The University of Amsterdam's Concept Detection System at ImageCLEF 2009

Koen E. A. van de Sande, Theo Gevers and Arnold W. M. Smeulders
Intelligent Systems Lab Amsterdam (ISLA), University of Amsterdam
ksande@uva.nl

Abstract

Our group within the University of Amsterdam participated in the large-scale visual concept detection task of ImageCLEF 2009. Our experiments focus on increasing the robustness of the individual concept detectors based on the bag-of-words approach, and less on the hierarchical nature of the concept set used. To increase the robustness of individual concept detectors, our experiments emphasize in particular the role of visual sampling, the value of color invariant features, the influence of codebook construction, and the effectiveness of kernel-based learning parameters. The participation in ImageCLEF 2009 has been successful, resulting in the top ranking for the large-scale visual concept detection task in terms of both EER and AUC. For 40 out of 53 individual concepts, we obtain the best performance of all submissions to this task. For the hierarchical evaluation, which considers the whole hierarchy of concepts instead of single detectors, using the concept likelihoods estimated by our detectors directly works better than scaling these likelihoods based on the class priors.

Categories and Subject Descriptors

H.3 [**Information Storage and Retrieval**]: H.3.1 Content Analysis and Indexing; H.3.4 Systems and Software; I.4.7 [**Image Processing and Computer Vision**]: Feature Measurement

General Terms

Performance, Measurement, Experimentation

Keywords

Color, Invariance, Concept Detection, Object and Scene Recognition, Bag-of-Words, Photo Annotation, Spatial Pyramid

1 Introduction

Robust image retrieval is highly relevant in a world that is adapting swiftly to visual communication. Online services like Flickr show that the sheer number of photos available online is too much for any human to grasp. Many people place their entire photo album on the internet. Most commercial image search engines provide access to photos based on text or other metadata, as this is still the easiest way for a user to describe an information need. The indices of these search engines are based on the filename, associated text or (social) tagging. This results in disappointing retrieval performance when the visual content is not mentioned, or properly reflected in the associated text. In addition, when the photos originate from non-English speaking countries, such as China, or the Netherlands, querying the content becomes much harder.

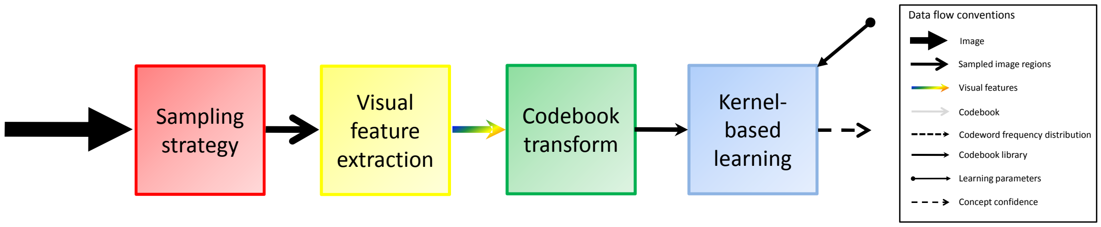

Figure 1: University of Amsterdam's ImageCLEF 2009 concept detection scheme, using the conventions shown on the right. The scheme serves as the blueprint for the organization of Section 2.

To cater for robust image retrieval, the promising solutions from literature are in majority concept-based [16], where detectors are related to objects, like a *telephone*, scenes, like a *kitchen*, and people, like *big group*. Any one of those brings an understanding of the current content. The elements in such a lexicon offer users a semantic entry by allowing them to query on presence or absence of visual content elements.

The Large-Scale Visual Concept Detection Task [12] evaluates 53 visual concept detectors. The concepts used are from the personal photo album domain: *beach holidays*, *snow*, *plants*, *indoor*, *mountains*, *still-life*, *small group of people*, *portrait*. For more information on the dataset and concepts used, see the overview paper [12].

Based on our previous work on concept detection [19, 15], we have focused on improving the robustness of the visual features used in our concept detectors. Systems with the best performance in image retrieval [11, 19] and video retrieval [22, 15] use combinations of multiple features for concept detection. The basis for these combinations is formed by good color features and multiple point sampling strategies.

This paper is organized as follows. Section 2 defines our concept detection system. Section 3 details our experiments and results. Finally, in section 4, conclusions are drawn.

2 Concept Detection System

We perceive concept detection as a combined computer vision and machine learning problem. Given an n-dimensional visual feature vector x_i, the aim is to obtain a measure, which indicates whether semantic concept ω_j is present in photo i. We may choose from various visual feature extraction methods to obtain x_i, and from a variety of supervised machine learning approaches to learn the relation between ω_j and x_i. The supervised machine learning process is composed of two phases: training and testing. In the first phase, the optimal configuration of features is learned from the training data. In the second phase, the classifier assigns a probability $p(\omega_j|x_i)$ to each input feature vector for each semantic concept.

2.1 Sampling Strategy

The visual appearance of a concept has a strong dependency on the viewpoint under which it is recorded. Salient point methods [17] introduce robustness against viewpoint changes by selecting points, which can be recovered under different perspectives. Another solution is to simply use many points, which is achieved by dense sampling. We summarize our sampling approach in Figure 2.

Harris-Laplace point detector In order to determine salient points, Harris-Laplace relies on a Harris corner detector. By applying it on multiple scales, it is possible to select the characteristic scale of a local corner using the Laplacian operator [17]. Hence, for each corner the Harris-Laplace detector selects a scale-invariant point if the local image structure under a Laplacian operator has a stable maximum.

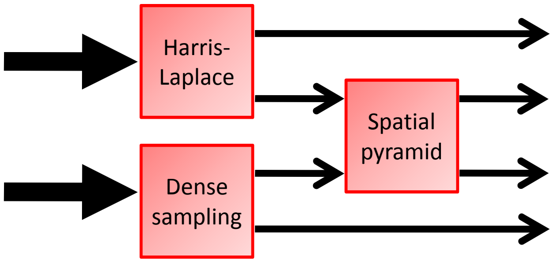

Figure 2: General scheme for sampling of image regions, including Harris-Laplace and dense point selection, and a spatial pyramid. Detail of Figure 1, using the same conventions.

Dense point detector For concepts with many homogenous areas, like scenes, corners are often rare. Hence, for these concepts relying on a Harris-Laplace detector can be suboptimal. To counter the shortcoming of Harris-Laplace, random and dense sampling strategies have been proposed [4, 6]. We employ dense sampling, which samples an image grid in a uniform fashion using a fixed pixel interval between regions. In our experiments we use an interval distance of 6 pixels and sample at multiple scales.

Spatial pyramid weighting Both Harris-Laplace and dense sampling give an equal weight to all keypoints, irrespective of their spatial location in the image frame. In order to overcome this limitation, Lazebnik *et al.* [7] suggest to repeatedly sample fixed subregions of an image, *e.g.* 1x1, 2x2, 4x4, *etc.*, and to aggregate the different resolutions into a so called spatial pyramid, which allows for region-specific weighting. Since every region is an image in itself, the spatial pyramid can be used in combination with both the Harris-Laplace point detector and dense point sampling [18]. Reported results using concept detection experiments are not yet conclusive in the ideal spatial pyramid configuration, some claim 2x2 is sufficient [7], others suggest to include 1x3 also [11]. We use a spatial pyramid of 1x1, 2x2, and 1x3 regions in our experiments.

2.2 Visual Feature Extraction

In the previous section, we addressed the dependency of the visual appearance of semantic concepts on the viewpoint under which they are recorded. However, the lighting conditions during photography also play an important role. We [19] analyzed the properties of color features under classes of illumination changes within the diagonal model of illumination change, and specifically for data sets consisting of Flickr images. In ImageCLEF, the images used also originate from Flickr. Here we summarize the main findings. We present an overview of the visual features used in Figure 3.

The features are computed around salient points obtained from the Harris-Laplace detector and dense sampling.

SIFT The SIFT feature proposed by Lowe [10] describes the local shape of a region using edge orientation histograms. The gradient of an image is shift-invariant: taking the derivative cancels out offsets [19]. Under light intensity changes, *i.e.* a scaling of the intensity channel, the gradient direction and the relative gradient magnitude remain the same. Because the SIFT feature is normalized, the gradient magnitude changes have no effect on the final feature. To compute SIFT features, we use the version described by Lowe [10].

OpponentSIFT OpponentSIFT describes all the channels in the opponent color space using SIFT features. The information in the O_3 channel is equal to the intensity information, while the

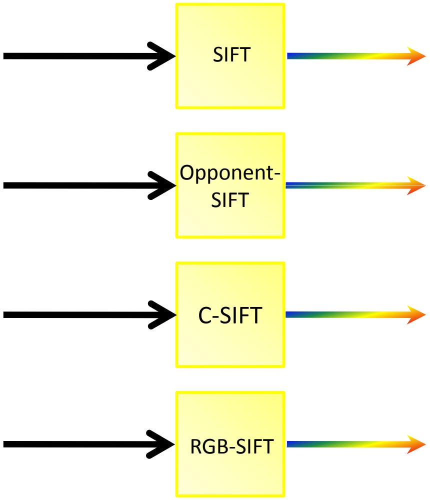

Figure 3: General scheme of the visual feature extraction methods used in our ImageCLEF 2009 experiments.

other channels describe the color information in the image. The feature normalization, as effective in SIFT, cancels out any local changes in light intensity.

C-SIFT The C-SIFT feature uses the C invariant [5], which can be intuitively seen as the gradient (or derivative) for the normalized opponent color space $O1/I$ and $O2/I$. The I intensity channel remains unchanged. C-SIFT is known to be scale-invariant with respect to light intensity. See [1, 19] for detailed evaluation.

RGB-SIFT For the RGB-SIFT, the SIFT feature is computed for each RGB channel independently. Due to the normalizations performed within SIFT, it is equal to transformed color SIFT [19]. The feature is scale-invariant, shift-invariant, and invariant to light color changes and shift.

2.3 Codebook Transform

To avoid using all visual features in an image, while incorporating translation invariance and a robustness to noise, we follow the well known codebook approach, see *e.g.* [8, 6, 23, 20, 19]. First, we assign visual features to discrete codewords predefined in a codebook. Then, we use the frequency distribution of the codewords as a compact feature vector representing an image frame. Two important variables in the codebook representation are *codebook construction* and *codeword assignment*. An extensive comparison of codebook representation variables is presented by Van Gemert *et al.* in [20]. Here we detail codebook construction and codeword assignment using hard and soft variants, following the scheme in Figure 4.

Codebook construction We employ k-means clustering. K-means partitions the visual feature space by minimizing the variance between a predefined number of k clusters. The advantage of the k-means algorithm is its simplicity. A disadvantage of k-means is its emphasis on clusters of dense areas in feature space. Hence, k-means does not spread clusters evenly throughout feature space. We fix the visual codebook to a maximum of 4000 codewords.

Hard-assignment Given a codebook of codewords, obtained from clustering, the traditional codebook approach describes each feature by the single best representative codeword in the code-

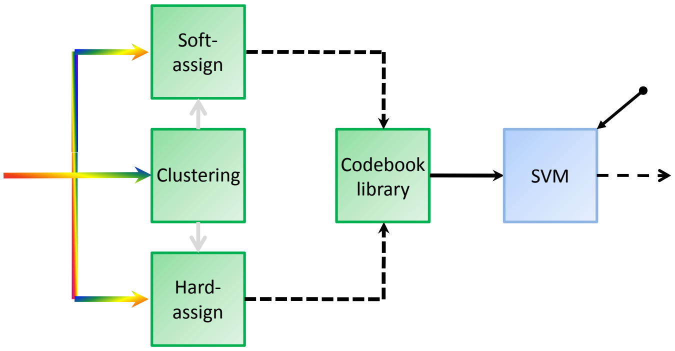

Figure 4: General scheme for transforming visual features into a codebook, where we distinguish between codebook construction using clustering and codeword assignment using soft and hard variants. We combine various codeword frequency distributions into a codebook library. This then forms the input to an SVM classifier.

book, *i.e.* hard-assignment. Basically, an image is represented by a histogram of codeword frequencies describing the probability density over codewords.

Soft-assignment In a recent paper [20], it is shown that the traditional codebook approach may be improved by using soft-assignment through kernel codebooks. A kernel codebook uses a kernel function to smooth the hard-assignment of image features to codewords. Out of the various forms of kernel-codebooks, we selected *codeword uncertainty* based on its empirical performance [20].

Codebook library Each of the possible sampling methods from Section 2.1 coupled with each visual feature extraction method from Section 2.2, a clustering method, and an assignment approach results in a separate visual codebook. An example is a codebook based on dense sampling of RGB-SIFT features in combination with hard-assignment. We collect all possible codebook combinations in a visual codebook library. Naturally, the codebooks can be combined using various configurations. For simplicity, we employ equal weights in our experiments when combining codebooks to form a library.

2.4 Kernel-based Learning

Learning robust concept detectors from large-scale visual codebooks is typically achieved by kernel-based learning methods. From all kernel-based learning approaches on offer, the support vector machine is commonly regarded as a solid choice. An overview is given together with the codebook transformations in Figure 4.

Support vector machine We use the support vector machine framework [21] for supervised learning of concepts. Here we use the LIBSVM implementation [2] with probabilistic output [13, 9]. The parameter of the support vector machine we optimize is C. In order to handle imbalance in the number of positive versus negative training examples, we fix the weights of the positive and negative class by estimation from the class priors on training data. It was shown by Zhang *et al.* [23] that in a codebook-approach to concept detection the earth movers distance and χ^2 kernel are to be preferred. We employ the χ^2 kernel, as it is less expensive in terms of computation.

3 Concept Detection Experiments

3.1 Submitted Runs

We have submitted five different runs. All runs use both Harris-Laplace and dense sampling with the SVM classifier. We do not use the EXIF metadata provided for the photos. Our system has been developed based on the PASCAL VOC [3] and TRECVID Sound and Vision datasets [14]. For ImageCLEF, we have learned new concept models based on the provided annotations. The only parameter specifically optimized for this dataset is the slack parameter C of the SVM. All other parameter settings are the same as in our PASCAL VOC 2008 system [19]. Extracting features, training models and applying those models on the test set was finished within 72 hours.

- **OpponentSIFT**: single color descriptor with hard assignment.
- **2-SIFT**: two color descriptors (OpponentSIFT and SIFT) with hard assignment.
- **4-SIFT**: four color descriptors (OpponentSIFT, C-SIFT, RGB-SIFT and SIFT) with hard assignment.
- **Rescaled 4-SIFT**: the same ordering of images as 4-SIFT, but with all concept detector outputs linearly scaled so the number of images with a score > 0.5 is equal to the concept prior probability in the training set.
- **Soft 4-SIFT**: four color descriptors (OpponentSIFT, C-SIFT, RGB-SIFT and SIFT) with soft assignment. The soft assignment parameters have been taken from our PASCAL VOC 2008 system [19].

3.2 Evaluation Per Concept

In table 1, the overall scores for the evaluation of concept detectors are shown. As for the evaluation of single detectors only the ranking of the images within a single concept matters, the rescaled version of 4-SIFT achieves the exact same performance as 4-SIFT. We note that the 4-SIFT run with hard assignment achieves not only the highest performance amongst our runs, but also over all other runs submitted to the Large-Scale Visual Concept Detection task.

In table 2, the Area Under the Curve scores have been split out per concept. We observe that the three aesthetic concepts have the lowest scores. This comes as no surprise, because these concepts are highly subjective: even human annotators only agree around 80% of the time with each other. For virtually all concepts besides the aesthetic ones, either the Soft 4-SIFT or the Hard 4-SIFT is the best run. This confirms our beliefs that these (color) descriptors are not redundant when used in combinations. Therefore, we recommend the use of these 4 descriptors instead of 1 or 2. The difference in overall performance between the Soft 4-SIFT or the Hard 4-SIFT run is quite small. Because the soft codebook assignment smoothing parameter was directly taken from a different dataset, we expect that the soft assignment run could be improved if the soft assignment parameter was selected with cross-validation on the training set. Together, our

Run name	Codebook	Average EER	Average AUC
4-SIFT	Hard-assignment	**0.2345**	**0.8387**
Rescaled 4-SIFT	Hard-assignment	**0.2345**	**0.8387**
Soft 4-SIFT	Soft-assignment	0.2355	0.8375
2-SIFT	Hard-assignment	0.2435	0.8300
OpponentSIFT	Hard-assignment	0.2530	0.8217

Table 1: Overall results of the University of Amsterdam evaluated over all concepts in the Large-Scale Visual Concept Detection Task using the equal error rate (EER) and the area under the curve (AUC).

Concept	4-SIFT	Soft 4-SIFT	2-SIFT	Opp.SIFT
Clouds	**0,958**	0,958	0,951	0,945
Sunset-Sunrise	0,953	**0,954**	0,947	0,946
Sky	0,945	**0,948**	0,935	0,930
Landscape-Nature	**0,944**	0,942	0,940	0,936
Sea	**0,935**	0,930	0,932	0,926
Mountains	**0,934**	0,931	0,930	0,922
Lake	0,911	0,903	**0,912**	0,900
Beach-Holidays	0,906	**0,907**	0,898	0,884
Trees	**0,903**	0,902	0,892	0,881
Water	0,901	**0,903**	0,892	0,886
Night	**0,898**	0,895	0,895	0,892
River	**0,897**	0,889	0,891	0,883
Outdoor	0,890	**0,896**	0,879	0,871
Food	**0,895**	0,895	0,881	0,877
Desert	**0,891**	0,865	0,891	0,884
Building-Sights	0,880	**0,882**	0,873	0,861
Big-Group	**0,881**	0,877	0,870	0,858
Plants	0,877	**0,881**	0,853	0,839
Flowers	0,868	**0,875**	0,846	0,836
Autumn	**0,870**	0,866	0,863	0,849
Portrait	**0,865**	0,864	0,857	0,846
Underexposed	0,858	**0,859**	0,857	0,854
No-Persons	0,850	**0,858**	0,837	0,826
Partly-Blurred	**0,852**	0,852	0,845	0,830
Winter	0,843	**0,846**	0,832	0,828
Snow	**0,846**	0,845	0,829	0,825
Day	0,841	**0,845**	0,831	0,824
No-Blur	0,843	**0,845**	0,836	0,823
No-Visual-Time	0,833	**0,835**	0,822	0,815
Indoor	0,830	**0,835**	0,823	0,810
Familiy-Friends	**0,834**	0,834	0,822	0,813
Partylife	**0,834**	0,834	0,831	0,819
Vehicle	**0,832**	0,832	0,832	0,822
Animals	0,818	**0,828**	0,811	0,797
Citylife	**0,826**	0,826	0,819	0,813
Still-Life	0,824	**0,825**	0,808	0,795
Spring	**0,822**	0,801	0,812	0,791
Canvas	**0,817**	0,810	0,803	0,790
Summer	**0,813**	0,813	0,791	0,782
Macro	**0,812**	0,791	0,805	0,795
No-Visual-Season	0,805	**0,806**	0,794	0,782
Small-Group	0,792	**0,795**	0,784	0,776
Single-Person	0,792	**0,795**	0,780	0,769
Out-of-focus	**0,792**	0,781	0,784	0,774
No-Visual-Place	**0,789**	0,786	0,781	0,779
Overexposed	**0,788**	0,782	0,777	0,771
Neutral-Illumination	0,778	**0,783**	0,775	0,774
Sunny	0,763	**0,765**	0,744	0,741
Motion-Blur	0,744	**0,747**	0,725	0,710
Sports	**0,695**	0,695	0,679	0,673
Aesthetic-Impression	0,658	**0,662**	0,657	0,657
Overall-Quality	0,656	0,656	0,653	**0,658**
Fancy	0,565	0,559	0,580	**0,583**
Average	**0,8387**	**0,8375**	**0,8300**	**0,8217**

Table 2: Results per concept for our runs in the Large-Scale Visual Concept Detection Task using the Area Under the Curve. The highest score per concept is highlighted using a grey background. The concepts are ordered by their highest score.

runs obtain the highest Area Under the Curve scores for 40 out of 53 concepts in the Photo Annotation task (20 for Soft 4-SIFT, 17 for 4-SIFT and 3 for the other runs). This analysis has shown us that our system is falling behind for concepts that correspond to conditions we have included invariance against. Our method is designed to be robust to unsharp images, so for *Out-of-focus*, *Partly-Blurred* and *No-Blur* there are better approaches possible. For the concepts *Overexposed*, *Underexposed*, *Neutral-Illumination*, *Night* and *Sunny*, recognizing how the scene is illuminated is very important. Because we are using invariant color descriptors, a lot of the discriminative lighting information is no longer present in the descriptors. Again, there should be better approaches possible for these concepts, such as estimating the color temperature and overall light intensity.

Our system was developed on other datasets, and only the concept models were specifically learned for the Photo Annotation dataset. Its good performance on this dataset, without changing the parameter settings, shows that it is generic and generalizes to multiple datasets. But, our system only performs well on this dataset because the train and test set come from the same source and have been obtained at the same time. Generalization across the boundary of multiple datasets is still an unsolved problem: for photos downloaded from Flickr in a different season or general web images, the performance will be significantly worse. However, all systems participating in the Photo Annotation task are 'overtrained' in this sense, and the models they learned too specific. An interesting avenue for future editions is to have a second test set with photos from a different source or moment in time, so this problem can be investigated further.

		Average Annotation Score	
Run name	Codebook	with agreement	without agreement
Soft 4-SIFT	Soft-assignment	**0.7831**	**0.7598**
4-SIFT	Hard-assignment	0.7812	0.7578
2-SIFT	Hard-assignment	0.7780	0.7544
OpponentSIFT	Hard-assignment	0.7705	0.7464
Rescaled 4-SIFT	Hard-assignment	0.7503	0.7312

Table 3: Results using the hierarchical evaluation measures for our runs in the Large-Scale Visual Concept Detection Task.

3.3 Evaluation Per Image

For the hierarchical evaluation, overall results are shown in table 3. When compared to the evaluation per concept, the Soft 4-SIFT run is now slightly better than the normal 4-SIFT run. Our attempt to improve performance for the hierarchical evaluation measure using a linear rescaling of the concept likelihoods has had the opposite effect: the normal 4-SIFT run is better than the Rescaled 4-SIFT run. Therefore, further investigation into building a cascade of concept classifiers is needed, as simply using the individual concept classifiers with their class priors does not work.

4 Conclusion

Our focus on invariant visual features for concept detection in ImageCLEF 2009 has been successful. It has resulted in the top ranking for the large-scale visual concept detection task in terms of both EER and AUC. For 40 individual concepts, we obtain the best performance of all submissions to the task. For the hierarchical evaluation, using the concept likelihoods estimated by our detectors directly works better than scaling these likelihoods based on the class priors.

Acknowledgements

This work was supported by the EC-FP6 VIDI-Video project.

References

[1] G. J. Burghouts and J. M. Geusebroek. Performance evaluation of local color invariants. *Computer Vision and Image Understanding*, 113:48–62, 2009.

[2] C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[3] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2008 (VOC2008) Results.

[4] L. Fei-Fei and P. Perona. A bayesian hierarchical model for learning natural scene categories. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 524–531, San Diego, USA, 2005.

[5] J. M. Geusebroek, R. van den Boomgaard, A. W. M. Smeulders, and H. Geerts. Color invariance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(12):1338–1350, 2001.

[6] F. Jurie and B. Triggs. Creating efficient codebooks for visual recognition. In *IEEE International Conference on Computer Vision*, pages 604–610, Beijing, China, 2005.

[7] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 2169–2178, New York, USA, 2006.

[8] T. K. Leung and J. Malik. Representing and recognizing the visual appearance of materials using three-dimensional textons. *International Journal of Computer Vision*, 43(1):29–44, 2001.

[9] H.-T. Lin, C.-J. Lin, and R. C. Weng. A note on Platt's probabilistic outputs for support vector machines. *Machine Learning*, 68(3):267–276, 2007.

[10] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[11] M. Marszałek, C. Schmid, H. Harzallah, and J. van de Weijer. Learning object representations for visual object class recognition, 2007. Visual Recognition Challenge workshop, in conjunction with IEEE International Conference on Computer Vision, Rio de Janeiro, Brazil.

[12] S. Nowak and P. Dunker. Overview of the clef 2009 large scale visual concept detection and annotation task. In *CLEF working notes 2009*, Corfu, Greece, 2009.

[13] J. C. Platt. Probabilities for SV machines. In A. J. Smola, P. L. Bartlett, B. Schölkopf, and D. Schuurmans, editors, *Advances in Large Margin Classifiers*, pages 61–74. MIT Press, 2000.

[14] A. F. Smeaton, P. Over, and W. Kraaij. Evaluation campaigns and TRECVid. In *ACM International Workshop on Multimedia Information Retrieval*, pages 321–330, Santa Barbara, USA, 2006.

[15] C. G. M. Snoek, K. E. A. van de Sande, O. de Rooij, B. Huurnink, J. C. van Gemert, J. R. R. Uijlings, and *et al.* . The MediaMill TRECVID 2008 semantic video search engine. In *Proceedings of the 6th TRECVID Workshop*, Gaithersburg, USA, November 2008.

[16] C. G. M. Snoek and M. Worring. Concept-based video retrieval. *Foundations and Trends in Information Retrieval*, 4(2):215–322, 2009.

[17] T. Tuytelaars and K. Mikolajczyk. Local invariant feature detectors: A survey. *Foundations and Trends in Computer Graphics and Vision*, 3(3):177–280, 2008.

[18] K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek. A comparison of color features for visual concept classification. In *ACM International Conference on Image and Video Retrieval*, pages 141–150, 2008.

[19] 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*, (in press), 2010.

[20] J. C. van Gemert, C. J. Veenman, A. W. M. Smeulders, and J. M. Geusebroek. Visual word ambiguity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (in press), 2010.

[21] V. N. Vapnik. *The Nature of Statistical Learning Theory.* Springer-Verlag, New York, USA, 2nd edition, 2000.

[22] D. Wang, X. Liu, L. Luo, J. Li, and B. Zhang. Video diver: generic video indexing with diverse features. In *ACM International Workshop on Multimedia Information Retrieval*, pages 61–70, Augsburg, Germany, 2007.

[23] J. Zhang, M. Marszałek, S. Lazebnik, and C. Schmid. Local features and kernels for classification of texture and object categories: A comprehensive study. *International Journal of Computer Vision*, 73(2):213–238, 2007.