Table II. Surprisingly the two BoV features that sampled interest-points by Harris-Laplace perform much worse than their dense sampled counter-parts with the linear classifiers. It is still unclear if this is a real effect or some mistake on our part.

F. Conclusions from the semantic indexing task

In this year's experiments we focused on replacing parts of the popular BoV-based features + SVM classifiers - concept detection pipeline with lighter, faster alternatives. This is partially motived by a recent research direction of our group, which concerns large amounts of streaming visual data, e.g. live TV broadcast on several channels, for which concept detection and analysis needs to be performed in real time.

First, we used the *Centrist* feature as a replacement for the much more computationally heavy BoV-based features. Once the concept detectors are trained, applying them to new images is relatively fast, but extracting features is something that needs to be done online for each new image. This motivates the usage of lighter features even though we retain the SVM-based classifiers. Our results show that the *Centrist* feature works quite well with reasonable performance, but doesn't quite achieve the same level as the BoV features. Considering that the *Centrist* feature can calculate in the order of 50 images per second, and SIFT-based ones (in our quick test) roughly 1 image per second, this is quite a good result.

Secondly, we tried to use faster linear classifiers instead of the kernel SVM. These classifiers are on average in the order of 60 times faster to train as reported in Section II-C. Unsurprisingly the single linear classifiers had much poorer performance than the non-linear ones, however fusing several classifiers with varying features we can get a reasonable performance. For example the best feature alone gives a performance of 0.0329, but combining the three best ones soars to 0.0827. By combining all 14 features using linear classifiers we can achieve a MXIAP of 0.1145.

Since the linear classifiers are much quicker to train we can afford to train many more of them; here we use at most about five times more features than with the SVM, retaining a more than 10-fold speed increase compared to the SVM baseline. Also the long training times of the SVM classifiers makes them very cumbersome in practice and e.g. mistakes or computer hardware failures take a long time to correct.

III. AUTOMATIC KNOWN-ITEM SEARCH

In the known-item search task we submitted four automatic runs. 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. This year we tried to improve the search by augmenting the metadata and automatic speech recognition text with the output of optical character recognition. Furthermore, we used a spell checking system to extend both queries and documents.

A. OCR

We used the *Tesseract*[4] OCR engine to perform optical character recognition on all keyframes (main keyframe + additional frames) in the test set. The results are very noisy, and often it does not succeed to recognise even relatively clear text, possibly due to the low quality and resolution of the keyframes. Still, in our tests with the TRECVID 2010 dataset, including the OCR gave a small improvement in retrieval performance overall.

B. Text search

For text search we used the *Lucene*[5] search engine based on the metadata, the automatic speech recognition results, and the OCR output from the videos. We used a single search index for all three modalities.

We processed both the video-wise textual documents and the query texts with spelling correction and either stemming

[4] *http://code.google.com/p/tesseract-ocr/*

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

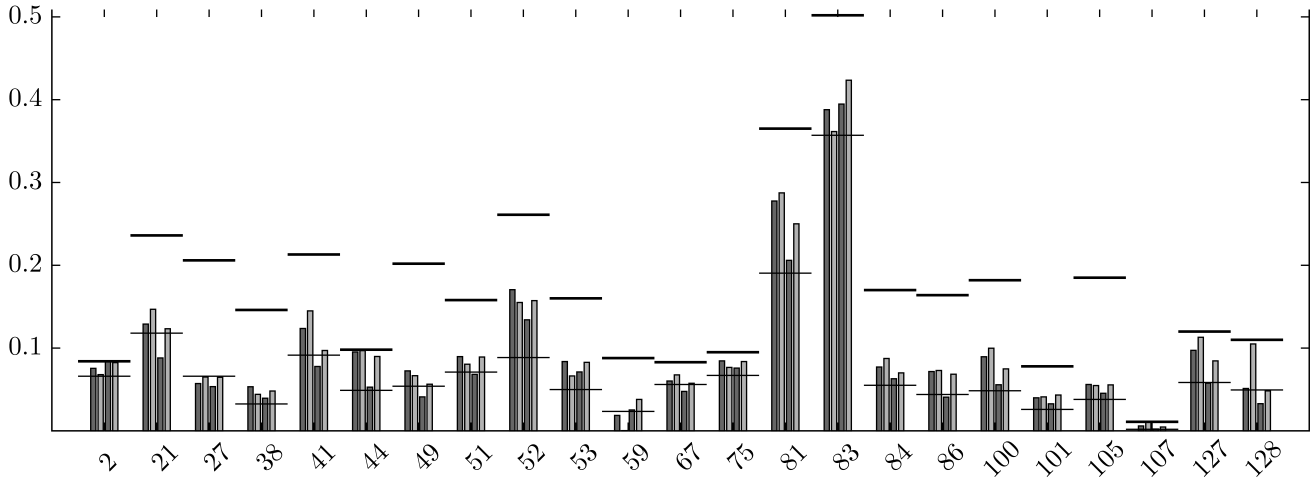

Fig. 1. The concept-wise XIAP results of our submitted runs for each evaluated concept in the light task. The order of the runs is as in Table I (i.e. the leftmost bar corresponds to `PicSOM_1`, etc.). The median and maximum values over all type-A submissions are illustrated as horizontal lines.

or lemmatisation. The spelling correction was performed using *GNU Aspell*[6] and adding the first suggestion made by Aspell for misspelled words to the queries and documents.

For stemming, we used the standard stemmer in the *EnglishAnalyzer* analyser in Lucene.

We also experimented with lemmatisation of the queries and metadata of the videos using the *MorphAdorner*[7] tool. Both the lemmatised and the original word forms were included in the resulting documents. In addition, during lemmatisation we doubled the weight of nouns in the query texts.

C. 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 it 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 a matching regular expression.

D. Pruning of concepts

From our experience in last years TRECVID search task, we saw that adding concepts would often even decrease the performance of the text-only search system. Taking into account this result, we decided to set a threshold in order to remove concepts with low detection scores. Since we did not have real evaluation scores for all concepts we took the cross-validation AP scores calculated during the cross-validation performed when selecting parameters for the SVM detectors. We got the threshold value, 0.25, by evaluating with the TRECVID 2010 queries and concepts.

E. Fusion of text and concepts

As concept detectors we took the mean of the detector outputs of the three best groups in the semantic indexing task: TokyoTech+Canon, University of Amsterdam (MediaMill) and Quaero. Since we only had access to the ordering of video shots, not the actual cores we took the score value distributions from our own detectors for the same concepts.

The selected concepts for a given query were combined with weighted geometric mean, where the weights were selected in an approach analogous to the inverse document frequency (idf) method used in information retrieval:

$$w_i^{\text{idf}} = \log \frac{N_t}{N_i}, \tag{3}$$

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. 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 90% of the Lucene score and 10% of the concept detector scores.

F. Black-and-white detection

We noticed in TRECVID 2010 that a few queries specified the target video to be in black-and-white. We created a simple black-and-white detector using a geometric mean of SVMs trained on three colour features with labels from the metadata. Then a regular expression was crafted (using the 2010 queries) to match searches which specified a "black and white" video as the query target. This matched 8 queries in 2011. For these queries we used the black-and-white detector to filter out results which were least likely to be black-and-white according to a threshold optimised with the 2010 queries.

[6] *http://aspell.net/*

[7] *http://morphadorner.northwestern.edu/morphadorner/*

TABLE III
AN OVERVIEW OF THE SUBMITTED AND ADDITIONAL RUNS IN THE KNOWN-ITEM SEARCH TASK. SEE TEXT FOR DETAILS.

#	run id	OCR	aspell	lemmatisation	concept threshold	MIR	MIR*
1	PicSOM_1	●	●		0.25	0.2682	0.2682
2	PicSOM_2	●	●		-	0.2720	0.2719
3	PicSOM_3			●	0.25	0.2200	0.2201
4	PicSOM_4			●	-	0.2231	0.2227
-	No concepts, no black & white detection	●	●		-		0.2723
-	All concepts, no black & white detection	●	●		0.00		0.2739

G. Results and conclusions from the automatic search task

Our submitted and additional runs in the known-item search task are summarised in Table III together with their mean reciprocal (or inverse) rank scores (MIR). Unfortunately we were unable to run the official evaluation script provided by TRECVID, and our own script was not able to produce exactly the same MIR scores. For this reason we have included as the last column in Table III the MIR score as calculated by us (MIR*), which differs from the official score only in the fourth decimal place.

All runs use a basic Lucene search on the metadata title, description and subjects plus the ASR data. In addition runs PicSOM_1 and PicSOM_2 use OCR and both the documents and the queries are augmented with spell correction suggestions from GNU Aspell. PicSOM_1 differs by using the visual concept detectors in addition. As can be seen, this unfortunately reduces the performance as compared to the pure text-based search.

The runs PicSOM_3 and PicSOM_4 instead augment the basic Lucene search by lemmatisation with MorphAdorner of both the documents and queries. The PicSOM_3 run also uses concepts in addition, with a similar slight decrease in performance. It is also clear that our attempt at lemmatisation was not successful compared to the simple stemming and spell checking approach of the two first runs.

We also performed some additional runs after the main TRECVID evaluation, which are shown as the last rows in Table III. The first, which uses no concepts at all, is identical to run PicSOM_2 except that it does not have the black and white detection. The improvement in performance clearly shows that black and white filtering was not a fruitful approach.

In order to find the optimal threshold value for filtering out concepts, we also performed some additional runs with different threshold settings. Contrary to our expectations, using all selected concepts (i.e. a threshold set to 0) showed the best result and it is even better than the "no concepts" setting which was the best result in our submitted runs. This run is shown on the last row of Table III.

As a point of comparison we made an "oracle" selection of concepts by selecting those for which the correct video was given a sufficiently high score. With the appropriate threshold we were able to achieve a MIR score of 0.3133. For example for query 501, "Find video featuring Newsreel clips of Nautilus Nuclear submarine entering NY harbor, a Hunter Killer helicopter and the first Pan Am commerical passenger jet" the oracle selection picked the concepts *3 Or More People, Airplane, Apartments, Boat/Ship, City, Cityscape, Crowd, Oceans, Vehicle*. With the exception of *Apartments*, all concepts are such that a human might have selected them. E.g. even though a crowd is not mentioned in the query, it is not far-fetched to assume that there will be crowds watching by the harbor or crowds of sailors. This is of course a very unrealistic situation, but it gives a hint what could be done with the best possible concept selection.

This year, as in 2010, we were not able to improve significantly upon the basic text search. It seems that known item search queries are so specific that it is very hard to select the correct general visual concepts and most improvements come from improving the text search itself.

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] Mats Sjöberg, Markus Koskela, Milen Chechev, and Jorma Laaksonen. PicSOM experiments in TRECVID 2010. In *Proceedings of the TRECVID 2010 Workshop*, Gaithersburg, MD, USA, November 2010.

[3] 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.

[4] 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.

[5] Jianxin Wu and James M. Rehg. CENTRIST: A visual descriptor for scene categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33:1489–1501, 2011.

[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.

[7] R.E. Fan, K.W. Chang, C.J. Hsieh, X.R. Wang, and C.J. Lin. Liblinear: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874, 2008.

[8] 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.

[9] 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.

[10] Emine Yilmaz, Evangelos Kanoulas, and Javed A. Aslam. A simple and efficient sampling method for estimating AP and NDCG. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR '08)*, pages 603–610, 2008.