Improving Bag-of-Features Action Recognition with Non-local Cues

Muhammad Muneeb Ullah
Muhammad.Muneeb.Ullah@inria.fr
Sobhan Naderi Parizi
Sobhan.Naderi_Parizi@inria.fr
Ivan Laptev
Ivan.Laptev@inria.fr

INRIA - Willow Project
Laboratoire d'Informatique
École Normale Supérieure
CNRS/ENS/INRIA (UMR 8548)

Abstract

Local space-time features have recently shown promising results within Bag-of-Features (BoF) approach to action recognition in video. Pure local features and descriptors, however, provide only limited discriminative power implying ambiguity among features and sub-optimal classification performance. In this work, we propose to disambiguate local space-time features and to improve action recognition by integrating additional non-local cues with BoF representation. For this purpose, we decompose video into region classes and augment local features with corresponding region-class labels. In particular, we investigate unsupervised and supervised video segmentation using (i) motion-based foreground segmentation, (ii) person detection, (iii) static action detection and (iv) object detection. While such segmentation methods might be imperfect, they provide complementary region-level information to local features. We demonstrate how this information can be integrated with BoF representations in a kernel-combination framework. We evaluate our method on the recent and challenging Hollywood-2 action dataset and demonstrate significant improvements.

1 Introduction

Local video descriptors in combination with Bag-of-Features (BoF) video classification have been shown successful for the task of action recognition [5, 10, 13, 15, 18, 21]. Pure local descriptors, however, should balance a trade-off between the discriminative power and the invariance needed to overcome irrelevant variations in video due to e.g. camera motion, lighting changes, projective effects and background clutter. Limited discriminative power of local descriptors may imply ambiguity of video representations and the resulting decrease of recognition performance.

The goal of this work is to improve discriminative power of local video features by integrating non-local cues available at the region-level of a video. For this purpose, we decompose video into region classes and augment local features with corresponding region-class labels. For motivation, consider regions of a parking lot and side walks in Figure 1. Such regions are likely to correlate with specific actions such as opening a trunk and running. As a consequence, propagating region labels to the local feature level in this example is expected to increase discriminative power of features with respect to particular actions.

BMVC 2010 doi:10.5244/C.24.95

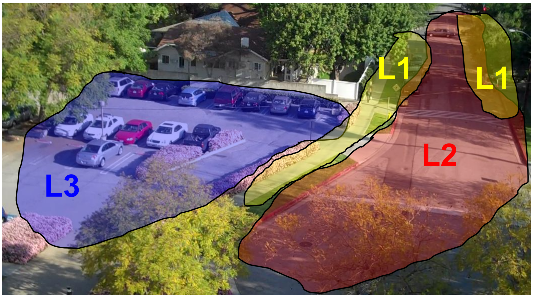

Figure 1: Regions in video such as road, side walk and parking lot frequently co-occur with specific actions (e.g. driving, running, opening a trunk) and may provide informative priors for action recognition.

To decompose a video into region classes, we in this paper resort to multiple and readily-available segmentation methods. In particular, we investigate unsupervised and supervised video segmentation using (i) motion-based foreground separation, (ii) person detection, (iii) static action detection and (iv) object detection. While such segmentation methods might be imperfect, they provides complementary region-level information to local features. Moreover, segmentation methods trained on additional training data (e.g. person and object detection) will introduce additional supervision into the BoF framework and will potentially increase its discriminative power.

Using different types of regions, we construct alternative video representations from the original set of local spatio-temporal features. We exploit complementarity of such representations and combine them within a multi-channel SVM framework [22]. We evaluate our method on the challenging Hollywood-2 human actions dataset [13] and demonstrate significant improvement with respect to the state of the art.

In summary, contributions of our work concern (i) increase of discriminative power of local features and associated BoF representations using region-level information and (ii) introduction of additional supervision into the BoF framework in the form of pre-trained region segmentation. The rest of the paper is organized as follows. Section 2 describes the BoF framework and its proposed extension. Section 3 presents details of alternative segmentation methods used in this work. Section 4 presents results while Section 5 concludes the paper.

2 Bag-of-Features with non-local cues

This section presents details of the BoF framework for action recognition and its extension.

2.1 Local features

We follow previous methods for action recognition [10, 18] and build upon Bag-of-Features (BoF) approach using local space-time features. To extract local features in video, we use the on-line implementation[1] of Spatio-temporal Interest Points (STIP) [9] combined with HOG/HOF descriptors [10]. Local features are extracted at multiple scale levels in space-time video pyramid. HOG and HOF descriptors are computed in the local space-time neighborhood of each feature and correspond to position-dependent histograms of spatial gradient and optical flow respectively. We concatenate HOG and HOF descriptors into a single vector describing appearance and motion of local neighborhoods.

[1] http://www.irisa.fr/vista/Equipe/People/Laptev/download.html

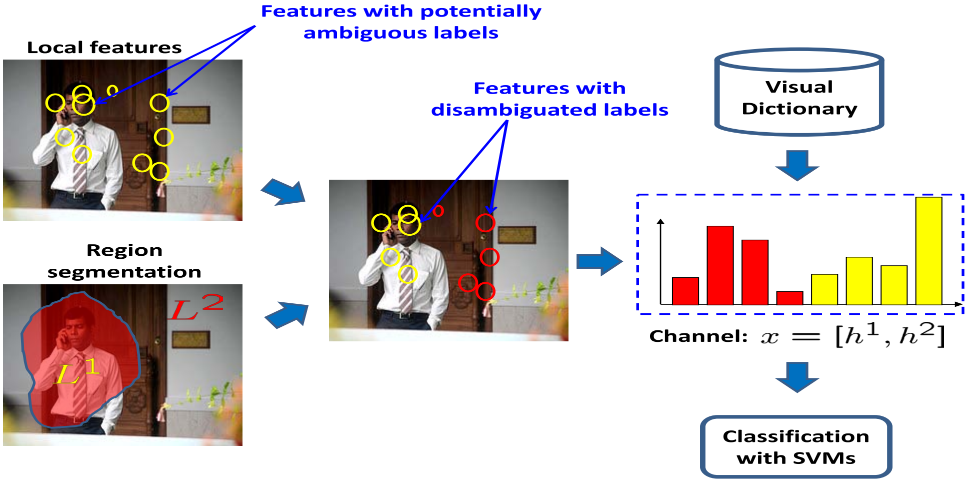
Features with potentially ambiguous labels
Local features
Features with disambiguated labels
Visual Dictionary
Region segmentation
L^2
L^1
Channel: $x = [h^1, h^2]$
Classification with SVMs

Figure 2: An illustration of our approach to disambiguate local descriptors with the help of semantic regions in video.

2.2 BoF representation

In the BoF framework, a video sequence is represented as a normalized frequency histogram of local space-time features. The histogram is computed over labels (or visual words [19]) associated with each local feature. Feature labels are commonly obtained by quantizing local feature descriptors according to a pre-learned dictionary. Following previous work [4, 10, 13, 19, 21], we construct a visual dictionary using K-Means with $K = 4000$ visual words. While K-Means is a conceptually simple and unsupervised approach to feature quantization, previous work [7, 14] aimed to improve image classification tasks by constructing *supervised* dictionaries. Here we follow [14] and use ERC-Forest to construct supervised visual dictionary for action classification.

ERC-Forest is an ensemble of randomly created clustering trees [14]. It predicts class labels c from local feature descriptors $\mathbf{d}$. It benefits from labeled training set $J = \{(\mathbf{d}_n, c_n), n = 1, \cdots, N\}$ with N descriptors $\mathbf{d}$ associated with class labels c and recursively builds random trees in a top-down manner. At each node, the labeled training set is divided into two halves such that the classes are separated well by maximizing the Shannon entropy:

$$S_c(J,T) = \frac{2 \cdot I_{C,T}(J)}{H_C(J) + H_T(J)} \tag{1}$$

where H_C denotes the entropy of the class distribution in J, H_T is the split entropy of the test T which splits the data into two partitions, and $I_{C,T}$ is the mutual information of the split (see [14] for further details). Following [14] we use ERC-Forest to build supervised visual vocabulary for local space-time features. We construct $M = 5$ multiple trees with 1000 leaf nodes each and assign M labels to each local feature according to each tree. ERC-Forests have been previously employed for image classification [12, 14, 16]. In Section 4 we demonstrate ERC-Forest to improve action recognition performance compared to K-means visual dictionary.

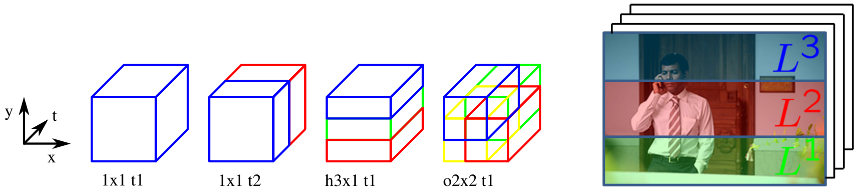

Figure 3: Left: Examples of spatio-temporal grids, Right: Illustration of video decomposition according to $h3 \times 1\ t1$ grid.

2.3 Extended BoF representation

We propose to extend BoF representation and to decompose video into a set of regions r assigned to labels l, $l \in \{L^1, \ldots, L^M\}$. A separate BoF histogram h^i is accumulated from quantized features within all regions with labels L^i. Following the terminology of [11], video signature, i.e. a *channel* is then constructed by concatenating BoF histograms for all region labels, i.e. $x = [h^1, \ldots, h^M]$ as illustrated in Figure 2. In this paper, we investigate different types of channels obtained with alternative video segmentation methods described in Section 3.

2.4 Mutli-channel SVM

For action classification we use non-linear Support Vector Machine (SVM)[3] with RBF kernel. To investigate combination of different channels, we use multi-channel kernel [22]

$$K(x_i, x_j) = exp\left(-\sum_c \frac{1}{\Omega_c} D\left(x_i^c, x_j^c\right)\right) \tag{2}$$

where $D(x_i^c, x_j^c)$ is χ^2 distance defined on histogram representations x_i^c, x_j^c obtained from videos i, j using feature channel c. We use normalization factor Ω_c computed as average channel distance [22]. For multi-class classification we use one-against-all approach.

3 Video segmentation

In this section, we describe alternative methods for decomposing video into region classes and providing means for disambiguating local features.

3.1 Spatio-temporal grids

Spatio-temporal video grids were introduced in [10] and showed promising results for action recognition. The basic idea is to divide a video into a set of predefined spatio-temporal regions. We follow this approach and use 24 spatio-temporal grids defined by the combination of six spatial subdivisions of a video: 1×1, 2×2, $h3 \times 1$, $v1 \times 3$, 3×3, $o2 \times 2$ and four temporal ones: $t1$, $t2$, $t3$ and $ot2$.[2] Figure 3 illustrates a few examples of grid configurations. All the grids together define a set of 24 channels which we denote by $STGrid24$.

[2] "1" stands for no subdivision, h, v and t denote subdivisions along horizontal, vertical and temporal axes of the video volume. "o" stands for cells with 50% overlap of their spatial or temporal extents.

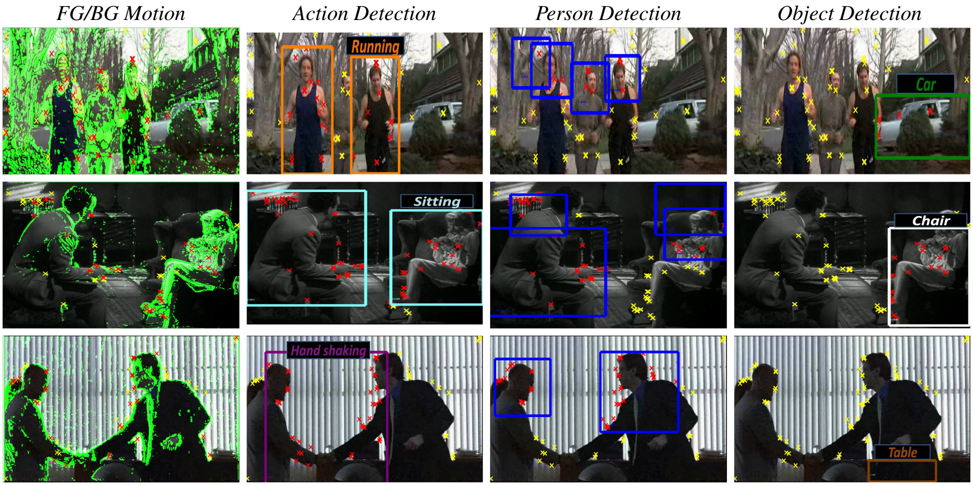

Figure 4: Illustration of proposed semantic region extraction in video according to (from left to right): motion region segmentation, action detection, person detection and object detection. Correct segmentation separates local features into meaningful groups denoted by yellow and red crosses. We also illustrated failures of automatic segmentation due to false negative detections (see e.g. missed running action in the first row) and false positive detections (see e.g. incorrect table detection in the third row).

3.2 Foreground/background motion segmentation

Segmenting local descriptors based on the foreground (FG) and background (BG) motions in video can be valuable in order to separate foreground features which are more likely to belong to the action from the background features which can help action recognition by capturing scene context. We use the Motion2D software [17] [3] to estimate regions of dominant motion in a video sequence. We then threshold (with four threshold values: 127, 150, 170, 200) the estimated foreground likelyhood maps and generate FG/BG masks. We use these masks to segment local descriptors into FG and BG classes. Figure 4 (1st column) shows FG masks (in green) together with the segmented features (in yellow and red). By separating features and building feature histograms according to FG and BG regions as well as for four different threshold values, we obtain 8 channels. We will refer to these eight channels as $Motion8$.

3.3 Action detection

The ability to localize actions in a video can be helpful for separating action-specific descriptors and thus building less ambiguous BoF representations for a particular action. Of course, all the remaining descriptors that belong to the background of action can form another complementary channel by capturing the context information.

The idea is to train an action specific detector on still images collected from the Internet and to perform action detections on the Hollywood-2 video sequences. Depending upon the availability of sufficient amount of action samples on the Internet, we investigate the idea for the following action classes: answering the phone, hugging, hand shaking, kissing, running, eating, driving a car, and sitting on sofa/chair. The last class corresponds to the action classes:

[3] http://www.irisa.fr/vista/Motion2D

Figure 5: Sample images collected from the Internet used to train the action detectors.

sitting down, standing up, and *sitting up*. Figure 5 presents sample images collected from the Internet. We train Felzenszwalb's object detector [6] for each action class (using 100-170 positive and approximately 9000 negative images for training) and run detector on the frames of Hollywood-2 videos (see Figure 4, 2nd column). The returned bounding boxes segment video into FG/BG corresponding to Action/Non-action regions. We then perform the following steps:

1. Threshold bounding boxes with six threshold values θ and divide each corresponding FG region into a 1×1 or 2×2 grid.

2. Compute 12 channels for six threshold values and two types of grid, i.e. $x_{\theta,1\times1} = \left[h^1, h^2\right]$ and $x_{\theta,2\times2} = \left[h^1, h^2, h^3, h^4, h^5\right]$.

We will refer to the 12 obtained channels as *Action12* for each of the eight aforementioned action classes.

3.4 Person detection

Separation of local descriptors on the basis of person/non-person region segmentation should potentially provide less ambiguous and more discriminative BoF representation for action recognition. We use the Calvin upper-body detector [1] which is a combination of the Felzenszwalb's object detector [6] and the Viola-Jones' face detector [20]. This detector returns bounding boxes fitting the head and upper half of the torso of the person (see Figure 4, 3rd column), which segment video into FG/BG corresponding to Person/Non-person regions. Following the steps of Section 3.3, we generate 12 channels. We will refer to these channels as *Person12*.

3.5 Object detection

Objects can provide a valuable context information for recognizing actions in video. For instance, the object *car* can be helpful to recognize the actions *driving a car* and *getting out of a car*, and the objects *chair* and *sofa* can be helpful for the classes *sitting down* and *standing up*. We investigate this concept by using Felzenszwalb's object detectors [6] [4] trained for the following object classes: *car, chair, table* and *sofa*, and perform separate detections on the Hollywood-2 sequences (see Figure 4, 4th column). The returned bounding boxes divide video into FG/BG corresponding to Object/Non-object regions. Again, following the steps of Section 3.3, we compute 12 channels per object class. We will refer to the corresponding 12 channels for each object class as *Objects12*.

[4] We use object detectors trained by the authors on VOC2008 dataset.

AnswerPhone DriveCar HandShake HugPerson Eat FightPerson

GetOutCar StandUp Run SitDown SitUp Kiss

Figure 6: Sample frames from video sequences of Hollywood-2 human actions dataset.

Channels	Performance (mean AP)
BoF with K-Means	0.479
BoF with ERC-Forest	**0.486**
STGrid24 with K-Means	0.504
STGrid24 with ERC-Forest	**0.518**

Table 1: Overall performance by the baseline channels.

4 Experimental results

This section presents our experimental results in detail. All the experiments have been performed on the Hollywood-2 dataset [13] using the *clean* training subset. The dataset is comprised of video sequences collected from 69 different Hollywood movies. It contains 12 action classes: answering a phone, driving a car, eating, fighting, getting out of a car, hand shaking, hugging, kissing, running, sitting down, sitting up, and standing up (see Figure 6). In total, there are 1707 action samples divided into a training set (823 sequences) and a test set (884 sequences). Train and test sequences are collected from different movies. For performance evaluation, we report the average precisions (APs) for individual classes as well as the mean average precision (mAP) over all the action classes.

4.1 Baseline performance

To get a baseline, we performed experiments with (i) the standard BoF method, and (ii) STGrid24 channels using the K-Means as well as ERC-Forest generated visual dictionaries. Table 1 compares their mean average precisions. It turns out that STGrid24 channel improves upon the standard BoF approach (consistent with the findings in [10]), and the two methods perform better (about 1%) with ERC-Forest generated dictionary. Therefore, from here on, we only present results obtained with ERC-Forest dictionary. Note that our baseline result for BoF with K-Means (mAP 0.479) is comparable to the best result (mAP 0.476) previously reported on this dataset in [21].

4.2 Improvements with channel combination

The performance by STGrid24 channels (0.518) will serve as the baseline result here. Table 2 (1st portion) reports results for the new channels (introduced in Section 3), with Action12 channels having the highest mAP (0.528). While most of our new channels do not outper-

Channels	Performance (mean AP)
Motion8	0.504
Person12	0.493
Objects12	0.499
Action12	**0.528**
STGrid24 + Motion8	0.532
STGrid24 + Person12	0.532
STGrid24 + Objects12	0.530
STGrid24 + Action12	**0.557**
STGrid24 + Motion8 + Action12 + Person12 + Objects12	**0.553**

Table 2: Overall performance of different channels individually as well as in combination with other complementary channels.

form the baseline, the advantage of all new channels becomes apparent when combined with the baseline channels STGrid24. As can be seen from Table 2, new channels combined with STGrid24 not only improve upon their individual performance but also improve the baseline result up to 0.557. This can be explained by the complementarity of channels adding different information to the BoF representation. Note, however, that the integration of Action12, Person12 and Object12 channels implies the use of additional training data which makes corresponding results not directly comparable to previous results reported on Hollywood-2 dataset. When combining all the four new channels with STGrid24 channels, we obtain 0.553 mAP, which is a significant improvement over the baseline. We also note that the channel combination STGrid24+Action12 (0.557) slightly outperforms combination of all channels. This behavior highlights the need for more sophisticated methods for kernel combination compared to the simple multi-channel approach (product of kernels) considered in this paper. We have tried learning kernel combination using Multiple Kernel Learning (MKL) framework [2], however, similar to previous findings [8], MKL did not improve results in our case.

In table 3, we present per-class average precision values corresponding to the baseline channels as well as the best performing new channels and their combinations. We note improvement of eleven out of twelve action classes (APs in bold in the last two columns) when combining new channels with the baseline channels. Distribution of the best class APs across three columns (corresponding to different channel(s)) points out the need to devise some sophisticated technique for class-specific channel(s) selection. Moreover, we observe significant performance gain for some classes (e.g. HandShake) and a relatively small improvement for some others classes (e.g. DriveCar). One reason, in addition to varying intra-class variation for each action class, could be the quality of the additional training data that we used to train different detectors. For instance, for HandShake, the collected images were quite representative of the Hollywood-2 samples (see HandShake in Figure 5 and 6). Whereas, for DriveCar, majority of the collected images were side-viewed, where in fact DriveCar is mostly frontal-viewed in Hollywood-2 (see DriveCar in Figure 5 and 6). We therefore believe that performance for the lower performing classes can be further improved by using better quality training images. Moreover, although the mean AP performance by the final channel combination (0.553) is slightly lower than that by the STGrid24+Action12 channels (0.557), yet it achieves the best results for seven action classes (APs in bold in the last column).

Channels	BoF	STGrid24	Action12	STGrid24 + Action12	STGrid24 + Motion8 + Action12 + Person12 + Objects12
mean AP	0.486	0.518	0.528	**0.557**	0.553
AnswerPhone	0.157	0.259	0.208	**0.263**	0.248
DriveCar	0.876	0.859	0.869	0.865	**0.881**
Eat	0.548	0.564	0.574	0.592	**0.614**
FightPerson	0.739	0.749	0.757	0.762	**0.765**
GetOutCar	0.334	0.440	0.383	0.457	**0.474**
HandShake	0.200	0.297	0.457	**0.497**	0.384
HugPerson	0.378	**0.461**	0.408	0.454	0.446
Kiss	0.521	0.550	0.560	0.590	**0.615**
Run	0.711	0.694	0.732	0.720	**0.743**
SitDown	0.590	0.589	0.596	**0.624**	0.613
SitUp	0.239	0.184	0.241	**0.275**	0.255
StandUp	0.533	0.574	0.549	0.588	**0.604**

Table 3: Per-class AP performance by different channels and channel combinations.

5 Conclusions

We have presented an extension to the standard BoF approach for classifying human actions in realistic videos. Our main idea is to segment videos into semantically meaningful regions (both spatially and temporally) and then to compute histogram of local features for each region separately. As we have shown experimentally, this separation helps to get significant improvement over our strong baseline by disambiguating histograms of local features. Our framework also enables introduction of additional supervision into BoF action classification in the form of region detectors that could be trained on related tasks. Our method thus provides the flexibility to utilize additional training data (Web images, annotation and images from PASCAL VOC dataset, etc.) to mitigate the problem of having limited training data within the Hollywood-2 dataset.

While we have shown significant improvement with the proposed method, we believe even higher improvement could be obtained by using a more appropriate procedure for combining different channels. Our main focus in this paper has been to demonstrate a notion of semantic level video segmentation and its advantage for action recognition. We plan to optimize channel combination and better adapt our method for each particular action class in the future work using frameworks such as Multiple Kernel Learning [2, 8].

6 Acknowledgements

We are grateful for financial support provided by the Quaero Programme, funded by OSEO as well as by the MSR-INRIA laboratory.

References

[1] http://www.vision.ee.ethz.ch/ calvin/calvin_upperbody_detector/.

[2] S. Canu A. Rakotomamonjy, F. Bach and Y. Grandvalet. Simplemkl. *Journal of Machine Learning Research*, 9:2491–2521, 2008.

[3] C. C. Chang and C. J. Lin. LIBSVM: a library for support vector machines, 2001.

[4] G. Csurka, C. Dance, L. Fan, J. Williamowski, and C. Bray. Visual categorization with bags of keypoints. In *ECCV Workshop Statistical Learning in Computer Vision*, 2004.

[5] P. Dollár, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. In *VS-PETS*, pages 65–72, 2005.

[6] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2009.

[7] B. Fulkerson, A. Vedaldi, and S. Soatto. Localizing objects with smart dictionaries. In *Proc. European Conference on Computer Vision*, pages I: 179–192, 2008.

[8] P. Gehler and S. Nowozin. On feature combination methods for multiclass object classification. In *Proc. International Conference on Computer Vision*, 2009.

[9] I. Laptev. On space-time interest points. *International Journal of Computer Vision*, 64 (2/3):107–123, 2005.

[10] I. Laptev, M. Marszałek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *Proc. Computer Vision and Pattern Recognition*, 2008.

[11] S. Lazebnik, C. Schmid, and J. Ponce. A sparse texture representation using local affine regions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27: 1265–1278, 2005.

[12] V. Lepetit, P. Lagger, and P. Fua. Randomized trees for real-time keypoint recognition. In *Proc. Computer Vision and Pattern Recognition*, 2005.

[13] M. Marszałek, I. Laptev, and C. Schmid. Actions in context. In *Proc. Computer Vision and Pattern Recognition*, 2009.

[14] F. Moosmann, E. Nowak, and F. Jurie. Randomized clustering forests for image classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(9): 1632–1646, 2008.

[15] J.C. Niebles, H. Wang, and L. Fei-Fei. Unsupervised learning of human action categories using spatial-temporal words. In *Proc. British Machine Vision Conference*, 2006.

[16] D. Nistér and H. Stewénius. Scalable recognition with a vocabulary tree. In *Proc. Computer Vision and Pattern Recognition*, 2006.

[17] J. M. Odobez and P. Bouthemy. Robust multiresolution estimation of parametric motion models. *Journal of Visual Comm. and Image Representation*, 6(4):348–365, 1995.

[18] C. Schüldt, I. Laptev, and B. Caputo. Recognizing human actions: A local svm approach. In *Proc. International Conference on Pattern Recognition*, 2004.

[19] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *Proc. International Conference on Computer Vision*, 2003.

[20] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proc. Computer Vision and Pattern Recognition*, 2001.

[21] H. Wang, M. M. Ullah, A. Klašer, I. Laptev, and C. Schmid. Evaluation of local spatio-temporal features for action recognition. In *Proc. British Machine Vision Conference*, 2009.

[22] J. Zhang, M. Marszałek, M. Lazebnik, and C. Schmid. Local features and kernel for classification of texture and object categories: A comprehensive study. *International Journal of Computer Vision*, 73:213–238, 2007.