Feature Tracking and Motion Compensation for Action Recognition

H. Uemura[1,2] S. Ishikawa[1] K. Mikolajczyk[2]
[1]Kyushu Institute of Technology, Japan, [2]University of Surrey, UK
uemura, ishikawa@ss10.cntl.kyutech.ac.jp k.mikolajczyk@surrey.ac.uk

Abstract

This paper discusses an approach to human action recognition via local feature tracking and robust estimation of background motion. The main contribution is a robust feature extraction algorithm based on KLT tracker and SIFT as well as a method for estimating dominant planes in the scene. Multiple interest point detectors are used to provide large number of features for every frame. The motion vectors for the features are estimated using optical flow and SIFT based matching. The features are combined with image segmentation to estimate dominant homographies, and then separated into static and moving ones regardless the camera motion. The action recognition approach can handle camera motion, zoom, human appearance variations, background clutter and occlusion. The motion compensation shows very good accuracy on a number of test sequences. The recognition system is extensively compared to state-of-the art action recognition methods and the results are improved.

1 Introduction

Significant progress has been made in classification of static scenes and action recognition is receiving more and more attention in computer vision community. Many existing methods [2, 5, 16, 19, 22, 25] obtain high classification score for simple action sequences with exaggerated motion, static and uniform background in controlled environment. Real action scenes represent a challenge which is rarely addressed in the literature as it is hard to compute a visual correspondence between the lab controlled action and the real action of the same category as their appearance, motion and clutter significantly differ. Recently, a space-time window descriptors combined with SVM were applied to classify actions in real movie sequences in [8], but the camera motion was not addressed there. The action recognition approach from [27], claimed to be the first one to deal with camera motion, explored multiview geometry. This solution however requires multiple camera setup or very similar actions captured from different viewpoints, which limits the range of possible applications. Other work relevant to camera motion estimation and dominant plane segmentation perform combined motion and image segmentation [24] or plane estimation [26], but these are concerned with either precise segmentation of moving regions or accurate reconstruction of 3D scene structure. Iterative estimation of dominant planes based on optical flow was also done for robot navigation in [20].

Frequently followed class of approaches to action recognition is based on spatio-temporal features computed globally [4] or locally [2, 5, 17, 22]. Global methods can-

not handle multiple actions performed simultaneously or localize them spatially. Spatio-temporal interest points [7] result in a very compact representation but are too few to build action models robust to camera motion, background clutter, occlusion, motion blur etc. It was demonstrated in [25] that as few as 5 to 25 spatio-temporal interest points give high recognition performance on standard test data. We argue that this number is insufficient for real actions where many of the extracted features are erroneous due to camera motion and background clutter. An interest point detector was combined with Gabor filter in [2] or a hybrid of spatio-temporal and static features was used in [17] to extract more action descriptors which improved the recognition performance.

The main contribution of this paper is a feature extraction approach that provides local appearance combined with motion information extracted from a video regardless the camera motion and background clutter. There exist advanced motion estimation methods [6, 24] which can handle camera motion and complex scenes but the advantage of our approach is its simplicity and integration with the action recognition system. The dominant motion compensation is based on the same features and motion tracks as the action recognition, which makes it also more efficient. Our action recognition approach follows the standard paradigm using local features, vocabulary based representation and voting, which is inspired by the results from the static object recognition [3]. Such systems have been very successful in retrieval and recognition of static images [14, 18]. Compared to existing approaches which usually focus on one of the issues associated with action recognition and make strong assumptions on static camera and uniform background, our system can deal with appearance variations, camera motion, scale change, background clutter and occlusion. Furthermore, unlike methods based on a single, small and flat codebook followed by SVM [2, 17, 22], our approach is based on many codebook trees and an efficient search method. In contrast to all the other systems the method proposed here can classify the whole sequence as well as recognize and localize actions within few frames of the sequence.

1.1 Overview

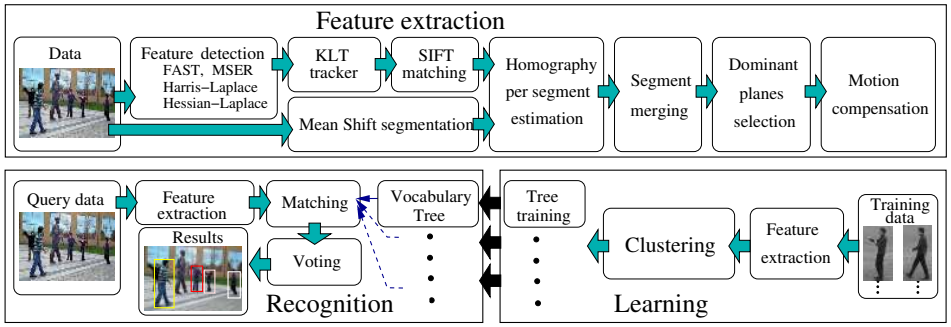


Figure 1. The main components of the feature extraction and the recognition system.

The principal components of our action recognition system are illustrated in Fig. 1. The main focus of this paper is on feature extraction and camera motion compensation via dominant plane segmentation displayed in Fig. 1(top). The feature extraction and tracking is discussed in section 2. We then explain in detail the dominant plane segmentation and motion compensation in section 3. The feature extraction module operates during training and recognition which are illustrated in Fig. 1(bottom) and discussed in section 4.

2 Feature Tracking

In this section we discuss features that are extracted from video frames and then tracked throughout several frames. These features and motion tracks are used by subsequent components of the system to estimate the plane homographies and to represent the actions.

2.1 Interest point detection

The features are provided by a number of state-of-the art interest point detectors. We have selected complementary detectors based on the results presented in [13]. In these experiments MSER detector [12] gave the most stable points in terms of invariance to viewpoint change and robustness to various types of noise. Harris-Laplace and Hessian-Laplace [13] are scale invariant detectors based on similar principles and provide corners and blobs, respectively. Finally, we use recently introduced FAST detector [21]. This detector gives a large number of features, which provide good coverage of the image. FAST features are not scale invariant but robust enough to handle small frame-to-frame transformations. In addition to the features provided by all the detectors we select points distributed on a regular grid with a step of 10 pixels, if a non zero image gradient is in the neighborhood of the point. This provides additional sparsely distributed points which are then used for dominant plane segmentation. Typically, up to 3000 features per frame are extracted. An example frame with a subset of detected interest points is displayed in Fig. 2(a).

2.2 Tracking

(a) *(b)* *(c)* *(d)*

Figure 2. A subset of detected point features and their tracks over 10 frames based on KLT tracker and SIFT. The frames illustrate tracks for (a) static camera, (b) handheld camera, (c) panning, (d) zooming.

KLT tracker. To represent an action it is essential to use a reliable local motion estimator. There has been a lot of work in the area of feature tracking with important contribution from [23]. We follow this approach and use a pyramidal implementation of the classical KLT tracker. The optical flow is computed at the lowest level of the pyramid and then propagated to the higher levels. Thus the lower level provides an initialization for tracking at the higher resolution. The number of pyramid levels is 4 and the used patch size is 15x15 pixels. These parameters provide a good tradeoff between the accuracy of the motion estimation and the robustness to large motion, zoom, and light changes.

SIFT matching. In addition to the optical flow based tracking, which is efficient but often gives erroneous motion vectors, we apply a SIFT [11] based verification. SIFT descriptors are computed for all interest points extracted from the sequence. The point neighborhood is defined by the scale of feature with the minimum size of 15x15 pixels. Every match candidate indicated by the optical flow is verified by computing similarity

score between SIFT features with the Bhattacharyya distance. It gives a similarity value in the range 0.0-1.0 which is convenient for estimating the threshold, but the L_2 norm could be used here as well. If the distance is larger than the threshold the pair of matches is removed from the set. A substantial number of points which are due to occlusion and background clutter are removed, it is therefore crucial to start with a large number of features so that a sufficient number remains after tracking and SIFT matching. Typically there are up to 1500 features per frame from all the detectors. Fig. 2 shows example frames with feature tracks for various types of camera motion. Note that for the hand-held camera displayed in Fig. 2(b) there is noticable motion even though the camera was held still. This shows the necessity of using dominant motion compensation otherwise it dominates local action motion and makes the recognition extremely difficult.

3 Motion compensation

Given a number of features with associated motion vectors extracted from consecutive frames the problem is to separate the local motions characterizing the actions, from the dominant camera motion. Single plane assumption is too strong as there is often the ground plane and the background plane in the outdoor scenes or even more planes in the indoor scenes. This requires image segmentation into dominant motion planes which can then be used to correct the local motion vectors. We approach this problem by combining color based image segmentation with estimation of dominant homographies based on local features.

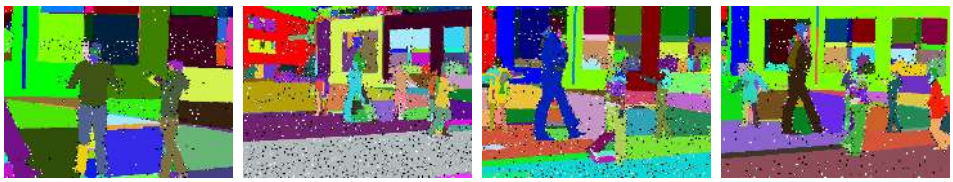


Figure 3. Mean Shift segmentation results with displayed features for different frames.

Image segmentation is done with Mean Shift [1] based on color. The Mean Shift is executed with the spatial window size of 11x11 pixels and the L_2 distance between the RGB colors. The purpose of the segmentation is to identify features that potentially belong to the same physical surface. Different planes are often separated by color or intensity gradients which are detected by the segmentation method. Figure 3 shows examples of segmented frames with displayed keypoint features. Given the detected features and the segmentation mask we allocate features to segments. A feature is allocated to a segment if a disk of 5 pixel radius centered on this feature overlaps with the segment. Feature-to-segment allocation is represented by $\mathbf{S_f}$ matrix (cf. Fig. 4(a)), where $\mathbf{S_f}(S_m, f_i) = 1$ indicates that feature f_i belongs to segment S_m. Note that a feature can belong to several segments.

Homography estimation is used here to model the dominant motion since perspective distortions are frequent in both indoor and outdoor scenes. The estimation is done for segments with more than 10 features by applying RANSAC to the features that belong to the segment. RANSAC samples 4 points at every iteration and estimates the homography. The homography is obtained if the number of inliers from the segment does not change for more than 10 iterations or the maximum number of 100 iterations is reached. After

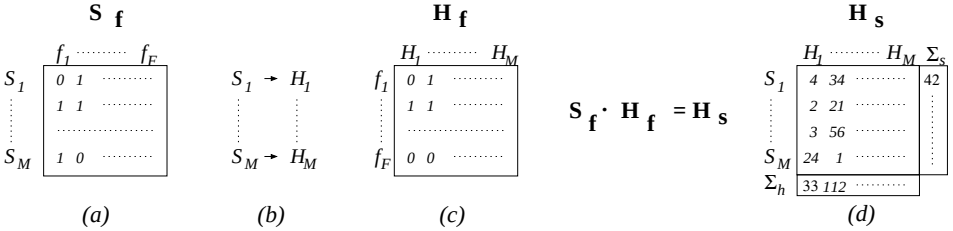


Figure 4. Inliers selection. Feature to segment and segment to homography labeling.

processing the segments we obtain a list of homographies (cf. Fig. 4)(b). The result of this operation is matrix $\mathbf{H_f}$ (cf. Fig. 4(c)) which indicates which features from the whole image are inliers to the homographies. Given the matrices the task is now to select the homography with the largest number of inliers. Matrix $\mathbf{H_S}$ is the product of matrices $\mathbf{S_f}$ and $\mathbf{H_f}$, and represents the number of inliers for every segment S_m and homography H_n (cf. Fig. 4(d)). The dominant homography is indicated by the column with the largest number of inliers: $\max(\sum_h)$. Given the dominant homography we iteratively merge segments which contain more than 80% of inliers to this homography $\mathbf{H_S}(S_m, H_n)/\sum_s \geq 0.8$, The merged segments are removed from matrix $\mathbf{S_f}$ by removing the corresponding rows. New matrix $\mathbf{H_S}$ is produced and the procedure is repeated to select the second dominant homography. This is repeated for 3 dominant homographies if there are more than 20% of the initial number of features remaining in $\mathbf{S_f}$ and $\mathbf{H_f}$ matrices. The remaining small segments are merged with the surrounding areas. These are usually the outliers representing local action motion. Fig. 5 shows the dominant plane segmentation for the frames presented in the other figures.

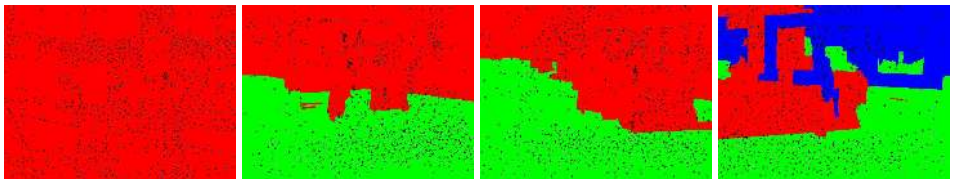


Figure 5. Segmentation into dominant planes.

Motion compensation is crucial when there is a background motion but it is often difficult to identify the foreground and the background from purely data driven segmentation in particular when there are close up views of human actions. Given the segmented dominant plane we use its homography to correct the motion of all local features that were allocated to this plane during the homography estimation. This is done for every dominant plane in the frame. The same holds for features on the boundaries between two dominant planes where both homographies are valid. The main risk with this approach is that one of the object-actions can be identified as a dominant plane and its motion would be then canceled. Some techniques avoiding this issue can be found in [6]. The impact of this on our action recognition results is less significant as: (1) we use only 3 dominant planes; (2) the most discriminative is the local motion. For example, the global forward motion in running or jogging is the main source of ambiguities between these actions, therefore canceling it can be beneficial in some cases.

Motion compensated features in our example frames are displayed in Fig. 6. It can be observed that only local motions remain which are then used to recognize the actions.

Figure 6. Motion compensated frames. Compare with frames in figure 2, respectively.

4 Action Recognition

Training. For action recognition we build on the system from [14], which is based on local features, vocabulary tree, and voting. The SIFT descriptors are extracted from the training data as described in sections 2 and 3. The vocabulary trees are then constructed with the agglomerative clustering similar to the one used in [9]. Initially each descriptor forms a cluster. Two nearest clusters in the whole set are merged at each iteration based on their Euclidean distance. We continue merging until the number of clusters is 1% of the initial number of descriptors. This results in a number of binary trees where every node is represented by the average of its children nodes. The trees are constructed in a similar way for every feature type provided by individual detectors. The leaves of the trees contain inverted files with occurrences on the training data. An occurrence includes a number of parameters: the position within object reference frame, the scale, the motion vector and the action category label. The leaf nodes of the trees can be considered a codebook similar to many other flat codebooks [2, 10, 16, 17, 22] however matching with vocabulary trees is much more efficient. Features from one motion track are in the same leaf cluster and a distribution of their motion vectors represents action motion.

Recognition. Every query feature is first compared to the top node of each tree of its type. If the distance from the query descriptor to the node center is less than a threshold the feature is accepted and compared to the children of this node. The nearest neighbor branch is followed then. This continues until the query feature reaches the leaf nodes of the tree. We then compare the motion of the feature to the occurrence vectors of the tree node. Separating appearance and motion information allows for more compact representation as the appearance can be shared between different actions. Thus, it increases matching efficiency when computing similarity score. An occurrence is allowed to cast a vote if the motion orientations agree within 10 degree margin. The corresponding bin in 3D voting space $(x, y, scale)$ is computed by comparing the occurrence position and scale in the leave node to the feature parameters [14]. This gives an estimate of the action center as well as indicates the category label. The votes from 5 consecutive video frames and from all trees are accumulated in the voting space. Local maxima in the voting space indicate the position, scale, and label of the action. The initial voting is based on features in motion only. This number is however insufficient to reliably recognize human actions. To improve the robustness of the method we also use the static features which increase the scores for initial motion based hypotheses. Thus the features with motion vectors focus the attention on candidate hypotheses and features with zero motion improve the ranking of the hypotheses. As previously demonstrated for still images in [14, 18] this simple recognition technique is very efficient and allows to use a large number of features, which make the system robust to background clutter, occlusion, camera motion, blur, low video quality etc. Further details on the tree construction and voting can be found in [15].

5 Experimental Results

This section discusses the experimental setup, results for motion compensation and performance of our action recognition system on different data.

5.1 Experimental Settings

Data. KTH action sequences were introduced in [22] and used in many action recognition papers [2, 16, 19, 25]. There are 6 human action categories: hand-clapping, hand-waving, boxing, running, jogging and walking. Each action is performed within a few seconds by individuals in rather uniform dark cloth on homogeneous background with static camera. There are 25 individuals in 4 indoor and outdoor scenarios, which results in 100 sequences per action. Example frames of the KTH sequences are displayed in Fig. 9(top row). We present our results for this data and compare to the other methods from the literature. However, recognition performance of other methods for the KTH data is already above the level of 90%, we therefore acquired a more challenging sequence of actions[1] which are included in the KTH set, but performed simultaneously with more complex background, occlusion and camera motion. Example frames are displayed in Fig. 9(middle and bottom rows). This sequence is used for testing only. To train and test the system we annotated every 5th frame of each sequence with bounding boxes using an interactive interface supported by color based Mean Shift tracking. In total, we annotated 11464 frames from 599 sequences of 6 KTH categories and 753 frames from multi-KTH sequence.

Performance measures. We evaluate the performance in a similar way to [2, 16, 19, 22, 25]. We refer to this test as 'classification'. In addition to that, we report results for detection of individual actions within frames, which we call 'localization'. The detection is correct if the intersection/union of the detected and the groundtruth bounding boxes is larger than 50% and the category label is correct. The detection results are presented with average precision, which is the area below precision-recall curve. All the experiments are done with leave-one-out cross-validation [2, 16, 25]. The results are averaged for all frames and all sequences of a given action.

5.2 Motion compensation

We first present the evaluation of the camera motion compensation. The experiment is done on 5 sequences[1] which show 2 indoor and 3 outdoor scenes with moving foreground objects[2], complex background, multiple dominant planes and camera motion. The sequences are annotated in a similar way to the other test data, by bounding boxes on humans performing actions. We extract features, perform tracking and dominant plane estimation. We compare the average magnitude of motion for foreground and background regions before and after motion compensation. We can observe in Fig. 7 a significant reduction of motion magnitude on the background while local foreground motion remains detectable and reliable. The background motion is reduced with factor 0.2 and the foreground with 0.7 only. The average ratio foreground/background magnitude before the compensation is 1.05 and increases to 4.1 for the compensated frames. This makes the action recognition task tractable even with significant camera motion.

[1] Available at http://personal.ee.surrey.ac.uk/Personal/K.Mikolajczyk/research.

[2] Sequence 1 is multi-KTH and sequence 4 contains multiple background planes without foreground motion.

	without motion compensation					with motion compensation					motion reduction
sequence	1	2	3	4	5	1	2	3	4	5	
foreground	6.3	7.5	9.2	.00	10.7	5.6	4.9	5.9	.00	5.8	0.7
background	3.2	9.6	12.9	3.7	14.1	0.7	1.1	2.2	.7	4.2	0.2

Figure 7. Motion compensation results for individual sequences. The table shows average motion magnitude in pixels.

5.3 Action recognition

Classification. We perform a classification experiment in order to compare the performance of our system to the other methods [2, 16, 25]. In this experiment we discard the action location information and assume there is only one action category in the sequence or none. Fig. 8(a) shows the confusion matrix for the KTH data. There is 2% of misclassification between hand clapping and waving. Higher classification error of 8% occurs

action / action	hand **clap**–ping	hand **wav**–ing	**box**–ing	**jog**–ging	**walk**–ing	**run**–ning
hand **clap**–ping	**.98**	.02	.00	.00	.00	.00
hand **wav**–ing	.02	**.96**	.00	.00	.00	.00
box–ing	.00	.00	**.98**	.00	.01	.00
jog–ing	.00	.00	.00	**.89**	.04	.08
walk–ing	.00	.00	.01	.04	**.94**	.03
run–ing	.00	.00	.00	.08	.03	**.87**

(a)

action / test	hand **clap**–ping	hand **wav**–ing	**box**–ing	**jog**–ging	**walk**–ing	**run**–ning
classification	.98	.96	.98	.89	.94	.87
state of the art	1.0 [25]	.93 [16]	1.0 [16]	.75 [25]	.90 [2]	.88 [16]
localization	.97	.96	.98	.80	.86	.79
motion features only	.83	.88	.84	.75	.81	.77
multi–KTH	.76	.81	.58	.51	.61	–
no motion compensation	.16	.18	.14	.31	.28	–

(b)

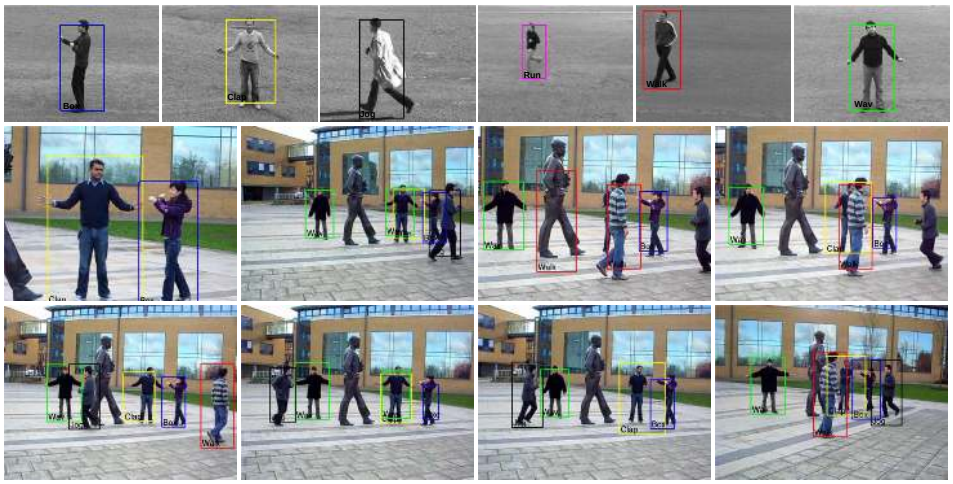
Figure 8. Human action recognition results. (a) Confusion matrix for KTH actions. (b) Results for KTH and multi-KTH sequences and comparison with other methods.

between jogging and running as well as walking of 4%. Other approaches also suffer from confusions between these categories. We improve the state-of-the art classification results for 3 of those categories, namely hand-waving, jogging and walking (see Fig. 8(b)). The results for other actions are comparable to the state-of-the art scores. However, in contrast to the other methods our system can also localize human actions and can handle various actions performed simultaneously. In addition to the class label it estimates the location and size of various actions, thus the number of possible false positives per image is very high compared to the classification systems from other papers. The results for localization test drop by 7-9% for walking, jogging and running mainly due to errors in size estimation (cf. Fig. 8(b)). Additional experiment shows the benefit of using the static features in addition to the moving ones as the performance for motion features only drops down.

For comparison we performed the localization test on multi-KTH sequence[3]. We used the KTH data to train the system and the detection results are displayed in Fig. 8(b) (multi-KTH). The results decrease by 15% for hand-waving up to by 40% for boxing. This demonstrates the difference in system performance one can expect when more realistic data is used. The drop is even more significant when no camera motion compensation is used (see Fig. 8(b) bottom row). Feature motion vectors are then unreliable and correct

[3]Demo sequence available at http://personal.ee.surrey.ac.uk/Personal/K.Mikolajczyk/research.

results are obtained only when the camera does not move. Fig. 9 shows example frames with the recognition results displayed by color boxes for different actions.



(a) *(b)* *(c)* *(d)*

Figure 9. Example results for KTH action sequences (top row) and for multi-KTH sequence (middle and bottom row). Different actions are color coded: red-walking, green-hand waving, yellow-hand clapping, blue-boxing, black-jogging, magenta-running. Note the scale and the viewpoint change from frame (a) to frame (d).

Conclusions

An approach to action recognition via local feature tracking and camera motion compensation has been introduced. The approach can recognize and localize various human actions. It deals with realistic video sequences with camera motion, background clutter and occlusion. The strength of the approach is in local features, robust tracking and combined image-motion segmentation. The system obtains an excellent performance on standard test data, compared to other approaches and improves state-of-the art results. We have also presented results on more challenging video sequence including various actions performed simultaneously in uncontrolled environment. Future research will investigate joint statistics of motion-appearance features and further use of other segmentation algorithms for motion compensation as well as recognition.

Acknowledgment. This research was supported by EU VIDI-Video IST-2-045547 and UK EPSRC EP/F0034 20/1 grants. We would like to thank Richard Bowden for his contribution to this work.

References

[1] D. Comaniciu and P. Meer. Mean Shift: A Robust Approach toward Feature Space Analysis. PAMI, 24(5):603 -619, 2002.

[2] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. In *VS-PETS*, 2005.

[3] M. Everingham and et. al. The PASCAL Visual Object Classes Challenge 2007.

[4] A. Efros, A. Berg, G. Mori, and J. Malik. Recognizing action at a distance. In *ICCV*, 2003.

[5] C. Fanti, L. Zelnik-Manor, and P. Perona. Hybrid models for human motion recognition. In *ICCV*, 2005.

[6] M. Irani and P. Anandan. A Unified Approach to Moving Object Detection in 2D and 3D Scenes. PAMI, 20(6):577–589, 1998.

[7] I. Laptev. On space-time interest points. IJCV, 64:107–123, 2005.

[8] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld Learning realistic human actions from movies In *ICCV*, 2007.

[9] B. Leibe, K. Mikolajczyk, B. Schiele. Efficient clustering and matching for object class recognition. In *BMVC*, 2006.

[10] B. Leibe, A. Leonardis, B. Schiele Robust object detection with interleaved categorization and segmentation. IJCV, 77(1-3):259–289, 2008.

[11] D. Lowe. Distinctive image features from scale-invariant keypoints. IJCV, 60(2):91–110, 2004.

[12] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In *BMVC*, 2002.

[13] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *PAMI*, 27(10):1615–1630, 2005.

[14] K. Mikolajczyk, B. Leibe, and B. Schiele. Multiple object class detection with a generative model. In *CVPR*, 2006.

[15] K. Mikolajczyk and H Uemura. Action Recognition with Motion-Apperance Vocabulary Forest. In *CVPR*, 2008.

[16] J.C. Niebles, H. Wang, and L. Fei-Fei. Unsupervised learning of human action categories using spatial-temporal words. In *BMVC*, 2006.

[17] J.C. Niebles and L. Fei-Fei. A hierarchical model of shape and appearance for human action classification. In *CVPR*, 2007.

[18] D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. In *CVPR*, 2006.

[19] S. Nowozin, G. Bakir, K. Tsuda. Discriminative subsequence mining for action classification. In *ICCV*, 2007.

[20] N. Ohnishi, A. Imiya. Model-Based Plane-Segmentation Using Optical Flow and Dominant Plane. In *LNCS*, vol. 4418, 2007.

[21] E. Rosten and T. Drummond Machine learning for high–speed corner detection In *ECCV*, 2006.

[22] C. Schuldt, I. Laptev, and B. Caputo. Recognizing human actions: A local SVM approach. In *ICPR*, 2004.

[23] J. Shi and C. Tomasi. Good Features to track. In *CVPR*, 1994.

[24] T. Veit, F. Cao, and P. Bouthemy. An a contrario decision framework for region-based motion detection. IJCV, 68(2):163-178, 2006

[25] S. Wong and R. Cipolla. Extracting spatiotemporal interest points using global information. In *ICCV*, 2007.

[26] A. Yang, S. Rao, A. Wagner, and Y. Ma. Segmentation of a Piece-Wise Planar Scene from Perspective Images. In *CVPR*, 2005.

[27] A. Yilmaz and M. Shah. Recognizing human actions in videos acquired by uncalibrated moving cameras. In *ICCV*, 2005.