The MediaMill TRECVID 2012 semantic video search engine

Snoek, C.G.M.; van de Sande, K.E.A.; Habibian, A.; Kordumova, S.; Li, Z.; Mazloom, M.; Pintea, S.L.; Tao, R.; Koelma, D.C.; Smeulders, A.W.M.

Publication date
2012

Document Version
Final published version

The MediaMill TRECVID 2012 Semantic Video Search Engine

C.G.M. Snoek, K.E.A. van de Sande, A. Habibian, S. Kordumova, Z. Li, M. Mazloom,
S.L. Pintea, R. Tao, D.C. Koelma, A.W.M. Smeulders
ISLA, University of Amsterdam
Amsterdam, The Netherlands
http://www.mediamill.nl

Abstract

In this paper we describe our TRECVID 2012 video retrieval experiments. The MediaMill team participated in four tasks: semantic indexing, multimedia event detection, multimedia event recounting and instance search. The starting point for the MediaMill detection approach is our top-performing bag-of-words system of TRECVID 2008-2011, which uses multiple color SIFT descriptors, averaged and difference coded into codebooks with spatial pyramids, and kernel-based machine learning. This year our concept detection experiments focus on establishing the influence of difference coding, the use of audio features, concept-pair detection using regular concepts, pair detection by spatiotemporal objects, and concept(-pair) detection without annotations. Our event detection and recounting experiments focus on representations using concept detectors. For instance search we study the influence of spatial verification and color invariance. The 2012 edition of the TRECVID benchmark has again been a fruitful participation for the MediaMill team, resulting in the runner-up ranking for concept detection in the semantic indexing task.

1 Introduction

Robust video retrieval is highly relevant in a world that is adapting swiftly to visual communication. Online services like YouTube and Vimeo show that video is no longer the domain of broadcast television only. Video has become the medium of choice for many people communicating via the Internet. Most commercial video search engines provide access to video based on text, 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, surrounding text, social tagging, closed captions, or a speech transcript. This results in disappointing retrieval performance when the visual content is not mentioned, or properly reflected in the associated text. In addition, when the videos originate from non-English speaking countries, such as China, or the Netherlands, querying the content becomes much harder as robust automatic speech recognition results and their accurate machine translations are difficult to achieve.

To cater for robust video retrieval, the promising solutions from the literature are mostly semantic [23, 27], where detectors are related to objects, like a *flag*, scenes, like a *beach*, people, like *female human face closeup*, and events like *landing a fish in*. Any one of those brings an understanding of the current content. The elements in such a lexicon of detectors offer users a semantic entry to video by allowing them to query on presence or absence of visual content elements. Last year we presented the *MediaMill 2011* semantic video search engine [26], which for the first time included event detection. This year, the MediaMill team participated in four tasks: semantic indexing, multimedia event detection, multimedia event recounting and instance search. Our semantic indexing experiments focus on establishing the influence of difference coding, the use of audio features, concept-pair detection using regular concepts, pair detection by spatiotemporal objects, and concept(-pair) detection without annotations. Our event detection and recounting experiments focus on representations using concept detectors [10, 19]. For instance search, we study the influence of spatial verification and color invariance. Taken together, the *MediaMill 2012* semantic video search engine provides users with robust semantic access to Internet video collections.

The remainder of the paper is organized as follows. We first define our bag-of-words foundation in Section 2. Then we highlight our detection approaches for concept(-pair)s in Section 3 We turn our attention to complex event detection in Section 4. Recounting video events is the topic of Section 5. Finally, we detail our instance search experiments in Section 6.

2 Bag-of-Words Foundation

Our TRECVID 2012 concept and event detection builds on previous editions of the MediaMill semantic video search engine [24, 26, 30, 32], which draws inspiration from the bag-of-words propagated by Schmid and her associates [13, 18, 40], as well as keypoint-based color descriptors [33], difference encoding [11, 20, 41], soft codebook representations [36, 38], and efficient algorithmic refinements [17, 31], a GPU implementation [34], and compute clusters.

2.1 Spatio-Temporal Sampling

The visual appearance of a semantic concept, an event or an instance in video has a strong dependency on the spatio-temporal viewpoint under which it is recorded. Salient point methods [29] 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. Appearance variations caused by temporal effects are addressed by analyzing video beyond the key frame level. By taking more frames into account during analysis, it becomes possible to recognize concepts that are visible during the shot, but not necessarily in a single key frame.

Temporal multi-frame selection In [25, 28] we demonstrated that a concept detection method that considers more video content obtains higher performance over key frame-based methods. We attribute this to the fact that the content of a shot changes due to object motion, camera motion, and imperfect shot segmentation results. Therefore, we employ a multi-frame sampling strategy for concept and event detection.

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 [7,12]. 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.* [13] 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. Similar to [18, 25] we use a spatial pyramid of 1x1 and 1x3 regions in our experiments.

2.2 Visual Descriptors

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 they are 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.* [33] 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.

SIFT The SIFT feature proposed by Lowe [16] 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 [33]. 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 [16].

TSIFT TSIFT is an unpublished color descriptor.

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 [9] 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 derivative) for the normalized opponent color space O_1/I and O_2/I. The I intensity channel remains unchanged. C-SIFT is known to be scale-invariant with respect to light intensity.

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 [33]. The feature is scale-invariant, shift-invariant, and invariant to light color changes and shift.

We compute the SIFT [16] and ColorSIFT [33] features around salient points obtained from the Harris-Laplace detector and dense sampling. All descriptors are then reduced to 80 dimensions with PCA.

2.3 Audio Descriptors

As low-level audio features, we extract Mel-frequency cepstral coefficients (MFCCs) over a 10ms window using CMU's Sphinx [1]. MFCCs are widely used in speech recognition: they describe the spectral shape of audio. The derivatives of the MFCCs (δ MFCC) and the second derivative ($\delta\delta$ MFCC) are also computed.

2.4 Word Encoding

To avoid using all low-level visual and audio features from a video, we follow the well known codebook approach.

Hard coding For both the visual and audio features, we first assign them to discrete codewords from a predefined codebook. Then, we use the frequency distribution of the codewords as a compact feature vector representing an image frame or audio window. By using a vectorized GPU implementation [34], our codebook transform process is an order of magnitude faster for the most expensive feature compared to the standard implementation. Two important variables in the codebook representation are *codebook construction* and *codeword assignment*. Based on previous experiments, balancing accuracy and performance, we employ codebook construction using k-means clustering in combination with hard codeword assignment and a maximum of 4,096 codewords.

Difference coding In is well known that the traditional hard-assignment may be improved by using soft-assignment through kernel codebooks [38]. A kernel codebook uses a kernel function to smooth the hard-assignment of (image) features to codewords by assign descriptors to multiple clusters, weighted by their distance to the center. Recently, many improved codeword assignment approaches have been proposed [11, 20, 41]. We employ difference coding. To be precise, we follow both the Fisher vector [20] and VLAD [11] schemes. The former is based on a Gaussian Mixture Model of the PCA-reduced descriptor space with 256 components, the latter is based on a k-means clustering of the PCA-reduced descriptor space with 1024 components. We also employ difference coding with Fisher vectors on the MFCCs.

The output of the word encoding is a bag-of-words vector, which forms the foundation for both concept detection and event detection.

3 Detecting Concepts in Video

Learning robust concept detectors from audiovisual features is typically achieved by kernel-based learning methods. Similar to previous years, we rely predominantly on the support vector machine framework [39] for supervised learning of semantic concepts. Here we use the LIBSVM implementation [5] with probabilistic output [15,21]. 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. We use the Histogram Intersection kernel and its efficient approximation as suggested by Maji *et al*. [17]. For difference coded bag-of-words we use a linear kernel [20, 41].

In general, we obtain good parameter settings for a support vector machine, by using an iterative search on both C and kernel function $K(\cdot)$ on cross validation data [37]. 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 concept detection performance [37].

3.1 Concept-Pair Detection

The new concept-pair detection task requires the detection of rare combinations of concepts in which the context is no longer informative (e.g. *animal & snow*, *dog & indoor*). We did a first attempt for spatiotemporal concept detection. Our method for concept-pair detection employs three main steps: (i) localized detection-by-tracking; (ii) BOW-features; and (iii) temporal fusion over frames.

Feature extraction: In our tradition of single concept detection, we extract RGB-SIFT descriptors densely at each frame and perform codebook projection. Undoubtedly, more features can be used, but given the dimension of the test data we restrict our attention for the moment to this color descriptor. For codebook construction and projection we follow the work of [30] by using Random Forests and a spatial pyramid.

Hierarchical segmentation: We perform object localization using our detector which has won the PASCAL VOC 2012 object detection task [35]. For the first frame only we perform hierarchical image segmentation using the method from [8] for retaining the candidate boxes. We obtain the initial over-segmented image, that is used for building the hierarchy, from [35]. We run an SVM classifier on each candidate box and retain only the best box for each class in the same manner as in [30].

Detection by tracking: In the subsequent frames we perform the first step — RGB-SIFT descriptor extraction and codebook projection over the complete image. We continue by tracking each of the previously detected best boxes. For each tracked best box we re-detect its content by running the corresponding classifier over the features extracted from it.

Temporal fusion: In the end, for each frame and each class of interest we have estimated detection scores. For the final ranking we need detection scores over the complete shot. In this step we perform $\mathbf{M}_k = (\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \boldsymbol{\pi}_k, \mathbf{T}_k)$. For each test shot we estimate all the class likelihoods — $l_k = p(\mathbf{X}|\mathbf{M}_k), \forall k \in \{1, ..9\}$. These scores are subsequently whitened and used for ranking.

We use the above system for the 9 localizable objects out of 20: *animal*, *bicycle*, *bird*, *car*, *dog*, *flag*, *person*, *table*, *telephone*. For the remaining concepts we use our regular concept detection.

The localized detection models are trained on Pascal-VOC 2007 data annotated with bounding boxes and another 600 frames per class, hand annotated from the TRECVID

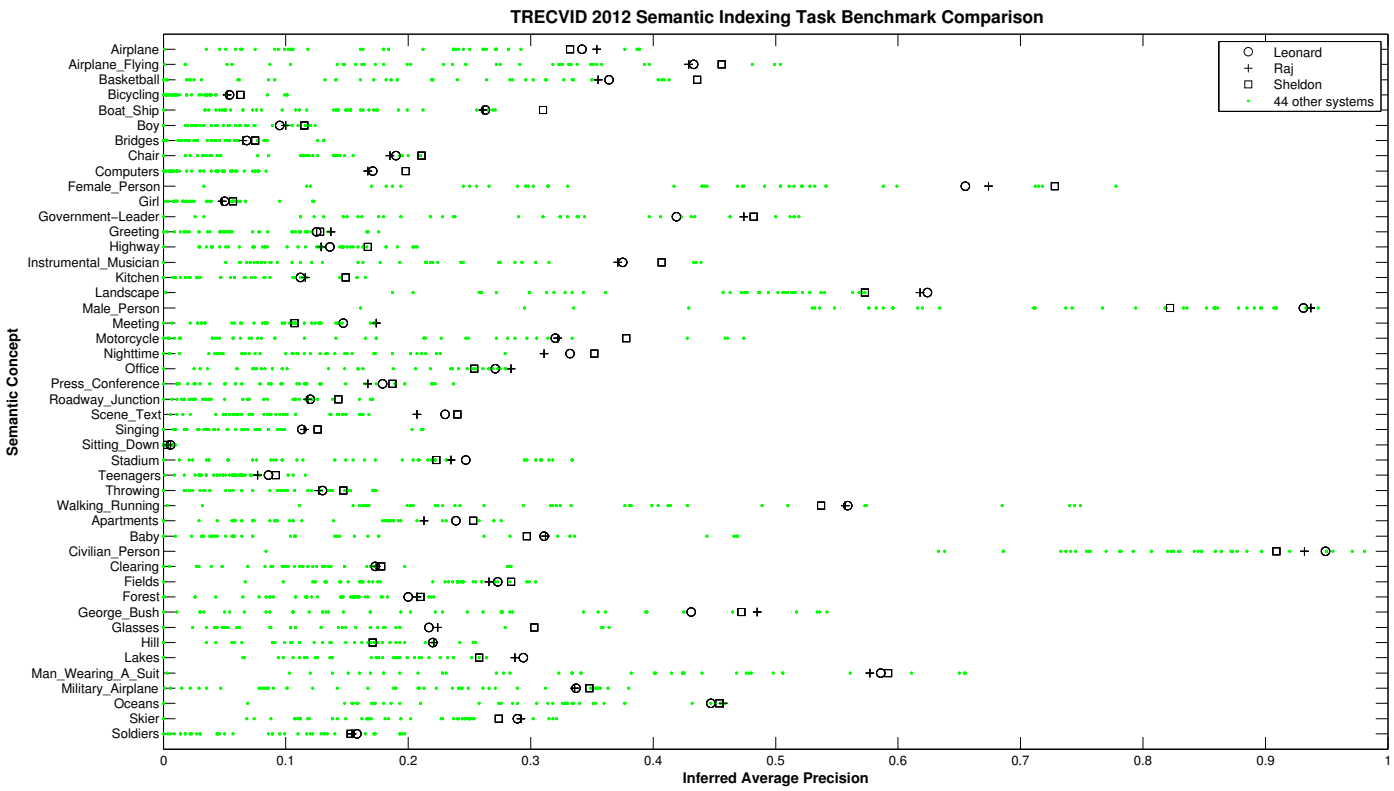

Figure 1: Comparison of MediaMill video concept detection experiments with other concept detection approaches in the TRECVID 2012 Semantic Indexing task.

training data for this task. For training the temporal models we use 600 shots per class from the TRECVID SIN task training data. We train a different model for each class and each possible sequence length (2 to 17 frames) and at test time we apply the model corresponding to the input sequence length. We use 9 hidden states per time step: $\mathbf{C}_t = (C_t^1, C_t^2, ..C_t^9)$ because we have 9 true classes which can be present in each frame.

At test time, we use a number of 5 frames around the central frame if the video is shorter than 100 frames and 17 frames otherwise. We also consider shots shorter than 5 frames. Selecting frames around the central frame is a viable choice because a large number of shots start and end in a camera-off state (contain black frames at the extremities of the shot). As a final concept-pair score we return the sum in log space of the two detection scores — the above method returns log-likelihoods and we take the log over the detection probabilities of the regular concept detectors.

3.2 Learning from web video

We experimented with concept detection without using the provided expert annotations. Given a query concept, we automatically download videos from YouTube, by sending queries with the concept name.

The downloaded videos serve as a raw training material and prevent us from dependence on any manual supervision during training. All harvested videos are shot segmented and the social tags associated to the videos are propagated to the middle keyframe of each shot. We select the important frames from the video using an unpublished algorithm. Once we have selected the important video frames per concept, we extract features and train concept models.

3.3 Submitted Concept Detection Results

Our experiments [3, 22] focus on establishing the influence of difference coding for concept detection, the use of audio features, concept-pair detection using regular concepts, pair detection by spatiotemporal objects, and concept(-pair) detection without annotations.

3.3.1 Semantic indexing task

An overview of our submitted concept detection runs is depicted in Figure 1.

Run: Leonard The *Leonard* run is our baseline. It is based on SIFT, TSIFT, and C-SIFT descriptors computed for a maximum of 10 frames per shot, each at least 4 frames apart. The descriptors are quantized using hard-assignment and VLAD difference coding. We learn concept scores using a non-linear SVM with histogram intersection kernel and a linear SVM. Fusion is performed using a simple AVG rule combination, the MAX per shot is the final score. This run

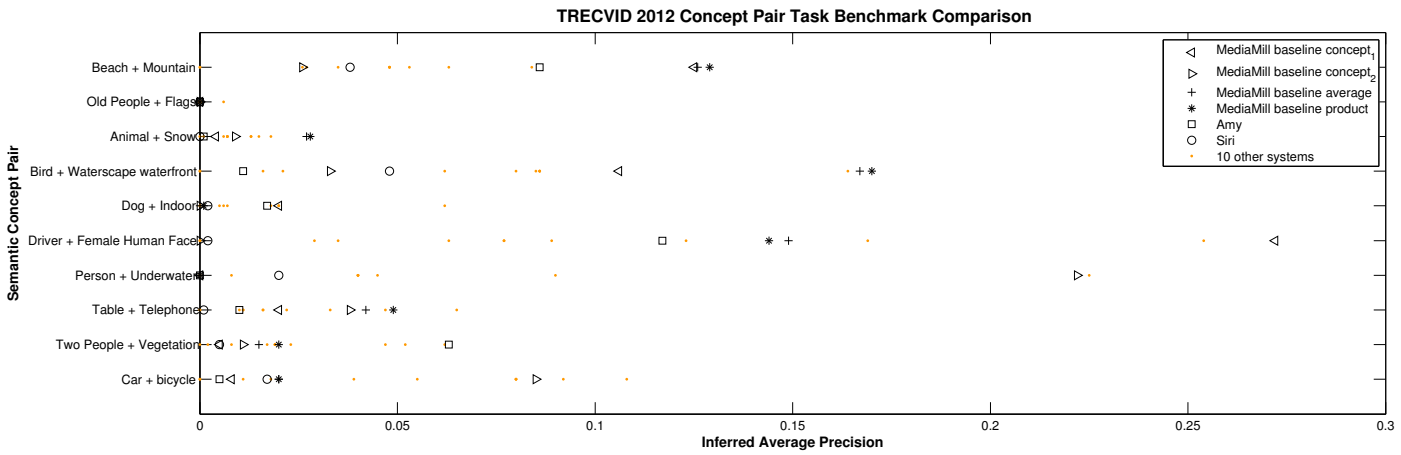

Figure 2: Comparison of MediaMill video concept pair detection experiments, including baselines, with other concept pair detection approaches in the TRECVID 2012 Concept Pair Detection task.

achieved a mean infAP of 0.289, with the overall highest infAP for the concepts: *landscape* and *lakes*.

Run: Raj The *Raj* run adds MFCC audio descriptors to our baseline. During training we consider a video positive if one of its shots is positive, during classification we consider all shots of a positive video as positive also. Fusion is performed using a simple AVG rule combination. This run achieved a mean infAP of 0.289 also, with the overall highest infAP for the concepts: *meeting* and *office*. We conclude that our audiovisual fusion is not optimally exploiting the benefit of audio yet.

Run: Sheldon The *Sheldon* run is based on the same color descriptors as the baseline, but uses Fisher difference coding in combination with a linear SVM only. The classifiers have been applied on a maximum of 10 frames per shot, each at least 4 frames apart. The final score is based on a simple MAX rule. This run achieved a mean infAP of 0.297, with the overall highest infAP for the concepts: *basketball*, *boat_ship*, *chairs*, *computers*, *nighttime*, and *scene_text*. We conclude that Fisher difference coding of color descriptors is more robust than VLAD combined with traditional coding.

3.3.2 No annotation task

We submitted three runs (partly) using training data obtained from YouTube. These are evaluated as part of the light condition of the SIN task.

Run: Bernadette The *Bernadette* run is most selective in obtaining training examples from YouTube. For concept detection it uses the implementation of *Leonard*. As expected it is not competitive compared to expert annotation. However, it outperforms our *Penny* run which uses more, but apparently noisy, YouTube annotations.

Run: Penny The *Penny* run is comparable to *Bernadette* but uses more training data from YouTube.

Run: Howard The *Howard* run is an average fusion of *Leonard*, using provided SIN task labels, and *Bernadette*. This simple fusion is always worse than the baseline.

3.3.3 Concept pair task

We provided pair detection baselines based on single concept detectors and submitted two concept pair detection runs, see Figure 2.

Run: Baselines

- *baseline-firstconcept:* this pair-run is based on a ranking of the first concept only.
- *baseline-secondconcept:* this pair-run is based on a ranking of the second concept only.
- *baseline-combine-sum:* this pair-run is based on a sum of the scores of concept 1 and concept 2.
- *baseline-combine-mul:* this pair-run is based on a product of the scores of concept 1 and concept 2.

The baselines are among the top performing runs, winning in total 4 concept pairs. Using a strategy which simply relies on the first concept of the pair only, results in the best retrieval result for the pair *Driver & Female Human Face*. In contrast, the second concept is more reliable for the pair *Person & Underwater*. As expected, the more rare concept in a pair is most suited for joint-detection. In terms of mean average precision the baselines using sum and product combinations of the individual concept detectors perform almost similar (0.055 vs 0.056). However, the product combination is the best performer for 3 pairs: *Beach & Mountain*, *Animal & Snow* and *Bird & Waterscape/Waterfront*.

Run: Amy Our *Amy* run exploits spatiotemporal pair detection for the pairs having concepts that can be localized. It falls back to the *Leonard* detectors for the other concepts. This run is the best performer for the pair *Two People & Vegetation*, but it should be noted that the detection is based on the detectors without spatiotemporal analysis. We attribute the relatively poor accuracy of the spatiotemporal analysis for pair detection to the lack of training data used for the local object detectors. More training data for localized objects in video is mandatory before spatiotemporal reasoning will be a viable approach.

Run: Siri For this run we created a training dataset by downloading videos from YouTube tagged with both concept names. For concept pairs for witch YouTube did not retrieve enough videos we modified the search query. For example instead of using a query consisting of *Bird* and *Waterscape waterfront* we used *Bird* and *Water*. Following the bi-concept approach [14], we identify the relevant videos and learn a joint detector for concept pairs directly. In addition, we calculate the 'tag informativeness' by measuring how consistent each tag is with the majority of other tags provided for the same video. We used Borda Count to combine the semantic and visual scores into a single ranking. The *Siri* run performs poorly for most pairs, which we attribute to the poor matching between tags and concept pair definitions. However, since the approach does not depend on individual concept detectors nor expensive manual expert annotations, we do consider the use of social tagged video for concept pair detection of interest for future version of this challenging task.

3.4 1,346 Concept Detectors

In addition to the 346 concept detectors from the TRECVID SIN task, we have also employed our *Sheldon* run setting on the entire concept set of the ImageNet Large Scale Visual Recognition Challenge 2011 [6], containing 1,000 object categories. All 1,346 detectors are included in the 2012 MediaMill semantic video search engine.

4 Detecting Events in Video

We participated in the multimedia event detection task. We explore two event representations, one founded on the same bag-of-words used for concept detection, the other based on a representation of concepts [10, 19]. In addition, within the SESAME team [2], we also investigated together with SRI International and the University of Southern California several additional multimedia approaches to video event detection.

4.1 Event as bag-of-words

Our baseline approach to visual event detection is based on the visual bag-of-words discussed in Section 2. Similar to concept detection we rely on the support vector machine framework [39] for supervised learning of events. We use the Histogram Intersection kernel and its efficient approximation as suggested by Maji *et al*. [17]. For difference coded bag-of-words we use a linear kernel [20, 41].

4.2 Event as bag-of-concepts

Our pipeline consists of three consecutive steps: concept detection, semantic video representation and learning the event model. In the concept detection step, we apply a set of predefined concept detectors on the extracted video frames. Each concept detector is a SVM classifier trained on low-level visual features so as to detect a particular concept. The concept detectors are trained on the 346 categories from the TRECVID semantic indexing task and 1,000 categories from ImageNet large scale visual recognition competition data. These 1,346 categories encompass various objects, scenes, people and actions. Each video frame is represented by a vector of 1,346 elements, obtained by applying all the detectors on the frames.

After the concept detection step, we aggregate the detection scores to reach a semantic representation for videos. This representation determines how confidently each concept has been detected throughout the video.

Finally, we create the event detector by training a classifier on the semantic representation of training videos. We use two different classifiers in our system: a non-linear SVM with approximated histogram intersection kernel and a random forest variant.

4.3 Submitted Event Detection Results

Run: LowLevel This run is based on a weighted fusion of event detectors based on color SIFT difference coding, color SIFT average coding and MFCC difference coding.

Run: HighLevel This run is based on semantic representations only and does not contain any low-level modalities. The ranking is obtained by late fusion of two classifiers: a non-linear SVM with approximated HIK kernel and a random forest variant. This is the best performing event detector run in MED2012 based on concept detectors only.

Run: AllLevel This run is the combination of our *LowLevel* and *HighLevel* runs. The output of different modalities are fused by weighted averaging. The weights are determined based on the average precision of each modality using a validation set in a 10-fold cross-validation setting. This is our best performing run, it shows that low-level and high-level event representations complement each other.

5 Recounting Events in Video

Our recounting system learns from the event kit description for each event category the relevant concepts per video clip.

The algorithm selects the concepts out of a lexicon of 1,346. After finding the best set of concepts for each event class, we sort them for each of the categories deemed important for the recounting, e.g., objects, actions, scene and people.

In addition to the selected concepts we report output from automatic speech recognition, video optical character recognition, camera motion and statistics related to the presence of frontal faces, which we obtained from SRI International and the University of Southern California within the SESAME team [2].

5.1 Submitted Event Recounting Results

The results from our participation in the multimedia event recouting task are described in the notebook paper by the SESAME team [2].

6 Searching Instances in Video

Our approach builds on the bag-of-words model. The framework consists of two components: offline indexing and online searching.

Offline indexing For each testing video clip, a set of frames are extracted with a fixed rate (1 frame every 2 seconds in our submissions). Then several types of local descriptors are extracted on each frame and quantized into visual word histograms. Inverted file structure is used to index the whole dataset for efficient online searching.

Online searching Each example frame of the instance is used as an independent query and the similarity score of a testing frame is accumulated. We use histogram intersection to measure the similarity between two frames. The maximum of the frame scores is taken as the video score, based on which a ranked list of videos is returned.

6.1 Submitted Instance Search Results

An overview of our 4 submissions is described as follows.

Run: Baihu The *Baihu* run is our baseline. Three versions of color SIFT features [33] are used, namely RGB-SIFT, OpponentSIFT and CSIFT. A large codebook with 500,000 visual words is constructed per descriptor. This run achieved a mean infAP of 0.093.

Run: Xuanwu The *Xuanwu* run adds a spatial verification step to re-rank the top 100 video clips returned by the *Baihu* run. A global homograph transformation is estimated using RANSAC and the initial result is re-ranked based on the number of inliers consistent with the estimated geometrical relation. This run achieved a mean infAP of 0.088, worse than the baseline, but it improved on 7 query topics.

Run: Zhuque The *Zhuque* run also uses the provided binary masks. In this run, each query is searched twice, one using visual words extracted from the whole query frame and the other only with the visual words inside the mask. The video scores of two trials are combined. This is a doubled version of the baseline. The mean infAP of this run is 0.118.

Run: Qinglong The *Qinglong* run is an extension of the *Zhuque* run by adding color invariance features [9]. This run achieved a mean infAP of 0.124, improving the *Zhuque* run on 7 topics.

Acknowledgments

The authors are grateful to NIST and the TRECVID coordinators for the benchmark organization effort. This research is supported by the STW SEARCHER project, the BeeldCanon project, FES COMMIT, and by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior National Business Center contract number D11PC20067. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoI/NBC, or the U.S. Government.

References

[1] CMU sphinx open source toolkit for speech recognition. `http://cmusphinx.sourceforge.net/`.

[2] M. Akbacak, R. C. Bolles, J. B. Burns, M. Eliot, A. Heller, J. A. Herson, G. K. Myers, R. Nallapati, S. Pancoast, J. van Hout, E. Yeh, A. Habibian, D. C. Koelma, Z. Li, M. Mazloom, S.-L. Pintea, K. E. A. van de Sande, A. W. M. Smeulders, C. G. M. Snoek, S. C. Lee, R. Nevatia, P. Sharma, C. Sun, and R. Trichet. The 2012 SESAME multimedia event detection (MED) system. In *Proceedings TRECVID Workshop*, Gaithersburg, USA, 2012.

[3] S. Ayache and G. Quénot. Video corpus annotation using active learning. In *European Conference on Information Retrieval*, pages 187–198, Glasgow, UK, 2008.

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

[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] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

[7] L. Fei-Fei and P. Perona. A bayesian hierarchical model for learning natural scence categories. In *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 524–531, 2005.

[8] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59(2):167–181, 2004.

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

[10] A. Habibian, K. E. A. van de Sande, and C. G. M. Snoek. Recommendations for video event recognition using concept vocabularies. In *Proceedings of the ACM International Conference on Multimedia Retrieval*, Dallas, Texas, USA, 2013.

[11] H. Jégou, F. Perronnin, M. Douze, J. Sanchez, P. Pérez, and C. Schmid. Aggregating local image descriptors into compact codes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(9):1704–1716, 2012.

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

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

[14] X. Li, C. G. M. Snoek, M. Worring, and A. W. M. Smeulders. Harvesting social images for bi-concept search. *IEEE Transactions on Multimedia*, 14(4):1091–1104, August 2012.

[15] 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.

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

[17] S. Maji, A. C. Berg, and J. Malik. Classification using intersection kernel support vector machines is efficient. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 619–626, Anchorage, Alaska, 2008.

[18] M. Marszałek, C. Schmid, H. Harzallah, and J. van de Weijer. Learning object representations for visual object class recognition, October 2007. Visual Recognition Challange workshop, in conjunction with ICCV.

[19] M. Mazloom, E. Gavves, K. E. A. van de Sande, and C. G. M. Snoek. Searching informative concept banks for video event detection. In *Proceedings of the ACM International Conference on Multimedia Retrieval*, Dallas, Texas, USA, 2013.

[20] F. Perronnin, J. Sánchez, and T. Mensink. Improving the fisher kernel for large-scale image classification. In *European Conference on Computer Vision*, 2010.

[21] 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. The MIT Press, Cambridge, USA, 2000.

[22] A. F. Smeaton, P. Over, and W. Kraaij. Evaluation campaigns and TRECVid. In *Proceedings of the ACM SIGMM International Workshop on Multimedia Information Retrieval*, pages 321–330, 2006.

[23] C. G. M. Snoek and A. W. M. Smeulders. Visual-concept search solved? *IEEE Computer*, 43(6):76–78, June 2010.

[24] C. G. M. Snoek, K. E. A. van de Sande, O. de Rooij, B. Huurnink, E. Gavves, D. Odijk, M. de Rijke, T. Gevers, M. Worring, D. C. Koelma, and A. W. M. Smeulders. The MediaMill TRECVID 2010 semantic video search engine. In *Proceedings of the 8th TRECVID Workshop*, Gaithersburg, USA, November 2010.

[25] C. G. M. Snoek, K. E. A. van de Sande, O. de Rooij, B. Huurnink, J. R. R. Uijlings, M. van Liempt, M. Bugalho, I. Trancoso, F. Yan, M. A. Tahir, K. Mikolajczyk, J. Kittler, M. de Rijke, J.-M. Geusebroek, T. Gevers, M. Worring, D. C. Koelma, and A. W. M. Smeulders. The MediaMill TRECVID 2009 semantic video search engine. In *Proceedings of the 7th TRECVID Workshop*, Gaithersburg, USA, November 2009.

[26] C. G. M. Snoek, K. E. A. van de Sande, X. Li, M. Mazloom, Y.-G. Jiang, D. C. Koelma, and A. W. M. Smeulders. The MediaMill TRECVID 2011 semantic video search engine. In *Proceedings TRECVID Workshop*, Gaithersburg, USA, December 2011.

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

[28] 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 *Proceedings of the IEEE International Conference on Multimedia & Expo*, Amsterdam, The Netherlands, July 2005.

[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] J. R. R. Uijlings. *The What and Where in Visual Object Recognition*. PhD thesis, University of Amsterdam, Amsterdam, The Netherlands, 2011.

[31] J. R. R. Uijlings, A. W. M. Smeulders, and R. J. H. Scha. Real-time visual concept classification. *IEEE Transactions on Multimedia*, 12(7):665–681, 2010.

[32] K. E. A. van de Sande. *Invariant Color Descriptors for Efficient Object Recognition*. PhD thesis, University of Amsterdam, Amsterdam, The Netherlands, 2011.

[33] 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, September 2010.

[34] K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek. Empowering visual categorization with the GPU. *IEEE Transactions on Multimedia*, 13(1):60–70, February 2011.

[35] K. E. A. van de Sande, J. R. R. Uijlings, T. Gevers, and A. W. M. Smeulders. Segmentation as selective search for object recognition. In *IEEE International Conference on Computer Vision*, 2011.

[36] 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*, 114(4):450–462, April 2010.

[37] J. C. van Gemert, C. J. Veenman, and J. M. Geusebroek. Episode-constrained cross-validation in video concept retrieval. *IEEE Trans. Multimedia*, 11(4):780–785, 2009.

[38] J. C. van Gemert, C. J. Veenman, A. W. M. Smeulders, and J. M. Geusebroek. Visual word ambiguity. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 32(7):1271–1283, 2010.

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

[40] J. Zhang, M. Marszalek, 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.

[41] X. Zhou, K. Yu, T. Zhang, and T. S. Huang. Image classiffication using super-vector coding of local image descriptors. In *European Conference on Computer Vision*, 2010.