Human Action Recognition from RGB-D Frames Based on Real-Time 3D Optical Flow Estimation

Gioia Ballin, Matteo Munaro, and Emanuele Menegatti

Department of Information Engineering of the University of Padova,
via Gradenigo 6B, 35131 - Padova, Italy
gioia.ballin@gmail.com, {munaro,emg}@dei.unipd.it

Abstract. Modern advances in the area of intelligent agents have led to the concept of cognitive robots. A cognitive robot is not only able to perceive complex stimuli from the environment, but also to reason about them and to act coherently. Computer vision-based recognition systems serve the perception task, but they also go beyond it by finding challenging applications in other fields such as video surveillance, HCI, content-based video analysis and motion capture. In this context, we propose an automatic system for real-time human action recognition. We use the Kinect sensor and the tracking system in [1] to robustly detect and track people in the scene. Next, we estimate the 3D optical flow related to the tracked people from point cloud data only and we summarize it by means of a 3D grid-based descriptor. Finally, temporal sequences of descriptors are classified with the Nearest Neighbor technique and the overall application is tested on a newly created dataset. Experimental results show the effectiveness of the proposed approach.

Keywords: Action recognition, 3D optical flow, RGB-D data, Kinect.

1 Introduction

The challenge of endowing robotic agents with human-like capabilities is currently addressed by the cognitive robotics research field. In cognitive robotics, the aim is to create smart agents able to efficiently perform complex tasks in partially observable environments. In order to achieve real-world goals, a cognitive robot is equipped with a processing architecture that combine the perception, cognition and action modules in the most effective way. Then, a cognitive robot is not only able to perceive complex stimuli from the environment, but also to reason about them and to act coherently. Furthermore, a cognitive robot should also be able to safely interact and cooperate with humans.

The interaction with humans and the interpretation of human actions and activities have recently gained a central role in the researchers' community since the spread of new robotic devices has reached real-life environments such as offices, homes and urban environments. In this context, we propose an automatic system for real-time human action recognition. Human action recognition is an active research area in computer vision. First investigations about this topic began in the seventies with pioneering studies accomplished by Johanssons [2]. From then on, the interest in the field grew increasingly, motivated by a number of potential real-world applications such as video

A. Chella et al. (Eds.): Biologically Inspired Cognitive Architectures 2012, AISC 196, pp. 65–74.
springerlink.com

surveillance, HCI, content-based video analysis and retrieval. Moreover, in recent years the task of recognizing human actions has gained increasingly popularity thanks to the emergence of modern applications such as motion capture and animation, video editing and service robotics.

Our system relies on the acquisition of RGB-D data and exploits the Robot Operating System [3] as a framework. We use the Microsoft Kinect sensor and the tracking system described in [4] and [1] to robustly detect and track people in the scene. Next, we estimate the 3D optical flow of the points relative to each person. For this purpose, we propose a novel technique that estimates 3D velocity vectors from point cloud data only, thus obtaining a real-time calculus of the flow. Then, we compute a 3D grid-based descriptor for representing the flow information within a temporal sequence and we recognize actions by means of the Nearest Neighbor classifier. We tested this technique on a RGB-D video dataset which contains six actions performed by six different actors. Our system is able to recognize the actions in the dataset with a 80% accuracy while running at a medium frame rate of 23 frames per second.

The remainder of the paper is organized as follows: Section 2 provides a complete review about the recent advances in human action recognition systems. Section 3 describes the proposed real-time computation of 3D optical flow, while Section 4 outlines the data structure used to summarize the estimated flow information. Experimental results are reported in Section 5 and Section 6 concludes the paper and outlines the future work.

2 Related Work

Most of the works on human action recognition rely on information extracted from 2D images and videos. These approaches mostly differ in the features representation. Popular global representations are edges [5], silhouettes of the human body [6] [7] [8], 2D optical flow [9] [10] [11] and 3D spatio-temporal volumes [6] [7] [8] [12] [13] [14]. Conversely, effective local representations mainly refer to [15] [16] [17] [18] [19] [20] and [21]. The recent spread of inexpensive RGB-D sensors has paved the way to new studies in this direction. Recognition systems that rely on the acquisition of 3D data could potentially outperform their 2D counterparts, but they still need to be investigated. The first work related to RGB-D action recognition is signed by Microsoft Research [22]. In [22], a sequence of depth maps is given as input to the system. Next, the relevant postures for each action are extracted and represented as a bag of 3D points. The motion dynamics are modeled by means of an action graph and a Gaussian Mixture Model is used to robustly capture the statistical distribution of the points. Recognition results state the superiority of the 3D silhouettes with respect to their 2D analogous.

Subsequent studies mainly refer to the use of two different technologies: Time of Flight cameras [23][24] and active matricial triangulation systems, in particular the Microsoft Kinect [25], [26], [27], [28], [29], [30]. [25][26] represent the first attempt to exploit skeleton tracking information to recognize human actions. This information is used to compute a set of features related to the human body pose and motion. Then, the proposed approach is tested on different classifiers: the SVM classification is compared with both the one-layer and the hierarchical Maximum Entropy Markov Model

classification. Finally, the dataset collected for testing purposes has been made publicly available by the authors. As for [25][26], the recently published work by Yang *et al.* [27] relies on the acquisition of skeleton body joints. The 3D position differences of body joints are exploited to characterize the posture and the motion of the observed human body. Finally, Principal Component Analysis is applied to compute the so-called *EigenJoints* and the Näives-Bayes-Nearest-Neighbor technique is used to classify these descriptors. This work consistently outperforms that of Li *et al.* [22] while using about a half their number of frames. A different approach is followed in [28] by Zhang and Parker, where the popular 2D spatio-temporal features are extended to the third dimension. The new features are called 4D spatio-temporal features, where the "4D" is justified by the 3D spatial components given by the sensor plus the time dimension. The descriptor computed is a 4D hyper cuboid, while Latent Dirichlet Allocation with Gibbs sampling is used as classifier. Another work in which typical 2D representations are extended to 3D is [29]. The authors extend the existing definitions of spatio-temporal interest points and motion history images to incorporate also the depth information. They also proposes a new publicly available dataset as test bed. For the classification purpose, SVMs with different kernels are used.

From the application point of view [25], [26], and [29] are targeted to applications in the personal robotics field, while [22] and [27] are addressed to HCI and gaming applications. Finally, [28] and [30] are primarily addressed to applications in the field of video surveillance. In [30], Popa *et al.* propose a system able to continuously analyze customers' shopping behaviours in malls. By means of the Microsoft Kinect sensor, Popa *et al.* extract silhouette data for each person in the scene and then compute moment invariants to summarize the features.

In [23][24], a kind of 3D optical flow is exploited for the gesture recognition task. Unlike our approach, Holte *et al.* compute the 2D optical flow using the traditional Lukas-Kanade method and then extend the 2D velocity vectors to incorporate also the depth dimension. At the end of this process, the 3D velocity vectors are used to create an annotated velocity cloud. 3D Motion Context and Harmonic Motion Context serve the task of representing the extracted motion vector field in a view-invariant way. With regard to the classification task, [23] and [24] do not follow a learning-based approach, instead a probabilistic Edit Distance classifier is used in order to identify which gesture best describes a string of primitives. [24] differs from [23] because the optical flow is estimated from each view of a multi-camera system and is then combined into a unique 3D motion vector field.

3 3D Optical Flow

In this section, we propose a novel approach to compute the 3D optical flow as an extension to the third dimension of the traditional 2D optical flow [9] [10] [11]. Computing the optical flow with real-time performances is really a challenging problem: traditional 2D approaches involve several computations on every pixel of the input images thus leading to poor temporal performances. On the contrary, we compute 3D optical flow only for relevant portions of the overall 3D scene.

In details, we associate a cluster to each tracked person by means of the underlying tracking-by-detection system [1]. Such a cluster is defined as a 4D point cloud representing an individual in the 3D world. The four dimensions represent the 3D geometric coordinates and the RGB color component of each point. With this information, we estimate the 3D optical flow associated to each identified cluster frame-by-frame. The first step of this process concerns storing the appropriate information. Indeed, at each frame F and for each track k, we store two elements: the cluster associated to k at frames F and $F-1$. The second step involves matching the two point clouds stored at each frame in order to find correspondences between points.

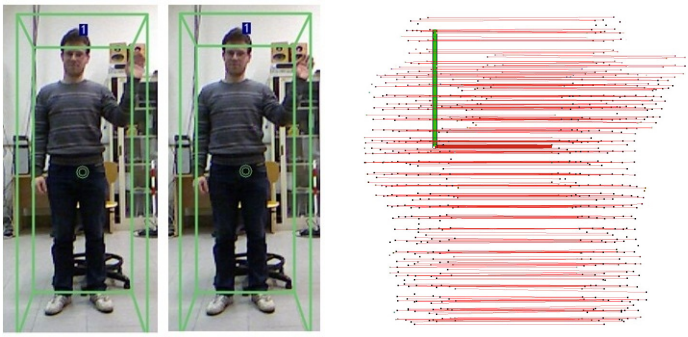
3.1 Points Matching

Matching cluster points relative to different time instants represents the true insight of this work. Since we deal with human motion, we cannot assume the whole person cluster to undergo a rigid transformation. For this reason, we exploit local matching techniques.

For each track k, let $A_k(F-1)$ be the cluster associated to k at the frame $F-1$ and let $A_k(F)$ the cluster associated to k at the frame F. Let P be a generic point in $A_k(F-1)$. If we can find the spatial location of P in $A_k(F)$, then we can also estimate the actual 3D velocity vector representing the movement of P. In order to find a match between the points in $A_k(F-1)$ and in $A_k(F)$, a two-way matching algorithm is applied. First, $A_k(F)$ is kept fixed and for each point in $A_k(F-1)$ a 1-nearest neighbor search is performed in order to find a matching point in $A_k(F)$. Next, the same pattern is repeated with $A_k(F-1)$ fixed instead of $A_k(F)$. This way, two different vectors of correspondences are returned, those vectors are then intersected and a final vector of correspondences is returned. Since clusters have an average of 300 points, kdtrees with FLANN searches are used to speed-up the computations. Furthermore, searches are driven by both the 3D geometric coordinates of the points and the RGB color information. In Fig. 1, we show the correspondences estimated while a person is hand waving. In Fig. 1(a) two consecutive images are shown, while in Fig. 1 (b) the corresponding cluster points are reported and the points correspondences are drawn in red. In this work, the 3D optical flow is seen as a set of estimated 3D velocity vectors that are obtained in constant time from the vector of estimated correspondences by dividing the spatial difference by the temporal difference. At an implementation level, the computed optical flow can be stored as an additional field for each point of $A_k(F)$, thus creating an annotated point cloud.

4 3D Grid-Based Descriptor

In order to recognize human actions from the 3D optical flow estimated in Section 3, first a suitable description for the flow is required. Indeed, the proposed approach generally returns a different number of velocity vectors in each frame. To achieve a fixed size descriptor we compute a 3D grid surrounding each cluster in the scene. The 3D grid provides an effective spatial partition of the cluster points. Furthermore, since each of the 3D velocity vectors is associated to a cluster point by means of an annotated



(a) RGB frames

(b) Cluster points and estimated correspondences

Fig. 1. Display of two consecutive RGB frames where the tracking output is drawn as a 3D bounding box (a) and the cluster points relative to the tracked person, together with the estimated correspondences (b) with regard to the hand waving action

point cloud, the grid provides also a 3D optical flow partition. In this work, the 3D grid represents the baseline data structure of the flow descriptor and it is defined by a fixed number of spatial partitions along the three geometric axes. The grid computation involves three basic steps. The first step is concerned with defining the minimum and maximum grid bounds along the three dimensions. Bounds are set so that the current cluster is centered in the grid, even if movements of the upper limbs occur. In the second step, the bounds of the grid are combined with the spatial divisions in order to define the minimum and maximum grid ranges associated to each 3D cube of the grid. The last step is devoted to place the right points into the right 3D cube. In particular, the current cluster is scanned and each point is put into the right cube based on its 3D geometric coordinates. We finally choose to have four partitions along the x, y, and z axis. This choice is justified by our will of keeping separate the right side of the human body from the left side, while also keeping limited the size of the descriptor. Such a 3D grid is shown in Fig. 2. The final descriptor is obtained by summarizing the flow information contained in each grid cube: the 3D average velocity vector is computed for each cube and all these vectors are concatenated in a column vector.

The 3D grid-based descriptor is calculated frame by frame, for each track k and frame F. For the classification purpose, we collect a sequence of n 3D grid-based descriptors and this sequence represents the final descriptor for the action at issue. Since an action actually represents a sequence of movements over time, considering multiple frames could potentially provides more discriminant information to the recognition task with respect to approaches in which only a single-frame classification is performed. In this work, we set the constant n to 10 and [31] provides a justification to our choice. Since we do not have the a priori knowledge about the action duration and since each different action is characterized by a different temporal duration, interpolation and sampling techniques are used to create the final descriptor. At the end, the final descriptor is normalized and used for training and testing purposes. Normalization enables to partially discard the presence of noise in the raw data.

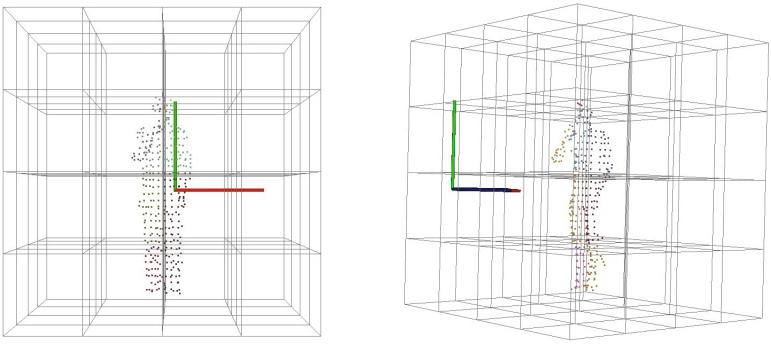


Fig. 2. Two different views of the computed 3D grid: 4 partions along the x, y and z axis are used

5 Experiments

In order to test the recognition performances of our descriptor we exploited a simple 1-Nearest Neighbor classifier. The test phase involves four main steps. In the first step we collect single-frame descriptors in the so called final descriptor until 10 frames are reached. When the 10 frames are exceeded, a window sampling is applied to obtain a 10-frames final descriptor. As a second step, the final descriptor is normalized, while in the third step we compute a distance between the test descriptor and all the training examples in the dataset. We used two types of distance function: the Euclidean distance function and the Mahalanobis distance function. Finally, the label related to the training descriptor that has the shortest distance to the test descriptor is chosen as predicted class.

5.1 Dataset and Setup

Our work is mainly targeted to video surveillance applications. Since no public RGB-D dataset devoted to recognize typical video surveillance actions is currently available, we collected a new RGB-D dataset in a lab environment in order to test our recognition system. The dataset contains six types of human actions: *standing*, *hand waving*, *sitting down*, *getting up*, *pointing*, *walking*. Each action is performed once by six different actors and recorded from the same point of view. We invited the six volunteers to naturally execute the actions, and we gave no indication to them about how to accomplish movements. Each of the segmented video samples spans from about 1 second to 7 seconds.

5.2 Results

This section discusses the experimental results achieved by performing the 1-Nearest Neighbor classification on 10-frames final descriptors. Tests have been executed by following the leave-one-out approach: first we chose an actor from the dataset to use its

recordings as test bed, then we trained the classifier with the examples related to the other five subjects. We collected the classification results related to the unseen actor with respect to the training samples. Finally, the process has been performed for each actor in the dataset. Results are provided in the form of confusion matrices and they are shown in Table 1 and Table 2. Table 1 is related to a Nearest Neighbor classification obtained by using the Mahalanobis distance, while Table 2 refers to a Nearest Neighbor classification based on the Euclidean distance computation. We are able to achieve an accuracy of 80% and a precision of 74% for the Euclidean-based classification, while we obtain an accuracy of 78% and a precision of 72% for the Mahalanobis-based classification. We can notice that the Mahalanobis distance led to worse results with respect to the Euclidean distance, suggesting that the number of training examples was not enough for computing reliable means and variances. Moreover, the computation of the covariance matrix and its inverse is costly when dealing with many dimensions. Experiments also show that our application is able to achieve good recognition results for those actions in which the movement of the entire human body is involved (e.g. *getting up* and *walking*), while fairly good performances are obtained from the recognition of actions characterized by the upper limbs motion only (e.g. *hand waving* and *sitting*).

Table 1. Confusion matrix related to a Nearest Neighbor classification obtained by using the Mahalanobis distance. In the matrix: **STA** stands for *standing*, **HAW** stands for *hand waving*, **SIT** stands for *sitting down*, **GET** stands for *getting up*, **POI** stands for *pointing* and finally **WAL** stands for *walking*.

	STA	**HAW**	**SIT**	**GET**	**POI**	**WAL**
STA	**0.67**	0.17			0.17	
HAW	0.17	**0.50**			0.17	
SIT			**0.83**			
GET		0.17		**1.00**		0.17
POI	0.17	0.17	0.17		**0.50**	
WAL						**0.83**

Table 2. Confusion matrix related to a Nearest Neighbor classification obtained by using the Euclidean distance. In the matrix: **STA** stands for *standing*, **HAW** stands for *hand waving*, **SIT** stands for *sitting down*, **GET** stands for *getting up*, **POI** stands for *pointing* and finally **WAL** stands for *walking*.

	STA	**HAW**	**SIT**	**GET**	**POI**	**WAL**
STA	**0.83**	0.33	0.17		0.33	
HAW		**0.50**				
SIT			**0.83**			
GET		0.17		**1.00**		
POI	0.17				**0.67**	
WAL						**1.00**

With regards to temporal performances, we run the application on a notebook equipped with a 2nd generation Intel Core i5 processor characterized by a processor speed that ranges from 2.4 GHz to 3 GHz if the Intel Turbo Boost technology is enabled. On this working station, the application runs in real-time with a medium frame rate of 23 frames per second.

6 Conclusions and Future Work

In this paper, we proposed a method for real-time human action recognition for a cognitive robot endowed with a RGB-D sensor. We focused on the features extraction step and in particular we exploited 3D optical flow information directly extracted from people point clouds to obtain a suitable representation of human actions. To this aim we also proposed a 3D grid-based descriptor to encode the 3D flow information into a single vector. The estimation of the 3D optical flow field proved to be effective to the recognition task with a Nearest Neighbor classifier: we achieved an accuracy of 80% and a precision of 74% on six basic actions performed of a newly collected RGB-D dataset. Furthermore, the application runs in real-time at a medium frame rate of 23 fps.

As future works, we envision to make the descriptor more discriminant by using histograms of 3D flow orientation instead of mean flow orientations. Moreover, we plan to use more sophisticated classifiers and to extend our dataset in order to include more actions, even in presence of partial occlusion.

References

1. Munaro, M., Basso, F., Menegatti, E.: Tracking people withing groups with rgb-d data. In: Proc. of the International Conference on Intelligent Robots and Systems (IROS), Vilamoura, Portugal (2012)
2. Johansson, G.: Visual perception of biological motion and a model for its analysis. Attention, Perception, & Psychophysics 14, 201–211 (1973), 10.3758/BF03212378
3. Quigley, M., Gerkey, B., Conley, K., Faust, J., Foote, T., Leibs, J., Berger, E., Wheeler, R., Ng, A.: Ros: an open-source robot operating system. In: Proceedings of the IEEE International Conference on Robotics and Automation, ICRA (2009)
4. Basso, F., Munaro, M., Michieletto, S., Pagello, E., Menegatti, E.: Fast and Robust Multi-People Tracking from RGB-D Data for a Mobile Robot. In: Lee, S., Cho, H., Yoon, K.-J., Lee, J. (eds.) Intelligent Autonomous Systems 12. AISC, vol. 193, pp. 269–281. Springer, Heidelberg (2012)
5. Carlsson, S., Sullivan, J.: Action recognition by shape matching to key frames. In: IEEE Computer Society Workshop on Models versus Exemplars in Computer Vision (2001)
6. Yilmaz, A., Shah, M.: Actions sketch: a novel action representation. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005, vol. 1, pp. 984–989 (June 2005)
7. Blank, M., Gorelick, L., Shechtman, E., Irani, M., Basri, R.: Actions as space-time shapes. In: Proc. Tenth IEEE Int. Conf. Computer Vision ICCV 2005, vol. 2, pp. 1395–1402 (2005)
8. Rusu, R.B., Bandouch, J., Meier, F., Essa, I.A., Beetz, M.: Human action recognition using global point feature histograms and action shapes. Advanced Robotics 23(14), 1873–1908 (2009)

9. Efros, A.A., Berg, A.C., Mori, G., Malik, J.: Recognizing action at a distance. In: Proceedings of the Ninth IEEE International Conference on Computer Vision, vol. 2, pp. 726–733 (October 2003)
10. Yacoob, Y., Black, M.J.: Parameterized modeling and recognition of activities. In: Sixth International Conference on Computer Vision, pp. 120–127 (January 1998)
11. Ali, S., Shah, M.: Human action recognition in videos using kinematic features and multiple instance learning. IEEE Transactions on Pattern Analysis and Machine Intelligence 32(2), 288–303 (2010)
12. Ke, Y., Sukthankar, R., Hebert, M.: Efficient visual event detection using volumetric features. In: Proc. Tenth IEEE Int. Conf. Computer Vision ICCV 2005, vol. 1, pp. 166–173 (2005)
13. Liu, J., Ali, S., Shah, M.: Recognizing human actions using multiple features. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2008, pp. 1–8 (June 2008)
14. Scovanner, P., Ali, S., Shah, M.: A 3-dimensional sift descriptor and its application to action recognition. In: Proceedings of the 15th International Conference on Multimedia, MULTIMEDIA 2007, pp. 357–360. ACM, New York (2007)
15. Laptev, I., Lindeberg, T.: Space-time interest points. In: Proc. Ninth IEEE Int. Computer Vision Conf., pp. 432–439 (2003)
16. Laptev, I., Marszalek, M., Schmid, C., Rozenfeld, B.: Learning realistic human actions from movies. In: Proc. IEEE Conf. Computer Vision and Pattern Recognition CVPR 2008, pp. 1–8 (2008)
17. Dollar, P., Rabaud, V., Cottrell, G., Belongie, S.: Behavior recognition via sparse spatio-temporal features. In: Proc. 2nd Joint IEEE Int. Visual Surveillance and Performance Evaluation of Tracking and Surveillance Workshop, pp. 65–72 (2005)
18. Schuldt, C., Laptev, I., Caputo, B.: Recognizing human actions: a local svm approach. In: Proc. 17th Int. Conf. Pattern Recognition ICPR 2004, vol. 3, pp. 32–36 (2004)
19. Niebles, J.C., Wang, H., Fei-Fei, L.: Unsupervised learning of human action categories using spatial-temporal words. Int. J. Comput. Vision 79, 299–318 (2008)
20. Kläser, A., Marszałek, M., Schmid, C.: A spatio-temporal descriptor based on 3d-gradients. In: British Machine Vision Conference, pp. 995–1004 (September 2008)
21. Weinland, D., Ronfard, R., Boyer, E.: Free viewpoint action recognition using motion history volumes. Comput. Vis. Image Underst. 104(2), 249–257 (2006)
22. Li, W., Zhang, Z., Liu, Z.: Action recognition based on a bag of 3d points. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 9–14 (June 2010)
23. Holte, M.B., Moeslund, T.B.: View invariant gesture recognition using 3d motion primitives. In: IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2008, March 31-April 4, pp. 797–800 (2008)
24. Holte, M.B., Moeslund, T.B., Nikolaidis, N., Pitas, I.: 3d human action recognition for multi-view camera systems. In: 2011 International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT), pp. 342–349 (May 2011)
25. Sung, J., Ponce, C., Selman, B., Saxena, A.: Human activity detection from rgbd images. In: Plan, Activity, and Intent Recognition. AAAI Workshops, vol. WS-11-16. AAAI (2011)
26. Sung, J., Ponce, C., Selman, B., Saxena, A.: Unstructured human activity detection from rgbd images. In: International Conference on Robotics and Automation, ICRA (2012)
27. Yang, X., Tian, Y.: Eigenjoints-based action recognition using naive-bayes-nearest-neighbor. In: IEEE Workshop on CVPR for Human Activity Understanding from 3D Data (2012)
28. Zhang, H., Parker, L.E.: 4-dimensional local spatio-temporal features for human activity recognition. In: 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 2044–2049 (September 2011)

29. Ni, P.B., Wang, G., Moulin, P.: Rgbd-hudaact: A color-depth video database for human daily activity recognition. In: 2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops), pp. 1147–1153 (November 2011)
30. Popa, M., Koc, A.K., Rothkrantz, L.J.M., Shan, C., Wiggers, P.: Kinect Sensing of Shopping Related Actions. In: Wichert, R., Van Laerhoven, K., Gelissen, J. (eds.) AmI 2011. CCIS, vol. 277, pp. 91–100. Springer, Heidelberg (2012)
31. Schindler, K., van Gool, L.: Action snippets: How many frames does human action recognition require? In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2008, pp. 1–8 (June 2008)