



Report on Benchmark Evaluation

Deliverable D.6.4

Month 21

Project Start:	01/02/2007
Project Duration:	36 Months
Priority area	2.3.6
Contract No.:	FP6-045547
Website:	http://www.vidivideo.info/

Due-Date:	01/11/2008
Delivery:	24/11/2008
Lead Partner:	CVC
Project Leader	UvA
Dissemination Level:	Public
Status:	Final
Approved:	Yes
Version:	1.0



Table of Contents

Table of Contents	2
1. Introduction.....	3
2. Detecting Concepts in Video	5
2.1. Spatio-Temporal Sampling	6
2.2. Visual Feature Extraction	8
2.3. Codebook Transform.....	11
2.4. Kernel-based Learning	13
2.5. Submitted Concept Detection Results	15
3. Detecting Concepts in Images.....	18
3.1. Visual Feature Extraction	18
3.2. Machine Learning	19
3.3. Submitted Concept Detection Results	19
References	22



1. Introduction

The objective of this report is the performance evaluation of the system being developed in WP6. Towards this end, different challenges or competitions have been organized recently in order to compare the results of novel approaches based on the same training and testing data. To allow for a quantitative evaluation of progress, we have participated in the TRECVID and VOC-PASCAL benchmarks in the second year of the project. Participation has been focused on (i) the concept detection task to see how the performance of each individual detector is compared to other state-of-the-art systems and (ii) the interactive retrieval task where the aim is to have users find the best result for a set of 24 information needs where for each need the interaction time is limited to 15 minutes. The latter will be a stepping stone to effective performance in the runtime interactive system. So this document details how the performance of the prototype has been evaluated in the TRECVID video retrieval benchmark and VOC-PASCAL object categorization challenge, yielding excellent results.

The [TREC VIDEO RETRIEVAL EVALUATION](#)¹ conference series is sponsored by the National Institute of Standards and Technology ([NIST](#)) with additional support from other U.S. government agencies. The goal of the conference series is to encourage research in information retrieval by providing a large test collection, uniform scoring procedures, and a forum for organizations interested in comparing their results. In 2001 and 2002 the TREC series sponsored a video "track" devoted to research in automatic segmentation, indexing, and content-based retrieval of digital video. Beginning in 2003, this track became an independent evaluation (TRECVID) with a 2-day workshop taking place just before TREC.

¹ <http://www-nlpir.nist.gov/projects/tv2008/tv2008.call.html>



The goal of the PASCAL Visual Object Classes Challenge (VOC) challenge² is to recognize objects from a number of visual object classes in realistic scenes (i.e. not pre-segmented objects). It is fundamentally a supervised learning learning problem in that a training set of labelled images is provided. For example, the twenty object classes that have been selected in 2008 are: person, bird, cat, cow, dog, horse, sheep, aeroplane, bicycle, boat, bus, car, motorbike, train, bottle, chair, dining table, potted plant, sofa, and tv/monitor.

To cater for robust video retrieval, the promising solutions from literature are in majority concept-based [24], where detectors are related to objects, like a *telephone*, scenes, like a *kitchen*, and people, like *singing*. Any one of those brings an understanding of the current content. The elements in such a lexicon offer users a semantic entry to video by allowing them to query on presence or absence of visual content elements. Last year UvA presented the *MediaMill 2007* semantic video search engine [22] using a 572 concept lexicon, albeit with varying performance. Rather than continuing to increase the lexicon size, our benchmark experiments during VIDI-Video will focus on increasing the robustness of a small set of concept detectors by using novel approaches that build upon recent findings in computer vision and pattern recognition.

² <http://pascallin.ecs.soton.ac.uk/challenges/VOC/voc2008/index.html>

2. Detecting Concepts in Video

We perceive concept detection in video as a combined computer vision and machine learning problem. Given an n -dimensional visual feature vector x_i , part of a shot i [18], the aim is to obtain a measure, which indicates whether semantic concept ω_j is present in shot 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.

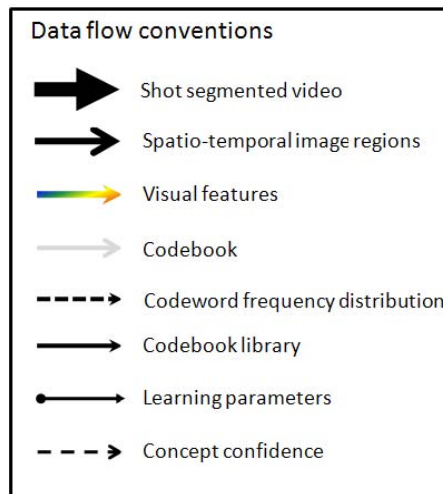


Figure 1: Data flow conventions as used in this Section. Different arrows indicate difference in data flows.

Our TRECVID 2008 concept detection approach builds on previous editions of the MediaMill semantic video search engine [22,23,26]. In addition, we draw inspiration from the work of Schmid and her associates [16, 36], extending their work by putting special emphasis on video sampling strategies, keypoint-based color features [4, 30], codebook representations [31, 33], and kernel-based machine learning. We detail our generic

concept detection scheme by presenting a component-wise decomposition. The components exploit a common architecture, with a standardized input-output model, to allow for semantic integration. The graphical conventions to describe the system architecture are indicated in Figure 1. Based on these conventions we follow the video data as they flow through the computational process, as summarized in the general scheme of our TRECVID 2008 concept detection approach in Figure 2, and detailed per component next.

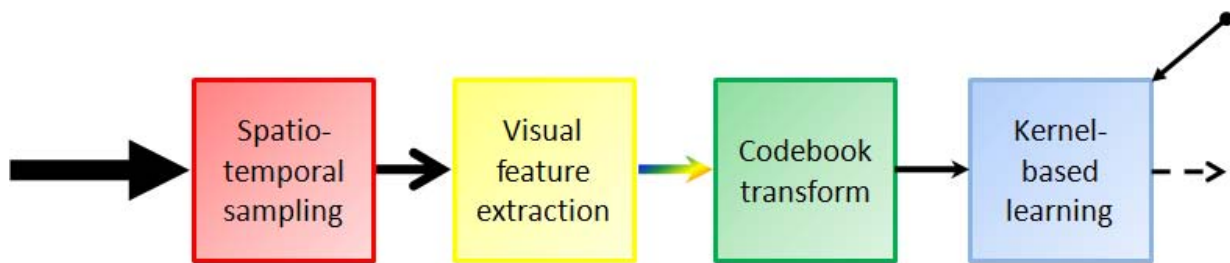


Figure 2: MediaMill TRECVID 2008 concept detection scheme, using the conventions of Figure 1.
The scheme serves as the blueprint for the organization of Section 2.

2.1. Spatio-Temporal Sampling

The visual appearance of a semantic concept in video has a strong dependency on the spatio-temporal viewpoint under which it is recorded. Salient point methods introduce robustness against viewpoint changes by selecting points which can be recovered under different viewpoints. Another solution is to simply use many points, which is done by dense sampling. Appearance variations caused by temporal effects are addressed by going beyond the key frame level. By taking more frames into account for the analysis, it becomes possible to recognize concepts that are visible in the shot, but not necessarily in a single key frame, more robustly. We summarize our spatio-temporal sampling approach in Figure 3.

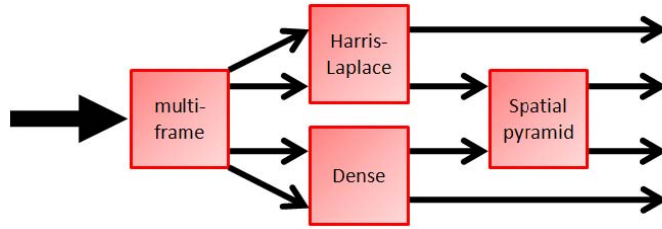


Figure 3: General scheme for spatio-temporal sampling of image regions, including temporal multi-frame selection, Harris Laplace and dense point selection, and a spatial pyramid, using the conventions of Figure 2.

Temporal multi-frame selection: We demonstrated in [25] that a concept detection method that considers more visual content obtains higher performance over key frame-based methods. This can be explained by the fact that the content of a shot changes due to object and camera motion, and imperfect shot segmentation results. Therefore, we employ a multi-frame sampling strategy. To be precise, we sample a maximum of 4 additional frames distributed around the (middle) key frame of each shot.

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 [29]. 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.

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 [6,11]. 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.* [12] 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. Reported results using concept detection experiments are not yet conclusive in the ideal spatial pyramid configuration, some claim 2x2 is sufficient [12], others suggest to include 1x3 also [16]. 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 in a video on the spatio-temporal viewpoint under which it is recorded. However, the lighting conditions during filming also play an important role. Burghouts and Geusebroek [4] analyzed the properties of color features under classes of illumination and viewing changes, such as viewpoint changes, light intensity changes, light direction changes, and light color changes. Van de Sande *et al.* [30] analyzed the properties of color features under classes of illumination changes within the diagonal model of illumination change, and specifically for data sets as considered within TRECVID. Another comparison of our invariant visual features, emphasizing discriminatory power, and efficiency of the feature representation is presented by Van Gemert *et al.* [33]. Here we summarize their main findings. We present an overview of the visual features used in Figure 4.

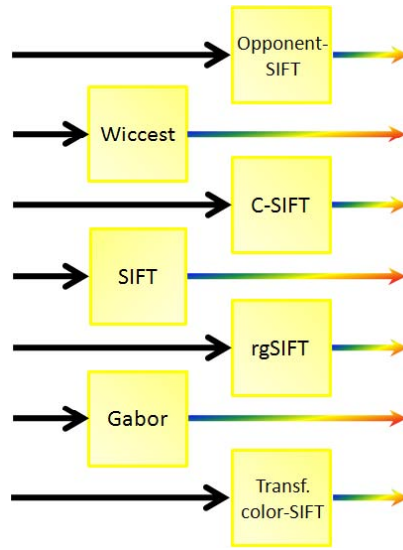


Figure 4: General scheme of the visual feature extraction methods used.

Wiccest: Wiccest features [7] utilize natural image statistics to effectively model texture information. Texture is described by the distribution of edges in a certain image. Hence, a histogram of a Gaussian derivative filter is used to represent the edge statistics. It was shown in [9] that the complete range of image statistics in natural textures can be well modelled with an integrated Weibull distribution. In effect, reducing a histogram to just two Weibull parameters, see [33]. The Wiccest features for an image region consist of the Weibull parameters for the color invariant edges in the region. Thus, the 2 Weibull parameters for the x -edges and y -edges of the three color channels yield a 12 dimensional feature.

Gabor: Gabor filters may be used to measure perceptual surface texture in an image [3]. Specifically, Gabor filters respond to regular patterns in a given orientation on a given scale and frequency, see [33]. In order to obtain an image region feature with Gabor filters we follow these three steps: 1) parameterize the Gabor filters 2) incorporate color invariance and 3) construct a histogram. First, the parameters of a Gabor filter consist of orientation, scale and frequency. We use four orientations, 0° , 45° , 90° , 135° , and two (scale, frequency) pairs: (2.828, 0.720), (1.414, 2.094). Second, color responses are

measured by filtering each color channel with a Gabor filter. The W color invariant is obtained by normalizing each Gabor filtered color channel by the intensity. Finally, a histogram of 101 bins is constructed for each Gabor filtered color channel.

SIFT: The SIFT feature proposed by Lowe [15] 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 [30]. 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, the version described by Lowe [15] is used.

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 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: In the opponent color space, the O_1 and O_2 channels still contain some intensity information. To add invariance to shadow and shading effects, we have proposed the C-invariant [8] which eliminates the remaining intensity information from these channels. The C-SIFT feature uses the C invariant, which can be intuitively seen as the gradient (or derivatives) for the normalized opponent color space $O1/I$ and $O2/I$. The I intensity channel remains unchanged. CSIFT is known to be scale-invariant with respect to light intensity. Due to the local comparison of colors, as effective due to the gradient, the color component of the feature is robust to light color changes. See [4, 30] for detailed evaluation.

rgSIFT: For the *rgSIFT* feature, features are added for the r and g chromaticity components of the normalized RGB color model, which is already scale-invariant [30]. In

addition to the r and g channel, this feature also includes intensity. Because the SIFT feature uses derivatives of the input channels, the rg SIFT feature becomes shift-invariant as well. However, the color part of the feature is not invariant to changes in illumination color.

Transformed color: SIFT For the transformed color SIFT, we normalize each RGB channel independently [30]. For every normalized channel, the SIFT feature is computed. The feature is scale-invariant, shift-invariant and invariant to light color changes and shift.

We compute the Wiccest and Gabor features on densely sampled image regions [33], the SIFT [15] and ColorSIFT [30] features are computed around salient points obtained from the Harris-Laplace detector and dense sampling. For all visual features we take several spatial pyramid configurations into account.

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. [11, 13, 21, 30, 31, 33]. First, we assign visual features to discrete codewords predefined in a codebook. Then, the frequency distribution of the codewords is used 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 [31, 33]. Here we detail codebook construction using clustering and codeword representation using hard-and soft-assignment, following the scheme in Figure 5.

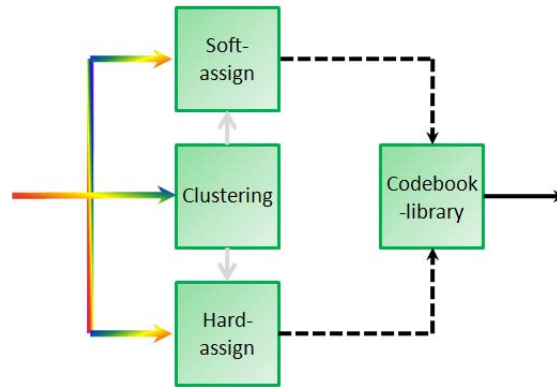


Figure 5: General scheme for transforming visual features into a codebook, where we distinguish between codebook constructions using clustering and codeword assignment using soft and hard variants. We combine various codeword frequency distributions into a codebook library.

Clustering: We employ two clustering methods: k -means and radius-based 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, biasing frequently occurring features. To overcome the limitation of k -means clustering, while maintaining efficiency, Jurie and Triggs [11] proposed radius-based clustering. The algorithm assigns visual features to the first cluster lying within a fixed radius of similarity r . Hence, the radius determines whether two visual features describe the same codeword. As an implementation of radius-based clustering we use Astrahans algorithm, see [33]. For both k -means and radius-based clustering 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 codebook, i.e. hard-assignment. Basically, an image is represented by a histogram of codeword frequencies that describes the probability density over codewords.

Soft-assignment: In a recent paper [31], we show 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* empirically due to its performance [31].

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 *rgSIFT* features in combination with *k*-means clustering and 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. We investigate the role of its parameters and how to select the optimal configuration for a concept, as detailed in Figure 6.

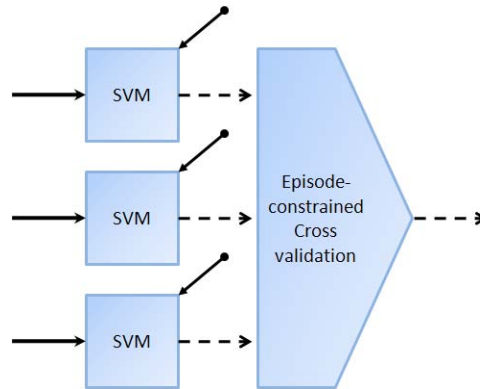


Figure 6: General scheme for kernel-based learning using support vector machines and episode-constrained cross-validation for parameters selection.

Support vector machine: Similar to previous years, we use the support vector machine framework [34] for supervised learning of semantic concepts. Here we use the LIBSVM implementation [5] with probabilistic output [14, 19]. It is well known that the parameters of the support vector machine algorithm have a significant influence on concept detection performance [1, 17, 26, 35]. The only parameters of the support vector machine we optimize are C and the kernel function $K(\cdot)$. 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. While the radial basis kernel function usually perform better than other kernels, it was recently shown by Zhang et al. [36] that in a codebook-approach to concept detection the earth movers distance [20] and χ^2 kernel are to be preferred. In general, we obtain good parameter settings for a support vector machine, by using an iterative search on both C and $K(\cdot)$.

Episode-constrained cross-validation: From all parameters q we select the combination that yields the best average precision performance, yielding q^* . We measure performance of all parameter combinations and select the combination that yields the best performance. We use a 3-fold cross validation to prevent over-fitting of parameters. Rather than using regular cross-validation for support vector machine parameter

optimization, we employ an *episode-constrained* cross-validation method, as this method is known to yield a less biased estimate of classifier performance [32].

The result of the parameter search over q is the improved model $p(\omega_j | x_i, q^*)$, contracted to $p^*(\omega_j | x_i)$, which we use to fuse and to rank concept detection results.

2.5. Submitted Concept Detection Results

We investigated the contribution of each component discussed in Sections 2.1–2.4, emphasizing in particular the role of sampling, the value of color invariance, the influence of codebook construction, and the effectiveness of kernel-based learning parameters. In our experimental setup we used the TRECVID 2007 development set as a training set, and the TRECVID 2007 test set as a validation set. The ground truth used for learning and evaluation are a combination of the common annotation effort [2] and the ground truth provided by ICT-CAS [28]. The positive examples from both efforts were combined using an OR operation and subsequently verified manually. Based on our extensive experiments (data not shown) we arrived at the conclusion that a codebook library employing dense sampling and Harris-Laplace salient points in combination with a spatial pyramid, one of the three following (color) SIFT features: SIFT, OpponentSIFT and transformed color SIFT, and a codebook representation based on k -means clustering and soft-assignment, is a powerful baseline for concept detection in video. This codebook library, consisting of 6 books in total, is our baseline. The baseline was not submitted for evaluation in the high-level feature extraction task, but post-TRECVID experiments indicate it would have received a mean infAP of 0.152. It was, however, the basis of all our TRECVID 2008 submissions. An overview of our submitted concept detection runs is depicted in Figure 7, and detailed next.

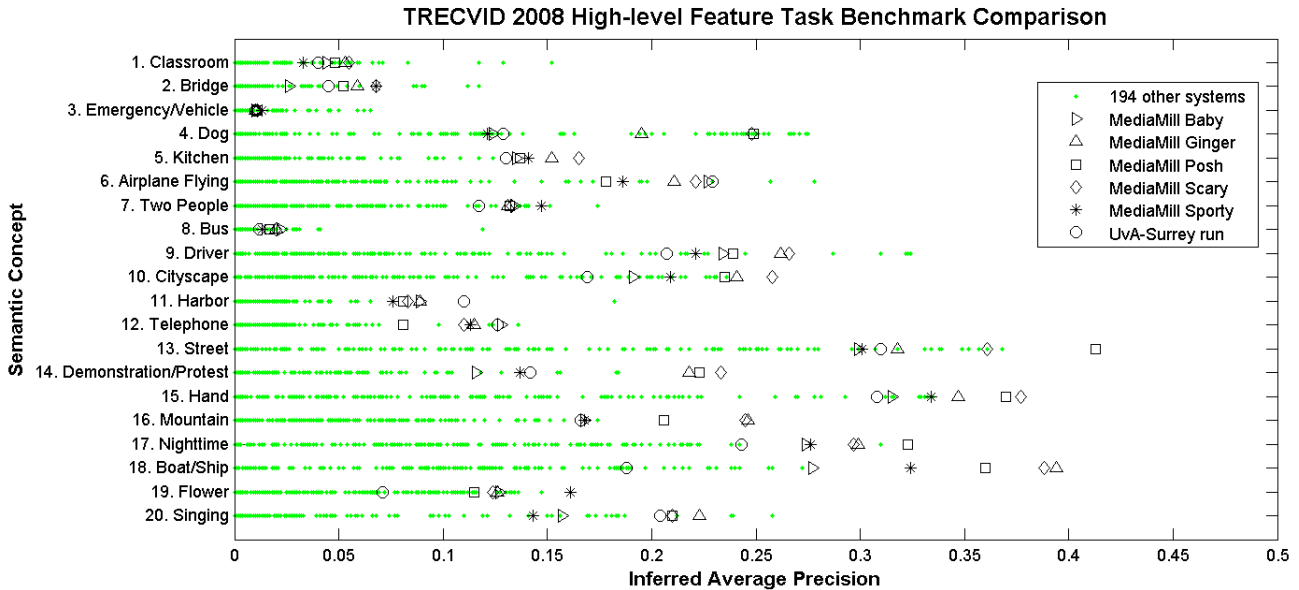


Figure 7: Comparison of MediaMill video concept detection experiments with present-day concept detection approaches in the TRECVID 2008 High-level Feature Task benchmark.

Baby run: The Baby run extends upon the baseline run by also including codebooks for the *rgSIFT* and *C-SIFT* features. This results in a codebook library of 10 books. This run achieved a mean infAP of 0.155. Indeed, only a small improvement over our baseline.

Sporty run: The codebook library used in the Sporty run extends upon the Baby run by also including the *Wiccest* and *Gabor* features, and their early fusion. We apply the standard sequential forward selection feature selection method [10] on this large codebook library. This run achieves the overall highest infAP for the concept *Flower*, and has a mean infAP of 0.159.

UvA-Surrey run: This run is a cooperation between the University of Amsterdam and the University of Surrey. It uses multiple kernel learning [27] on the codebook library of the Baby run together with another codebook library based on *SIFT* only. The weights of the kernels, i.e., the relative importance of the 2 codebook libraries, are learnt from the training data. It achieved a mean infAP of 0.148.

Ginger run: The Ginger run extends the codebook library of 6 books from the baseline run to the temporal domain. For every shot, up to 5 frames are processed, and the results are averaged. This run achieves the overall highest infAP for the concepts *Mountain* and *Boat/Ship*, and has a mean infAP of 0.185.

Posh run: The Posh run is based on a codebook library in a temporal setting. The codebooks to use per concept were chosen on the basis of hold-out performance on the validation set. There were 3 sets of codebooks to choose from, together with the method for temporal aggregation, which could be either the average or the maximum concept likelihood. This run achieves the overall highest infAP for the *Street* and *Nighttime* concepts, and has a mean infAP of 0.184.

Scary run: The Scary run applies the standard sequential forward selection feature selection method on several codebook libraries, all of which have been applied spatiotemporally to up to 5 frames per shot. This run achieved the overall highest mean infAP in the TRECVID2008 benchmark (0.194), with the overall highest infAP for 4 concepts: *Kitchen*, *Cityscape*, *Demonstration or protest*, and *Hand*.

3. Detecting Concepts in Images

Similar to our approach to concept detection in video, we perceive concept detection in images as a combined computer vision and machine learning problem.

3.1. *Visual Feature Extraction*

Our concept detection approach for the PASCAL VOC Challenge 2008 [37] builds upon the same visual feature extraction methods as our TRECVID 2008 approach. We will list the methods used for spatial sampling and visual feature extraction:

Spatial Sampling (see also section 2.1):

- Harris-Laplace point detector
- Dense point detector
- Spatial pyramid weighting using pyramids of 1x1, 2x2 and 1x3

Visual Feature Extraction (see also section 2.2):

- SIFT
- OpponentSIFT
- C-SIFT
- *rg*SIFT
- Transformed Color SIFT

For the codebook transform, we employ only the *k*-means algorithm with soft assignment, due to its experimental performance on the TRECVID 2008 benchmark. Similar to our approach in that benchmark, we again use a codebook library which collects all possible

codebook combinations using various configurations of spatial sampling and visual feature extraction.

3.2. Machine Learning

The machine learning part of our concept detection approach for the PASCAL VOC Challenge 2008 uses two machine learning algorithms. The first is the support vector machine framework (see section 2.4), also employed in TRECVID 2008. The second algorithm is Spectral Regression Kernel Discriminant Analysis (SRKDA) proposed by Deng *et al* [38].

SRKDA is one of the many approaches that uses kernel-based non-linear extensions of Linear Discriminant Analysis (LDA). This approach maps discriminant analysis into a regression framework and thus avoids eigen-decomposition of the kernel-matrix. That results in saving huge computation cost. It has been shown [38] that SRKDA is 27 times faster than traditional KDA and also error rates are better than state-of-art approaches such as the support vector machine framework.

3.3. Submitted Concept Detection Results

We investigated the contribution of each component discussed in Sections 3.1 and 3.2, emphasizing in particular the role of sampling, the value of color invariance and the effectiveness of different machine learning algorithms. The PASCAL VOC Challenge 2008 provides an experimental setup with a predefined training set, validation set and ground truth. Based on our extensive experiments (data not shown) we arrived at the conclusion that a codebook library employing dense sampling and Harris-Laplace salient points in combination with a spatial pyramid, the five (color) SIFT features, and a support



vector machine classifier, is a powerful baseline for concept detection in photo collections.

The results of our submission to the PASCAL VOC Challenge 2008 are shown in figure 8. The submitted runs are discussed next.

Soft5ColorSIFT run: This run is the baseline mentioned above. This run obtained the best overall results in the competition for the *potted plant* and *TV/monitor* concepts.

TreeSFS run: This run combines the baseline with a random forest approach to codebook transform. This run obtained the best overall results in the competition for the *bird*, *horse*, *person* and *sofa* concepts.

SRKDA run: This run uses the same 30 codebooks from the codebook library as the Soft5ColorSIFT run. However, instead of learning a single SVM classifier, it learns 30 simpler classifiers using the SRKDA algorithm. This gives a posterior probability for each codebook configuration. These probabilities are then simply averaged. This run obtains the highest overall precision of all entries into the PASCAL VOC 2008 benchmark. It also obtained the highest performance for the following 8 concepts: *bicycle*, *bottle*, *bus*, *car*, *chair*, *cow*, *sheep* and *train*.

Object Category	SurreyUvA_SRKDA	UvA_Soft5ColorSift	UvA_TreeSFS
Aeroplane	79,5	79,7	80,8
Bicycle	54,3	52,1	53,2
Bird	61,4	61,5	61,6
Boat	64,8	65,5	65,6
Bottle	30,0	29,1	29,4
Bus	52,1	46,5	49,9
Car	59,5	58,3	58,5
Cat	59,4	57,4	59,4
Chair	48,9	48,2	48,0
Cow	33,6	27,9	30,1
Dining table	37,8	38,3	39,6
Dog	46,0	46,6	45,0
Horse	66,1	66,0	67,3
Motorbike	64,0	60,6	60,4
Person	86,8	87,0	87,1
Potted plant	29,2	31,8	30,1
Sheep	42,3	42,2	41,5
Sofa	44,0	45,3	45,4
Train	77,8	72,3	74,3
TV/Monitor	61,2	64,7	59,8
MAP	54,9	54,1	54,4

Figure 8: Comparison of image concept detection experiments with present-day concept detection approaches in the VOC2008 Challenge. Yellow marks a concept detection result equal to the best entry into the VOC competition.

References

- [1] A. Amir, M. Berg, S.-F. Chang, W. Hsu, G. Iyengar, C.-Y. Lin, M. R. Naphade, A. P. Natsev, C. Neti, H. J. Nock, J. R. Smith, B. L. Tseng, Y. Wu, and D. Zhang. IBM research TRECVID-2003 video retrieval system. In *Proc. TRECVID Workshop*, NIST Special Publication, Gaithersburg, USA, 2003.
- [2] S. Ayache and G. Qu'énoc. Video corpus annotation using active learning. In *European Conference on Information Retrieval*, pages 187–198, Glasgow, Scotland, 2008.
- [3] A. C. Bovik, M. Clark, and W. S. Geisler. Multichannel texture analysis using localized spatial filters. *IEEE Trans. PAMI*, 12(1):55–73, 1990.
- [4] G. J. Burghouts and J. M. Geusebroek. Performance evaluation of local color invariants. *Computer Vision and Image Understanding*.
- [5] 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/>.
- [6] L. Fei-Fei and P. Perona. A bayesian hierarchical model for learning natural scene categories. In *Proc. IEEE CVPR*, pages 524–531, 2005.
- [7] J.-M. Geusebroek. Compact object descriptors from local colour invariant histograms. In *British Machine Vision Conference*, Edinburgh, UK, 2006.
- [8] J. M. Geusebroek, R. Boomgaard, A. W. M. Smeulders, and H. Geerts. Color invariance. *IEEE Trans. PAMI*, 23(12):1338–1350, 2001.
- [9] J. M. Geusebroek and A. W. M. Smeulders. A six-stimulus theory for stochastic texture. *IJCV*, 62(1/2):7–16, 2005.
- [10] A. K. Jain, R. P. W. Duin, and J. Mao. Statistical pattern recognition: A review. *IEEE Trans. PAMI*, 22(1):4–37, 2000.

- [11] F. Jurie and B. Triggs. Creating efficient codebooks for visual recognition. In *Proc. IEEE ICCV*, pages 604–610, 2005.
- [12] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Proc. IEEE CVPR*, volume 2, pages 2169–2178, New York, USA, 2006.
- [13] T. Leung and J. Malik. Representing and recognizing the visual appearance of materials using three-dimensional textons. *IJCV*, 43(1):29–44, 2001.
- [14] 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.
- [15] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60:91–110, 2004.
- [16] M. Marszałek, C. Schmid, H. Harzallah, and J. van de Weijer. Learning object representations for visual object class recognition, October 2007. Visual Recognition Challenge workshop, in conjunction with ICCV.
- [17] M. R. Naphade. On supervision and statistical learning for semantic multimedia analysis. *Journal of Visual Communication and Image Representation*, 15(3):348–369, 2004.
- [18] C. Petersohn. Fraunhofer HHI at TRECVID 2004: Shot boundary detection system. In *Proc. TRECVID Workshop*, NIST Special Publication, Gaithersburg, USA, 2004.
- [19] 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.
- [20] Y. Rubner, C. Tomasi, and L. J. Guibas. The earth mover’s distance as a metric for image retrieval. *IJCV*, 40(2):99–121, 2000.
- [21] J. Sivic and A. Zisserman. Efficient visual search for objects in videos. *Proc. IEEE*, 96(4):548–566, 2008.

- [22] C. G. M. Snoek, I. Everts, J. C. van Gemert, J.-M. Geusebroek, B. Huurnink, D. C. Koelma, M. van Liempt, O. de Rooij, K. E. A. van de Sande, A. W. M. Smeulders, J. R. R. Uijlings, and M. Worring. The MediaMill TRECVID 2007 semantic video search engine, 2007.
- [23] C. G. M. Snoek, J. C. van Gemert, T. Gevers, B. Huurnink, D. C. Koelma, M. van Liempt, O. de Rooij, K. E. A. van de Sande, F. J. Seinstra, A. W. M. Smeulders, A. H. C. Thean, C. J. Veenman, and M. Worring. The MediaMill TRECVID 2006 semantic video search engine, 2006.
- [24] C. G. M. Snoek and M. Worring. Concept-based video retrieval. *Foundations and Trends in Information Retrieval*, 2009. submitted.
- [25] C. G. M. Snoek, M. Worring, J.-M. Geusebroek, D. C. Koelma, and F. J. Seinstra. On the surplus value of semantic video analysis beyond the key frame. In *Proc. IEEE ICME*, Amsterdam, The Netherlands, 2005.
- [26] C. G. M. Snoek, M. Worring, J.-M. Geusebroek, D. C. Koelma, F. J. Seinstra, and A. W. M. Smeulders. The semantic pathfinder: Using an authoring metaphor for generic multimedia indexing. *IEEE Trans. PAMI*, 28(10):1678–1689, 2006.
- [27] S. Sonnenburg, G. Rˆatsch, C. Schˆafer, and B. Schˆolkopf. Large scale multiple kernel learning. *Journal of Machine Learning Research*, 7:1531–1565, 2006.
- [28] S. Tang et al. TRECVID 2008 high-level feature extraction by MCG-ICT-CAS. In *Proc. TRECVID Workshop*, NIST Special Publication, Gaithersburg, USA, 2008.
- [29] T. Tuytelaars and K. Mikolajczyk. Local invariant feature detectors: A survey. *Foundations and Trends in Computer Graphics and Vision*, 3(3):177–280, 2008.
- [30] K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek. Evaluation of color descriptors for object and scene recognition. In *Proc. IEEE CVPR*, Anchorage, Alaska, 2008.
- [31] J. C. van Gemert, J. M. Geusebroek, C. J. Veenman, and A. W. M. Smeulders. Kernel codebooks for scene categorization. In *European Conference on Computer Vision*, Marseille, France, 2008.

- [32] J. C. van Gemert, C. G. M. Snoek, C. Veenman, and A. W. M. Smeulders. The influence of cross-validation on video classification performance. In *Proc. ACM Multimedia*, pages 695–698, Santa Barbara, USA, 2006.
- [33] J. C. van Gemert, C. G. M. Snoek, C. J. Veenman, A. W. M. Smeulders, and J.-M. Geusebroek. Comparing compact codebooks for visual categorization. *Computer Vision and Image Understanding*, 2009. Submitted.
- [34] V. N. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag, New York, USA, 2nd edition, 2000.
- [35] D. Wang, X. Liu, L. Luo, J. Li, and B. Zhang. Video Diver: generic video indexing with diverse features. In *Proc. ACM SIGMM MIR Workshop*, pages 61–70, Augsburg, Germany, 2007.
- [36] J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid. Local features and kernels for classification of texture and object categories: A comprehensive study. *IJCV*, 73(2):213–238, 2007.
- [37] M. Everingham, L. Van-Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2008 (VOC2008) Results. <http://www.pascal-network.org/challenges/VOC/voc2008/workshop/>
- [38] D. Cai, X. He and J. Han. Efficient Kernel Discriminant Analysis via Spectral Regression. International Conference on Data Mining, 2007