Qualcomm Research and University of Amsterdam at TRECVID 2015: Recognizing Concepts, Objects, and Events in Video

Snoek, C.G.M.; Cappallo, S.; Fontijne, D.; Julian, D.; Koelma, D.C.; Mettes, P.; van de Sande, K.E.A.; Sarah, A.; Stokman, H.; Towal, R.B.

Publication date
2015

Document Version
Final published version

Published in
2015 TREC Video Retrieval Evaluation: notebook papers and slides

Qualcomm Research and University of Amsterdam at TRECVID 2015: Recognizing Concepts, Objects, and Events in Video

Cees G.M. Snoek[†*], Spencer Cappallo[*], Daniel Fontijne[†], David Julian[‡], Dennis C. Koelma[*], Pascal Mettes[*], Koen E.A. van de Sande[†], Anthony Sarah[‡], Harro Stokman[†], R. Blythe Towal[‡]

[†]Qualcomm Research Netherlands
Amsterdam, The Netherlands

[‡]Qualcomm Research
San Diego, USA

[*]University of Amsterdam
Amsterdam, The Netherlands

Abstract

In this paper we summarize our TRECVID 2015 [12] video recognition experiments. We participated in three tasks: concept detection, object localization, and event recognition, where Qualcomm Research focused on concept detection and object localization and the University of Amsterdam focused on event detection. For concept detection we start from the very deep networks that excelled in the ImageNet 2014 competition and redesign them for the purpose of video recognition, emphasizing on training data augmentation as well as video fine-tuning. Our entry in the localization task is based on classifying a limited number of boxes in each frame using deep learning features. The boxes are proposed by an improved version of selective search. At the core of our multimedia event detection system is an Inception-style deep convolutional neural network that is trained on the full ImageNet hierarchy with 22k categories. We propose several operations that combine and generalize the ImageNet categories to form a desirable set of (super-)categories, while still being able to train a reliable model. The 2015 edition of the TRECVID benchmark has been a fruitful participation for our team, resulting in the best overall result for concept detection, object localization and event detection.

1 Task I: Concept Detection

Up to 2014 the best video concept detection systems in TRECVID combined traditional encodings with deep convolutional neural networks [16,17], this year we present our system entry that is based on deep learning only. We start from the very deep networks that excelled in the ImageNet 2014 competition [13] and redesign them for the purpose of video recognition. Each of our runs was a mixture of Inception Style [18] and VGG Style networks [15]. The input for each network is raw pixel data, the output are concept scores. The networks are trained using error back propagation. However, in contrast to ImageNet, there are too few labeled examples in the TRECVID SIN 2015 set [1] for deep learning to be effective. To improve the results, we took networks that had already been trained on ImageNet and re-trained them for the 60 TRECVID 2015 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.

Inception Style Networks The GoogLeNet/Inception architecture [18] with batch normalization [5] was used as the foundation for the Inception Style approaches. These models were pre-trained in-house on various selections of the ImageNet 'fall 2011' dataset. For fine-tuning Inception models, an 'Alex-style' [8] fully connected head was placed on top of the Inception 5b layer. These models were then fine-tuned on different sets of TRECVID data with different sets of augmentation, including, scale, vignetting, color-casting and aspect-raio distortion as in [22]. This resulted in a total of 42 networks.

VGG Style Networks There were several VGG architectures [15] used for the TRECVID entry based on a mixture of VGG Net D and VGG Net E networks. The initial weights for the networks were obtained from VGGs ImageNet trained models. These models were then fine-tuned on different sets of 2014 and 2015 TRECVID data with different sets of augmentation, including, scale, vignetting, color-casting and aspect-raio distortion as in [22]. This resulted in a total of 14 networks.

1.1 Submitted Runs

We submitted four runs in the SIN task, which we summarize in Figure 1. Our *Gargantua* run uses a non-weighted fusion of all available models. It scores an MAP of 0.360 and is the best performer for 7 out of 30 concepts. The *Mann* run uses a weighted fusion of all models per category. This run obtains an MAP of 0.359 and is the best performer for 6 concepts. Our other runs are based on fewer models, selected based on their validation set performance. The *Edmunds* run is a non-weighted fusion of 32 models and scores 0.349 MAP (best for 3 concepts). Our *Miller* run uses only 7 models and obtains the best overall MAP of 0.362, with the highest score for 12 out of 30 concepts. In this run the internal validation set was also added during learning, without verifying its effectiveness at training time. Taken together our runs are the best performer for 20 out of 30 concepts,

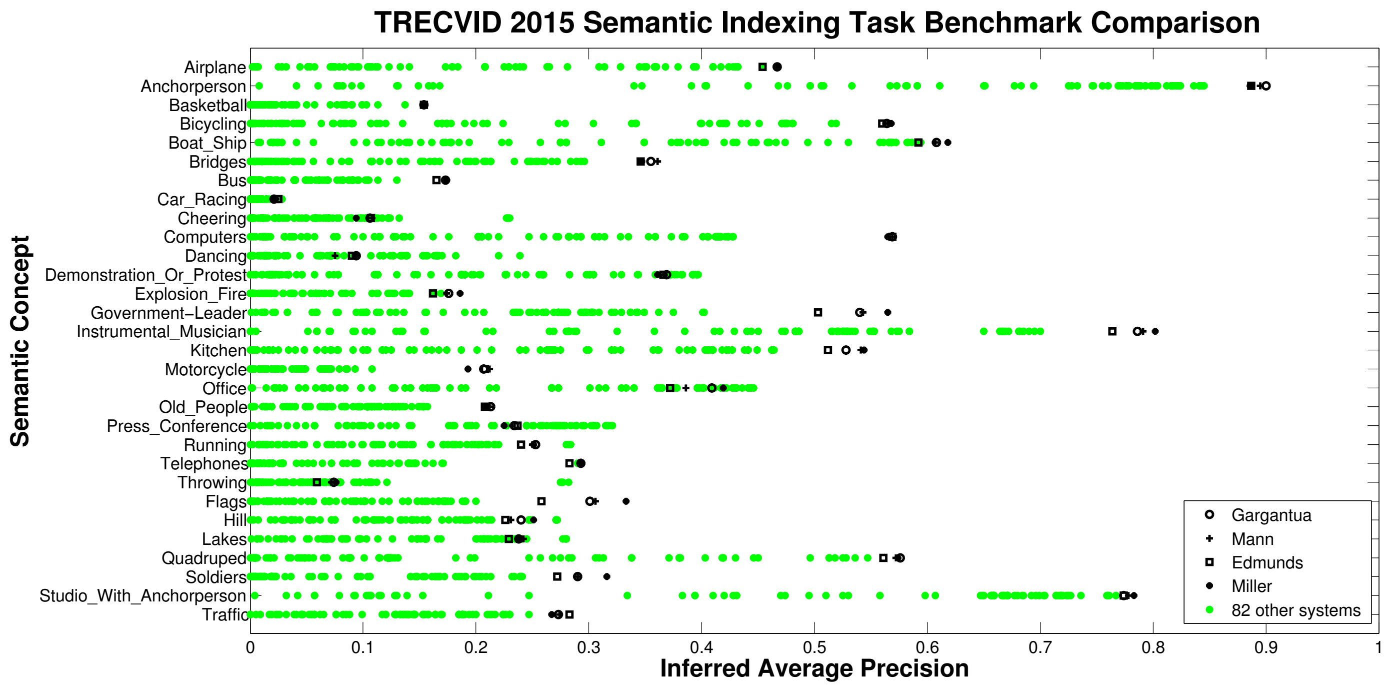

Figure 1: Comparison of Qualcomm Research video concept detection experiments with other concept detection approaches in the TRECVID 2015 Semantic Indexing task.

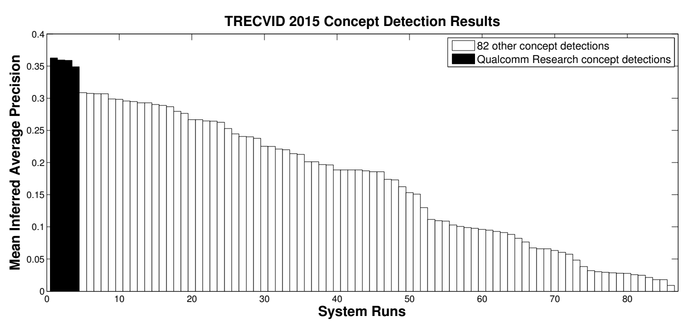

Figure 2: Qualcomm Research video concept detection runs compared with other concept detection approaches in the TRECVID 2015 Semantic Indexing task.

and the four best performers amongst all submissions, see Figure 2.

2 Task II: Object Localization

Up to 2014 the best video object localization systems in TRECVID combined box proposals [19] with traditional encodings and deep convolutional neural networks [16, 17, 20], this year we present our system entry that is based on box proposals encoded with deep learning only.

Deep learning features for boxes The deep learning features are extracted using two of the Inception deep neural networks from the SIN taks submission. Compared to a standard AlexNet (29.9 MAP on our validations set), the use of an Inception network brings us an extra 7.4% MAP (37.3 MAP). One network is trained to recognize 2,048 ImageNet categories deemed relevant to TRECVID, the other to recognize 4,096 categories. Compared to a more standard 1,000 ImageNet category network (37.3 MAP), these obtain 39.8/40.3 MAP on our internal validation set of box-annotated TRECVID keyframes. When combined, the two features give us a 43.7 MAP. This is a significant improvement over last years Fisher with FLAIR features [16, 20], which scored 26.5 MAP on our internal validation set.

Table 1: Overview of Qualcomm Research object localization experiments on our internal validation set. Note the MAP improvement of our deep learning system over last years best TRECVID performer using Fisher with FLAIR [20].

Method	Box proposals	MAP
Fisher with FLAIR (TZIFT)	Selective Search	20.3
Fisher with FLAIR (ZIFT)	Selective Search	24.1
Fusion of Fisher with FLAIR (ZIFT+TZIFT)	Selective Search	26.5
SVM on AlexNet 1,000 features	Selective Search	29.9
SVM on Inception 1,000 features	Selective Search	37.3
SVM on Inception 2,048 features	Selective Search	39.8
SVM on Inception 2,048 features	Selective Search++	40.2
SVM on Inception 4,096 features	Selective Search	40.3
SVM on Inception 4,096 features	Selective Search++	42.4
Fusion of Inception 2,048 & 4,096	Selective Search	43.7
Fusion of Inception 2,048 & 4,096	Selective Search++	45.3

Box proposals Our entry in the TRECVID 2015 localization task is based on classifying a limited number of boxes

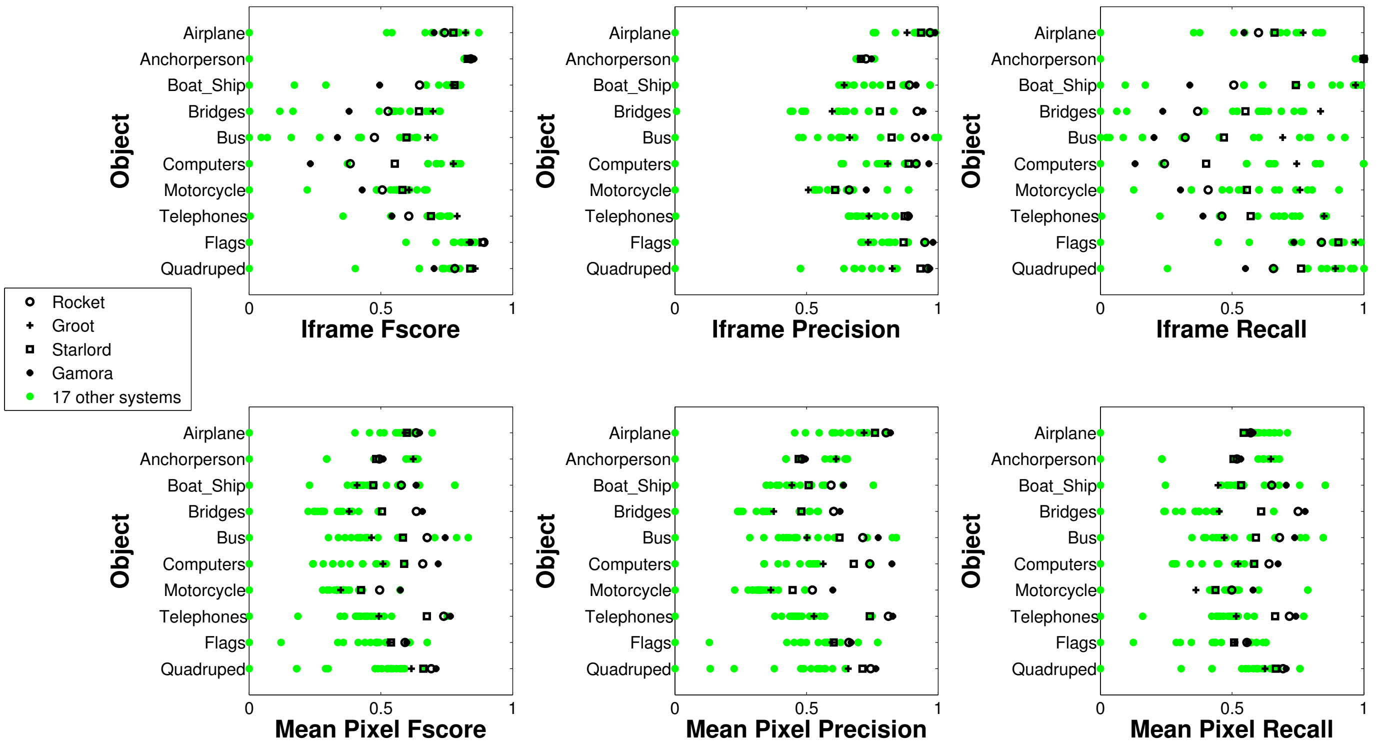

Figure 3: Comparison of Qualcomm Research video object localization experiments with other localization approaches in the TRECVID 2015 Object Localization task.

Table 2: Overview of Qualcomm Research object localization runs on our internal validation set.

Run	Threshold	Max boxes	Recall	Precision	F-scores	MAP
Gamora	0.5	1	34%	55%	0.42	30.9
Rocket	0.0	1	41%	42%	0.41	35.0
Starlord	-0.5	1	47%	24%	0.32	38.1
Groot	-1.1	3	64%	7%	0.12	43.5

in each frame using deep learning features. The boxes are proposed by an improved version of selective search. In Table 1, the difference between the standard proposal method, known as selective search fast or quick in the literature [19], and the improved selective search, Selective Search++, is 1.6% MAP: from 43.7 to 45.3 MAP on our internal validation set.

Localization system training For training an SVM model to classify boxes, we obtain positive object boxes through human annotation. The negative examples are picked randomly and then we follow the commonly used hard negative mining approach to collect extra negative examples [19, 20]. With the trained SVM models, we classify the box proposals generated by selective search. This forms a localization system that for each frame outputs a number of boxes together with confidence scores per box.

2.1 Submitted Runs

All our runs are based on the same set of boxes and confidences (those from the setting which achieved 45.3 MAP), with different thresholds and limits on the number of boxes applied. The different choices aim to optimize either precision or recall, or to strike a balance between both. The different runs are listed in Table 2 with their characteristics on our internal validation set. The results for the 10 object categories evaluated over 6 different measures is shown in Figure 3.

The *Groot* run is aimed at high recall: it predicts up to 3 boxes per image, to account for multiple object instances. However, this run has a worse pixel recall than those that predict only a single box (*Starlord* run). In the evaluation only one box is annotated by NIST, and there is a penalty for predicting 3 boxes if there is only one instance. Even though this run will find more object instances, it does not outweigh the penalty for two 'false positives'. In terms of iframe recall, it does score better than *Starlord*. Our *Gamora* run aims at high precision. It obtains the highest score in 19 out of 60 cases, especially in iframe precision, pixel precision, pixel recall and pixel fscore. Our *Rocket* run is in between *Gamora* and *Starlord* in terms of the threshold. It is meant to balance precision and recall, but is almost always outperformed by *Gamora* (on precision/f-score) or *Starlord* (on recall). Overall, given the 10 objects and 6 different measures, we have one run with the highest scores in 19 cases, and a total of 23 best scores when considering

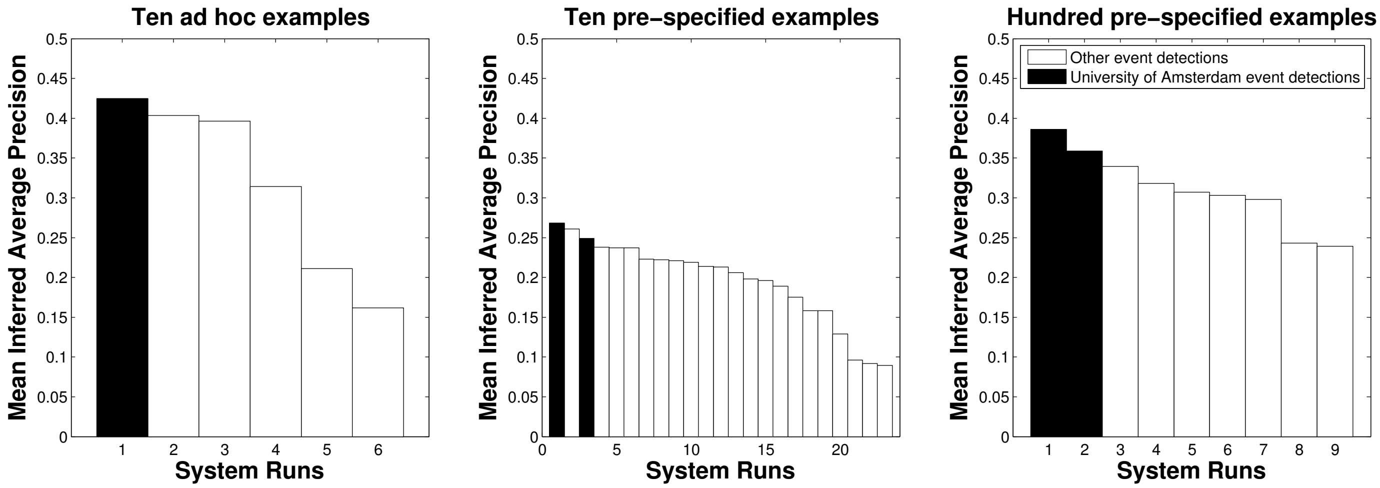

Figure 4: Comparison of University of Amsterdam's video event detection experiments with other event detection approaches in the TRECVID 2015 Multimedia Event Detection task, as learned from ten ad hoc examples, ten pre-specified examples or hundred pre-specified examples.

all 4 runs.

3 Task III: Event Recognition

Last year, our event recognition system was founded on a VideoStory embedding [3]. Rather than relying on predefined concept detectors, and annotations, for the video representation [4,7,9], VideoStory learns a representation from web-harvested video clips and their descriptions. This year our event detection efforts focus on deep learning. The network, Google's Inception network [18], is trained on a large personalized selection of ImageNet concepts [13] and applied to the frames of the event videos. Below, we outline how the deep network is used in all submissions and fused with other modalities.

Event detection without examples For the event detection submission without using any video training examples, we employ a semantic embedding space to translate video-level concept probabilities into event-specific concepts, as also suggested in [2, 6]. The probabilities are computed by averaging frame-level scores from the probability layer of the deep network. The event-specific concepts are taken as the top-ranking terms from the event kit, based on *tf-idf*. The embedding space is a word2vec model [11]. The score of a test video is calculated as the maximum concept score across the event-specific concepts. To improve performance, we a apply a transformation that re-weights concepts based on concept inter-relatedness. This creates a higher prior for the concepts integral to the event. We use the similarity in the word2vec space to generate these weights.

Event detection with ten examples For the event detection submission based on ten examples, we consider two runs. A run using only the deep learning features and a fusion run with several other modalities. For the deep learning features, we compute frame representations twice per second at both the pool5 layer and the probability layer. For both layers, the features are averaged per video and then normalized. A histogram intersection kernel SVM model is trained on the representations from both layers and the scores for a test video are summed. For the fusion, we combine the two deep learning features with two additional modalities. The first modality is based on motion features. MBH and HOG descriptors are computed along improved dense trajectories for each video [21]. The motion descriptors are then aggregated into a video representation using Fisher Vectors [14] and classified using a linear SVM. The second modality is based on audio features. MFCC coefficients and their first and second order derivatives are computed in each video and again aggregated using Fisher Vectors. Here, a histogram intersection kernel SVM model is trained on the audio representations. All four models are fused by summing the scores.

Event detection with hundred examples For the event detection submission based on hundred examples, we also consider a run based on deep learning features only and a fusion run. The deep learning run is identical to the ten example run. For the fusion, we use the four representations as explained above, along with a fifth representation based on the bag-of-fragments model [10]. The bag-of-fragments model re-uses the pool5 layer for the frame representations. For each event, the most discriminative video fragments are discovered from the hundred training examples and these fragments are max-pooled over a video to obtain the fragment-based video representation.

3.1 Submitted Runs

For event detection without examples, our system yields an inferred Average Precision score of 0.039 on the full test set. The main results for ten and hundred examples are shown in Figure 4 using the Mean Inferred AP score. For both the ad-hoc and pre-specified runs, our system is the top performer. For the ten ad hoc examples, our system obtains a score of 0.425. For the ten pre-specified examples, our fusion run yields the best overall result, while the run using only the deep learning features is competitive. Finally, for event detection with hundred pre-specified examples, our fusion run is the top performer and the run based on deep learning features only is the runner-up, further indicating its effectiveness.

Acknowledgments

The authors are grateful to NIST and the TRECVID coordinators for the benchmark organization effort. The University of Amsterdam acknowledges support by the STW STORY project and the Dutch national program COMMIT.

References

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

[2] S. Cappallo, T. Mensink, and C. G. M. Snoek. Image2emoji: Zero-shot emoji prediction for visual media. In *MM*, 2015.

[3] 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.

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

[5] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015.

[6] M. Jain, J. C. van Gemert, T. Mensink, and C. G. M. Snoek. Objects2action: Classifying and localizing actions without any video example. In *ICCV*, 2015.

[7] M. Jain, J. C. van Gemert, and C. G. M. Snoek. What do 15,000 object categories tell us about classifying and localizing actions? In *CVPR*, 2015.

[8] A. Krizhevsky, I. Sutskever, and G. Hinton. ImageNet classification using deep convolutional neural networks. In *NIPS*, 2012.

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

[10] P. Mettes, J. C. van Gemert, S. Cappallo, T. Mensink, and C. G. M. Snoek. Bag-of-fragments: Selecting and encoding video fragments for event detection and recounting. In *ICMR*, 2015.

[11] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*, 2013.

[12] P. Over, G. Awad, M. Michel, J. Fiscus, G. Sanders, W. Kraaij, A. F. Smeaton, G. Quenot, and R. Ordelman. Trecvid 2015 – an overview of the goals, tasks, data, evaluation mechanisms and metrics. In *TRECVID*, 2015.

[13] 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. *IJCV*, 115(3):211–252, 2015.

[14] J. Sánchez, F. Perronnin, T. Mensink, and J. Verbeek. Image classification with the fisher vector: Theory and practice. *IJCV*, 105(3):222–245, 2013.

[15] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.

[16] 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, and A. W. M. Smeulders. MediaMill at TRECVID 2014: Searching concepts, objects, instances and events in video. In *TRECVID*, 2014.

[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] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *CVPR*, 2015.

[19] 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.

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

[21] H. Wang and C. Schmid. Action recognition with improved trajectories. In *ICCV*, 2013.

[22] R. Wu, S. Yan, Y. Shan, Q. Dang, and G. Sun. Deep Image: Scaling up Image Recognition. *CoRR*, 2015.