The MediaMill TRECVID 2011 Semantic Video Search Engine

C.G.M. Snoek, K.E.A. van de Sande, X. Li, M. Mazloom, Y.-G. Jiang*, 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 2011 video retrieval experiments. The MediaMill team participated in two tasks: semantic indexing and multimedia event detection. The starting point for the MediaMill detection approach is our top-performing bag-of-words system of TRECVID 2010, which uses multiple color SIFT descriptors, sparse codebooks with spatial pyramids, and kernel-based machine learning. All supported by GPU-optimized algorithms, approximated histogram intersection kernels, and multi-frame video processing. This year our experiments focus on 1) the soft assignment of descriptors with the use of difference coding, 2) the exploration of bag-of-words for event detection, and 3) the selection of informative concepts out of 1,346 concept detectors as a representation for event detection. The 2011 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 literature are mostly semantic [24], 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 2010* semantic video search engine [21], which made our robust (concept) detection system more efficient [12, 28, 31], and leading us to conclude that progress in visual concept search has doubled in just 3 years [20]. This year our experiments focus on 1) the soft assignment of descriptors with the use of difference coding, 2) the exploration of bag-of-words for event detection, and 3) the selection of informative concepts out of 1,346 concept detectors as a representation for event detection. Taken together, the *MediaMill 2011* 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 concepts in Section 3 We summarize our efforts in the multimedia event detection task in Section 4.

2 Bag-of-Words Foundation

Our TRECVID 2011 concept and event detection builds on previous editions of the MediaMill semantic video search engine [21–23,27,29], which draws inspiration from the bag-of-words propagated by Schmid and her associates [8,13,36], as well as recent advances in keypoint-based color features [30], codebook representations [32, 34], and efficient algorithmic refinements [12, 28], a GPU implementation [31], and compute clusters. In our description of the bag-of-words, we follow the video data as it flows through the computational process, as summarized in Figure 1, and detailed per component next.

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 [26] introduce robustness against viewpoint changes by selecting points, which can be recovered under different perspectives.

*Currently at: Fudan University, Shanghai, China.

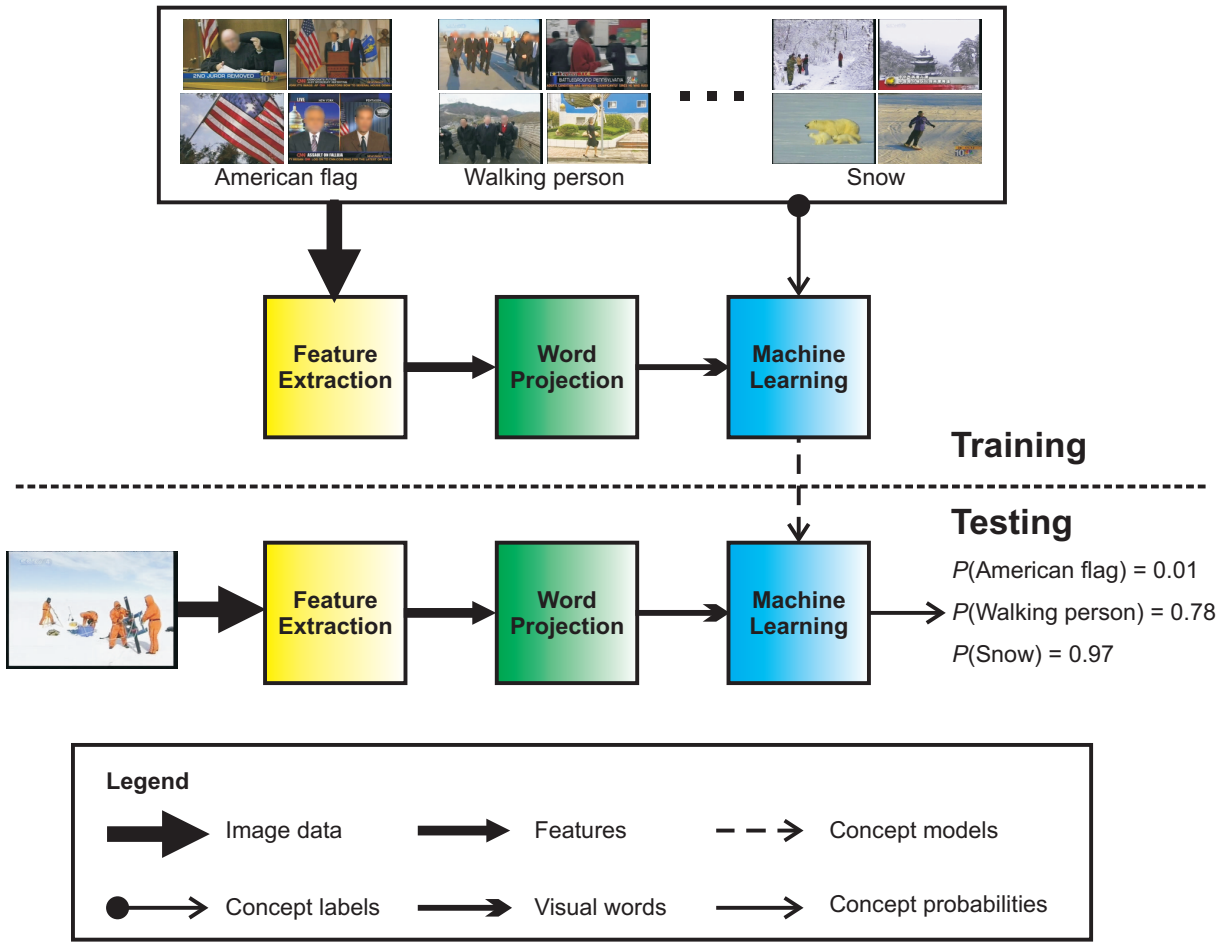

Figure 1: MediaMill TRECVID 2011 concept and event detection scheme, founded on bag-of-words, which serves as the blueprint for the organization of Section 2.

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 [22, 23, 25] 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. To be precise, we sample up to 6 additional i-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 [26]. 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, 7]. 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.* [8] 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 [13, 22, 23] 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 [3] 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.

SIFT The SIFT feature proposed by Lowe [11] 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, we use the version described by Lowe [11].

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.

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

We compute the SIFT [11] and ColorSIFT [30] features around salient points obtained from the Harris-Laplace detector and dense sampling. For all visual features we employ a spatial pyramid of 1x1 and 1x3 regions.

2.3 Word Projection

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.*, [7, 9, 18, 32, 34]. 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. By using a vectorized GPU implementation [31], 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.

In is well known that the traditional hard-assignment may be improved by using soft-assignment through kernel codebooks [34]. 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 soft assignment approaches have been proposed [14, 37]. In [14] Peronnin *et al*. train a Gaussian Mixture Model, where each codebook element has its own sigma one per dimension. They do not store the assignment, but the differences in all descriptor dimensions. Super Vector Coding by Zhou *et al*. [37] also counts the dimension-wise difference of a descriptor to a visual word. While these methods propose many new components and algorithms, we consider the difference coding their main contribution. We employ difference coding also.

Kernel 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 k-means clustering and hard assignment. We collect all possible codebook combinations in a (visual) kernel library. By using a GPU implementation [31], this kernel library can be computed efficiently. Naturally, the codebooks can be combined using various configurations. Depending on the kernel-based learning scheme used, we simply employ equal weights in our experiments or learn the optimal weight using cross-validation.

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

3 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 [15], 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

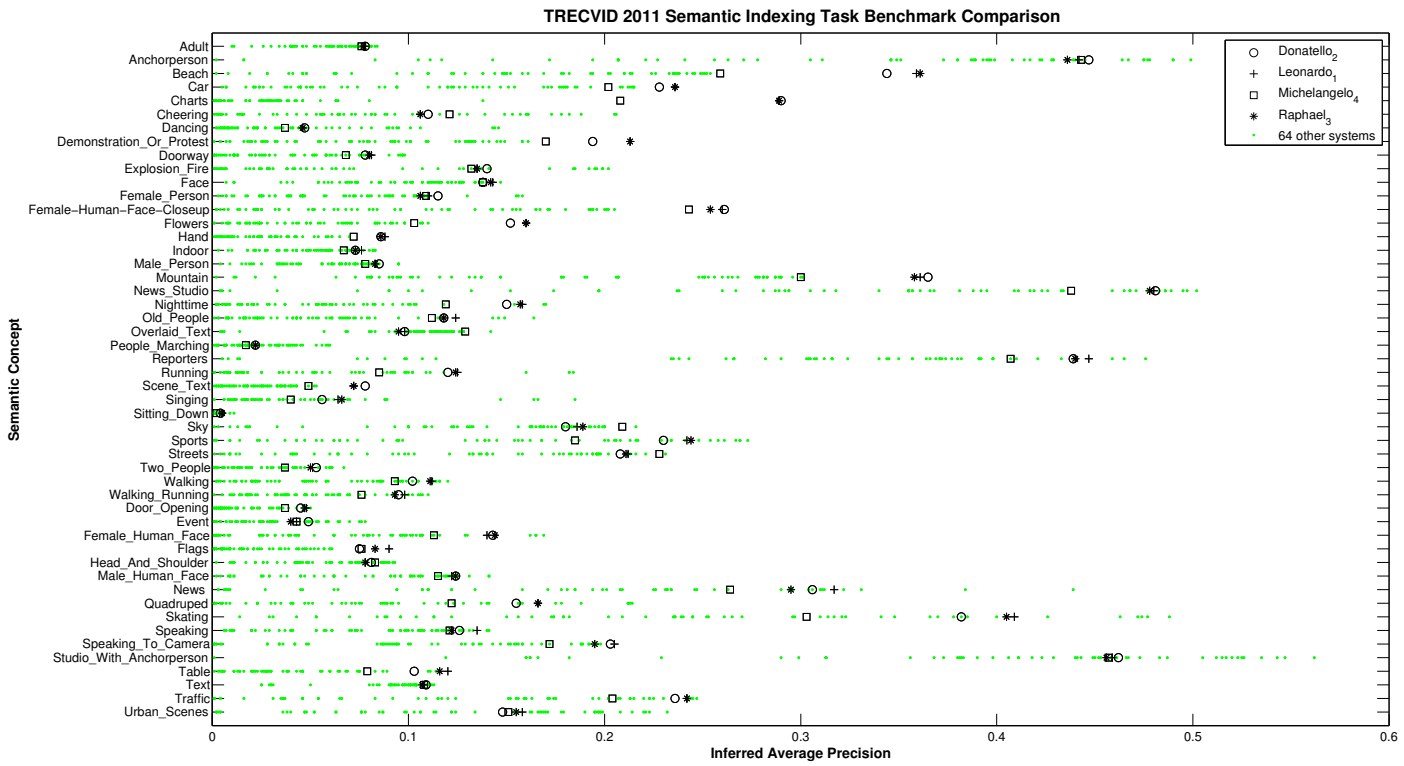

Figure 2: Comparison of MediaMill video concept detection experiments with other concept detection approaches in the TRECVID 2011 Semantic Indexing task.

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.

Learning robust concept detectors from visual features is typically achieved by kernel-based learning methods. Similar to previous years, we rely predominantly on the support vector machine framework [35] for supervised learning of semantic concepts. Here we use the LIBSVM implementation [4] with probabilistic output [10,16]. 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 χ^2 kernel function usually performs better than other kernels [36], it is computationally demanding when classifying multiple frames per shot. Therefore, we use the Histogram Intersection kernel and its efficient approximation as suggested by Maji *et al.* [12]. For difference coded bag-of-words we use a linear kernel [14,37].

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 [33]. 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 [33].

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.

3.1 Submitted Concept Detection Results

Our experiments [2,19] focus on establishing the influence of difference coding for concept detection. An overview of our submitted concept detection runs is depicted in Figure 2, and detailed next.

Run: Michelangelo The *Michelangelo* run is our baseline. It is based on multiple (visual) kernel libraries using both hard-assignment and difference coding on SIFT, OpponentSIFT, and RGB-SIFT descriptors, which have been applied on a single keyframe per shot. Fusion is performed using a simple AVG rule combination. This run achieved a mean infAP of 0.150.

Run: Donatello The *Donatello* run is a multi-frame version of the baseline. Here we have classified up to 6 additional i-frames per shot in combination with a MAX rule, before averaging the hard-assigned version with the difference coding version. This run achieved a mean infAP of

0.168, with the overall highest infAP for 4 concepts: *charts*, *female human face closeup*, *mountain*, and *scene text* .

Run: Raphael The *Raphael* run is similar in spirit to our best performing run of last year. It is based on multiple (visual) kernel libraries using hard-assigned SIFT, OpponentSIFT, and RGB-SIFT descriptors, which have been applied spatio-temporally with up to 10 additional i-frames per shot in combination with a MAX rule combination. This run achieved a mean infAP of 0.170, with the overall highest infAP for 4 concepts: *beach*, *car*, *demonstration*, and *flowers*.

Run: Leonardo The *Leonardo* run is similar to the *Donatello* run, with the only exception that 10 additional i-frames per shot are classified. This run achieved a mean infAP of 0.172, with the overall highest infAP for 7 concepts: *car*, *demonstration*, *flowers*, *hand*, *flags*, *speaking to camera*, and *table*.

3.2 1,346 Concept Detectors

In addition to the 346 concept detectors from the TRECVID SIN task, we have also employed our *Raphael* run setting on the entire concept set of the ImageNet Large Scale Visual Recognition Challenge 2011 [5], containing 1,000 object categories. All 1,346 detectors are included in the 2011 MediaMill semantic video search engine.

4 Detecting Events in Video

We participated in the multimedia event detection task using a visual-only approach. We explore two event representations, one founded on the same bag-of-words used for concept detection, the other based on a representation of *informative* concepts. Event representations based on multimedia fusion are investigated together with SRI International and the University of Southern California within the SESAME team [1].

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 [35] for supervised learning of events. We use the Histogram Intersection kernel and its efficient approximation as suggested by Maji *et al.* [12]. For difference coded bag-of-words we use a linear kernel [14, 37].

4.2 Event as bag-of-concepts

We investigate whether we can learn for a given event what concepts are most suited for its representation. We start from a large bag of concept detectors, in our case as many as 1,346, but rather than using all detectors simultaneously, we propose to select the most *informative* ones to represent and describe an event, as learned from training data with the aid of cross-entropy [17].

4.3 Submitted Event Detection Results

As training data we use keyframes sampled from the event kits, verified to contain the event, and not a black frame for example, by a human annotator. Classification of the test set is done on keyframes with a maximum of 6 extra frames per shot for the runs which view an event as bag-of-words. For the event as bag-of-concepts we classify 1 key frame per shot. The score of the video is the maximum score of the frames classified within that video. We use the video score to rank all videos in the collection.

In order to return a limited number of videos presumably containing the event of interest, we set a cut-off threshold, such that videos whose scores below the threshold will not be considered. We design our threshold selection such that the Normalized Detection Cost on unseen test data will be minimized. We use a regression model that interpolates between the confidence scores of videos on position at 1% and the position on 2% of the ranked list. The two weights are optimized by cross-validation. As it directly takes detection scores as input, the proposed model is adaptive to test data. With only one parameter to optimize, the model is simple and robust.

Our experiments focus on establishing the influence of event representations based on bag-of-words and bag of informative concepts. An overview of our submitted concept detection runs is depicted in Figure 3, and detailed next.

Run: Shield The *Shield* run is based on an event representation of informative concepts. While a representation based on concept scores is always worse than bag-of-words, we do believe the results are promising. Especially for the events *Parkour* and *Getting a vehicle unstuck* we obtain reasonable detection results. Our approach fails for *Birthday Party*.

Run: Thor The *Thor* run resembles the *Raphael* run for concept detection. We consider it our baseline. It is based on multiple (visual) kernel libraries using hard-assigned SIFT, OpponentSIFT, and RGB-SIFT descriptors, which have been applied spatio-temporally with up to 6 additional i-frames per shot in combination with a MAX rule combination per video. While this run outperforms the *Shield* run it is almost always worse than both *IronMan* and *CaptainAmerica*. The only exception being the event *Birthday Party*.

Run: IronMan The *IronMan* run is based on multiple (visual) kernel libraries using difference coding on SIFT, OpponentSIFT, and RGB-SIFT descriptors. Classification of the test set is done using a linear SVM on keyframes with a maximum of 6 extra frames per shot. The score of the video

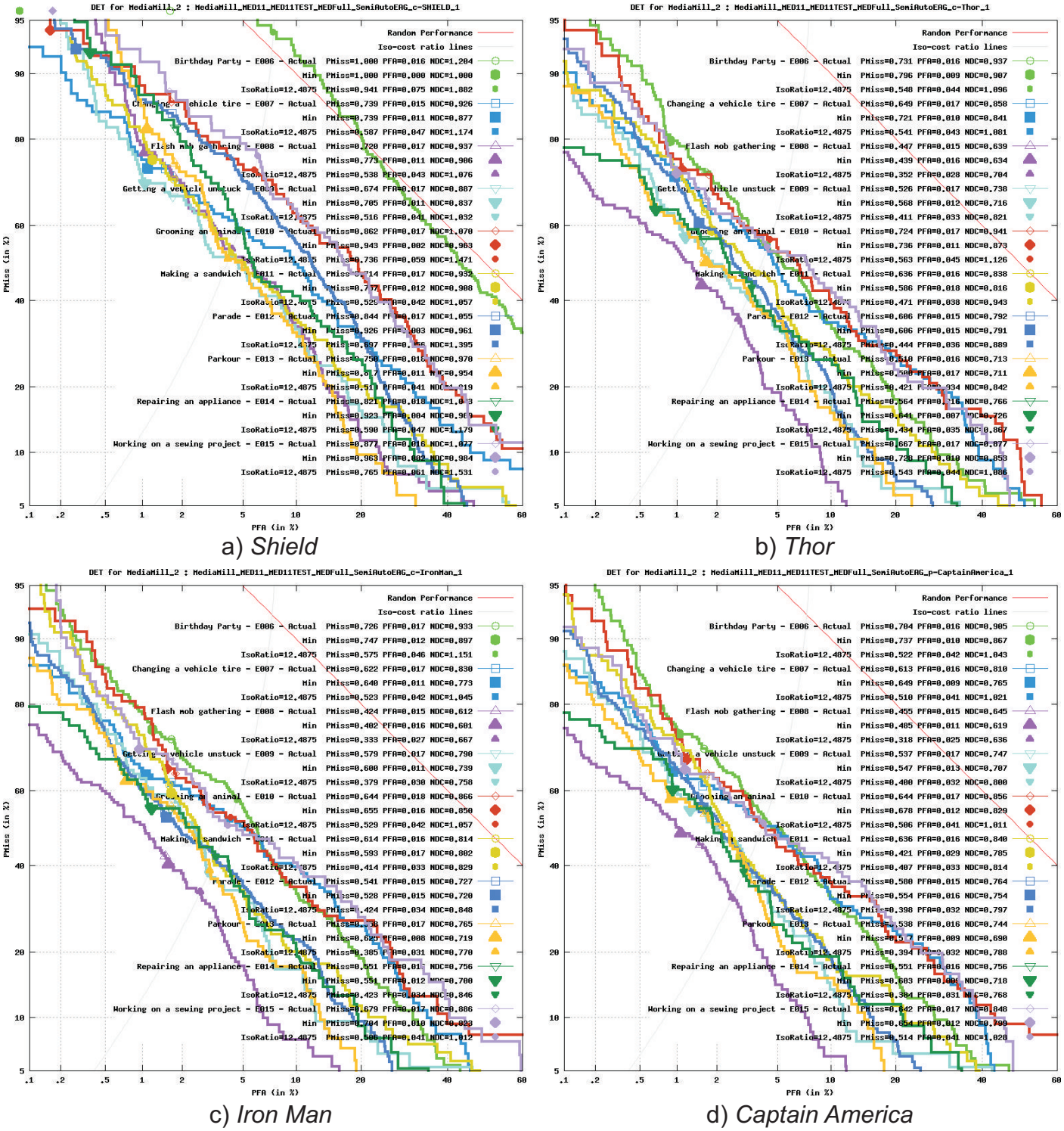

Figure 3: Overview of the MediaMill runs in the 2011 TRECVID event detection task benchmark.

is the maximum score of the frames classified within that video. This run outperforms *Thor* for almost all events, indicating the added value of difference coded bag-of-words for event detection.

Run: CaptainAmerica The *CaptainAmerica* run resembles the *Donatello* run for concept detection. It is based on multiple (visual) kernel libraries using both hard-assignment and difference coding on SIFT, OpponentSIFT, and RGB-SIFT descriptors, which have been applied spatio-temporally with up to 6 additional i-frames per shot in com-

bination with a MAX rule combination per video, before averaging the hard-assigned version with the difference coding version. Similar to concept detection, this is our best event detection run, outperforming the other runs for almost all events.

5 Conclusion

TRECVID continues to be a rewarding experience in gaining insight in the difficult problem of semantic video retrieval. The 2011 edition has again been a successful participation for the MediaMill team resulting in runner-up ranking for concept detection and a first exploration of the challenging problem of event detection.

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] M. Akbacak, R. C. Bolles, J. B. Burns, M. Eliot, A. Heller, J. A. Herson, D. C. Koelma, X. Li, M. Mazloom, G. K. Myers, R. Nallapati, R. Revatia, A. W. M. Smeulders, P. Sharma, C. G. M. Snoek, C. Sun, R. Trichet, K. E. A. van de Sande, and E. Yeh. The TRECVID SESAME MED system. In *Proc. TRECVID Workshop*, Gaithersburg, USA, 2011.

[2] S. Ayache and G. Quénot. Video corpus annotation using active learning. In *ECIR*, 2008.

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

[4] 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/`.

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

[6] L. Fei-Fei and P. Perona. A bayesian hierarchical model for learning natural scence categories. In *CVPR*, 2005.

[7] F. Jurie and B. Triggs. Creating efficient codebooks for visual recognition. In *ICCV*, 604–610, 2005.

[8] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR*, 2006.

[9] T. Leung and J. Malik. Representing and recognizing the visual appearance of materials using three-dimensional textons. *Int'l J. Computer Vision*, 43(1):29–44, 2001.

[10] 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.

[11] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *Int'l J. Computer Vision*, 60:91–110, 2004.

[12] S. Maji, A. C. Berg, and J. Malik. Classification using intersection kernel support vector machines is efficient. In *CVPR*, 2008.

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

[14] F. Perronnin, J. Sánchez, and T. Mensink. Improving the fisher kernel for large-scale image classification. In *ECCV*, 2010.

[15] C. Petersohn. Fraunhofer HHI at TRECVID 2004: Shot boundary detection system. In *Proc. TRECVID Workshop*, Gaithersburg, USA, 2004.

[16] 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.

[17] R. Y. Rubinstein and D. P. Kroese. *The Cross-Entropy Method: A Unified Approach to Combinatorial Optimization, Monte-Carlo Simulation and Machine Learning*. Information Science and Statistics. 2004.

[18] J. Sivic and A. Zisserman. Efficient visual search for objects in videos. *Proc. IEEE*, 96(4):548–566, 2008.

[19] A. F. Smeaton, P. Over, and W. Kraaij. Evaluation campaigns and TRECVid. In *MIR*, 2006.

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

[21] 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 *Proc. TRECVID Workshop*, Gaithersburg, USA, 2010.

[22] 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 *Proc. TRECVID Workshop*, Gaithersburg, USA, 2009.

[23] C. G. M. Snoek, K. E. A. van de Sande, O. de Rooij, B. Huurnink, J. C. van Gemert, J. R. R. Uijlings, J. He, X. Li, I. Everts, V. Nedović, M. van Liempt, R. van Balen, F. Yan, M. A. Tahir, K. Mikolajczyk, J. Kittler, M. de Rijke, J.-M. Geusebroek, T. Gevers, M. Worring, A. W. M. Smeulders, and D. C. Koelma. The MediaMill TRECVID 2008 semantic video search engine. In *Proc. TRECVID Workshop*, Gaithersburg, USA, 2008.

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

[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 *ICME*, 2005.

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

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

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

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

[30] K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek. Evaluating color descriptors for object and scene recognition. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 32(9):1582–1596, 2010.

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

[32] 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, 2010.

[33] 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.

[34] 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.

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

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

[37] X. Zhou, K. Yu, T. Zhang, and T. S. Huang. Image classiffication using super-vector coding of local image descriptors. In *ECCV*, 2010.