Brno University of Technology at TRECVid 2011 SIN, CCD

Brno University of Technology
Faculty of Information Technology
Department of Computer Graphics and Multimedia
Božetěchova 2, 612 66 Brno, CZ

Semantic indexing

Michal Hradiš, Ivo Řezníček, Kamil Behúň, Lubomír Otrusina
ihradis, ireznicek, iotrusina@fit.vutbr.cz, xbehun03@stud.fit.vutbr.cz

1. The runs differ in the types of features used. All runs use several bag-of-word representations fed to separate linear SVMs and the SVMs were fused by logistic regression. Visual and audio features were used as well as metadata. We added contextual features extracted from the video from which a shot originated.

- F_A_brno.run1 (run1) – Only visual information. Dense sampling and Harris-Laplace detector with SIFT and RGB-SIFT descriptors
- F_A_brno.run1 (run2) – The same as in run1 with added features from audio and metadata.
- F_A_brno.run3 (run3) – The same as in run2 with added contextual features extracted from the whole video.

2. Audio and metadata significantly improves results. Even grater improvement was achieved by using the contextual features.

Content-based Copy Detection

Vítězslav Beran, Ivo Řezníček
beranv, ireznicek@fit.vutbr.cz

1. One run submitted in two versions (the difference is only in relevance threshold setting)
   - brnoccd: SIFT and SURF combination, bag-of-words (visual codebook: 100k size, 4 nearest neighbors used in soft-assignment), inverted file index, geometry (homography) based image similarity metric
2. What if any significant differences (in terms of what measures) did you find among the runs?
   - only one setting used – no differences
3. Based on the results, can you estimate the relative contribution of each component of your system/approach to its effectiveness?
   - slow search in reference dataset due to pure indexing effectiveness
4. Overall, what did you learn about runs/approaches and the research question(s) that motivated them?
   - change the way of describing the video content – frame based (or key-frame based) approach is not sufficient

Semantic indexing

Our approach to semantic indexing combines supervised machine learning, classifier fusion and description of video shots in terms of frequencies of local visual and audio primitives, and frequencies of words in associated metadata. Similar approaches were previously shown to be suitable for this type of tasks [Lazebnik et al., 2006; van de Sande et al., 2010; Snoek et al., 2009]. Additionally, our approach utilizes contextual information extracted from the whole video in addition to the information extracted from the particular shot. The processing pipeline without the contextual features is shown in Figure 1.

To be able to process audio information the same way as image information, spectrograms were computed for short audio segments. The construction of audio and video shot descriptors can be then divided into three separate steps. First, a sampling was used to select parts of the video which are of interest. The appearance of the selected video parts was then expressed by a multidimensional feature vector which is resistant to small displacements and other local transformations while retaining most of the useful information. Based on the local descriptors, a bag-of-word representation describing the whole shot was created. A shot was represented as multiple bag-of-word vectors, each based on different combination of sampling and appearance description. Linear support vector machine (SVM) classifiers were trained separately on these bag-of-word representations and their predictions were fused by logistic regression. An overview of the whole processing pipeline is shown in Figure 1.

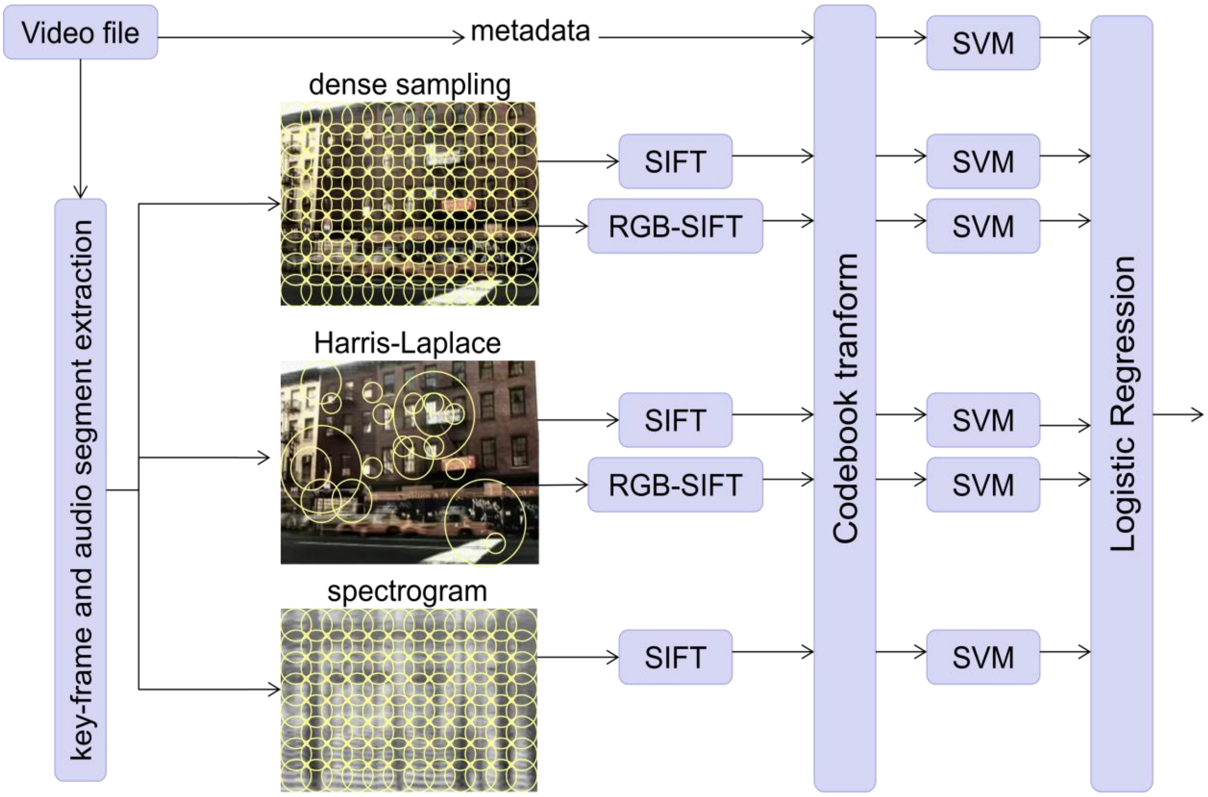

Figure 1 – The processing pipeline without contextual features. Key-frames and audio segments (spectrograms) are extracted from a video shot. Spatial sampling is performed. Local patches are described by SIFT and RGB-SIFT. BOW representations are computed by codebook transform. SVM classifiers are trained for each representation and the classifiers are fused by logistic regression.

The contextual features were computed from the whole video as histograms of responses of the classifiers described in the previous paragraph. These features give information about the distribution of semantic concepts in the video which is intuitively important for shot classification as the semantic concepts presents in shots are correlated in a video.

The following text explains in detail each part of the processing pipeline and also the results achieved in TRECVID 2010 evaluations. First, the image sampling is explained together with appearance description. Next, the transformation to bag-of-word representation is discussed followed by explanation of audio and metadata feature extraction. The subsequent part then gives details on the machine learning. Next, the context features are explained. Finally, the achieved results are presented together with discussion of contributions of the individual parts of the pipeline.

Local patch description

The local patches were parametrized using the original SIFT descriptor by Lowe [Lowe, 1999] and RGB-SIFT [van de Sande et al., 2010]. The SIFT descriptor computes Histograms of Oriented Gradients (HOG) on a 4 x 4 grid centered on an image patch. The computed descriptors are vectors of 128 values created by concatenating the 16 histograms. The magnitude of a single pixel is distributed between neighboring histograms according to a spatial Gaussian filter which alleviates the boundary effect. The SIFT descriptor is invariant to shifts and scaling of the image intensity channel. It encodes the shape of an image patch while being resistant to small displacements and geometric transformations.

The RGB-SIFT descriptor [van de Sande et al., 2010] computes SIFT independently on R, G and B image channels. For computational reasons, Principal Component Analysis was used to reduce dimensionality of the computed descriptors to 198.

Codebook transform

Codebook transformation creates compact, yet powerful representation. In the original form [Lazebnik et al., 2006], the visual features are assigned each to the most similar visual word based on distance in the visual descriptor space. The prototypes of the visual words together form a codebook – thus the name codebook transformation. The codebook transformation produces bag-of-word representation which captures occurrence frequencies of the visual words in a document while discarding any spatial information. Simple ways how to retain some of the spatial information exist [Lazebnik et al., 2006], but these were not used in our system. The further text explains the specifics of our approach.

Separate codebooks were created of each combination of sampling and descriptor. The codebooks were constructed by running 15 iterations of k-means algorithm on 600 MB of randomly selected local features from the training dataset. The size of all codebooks was 4098.

To minimize the amount of lost information, the local features were translated to visual words by soft-assignment instead of hard-assignment. Particularly, codeword uncertainty 0 was used. The kernel was Gaussian and its size represented by standard deviation was equal to average distance of the closest words in the particular codebook. The resulting histograms were not normalized

Audio feature extraction

For parametrization of the audio information, an approach similar to parametrization of the visual information was used. An audio segment representing each shot was extracted from the audio stream. The minimum length of the segments was 10 seconds and overlap was allowed as necessary.

Mel-frequency spectrograms with 128 frequency bands, maximum frequency 8KHz, window length 100ms and overlap 80ms were computed from these segments. Dynamic range of the spectrograms was reduced to fit into 8-bit resolution. The spectrograms were then processed as images by dense sampling and SIFT descriptor. BOW representation was constructed for the spectrograms by codebook transform the same way as for images. The audio features are denoted as SPEC in the results.

Metadata

From the metadata, XML tags were removed together with any non-alphabetical characters and words where lower-case character was followed by upper-case character were split. Stemming was not performed on the data. Although, the data includes several non-English videos, we did not employ any machine translation, as the ratio of the non-English videos is relatively small and it should not seriously influence the results. The words were then converted to BOW representation.

In addition to the simple BOW representation for metadata, Explicit Semantic Analysis (ESA) was used. ESA is a vector space model for computing the similarity of texts based on the Wikipedia. ESA maps each word to the space of Wikipedia articles. The machine learning algorithm is used to assess the strength of association between each word and each concept from the Wikipedia. It computes word similarity in the dimensions defined by the articles. For example, a word can obtain vector (1.2, 0.1, 3.5, …) where the similarity between the word and the first article in Wikipedia is 1.2, the similarity between the word and the second article is 0.1 and so on. The cosine of the angle between the vectors is used to measure similarity of individual texts. For more details on ESA, see [Gabrilovich, 2007].

ESA was used for two types of features. One type of features comprises of similarities to the semantic concepts for which classifiers should be created (METADATA ESA VIDEO). The other types comprises of similarities to all videos from the training set (METADATA ESA VIDEO). The simple BOW representation is denoted in results as METADATA BOW.

Note that when using the metadata features, all shots from a video are represented by the same feature vector.

Classification

The schema of the classification is shown in Figure 2. The main issues for the machine learning part were how to merge information from multiple sources and how to manage relatively large dataset with 130 classes. Generally, SVM is the most common choice of learning algorithm for classification problems where the feature vectors are bags-of-visual-words [Lazebnik et al., 2006; van de Sande et al., 2010; Snoek et al., 2009] and information from different sources is usually merged in kernel [Snoek et al., 2009]. Another possibility is to perform late fusion of separate classifiers each based on the individual information source.

For computational reasons, we decided to use linear SVM to learn separate classifiers for each type of bag-of-word representation and to fuse the separate models linearly by adapting weights of individual models by logistic regression. This approach allowed us to utilize all annotated samples from the training set.

LIBLINEAR [Fan et al. 2008] implementation of SVM solver and logistic regression was used to learn all models. The library was slightly modified to allow terminating computation after a fixed number of iterations.

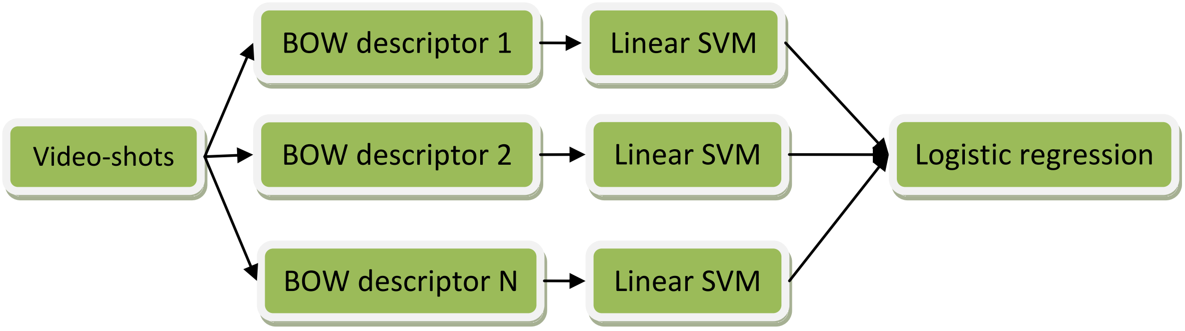

Figure 2 SIN - Classification schema

The soft margin parameter of SVM and the regularization parameter of the logistic regression were both selected separately by grid search with 5-fold cross-validation. The objective function for this parameter optimization was the average precision and the parameters were optimized for each class separately. To utilize all training data for both SVM learning and for the logistic regression, the five SVM classifiers created in cross-validation produced responses for the samples from the training set which were not used to train the particular classifier. Logistic regression was than trained on this whole dataset merged from responses of the five classifiers. The final SVM classifiers were trained on the whole training set and the final fusion was also learned on the full dataset.

Contextual Features

The classifiers described in the previous text use information only about single shot, even though the shots in a video and their semantic concepts are strongly correlated. In order to exploit this correlation, we applied the semantic classifiers for shots to the whole dataset and computed histograms of their responses for each video. These histograms were then used as features vectors fro shot classification as the other features described earlier.

The histograms consisted of 8 equidistant bins with the outer bins set to 5% quantiles. The dimension of the resulting feature vectors obtained by concatenating the histograms of individual semantic classes was 2760.

Note that with the contextual features, all shots from a video are represented by the same feature vector.

The runs

We submitted three different runs which differ in types of features used. The complete overview of the features used is summarized in Table 1.

- **RUN1** Used only visual information. It combined HARLAP, DENSE8 and DENSE16 sampling with SIFT and rgb-SIFT descriptors.
- **RUN2** extended RUN1 with audio information and metadata.
- **RUN3** extended RUN1 with the contextual features.

Results

The results achieved on training set in five-fold cross-validation by separate types of visual features are shown in Table 1. The results were measured as mean average precision across all semantic classes. It can be clearly seen that dense sampling provides generally better results than Harris-

Laplace detector. However, these observations are no longer always valid when looking at performance on separate classes. The audio feature perform significantly worse than image featurs. The metadata provides results comparable to the Harris-Laplace sampling of images except METADATA ESA CONCEPT which gave the worst results.

The contextual features are in all cases better than the features extracted from a shot. This is surprising as with the contextual features, all shots from a video are represented by the same feature vector and, therefore, get the same classifier responses.

Results of the official runs assessed by NIST are shown in Table 2. The results show that adding the audio and metadata modalities (RUN2) improves results significantly over the visual features (RUN1). Adding the contextual features (RUN3) improves results significantly as well. The achieved results on the test set are worse than the best results achieved in the evaluations. Most of this performance gap can be explained by the fact that we use only linear SVM. From our other experiments, we expect that switching to non-linear SVM would improve the results by 50–70 %. Also, we use only training data specific to TRECVID 2011 and utilizing past data or data from other sources would improve the results as well. Detectors of object classes such as faces, people and cars used as feature extractors would also improve results and we plant to follow this idea in the future.

Descriptor	Mean AP	Contextual
HARLAP SIFT	0.165	0.221
HARLAP RGB-SIFT	0.170	0.227
DENSE8 SIFT	0.197	0.214
DENSE8 RGB-SIFT	0.215	0.222
DENSE16 SIFT	0.215	0.222
DENSE16 RGB-SIFT	0.230	0.231
SPEC. DENSE8 SIFT	0.124	0.155
SPEC. DENSE16 SIFT	0.123	0.159
METADATA BOW	0.162	
METADATA ESA CONCEPT	0.107	
METADATA ESA VIDEO	0.160	

Table 1 Results of individual feature types on the training set from TRECVID 2011 in cross-validation achieved by the separate types of features on all 345 classes.

RUN name	Full	Lite
RUN1	0.089	0.068
RUN2	0.108	0.080
RUN3	0.128	0.099
TRECVID best	0.173	0.147
TRECVID median	0.108	0.055

Table 2 Results of the official runs submitted to TRECVID 2011 SIN. Results for Full (evaluated 50 features) and Lite (evaluated 23 features) evaluation are shown as Mean Average Precision.

Content-based Copy Detection

Our CCD system is composed from three main parts: *key-frame detection*, *image retrieval* and *copy candidate verification*. The goal of the system is to find the possible existence of the part of the query video in the reference dataset and detect the positions of the similar video-segments.

Having the reference database prepared, the query video is processed and query key-frames are detected, described and searched in database. The candidates (returned reference key-frames) of adjacent query key-frames are grouped into larger segments when possible. The candidate segments are then verified using more precise frame-content analysis with geometrical constraints and the location of the detected copy segment is refined.

The presented system describes the visual content of video frames using low-level local image features (SURF [Bay et al., 2006] and SIFT [Lowe, 2004]) and bag-of-words representation of those features (visual codebook approach [Sivic and Zisserman, 2003]). Image features such as color histograms, texture analysis, gradient distribution, etc. are employed only for key-frame detection.

Key-frame detection

The reference and query videos are firstly processed to analyze and detect key-frames. Besides the given key-frames by NIST, we are applying the same algorithm for key-frame detection to both, reference and query data, to increase the probability of detection of the key-frame with similar visual features. Our method analyses the difference between frame's features; Euclidean distance for metric features and cosine distance for bag-of-words representation. The differences are evaluated over flowing window (10 samples in average). Figure 3 shows the output signal with candidates shot boundary candidates (red lines) and key-frame candidates (blue lines).

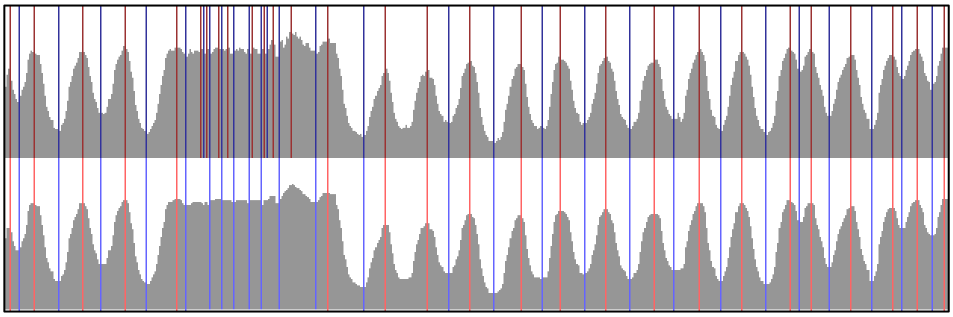

Figure 3 Similarity of visual content of adjacent video frames with shot boundary (red) and key-frame (blue line) detected candidates.

The candidates are then chosen after applying more constraints: too short shots, similar adjacent key-frames, etc.

Image retrieval

The **visual codebook** [Sivic and Zisserman, 2003] is trained in off-line stage using the descriptors from training data. The goal of the visual codebook is to represent the distribution of descriptors in the descriptor space. From the amount of existing approaches, the presented system uses visual vocabulary based on *k-mean* clustering with *kd-tree* search structure and *soft-assignment* schema [Beran et al., 2010a]. The visual codebook and translation procedure has the following setting:

- descriptors space dimensionality (128 for SIFT and 64 for SURF),

- 100.000 visual words in the codebook,
- 4 nearest neighbors in soft-assignment [Philbin et al., 2008],
- standard TF-IDF weighting schema using logarithmic function [Manning and Hinrich, 2008].

The bag-of-words of the key-frames are efficiently stored in database (PostgreSQL) and Generalized Inverted (document) Index [PostgreSQL, 2008] is used to speed up the queries. The *cosine distance* is used as the similarity metric for bag-of-word comparison.

Fusion of retrieved candidates (two retrieval results – SIFT based and SURF based) favors the candidates retrieved in both results. Having the sorted list of candidates from more retrieval runs x_k^{REF} (where k is number of retrieval results for particular reference video REF), increase the confidence of the best candidate by simultaneous application of the following equation for $i \in \{2, k\}$:

$$x_1^{REF} = x_1^{REF} + (1 - x_1^{REF}) * x_i^{REF}$$

Experimenting with more fusion functions, this approach gives the best results.

Copy candidate verification

Having the list of candidate key-frames from the reference dataset for each key-frame from the query video, first, the block segments are constructed from adjacent query key-frames referencing to the similar video. Then each candidate reference segment (reference video segment) of the each query segment (query video segment) is analyzed using following procedure

1. The reference-frame to query-frame homography H_{RQ} is computed and classified. When H_{RQ} is classified as *too-distorted*, the candidate segment is refused and next candidate is analyzed. The H_{RQ} is applied to all query video frames to refine the possible geometrical transformation between these video sequences.
2. Search for left and right cuts in both, reference and query sequences. The cut probability is evaluated using the same metric as in the key-frame detection method.
3. Evaluate the visual similarity of the overlapped part of the reference and the query sequence. If *too-different*, discard the candidate and process the next one.
4. Refine the time offset between reference and query sequences [Beran et al., 2010b].
5. Find final borders of overlapping reference and query segments.

Each query segment might reference to ~2000 candidates, but most of them are refused at the very beginning of the verification procedure. The reference candidates are then sorted according to accumulated characteristics (scores and errors) and reported.

Results

The experiments with the system reveal two major parts of interest: *visual codebook setting* correlating to index performance and *video-content description* approach. Visual codebook setting seems to be the crucial for index performance. The used setting (codebook size, soft-assignment parameters, etc.) caused that the usage of inverted-file index poor efficiency. Video-content description method was taken from image retrieval system with no adaptation to video (temporal) data. Also, the system was not especially re-designed for the introduced TRECVID set transformations, so e.g. vertical flip transformation is undetectable by our CCD system.

Our CCD system shows very good results in precision of detected copy segments measured by defined F1 metric [Smeaton et al., 2006] for positively detected samples.

Acknowledgements

This work has been supported by the research project "Security-Oriented Research in Information Technology" CEZMSMT, MSM0021630528, EU FP7 project TA2: Together Anywhere, Together Anytime ICT-007-214793, grant no 214793 and "Tools and Methods for Video and Image Processing for the Fight against Terrorism", VG20102015006.

References

[Bay et al., 2006] Bay, H., Tuytelaars, T., and Gool, L. V. Surf: Speeded up robust features. In In ECCV, pp. 404-417, 2006.

[Beran et al., 2010a] Beran, V., Zemčík, P. Visual Codebooks Survey for Video On-line Processing, In: Computer Vision and Graphics: Proc. ICCVG 2010, Warsaw, PL, Springer, p. 10, 2010.

[Beran et al., 2010b] Beran, V., Zemčík, P., Herout, A. On-line Video Synchronization Based on Visual Vocabularies, In: International Journal of Signal and Image Processing, Vol. 2010, No. 2, TN, p. 7, ISSN 1737-9253, 2010.

[Fan et al. 2008] R.-E. Fan et al.: LIBLINEAR: A library for large linear classification. Journal of Machine Learning Research 9(2008), pp. 1871-1874, 2008.

[Gabrilovich, 2007] Gabrilovich, E. and Markovitch, S. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In Proceedings of the 20th International Joint Conference on Artificial Intelligence, 2007.

[Gemert et al., 2010] Jan C. van Gemert, Cor J. Veenman, Arnold W.M. Smeulders, Jan-Mark Geusebroek: Visual Word Ambiguity, PAMI, pp. 1271-1283, July, 2010.

[Laptev and Lindeberg, 2003] I. Laptev and T. Lindeberg: Space-time interest points. In ICCV, 2003.

[Lazebnik et al., 2006] Lazebnik, S., Schmid, C., Ponce, J.: Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories, Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on , vol.2, no., pp. 2169- 2178, 2006.

[Lowe, 1999] Lowe, D. G. Object Recognition from Local Scale-Invariant Features. In ICCV ’99: Proceedings of the International Conference on Computer Vision-Volume 2, page 1150,Washington, DC, USA. IEEE ComputerSociety, 1999.

[Lowe, 2004] Lowe, D. G. Distinctive Image Features from Scale-Invariant Keypoints, International Journal of Computer Vision, 60, 2, pp. 91-110, 2004.

[Manning and Hinrich, 2008] Manning Christopher D., R. P., and Hinrich, S. Introduction to Information Retrieval. Cambridge University Press, 2008.

[Mikolajczyk and Schmid, 2004] Mikolajczyk, K. and Schmid, C.: Scale & affine invariant interest point detectors. International Journal on Computer Vision 60(1):63-86, 2004.

[Philbin et al., 2008] Philbin, J., Chum, O., Isard, M., Sivic, J., and Zisserman, A. Lost in quantization: Improving particular object retrieval in large scale image databases. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2008.

[PostgreSQL, 2008] PostgreSQL Global Development Group. 2008. PostgreSQL 8.3 Documentation: GIN Indexes. http://www.postgresql.org/docs/8.3/static/gin.html.

[van de Sande et al., 2010] Koen E. A. van de Sande, Theo Gevers and Cees G. M. Snoek: Evaluating Color Descriptors for Object and Scene Recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence, volume 32 (9), pages 1582-1596, 2010.

[Sivic and Zisserman, 2003] Sivic, J., and Zisserman, A. Video google: A text retrieval approach to object matching in videos. In ICCV '03: Proceedings of the Ninth IEEE International Conference on Computer Vision (Washington, DC, USA, 2003), vol. 2, IEEE Computer Society, pp. 1470-1477, 2003.

[Smeaton et al., 2006] Smeaton, A. F., Over, P., and Kraaij, W. Evaluation campaigns and TRECVid. In Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval (Santa Barbara, California, USA, October 26 - 27, 2006). MIR '06. ACM Press, New York, NY, 321-330. DOI= http://doi.acm.org/10.1145/1178677.1178722

[Snoek et al., 2009] C.G.M. Snoek et al.: The MediaMill TRECVID 2009 Semantic Video Search Engine. TRECVID 2009: Participant Notebook Papers and Slides. National Institute of Standards and Technology, Gaithersburg, MD, US, 2009.

[Willems et al., 2008] G. Willems et al.: An efficient dense and scale-in variant spatio-temporal interest point detector. In ECCV, 2008