MediaMill at TRECVID 2014: Searching Concepts, Objects, Instances and Events in Video

Snoek, C.G.M.; van de Sande, K.E.A.; Fontijne, D.; Cappallo, S.; van Gemert, J.; Habibian, A.; Mensink, T.; Mettes, P.; Tao, R.; Koelma, D.C.; Smeulders, A.W.M.

Publication date
2014

Document Version
Final published version

Published in
2014 TREC Video Retrieval Evaluation: notebook papers and slides

MediaMill at TRECVID 2014: Searching Concepts, Objects, Instances and Events in Video

C.G.M. Snoek[†‡], K.E.A. van de Sande[‡], D. Fontijne[‡], S. Cappallo[†], J. van Gemert[†], A. Habibian[†], T. Mensink[†], P. Mettes[†], R. Tao[†], D.C. Koelma[†], A.W.M. Smeulders[†]

[†]ISLA, University of Amsterdam
Amsterdam, The Netherlands

[‡]Qualcomm Technologies
Amsterdam, The Netherlands

http://www.mediamill.nl

Abstract

In this paper we summarize our TRECVID 2014 [12] video retrieval experiments. The MediaMill team participated in five tasks: concept detection, object localization, instance search, event recognition and recounting. We experimented with concept detection using deep learning and color difference coding [17], object localization using FLAIR [23], instance search by one example [19], event recognition based on VideoStory [4], and event recounting using COSTA [10]. Our experiments focus on establishing the video retrieval value of these innovations. The 2014 edition of the TRECVID benchmark has again been a fruitful participation for the MediaMill team, resulting in the best result for concept detection and object localization.

1 Task I: Concept Detection

Last year we introduced deep convolutional neural networks for video concept detection and localization, and demonstrated their complementary power when used with Fisher vectorized bag-of-words [17]. This year we included several tangential improvements for concept detection, including coordinate coding in bag-of-words, pre-training on ImageNet objects, and the use of fully connected output layers as SVM features. Since deep learning is critically dependent on labeled examples and suffers from noisy and incomplete annotations, as common in TRECVID [2, 16], we manually extended the collaborative annotations.

Color Difference Coding Our baseline concept detection system uses a bag-of-words with color point descriptors only. For point sampling we rely on dense sampling, with an interval distance of six pixels and sampled at multiple scales. We used a spatial pyramid of 1x1 and 1x3 regions in our experiments. We used a mixture of SIFT, TSIFT, and C-SIFT descriptors [22]. We compute the descriptors around points obtained from dense sampling, and reduce the dimensionality with principal component analysis. We encode the color descriptors with the aid of difference coding using Fisher vectors with a Gaussian Mixture Model codebook [13]. We encode spatial information into the Fisher vector akin to [15]. For efficient storage we perform product quantization [6] on the features. The classifier is a linear SVM, which we apply on the keyframe and six additional frames per shot, we take the maximum response as the score per shot.

Convolutional Neural Network Our deep learning concept detection system is a convolutional neural network with eight layers with weights [24]. The input is raw pixel data, the output are concept scores. The network is trained using error back propagation. However, in contrast to ImageNet, there are too few labeled examples in the TRECVID SIN 2014 set for deep learning to be effective. We studied how additional examples for 15K objects from ImageNet [14] can be exploited to better train our networks. To improve the results, we took a network that had already been trained on ImageNet and re-trained it for the 60 TRECVID 2014 SIN concepts. We train a network and apply it on the keyframe and six additional frames per shot, we take the maximum response as the score per shot. We repeat this for a total of eight networks and average the scores per shot.

1.1 Submitted Runs

We submitted four runs in the regular SIN task. We summarize our regular SIN task submission in Figure 1.

Cersei is our baseline run. It is based on eight deep CNN networks, pretrained on ImageNet and tuned for the SIN task concepts. It achieves an mAP of 0.318 and is the best performer for 1 out of 30 concepts. This run came out third in terms of overall system performance.

Tyrion is our first hybrid system that fuses deep learning and color difference coding by a simple weighted average obtained by cross-validation. It combines our best single deep CNN network with color difference coding using Fisher vectors and a spatially improved color difference coding using Fisher vectors. It achieves an mAP of 0.316 and is the best

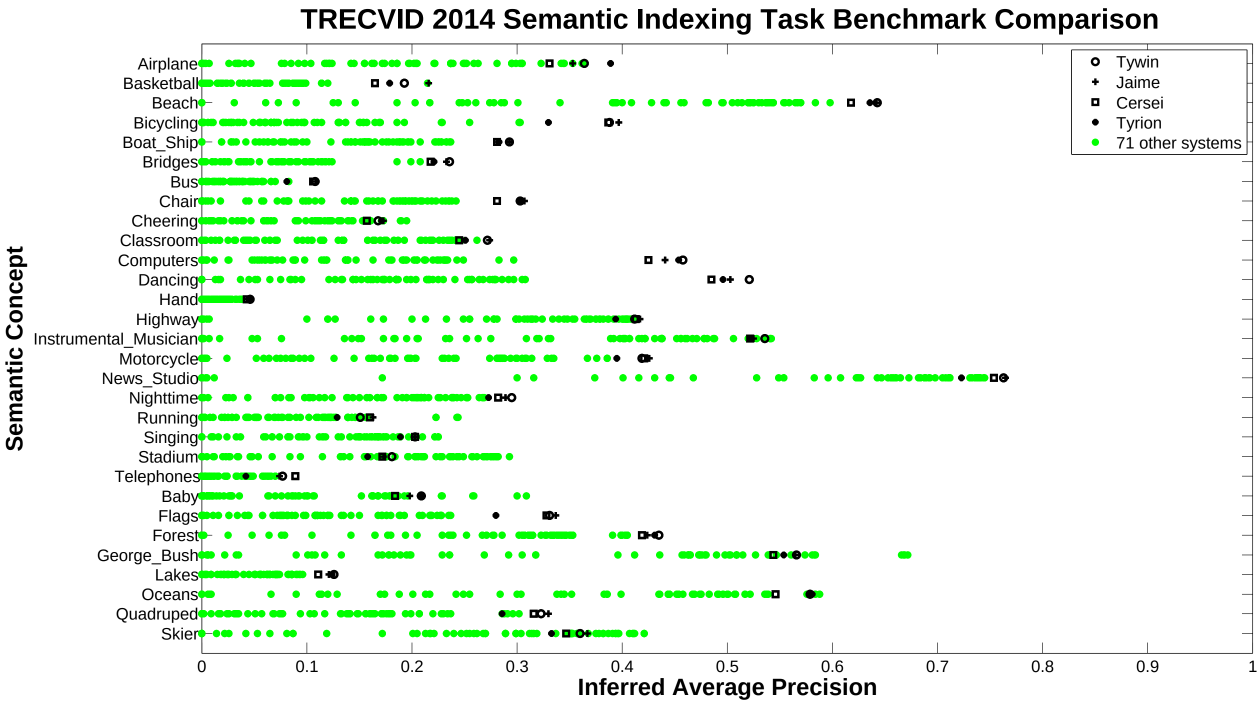

Figure 1: Comparison of MediaMill video concept detection experiments with other concept detection approaches in the TRECVID 2014 Semantic Indexing task.

performer for 2 out of 30 concepts. This run came out fourth in terms of overall system performance.

Jaime is our second hybrid system that fuses deep learning and color difference coding by a simple weighted average obtained by cross-validation. It combines the *Cersie* baseline run with color difference coding using Fisher vectors and a spatially improved version thereof. It achieves an mAP of 0.331 and is the best performer for 11 out of 30 concepts. This run came out second in terms of overall system performance.

Tywin is our third hybrid system that combines all components from the above runs. It achieves an mAP of 0.332 and is the best performer for 9 out of 30 concepts. This run came out as the system with best overall performance in the SIN task of TRECVID 2014.

2 Task II: Object Localization

We perceive object localization in video as a supervised learning problem. So we require bounding box annotations for objects of interest. We have refined a subset of the global image annotations for 10 (global) concepts to object-level by adding their bounding boxes. A major computational bottleneck in many current localization algorithms is the evaluation of arbitrary boxes. Dense local analysis and powerful bag-of-word encodings, such as Fisher vectors and VLAD, lead to improved accuracy at the expense of increased computation time. Where a simplification in the representation is tempting, we prefer our recently proposed FLAIR - Fast Local Area Independent Representation [23], which reduces computation time while maintaining accuracy.

Fisher with FLAIR By representing the picture as sparse integral images, one per codeword, FLAIR allows for very fast evaluation of any box encoding and still enables spatial pooling. In FLAIR we achieve exact Fisher vector coding, even with l2 and power-norms. Finally, by multiple codeword assignments, we achieve exact and approximate Fisher vectors with FLAIR. The results are a 18x speedup, which we leverage to perform video concept localization. Given a ranking of a video collection provided by our concept detection runs, we start from state-of-the-art fast selective search [21] to generate object proposals for each of the provide iframes. As descriptor for each image we use dense SIFT, OpponentSIFT and C-SIFT [22], sampled at every 2 pixels at 3 scales. The dimensionality of each descriptor is reduced to $D = 80$ with PCA. We employ Fisher with FLAIR using a codebook of size $K = 256$. Because evaluating multiple boxes is computationally cheap in FLAIR, we use a spatial pyramid with 30 cells (1x1, 2x2, 3x3 and 4x4). For each object we train a linear SVM classifier, where the positive examples come from our ground truth annotations. We follow a hard negative mining protocol as is common in the literature [23].

2.1 Submitted Runs

For each of the four submitted runs in the concept detection task we analyzed the top-1,000 shots using Fisher with FLAIR. We observe that *Cersei with FLAIR* provides us with the best results for all the provided metrics, indicating it is best to use an initial ranking provided by deep convolutional networks, before employing Fisher vectors with FLAIR. This run came out as the system with best overall performance in the SIN localization task of TRECVID 2014.

3 Task III: Instance Search

Our instance search system is built upon the recent advances in the literature. We use three interest point detectors, MSER, Harris-Affine and Hessian-Affine, and two local descriptors, RootSIFT [1] and a SIFT descriptor combined with a 64-dimensional color feature. We employ VLAD encoding [7] with a large vocabulary containing 20,000 visual words. Normalization per visual word is applied to tackle burstiness [20]. Exponential similarity proposed by Tao *et al*. [19] is used to compute the similarity of two frames on each visual word. The exponential similarity puts disproportional high weights on close matches in the feature space, which is advantageous for instance search.

3.1 Submitted Runs

We participated in two subtasks, using example one only and using all four examples. We submitted three runs for each subtask. The difference among the three runs lies in how the query examples are used.

Run1 In this run, we only consider the foreground region of the query. This run achieves an mAP of 0.125 in the one-example case, and 0.227 in the four-example case.

Run2 This run uses the entire query image, scoring 0.106 and 0.161 respectively in the one-example subtask and the four-examples subtask.

Run3 This run fuses the results of the above two runs. The performance is 0.133 and 0.221 respectively.

4 Task IV: Event Recognition

Our event recognition system is founded on our recently proposed VideoStory [4] embedding. Rather than relying on predefined concept detectors, and annotations, for the video representation [5, 8, 9], VideoStory learns a representation from web-harvested video clips and their descriptions.

VideoStory embedding We modeled the representation learning as a multi-step embedding process, where the high-dimensional raw pixel values from the video are embedded into low-dimensional semantic features. Our VideoStory embedding process is made of two consecutive steps: a pixel to feature embedding, where the frame pixels [18] are projected into a non-semantic feature space by applying a deep CNN, which is pre-trained on the ImageNet data. We feed the video frames into the deep CNN and take the responses from the middle layers as features. We experimentally selected the layers which generate more effective features and used them for the second step of generating the meta-data. In our submissions, we use the responses from the second fully connected layer.

Although the extracted features can be used to train event classifiers directly, we keep on embedding them into a lower dimensional and more semantic feature space. We follow three goals in performing this second step embedding: first, to make the features semantically interpretable, which is desired for recounting purposes. Second, to reduce the features size, which makes the event training and detection even more efficient. Third, to transfer some prior information relevant to the event to generate more effective features. For this purpose, we apply VideoStory embedding [4], which learns to embed video features into their semantic description. We train VideoStory on a collection of 46K YouTube videos and their titles, provided by [4]. We use the same VideoStory embedding for the 0ex, 10ex, and 100ex submmission.

In addition, within the SESAME team [3, 11], we also investigate together with SRI International and the University of Southern California several additional multimedia approaches to video event detection.

4.1 Submitted Runs

Pre-Specified In our pre-specified run we score 3.6 mAP for Semantic Query without examples, 14.9 mAP when using 10 examples to train events, and 24.1 mAP when using 100 examples. Note that we use a single visual feature.

AdHoc In our adhoc run we score 1.8 mAP for Semantic Query without examples, 9.0 mAP when using 10 examples to train events, and 18.6 mAP when using 100 examples. Note that we use a single visual feature.

5 Task V: Event Recounting

The goal of event recounting is to provide key evidence to semantically explain the MED classification. This key evidence consists of a number of video snippets with the corresponding semantic concepts detected in these snippets. Each piece of evidence is further supplemented with a confidence score of the observed concepts.

The MediaMill MER submission is built upon the 15K concept scores provided by the CNN outlined in Section 1.

As a result, the submission is focused on visual evidence of concepts and their temporal locations. To limit the influence of the large number of irrelevant concepts, we automatically subselect a limited number of concepts using a ranking approach inspired by COSTA [10].

To yield the temporally localized evidence, we split each video into a number of segments. During training, we discover the most informative segments from the ten provided positive videos. These parts are matched with the segments of a test video and the top-ranked matches are presented as the key visual evidence. Exactly how many pieces of evidence are chosen is a function of the video length. For each piece of evidence, the highest scoring concepts are used as the semantic recounting.

5.1 Submitted Run

The MER evaluation consists of three criteria: the percentage of the video used in the recounting, the conciseness of the recounting, and how convincing the presented key evidence is. The first criterion measures the brevity of the evidence, and the second evaluates how well the selected concepts represent the event. Finally, the third criterion states how well the key evidence demonstrates that the video contains the event. In the MediaMill system, on average 15.4% of the video was used as key evidence. Furthermore, for 55.0% of the recountings, the key evidence was deemed convincing by the judges.

Acknowledgments

The authors are grateful to NIST and the TRECVID coordinators for the benchmark organization effort. This research is supported by the STW STORY project and the Dutch national program COMMIT.

References

[1] R. Arandjelovic and A. Zisserman. Three things everyone should know to improve object retrieval. In *CVPR*, 2012.

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

[3] R. C. Bolles, J. B. Burns, J. A. Herson, G. K. Myers, J. van Hout, W. Wang, J. Wong, E. Yeh, A. Habibian, D. C. Koelma, T. Mensink, A. W. M. Smeulders, C. G. M. Snoek, R. Kovvuri, A. Aggarwal, K. Chen, P. Sharma, S. Cao, and R. Nevatia. The 2014 SESAME multimedia event detection and recounting system. In *TRECVID*, 2014.

[4] A. Habibian, T. Mensink, and C. G. M. Snoek. Videostory: A new multimedia embedding for few-example recognition and translation of events. In *MM*, 2014.

[5] A. Habibian and C. G. M. Snoek. Recommendations for recognizing video events by concept vocabularies. *CVIU*, 124:110–122, 2014.

[6] H. Jégou, M. Douze, and C. Schmid. Product quantization for nearest neighbor search. *TPAMI*, 33(1):117–128, 2011.

[7] H. Jégou, M. Douze, C. Schmid, and P. Pérez. Aggregating local descriptors into a compact image representation. In *CVPR*, 2010.

[8] M. Mazloom, E. Gavves, and C. G. M. Snoek. Conceptlets: Selective semantics for classifying video events. *TMM*, 16(8):2214–2228, 2014.

[9] M. Mazloom, A. Habibian, and C. G. M. Snoek. Querying for video events by semantic signatures from few examples. In *MM*, 2013.

[10] T. Mensink, E. Gavves, and C. G. M. Snoek. Costa: Co-occurrence statistics for zero-shot classification. In *CVPR*, 2014.

[11] G. K. Myers, R. Nallapati, J. van Hout, S. Pancoast, R. Nevatia, C. Sun, A. Habibian, D. C. Koelma, K. E. A. van de Sande, A. W. M. Smeulders, and C. G. M. Snoek. Evaluating multimedia features and fusion for example-based event detection. *MVA*, 25(1):17–32, 2014.

[12] P. Over, G. Awad, M. Michel, J. Fiscus, G. Sanders, W. Kraaij, A. F. Smeaton, and G. Quenot. Trecvid 2014 – an overview of the goals, tasks, data, evaluation mechanisms and metrics. In *TRECVID*, 2014.

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

[14] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. Imagenet large scale visual recognition challenge, 2014. arXiv:1409.0575.

[15] J. Sanchez, F. Perronnin, and T. E. de Campos. Modeling the spatial layout of images beyond spatial pyramids. *PRL*, 33(16):2216–2223, 2012.

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

[17] C. G. M. Snoek, K. E. A. van de Sande, D. Fontijne, A. Habibian, M. Jain, S. Kordumova, Z. Li, M. Mazloom, S.-L. Pintea, R. Tao, D. C. Koelma, and A. W. M. Smeulders. MediaMill at TRECVID 2013: Searching concepts, objects, instances and events in video. In *TRECVID*, 2013.

[18] S. Strassel, A. Morris, J. Fiscus, C. Caruso, H. Lee, P. Over, J. Fiumara, B. Shaw, B. Antonishek, and M. Michel. Creating HAVIC: Heterogeneous audio visual internet collection. In *LREC*, 2012.

[19] R. Tao, E. Gavves, C. G. M. Snoek, and A. W. M. Smeulders. Locality in generic instance search from one example. In *CVPR*, 2014.

[20] G. Tolias, Y. Avrithis, and H. Jégou. To aggregate or not to aggregate: Selective match kernels for image search. In *ICCV*, 2013.

[21] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, and A. W. M. Smeulders. Selective search for object recognition. *IJCV*, 104(2):154–171, 2013.

[22] K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek. Evaluating color descriptors for object and scene recognition. *TPAMI*, 32(9):1582–1596, 2010.

[23] K. E. A. van de Sande, C. G. M. Snoek, and A. W. M. Smeulders. Fisher and VLAD with FLAIR. In *CVPR*, 2014.

[24] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *ECCV*, 2014.