Action Detection with Improved Dense Trajectories and Sliding Window

Zhixin Shu(✉), Kiwon Yun, and Dimitris Samaras

Stony Brook University, Stony Brook, NY 11794, USA
{zhshu,kyun,samaras}@cs.stonybrook.edu

Abstract. In this paper we describe an action/interaction detection system based on improved dense trajectories [19], multiple visual descriptors and bag-of-features representation. Given that the actions/interactions are not mutual exclusive, we train a binary classifier for every predefined action/interaction. We rely on a non-overlapped temporal sliding window to enable the temporal localization. We have tested our system in ChaLearn Looking at People Challenge 2014 Track 2 dataset [1,2]. We obtained 0.4226 average overlap, which is the 3rd place in the track of the challenge. Finally, we provide an extensive analysis of the performance of this system on different actions and provide possible ways to improve a general action detection system.

Keywords: Video analysis · Action recognition · Action detection · Dense trajectories

1 Introduction

Human activity analysis has received considerable attention over the last two decades [3,4]. It is important in many computer vision applications, including video surveillance, content-based video retrieval, human computer interactions, etc. Early attempts on this problem focused on simple actions performed by a single person (e.g. walking, waving and hopping) [5–8]. However, most recent research has been extended to more complex activities such as actions in daily life [9,10], and interactions between multiple persons or objects [11–14]. Much of the state-of-the-art work in action recognition is based on local spatiotemporal features [15,16], trajectories [17–20] or mid-level features [21–23] (e.g. pose and parts).

The ChaLearn Looking at People (LAP) Challenge 2014 [1] is designed to encourage researchers to evaluate and optimize most recent techniques from three different tracks such as human body pose recovery, action and interaction recognition, and gesture recognition. This work is our participation of the Track 2 - action/interaction recognition on RGB data. The Challenge provides videos of 235 action performances from 17 users corresponding to 11 action categories including both natural isolated activities performed by a single person (e.g. waving, pointing, walking, etc.) and interactions between multiple persons

L. Agapito et al. (Eds.): ECCV 2014 Workshops, Part I, LNCS 8925, pp. 541–551, 2015.
DOI: 10.1007/978-3-319-16178-5_38

(e.g. shaking-hands, hugging, fighting, etc.). The goal of the challenge is to recognize the performing action in videos by labeling each frame as an action category. To achieve this goal, we used improved dense trajectory features proposed by Wang and Schmid [19], and applied a sliding window fashion. Even though improved dense trajectory features provide the state-of-the-art performance on a variety of datasets for action classification [19,23,24], applying the feature for temporal localization using a sliding window is not well explored. We show this simple approach can perform well in the ChaLearn LAP dataset. The average Jaccard index obtained on the testing set is 0.4226, which we achieved 3rd place in the challenge.

2 Detecting Actions in Video with Improved Trajectories

In this section, we describe the framework and method of the system that we use for action detection. On the basis of trajectory features, feature descriptors and bag-of-features coding method, we train a binary SVM for every action that is previously defined. All the classifiers are trained independently. A sliding window is applied for the purpose of localizing actions.

2.1 System Framework

The framework of the system is illustrated in Fig.1. In both training and testing stage, we apply a temporal sliding window on the video data to generate video segments. The training data are human-labeled videos where the labels are the actions/interactions in the video and the exact time when it takes place. From the training set, we extract and process the visual features, according to which binary classifier is independently trained for each action. At the stage of testing, the same process of sliding window and feature extraction is applied on the unlabeled video. The trained classifiers are used to detect the existence of every action in each video segment.

2.2 Visual Features

Video data is usually in large size and contains redundant information. Most information in the image sequences is background and noise. To recognize human actions in videos, we shall use compact and efficient features to represent the information that we are interested in.

Videos are essentially image sequences which can be seen as pixels aligned in 3D space, which consists of 2 dimensions in the image and a third dimension of time. The space-time interest points[16] is a natural generalization of the local image feature of interest points from 2D to the 3D space. The basic idea is to make use of the idea from 2D interest points like Harris[25] corners and generalize it to 3D case. Laptev[16] showed that STIP features can capture some interesting events in spatial-temporal spaces that can be used for a compact representation of video data as well as for interpretation of spatio-temporal events.

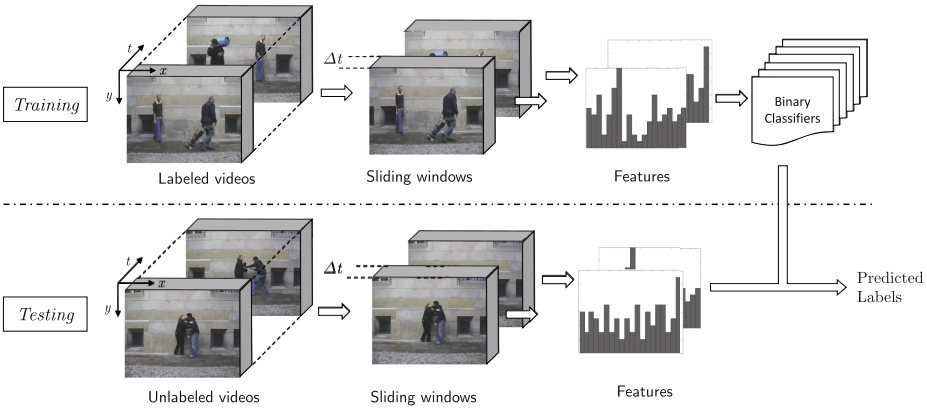

Fig. 1. The framework for action detection

Another successful feature for video representation is dense trajectory[18] proposed by Wang et al. In this approach, dense points are sampled from each frame in videos and tracked on a dense optical flow field. Dense trajectory features are able to cover most of the motion features of a video and therefore can be used as a tool to capture the motion patterns and the local features of motions together with local image features. A main problem about trajectory-based features is that trajectories also capture the camera motion. The trajectories generated by actions or events will be severely affected by the camera motion trajectories. There are works trying to separate the trajectories from action in the video with the trajectories brought by camera movement. On the basis of dense trajectory, Jain et al.[26] proposed a variation by decomposing visual motion into dominant and residual motions. Wang's improved dense trajectories[19] provided a SURF feature based camera motion compensation method to address the problem.

In our system, we use the improved dense trajectories[19] as visual features for action classification. The type of feature we used is selected experimentally. We have tested STIP features, dense trajectories and improved dense trajectories on the ChaLearn LAP dataset, and the improved dense trajectories perform better than other features. The camera is mostly static in ChaLearn LAP dataset where the trajectory features generally performs better than STIP features on capturing actions. Few video clips in the dataset contain camera shake. Comparing to the original dense trajectories, the improved trajectories compensate on camera movement, which aids in the suppression of camera shake artifacts.

2.3 Action Representation

Features like space-time interest points or trajectories contain the information of events or motion patterns in videos. However, in order to use such feature information for the task of classification, we should find a way to describe the features. Similar to previous works[18][19][26], the visual descriptors used in our

system, including Histogram of gradient(HOG)[27], histogram of flow(HOF)[28], motion boundary histograms(MBH)[29] and dense trajectories[19].

HOG encodes the appearance by using the intensity gradient orientations and magnitudes. HOG is an image appearance descriptor which is not formally used as motion description in the video. However, it is also useful to make a distinction of local image features in the video since the descriptor is based on the position of features. Moreover, for human action recognition, most features are located at the human body area. The HOG features are, therefore, capturing human appearances in the frame as well as the human pose related information, which is also a strong cue for action classification. HOF is a statistical description of the orientation and magnitude of the optical flow field. Hence, this descriptor mainly captures the motion information between frames. MBH is designed to capture the gradient of horizontal and vertical components of the flow. The motion boundary encodes the relative pixel motion and, therefore, suppresses camera motion. In the trajectory based method, the trajectory itself plays a significant role which encodes the shape of the trajectory represented by normalized relative coordinates of the successive points forming the trajectory. This description depends on the dense flow used for tracking points.

Feature coding is the final step of action representation in our system, which unifies the form of the representations by building statistics upon the descriptors. Bag-of-features representation is widely used in the context of action detection in computer vision. It is originated from the natural language processing field that they use bag-of-words to represent documents. On the basis of visual descriptors computed from the training set, we constructed the bag-of-features codebook with k-means clustering. There are two ways to build bag-of-features from multiple descriptors. One is building separate codebook for every descriptor and concatenate the coding result for every video segment to form the representation. The other way is to concatenate the descriptors of a feature as one big descriptor and build the codebook upon that. By experimental comparisons between two methods with different vocabulary size, we decide to use the second way to construct bag-of-features and the size of the feature dictionary is determined to be 4096.

2.4 From Recognition to Detection

The action detection is built upon an action classification procedure that not only predicts the labels of actions in videos but also their time and duration. In the system, we use a non-overlapped sliding window to locate the time of actions in video. A video, as shown in Fig.1, can be seen as a 3D image volume. The sliding window is applied on the time direction to segment out a series of video segments of length Δt. We train classifiers on the video segments generated from training data. During testing, we predict on the same length of video segments. Therefore, $1/\Delta t$ can be seen as the temporal resolution of the system. Since the unit represents a temporal position in the video, it will be included as a part of the detection results which tells us when does the actions/interactions take place.

There are two problems about the sliding window approach: 1) how to decide the size of window, and 2) how to label the video segments.

The size of the sliding window Δt is related to the temporal scale of the temporal feature that we extracted. In trajectory features, it is associated with the length of the trajectories. However, the length of trajectories is upper bounded but not necessarily fixed. Therefore, the choice of Δt is made in the validation stage as a model parameter. On the other hand, different actions entail different motion patterns and the model for every action is trained independently, so the optimal Δt for every action is not necessarily the same.

The bag-of-features representation is a feature statistic constructed on a video segment of length Δt. However, in training data, the actions are not labeled with a fixed length. The length of continuous frame sequences labeled as one action varies from 7 to more than 40. In our system, we do not align the video segments to action labels because the start and end of actions are unknown in testing data.

The training and prediction of every action is conducted independently. For an action A and a video segment V, where V_i represents the ith frame of V such that $1 \leq i \leq \Delta t$. The video segment V is labeled $+1$ for action A if there exists i such that V_i is labeled $+1$ for action A. Otherwise V is labeled -1 for action A. During training, we build a bag-of-features representation for every action in every video segment. If V is labeled -1 for action A, the BoF representation is the statistics of feature descriptions from all the trajectories in all frames V_i where $1 \leq i \leq \Delta t$. If V is labeled $+1$, the BoF representation is the statistics of feature descriptions from the frames which are labeled $+1$. During testing, since no label is given, we build the BoF of a video segment with the descriptors in all frames. The classifiers will predict a video segment with a single label, either positive or negative, for every action. We assign the predicted label of the video segment to every frame in it as the result of detection.

3 Experiments

In this section, we show the experimental result of our system on ChaLearn LAP dataset[2].

3.1 Dataset

The data consists of 9 videos, each of which is approximately 60 seconds long. 7 videos are used for training and 2 are used for testing. In these videos, 235 performances from 17 users corresponding to 11 action/interactions categories are recorded and manually labeled. The categories include *Wave*, *Point*, *Clap*, *Crouch*, *Jump*, *Walk*, *Run*, *Shake Hands*, *Hug*, *Kiss,* and *Fight.* Actions are performed by one or more actors at the same time, meaning that the vocabulary of 11 actions is formed by either single actor or multiple actor actions.

3.2 Result and Analysis

The evaluation method is Jaccard index which is defined as

$$J_i = \frac{T_i \cap P_i}{T_i \cup P_i} \tag{1}$$

For the ith action, the Jaccard of our detection system J_i is the overlap of the human labeled ground truth T_i and system prediction P_i in the test video sequences. The average Jaccard index is the algebraic average of J_i of all actions. In the ChaLearn dataset, the chance of overlap is around 0.06.

Kernel SVMs are used as classifiers for the system. Since the actions are not mutually exclusive, we train classifiers for each action in a one vs. all fashion. There are 11 actions in this dataset, therefore, we train 11 SVMs for each of the actions independently. The BoF statistics of every labeled video segments are normalized and used to train the SVMs. The kernel used is RBF-χ^2 kernel and the regularization as well as the window size is determined by validation data. We choose $\Delta t = 15$ and $C = 100$ in SVM for all actions.

The average Jaccard index obtained from our system on the testing set (Seq05 and Seq07) of ChaLearn LAP data is 0.4226. Table 1 shows the Jaccard value we obtained for all actions and their average.

Table 1. Detection results by Jaccard index. While shake-hands, crouch, jump and walk have higher accuracy, kiss, hug, point have lower accuracy.

Wave	Point	Clap	Crouch	Jump	Walk
0.4691	0.2907	0.3267	0.5441	0.5224	0.5064
Run	Shake hands	Hug	Kiss	Fight	**Average**
0.4886	0.5610	0.2883	0.1616	0.4892	**0.4226**

Examples of detection results of our system on test data can be found in Fig. 2 (correct detections) and Fig. 3 (incorrect detections). In our observations, *point* is one of the most difficult action category to detect among all 11 actions given in the data set. For example, in Fig 3-(4), no label is returned when the actor is *pointing*. In Fig 3-(7), *point* is detected as both *wave* and *point*. In Fig 3-(2), *wave* is detected as both *wave* and *point*. The interaction *kissing* is also a difficult case. In Fig. 3-(8) *kiss* is detected as *hug*. There are two reasons why some pairs of actions are confused each other. The first is the motion patterns of these actions are very similar to each other, but our features are based on motion tracking. The other reason is there exists some confusion between actions similar on human pose. For example, *point* and *wave* are more similar to each other on human pose than any other actions. Similarly, *hug* and *kiss* are similar on pose. In our system, the pose information captured by local image descriptors (HOG) also plays an important role in classifying actions.

Another interesting incorrect detection result is between *walk* and *run*. These two actions are rarely confused with other actions but often detected at the same

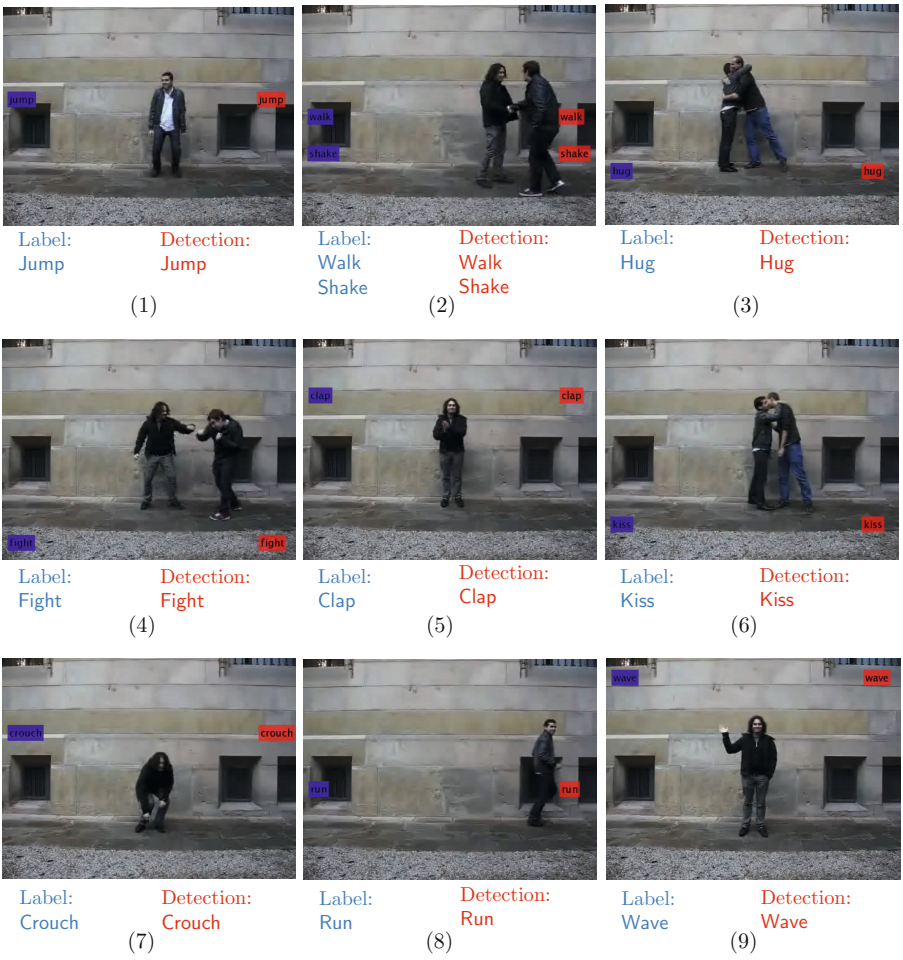

Fig. 2. Correct detection results. The blue labels at the left of the frame are the human labeled ground truth provided in the dataset. At the right side, the red labels reflect the detection result computed by our system.

time, as showed in Fig. 3-(3). Intuitively, *walk* and *run* have a similar motion pattern. The main difference between them is on the velocity of motion and pose. The poses of *walk* and *run* are more closer to each other compared with other actions. This observation tells us that the trajectory-like features might be strong in differentiating *walk* and *run* with other actions but not between these two. It might be helpful to build a hierarchy of classifiers or make use of other features to classify between *walk* and *run*. Considering the objective actions/iteractions we are handling in this system, two actions can be performed at the same time, meaning that some actions/interactions are not mutually-exclusive. For example,

Fig. 3. Incorrect detection results. The blue labels at the left of the frame are the human labeled ground truth provided in the dataset. At the right side, the red labels reflect the detection result computed by our system.

as shown in Fig. 2-(2), two people are *walking* and *shaking* at the same time. That is why we use 11 binary classifiers instead of one multi-class classifier in our system. However, some actions/interactions are very unlikely to appear together. For example, for a single person, it is very unusual to walk and run at the same time. This is a human experience imposed prior. Practically, we can include this prior to the system, but it also will be interesting to automatically discover the exclusiveness of actions from data and build the detection system with a hierarchical structure. We will leave this as a future work.

In Fig. 3-(5), we show a result where the action was labeled with *clap* but detected as *walk*. However, as we can see from the frame, one person is *clapping* and the other is *walking* in the scene. We should notice that even the human labeling is not perfect, but still this is a failure case in our system because the *clap* is not detected. In our system, we take the statistics related to all the motions detected, which does not depend on the number or location of objects in the scene. When the scene contains more than one action, it might be useful to make use of a human detection system to improve performance.

4 Conclusion

We implement a supervised action/interaction detection system to participate in the 2014 ChaLearn LAP challenge. Our framework for detecting actions in videos is improved dense trajectories-based action classification applied on a sliding window fashion. We independently trained 11 one-versus-all kernel SVMs on the labeled training set for 11 different actions. The feature and feature descriptions we used are improved dense trajectories, HOG, HOF, MBHx and MBHy.

We have discussed two possible directions for future work in the last section where we suggest that it will be interesting to automatically discover the dependencies of actions and make use of human detectors to improve the performance. Besides that, future work could be on understanding the actions not only using motion but also many other properties of the action. For example, people can recognize actions from single image without motion information, which indicates that the human posture and the image background are strong cues for humans to understand actions. On the other hand, the detection of action from video data does not necessarily rely on a sliding window. There are researchers[30] show work on detect a sparse key frames from video and uses them to represent video events.

Acknowledgments. The implementation of kernel SVMs that we use is Krzysztof Sopylas chi-square kernel implementation[1] which is build on the basis of libsvm[2]. For the computation of features and feature descriptions, we directly use the code of improved dense trajectories from Heng Wang [3].

This work was partially supported by NSF IIS-1161876, IIS-1111047, NIH R21 DA034954 and the DIGITEO Institute, France.

References

1. Escalera, S., et al.: ChaLearn looking at people challenge 2014: dataset and results. In: Bronstein, M., Agapito, L., Rother, C. (eds.) Computer Vision - ECCV 2014 Workshops. LNCS, vol. 8925, pp. 459–473. Springer, Heidelberg (2015)

[1] http://wmii.uwm.edu.pl/~ksopyla/projects/libsvm-with-chi-squared-kernel/
[2] http://www.csie.ntu.edu.tw/~cjlin/libsvm
[3] https://lear.inrialpes.fr/people/wang/improved_trajectories

2. Snchez, D., Bautista, M., Escalera, S.: HuPBA 8k+: Dataset and ECOC-GraphCut based Segmentation of Human Limbs. Neurocomputing (2014)
3. Poppe, R.: A survey on vision-based human action recognition. Image and Vision Computing **28**(6), 976–990 (2010)
4. Aggarwal, J., Ryoo, M.: Human activity analysis: A review. ACM Comput. Surv. **43**(3), 16:1–16:43 (2011)
5. Blank, M., Gorelick, L., Shechtman, E., Irani, M., Basri, R.: Actions as space-time shapes. In: Proceedings of the International Conference On Computer Vision, ICCV (2005)
6. Dollár, P., Rabaud, V., Cottrell, G., Belongie, S.: Behavior recognition via sparse spatio-temporal features. In: 2nd Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance, PETS (2005)
7. Niebles, J., Wang, H., Fei-Fei, L.: Unsupervised learning of human action categories using spatial-temporal words. International Journal of Computer Vision **79**(3), 299–318 (2008)
8. Schuldt, C., Laptev, I., Caputo, B.: Recognizing human actions: a local svm approach. In: Proceedings of the 17th International Conference on Pattern Recognition, ICPR (2004)
9. Messing, R., Pal, C., Kautz, H.: Activity recognition using the velocity histories of tracked keypoints. In: Proceedings of the International Conference On Computer Vision, ICCV (2009)
10. Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., Serre, T.: Hmdb: a large video database for human motion recognition. In: Proceedings of the International Conference On Computer Vision, ICCV (2011)
11. Ayazoglu, M., Yilmaz, B., Sznaier, M., Camps, O.: Finding causal interactions in video sequences. In: Proceedings of the International Conference On Computer Vision, ICCV (2013)
12. Ryoo, M.S., Aggarwal, J.K.: Spatio-temporal relationship match: video structure comparison for recognition of complex human activities. In: Proceedings of the International Conference On Computer Vision, ICCV (2009)
13. Yun, K., Honorio, J., Chattopadhyay, D., Berg, T.L., Samaras, D.: Two-person interaction detection using body-pose features and multiple instance learning. In: IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPRW (2012)
14. Yao, B., Fei-Fei, L.: Modeling mutual context of object and human pose in human-object interaction activities. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR (2010)
15. Wang, H., Ullah, M.M., Kläser, A., Laptev, I., Schmid, C.: Evaluation of local spatio-temporal features for action recognition. In: British Machine Vision Conference, BMVC (2009)
16. Laptev, I.: On space-time interest points. International Journal of Computer Vision **64**(2-3), 107–123 (2005)
17. Ali, S., Basharat, A., Shah, M.: Chaotic invariants for human action recognition. In: Proceedings of the International Conference On Computer Vision, ICCV (2007)
18. Wang, H., Kläser, A., Schmid, C., Liu, C.L.: Action recognition by dense trajectories. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR (2011)
19. Wang, H., Schmid, C.: Action recognition with improved trajectories. In: Proceedings of the International Conference On Computer Vision, ICCV (2013)

20. Fathi, A., Mori, G.: Action recognition by learning mid-level motion features. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR (2008)
21. Raptis, M., Kokkinos, I., Soatto, S.: Discovering discriminative action parts from mid-level video representations. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR (2012)
22. Zhang, W., Zhu, M., Derpanis, K.: From actemes to action: a strongly-supervised representation for detailed action understanding. In: Proceedings of the International Conference On Computer Vision, ICCV (2013)
23. Oneata, D., Verbeek, J., Schmid, C.: Efficient action localization with approximately normalized fisher vectors. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR (2014)
24. Simonyan, K., Zisserman, A.: Two-Stream Convolutional Networks for Action Recognition in Videos. arXiv:1406.2199v1 (2014)
25. Harris, C., Stephens, M.: A combined corner and edge detector. In: Alvey Vision Conference, vol. 15, p. 50 (1988)
26. Jain, M., Jgou, H., Bouthemy, P.: Better exploiting motion for better action recognition. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR (2013)
27. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR (2005)
28. Laptev, I., Marszaek, M., Schmid, C., Rozenfeld, B.: Learning realistic human actions from movies. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR (2008)
29. Dalal, N., Triggs, B., Schmid, C.: Human detection using oriented histograms of flow and appearance. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3952, pp. 428–441. Springer, Heidelberg (2006)
30. Raptis, M., Sigal, L.: Poselet key-framing: a model for human activity recognition. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR (2013)