This CVPR2013 Workshop paper is the Open Access version, provided by the Computer Vision Foundation.
The authoritative version of this paper is available in IEEE Xplore.

Fusing Spatiotemporal Features and Joints for 3D Action Recognition

Yu Zhu, Wenbin Chen, and Guodong Guo
West Virginia University, Dept. of CSEE, Morgantown, WV 26506
yzhu4@mix.wvu.edu, wnchen@mix.wvu.edu, guodong.guo@mail.wvu.edu

Abstract

We present a novel approach to 3D human action recognition based on a feature-level fusion of spatiotemporal features and skeleton joints. First, 3D interest points detection and local feature description are performed to extract spatiotemporal motion information. Then the frame difference and pairwise distances of skeleton joint positions are computed to characterize the spatial information of the joints in 3D space. These two features are complementary to each other. A fusion scheme is then proposed to combine them effectively based on the random forests method. The proposed approach is validated on three challenging 3D action datasets for human action recognition. Experimental results show that the proposed approach outperforms the state-of-the-art methods on all three datasets.

1. Introduction

Human action recognition is to automatically recognize ongoing actions performed by humans. Human action recognition has a variety of applications in real world, such as Human Computer Interaction (HCI), video surveillance, video retrieval and video games. In the past decades, many approaches have been proposed to solve the action recognition problem on the visible light or RGB video sequences [19, 1, 24]. Recently, with the launch of the Kinect sensor [16], human action recognition in 3D data has become a very active research topic in computer vision. Besides, in Shotton *et al.* [16]'s work, a quick estimation is feasible to obtain 3D human skeleton joint positions from the depth maps. Thus, using the Kinect sensor, three channels (RGB, depth and skeleton joint positions) of information can be provided, which can benefit robotics and human centered computing problems, and bring a broader scope for action recognition research [4].

There are several representative works for 3D action recognition. Li *et al.* [10] proposed an action graph for depth action recognition. A bag of 3D points sampled on depth data is used to encode the action posture, and action graph is used to model the dynamics of the actions. Wang *et al.* [23] proposed to combine the skeleton feature and local occupation feature, then learned an actionlets ensemble model to represent actions. A multiple kernel learning method is used to combine the actionlets. In [22], Wang *et al.* proposed a semi-local feature called Random Occupancy Patterns (ROP), which is extracted from 4D volumes. Sparse coding is utilized to encode the features and the SVM is used for classification. Vieira *et al.* [20] proposed a space-time occupancy patterns to represent depth sequences. Both space and time axes are divided into multiple segments. Occupancy feature is computed in each cell, and a nearest neighbor classifier is applied for action recognition. Different approaches based on Motion History Images (MHI) are proposed by Yang *et al.* [28, 27, 29]. The main idea is to use accumulated depth maps and compute histogram of gradients (HOG) features, to represent human actions. Xia *et al.* [26] proposed an alternative feature called HOJ3D based on the skeleton joints. A coordinate based on skeleton joints is constructed, and multiple 3D bins are used to extract histogram features, by counting the number of joints in each bin. A Hidden Markov Model is used for action classification. Similarly, Miranda *et al.* [11] used the pose descriptor in a torso-based coordinate system and the SVM classifier to learn key poses and a decision forest is used to recognize the action classes. More recently, Oreifej *et al.* [14] described the depth sequence as a histogram (HON4D) captured in the 4D space of time, depth and spatial coordinates. A 600-cell polychorons is used to quantize and represent the features. SVM classifier is used and showed a good performance for action recognition. Sung *et al.* [17] combined both the RGB and depth channels for action recognition. Hand positions, body pose and motion features are extracted from skeleton joints. HOG is used as the descriptor for both RGB and depth images. A two-layer maximum-entropy Markov model is trained for classification. There are some other works on human motion analysis using depth imagery, see [4] for details.

On the other hand, the spatiotemporal interest points (STIPs) features have shown promising results in regular visible light action datasets. Several local space-time features for visible light action recognition have been proposed. Laptev *et al.* [8] extends Harris corner detection[6] to space and time, and proposed some effective methods to make spatiotemporal interest points (STIPs) velocity-adaptive. Dollar *et al.*[5] proposed an alternative interest point detector which applied Gabor filter on the spatial and temporal dimensions. In Willems *et al.*'s work [25], a 3D Hessian matrix is used to determine the interest points by its matrix determinant. They also extends SURF descriptor to space-time as the feature descriptor. Recently, Wang *et al.* [21] made a comprehensive evaluation with various detectors and descriptors on different RGB action datasets. However, only a few work has utilized the STIPs features for depth action recognition. Zhang *et al.* [30] in their approach extends the STIPs approach by Dollar *el al.* [5] to the fourth dimension. Ni *et al.* [12] extends the Harris3D detector and HOG/HOF descriptor [9], by adding depth information to 2D features. This feature has also been evaluated by Ofli *et al.* [13] on depth data in introducing their multimodal action dataset. Zhao *et al.* [31] proposed to adapt the STIPs detected on RGB data to the depth data and combine the two channels for action recognition.

The local features extracted by spatiotemporal interest points methods can capture complex motions in the video, however, some interest points may be detected on the unrelated areas such as the background, because of the noise in depth imagery. The STIPs methods usually work on actions with large motions. While the skeleton joints features represent human postures, which can capture the spatial information well in human actions. Nevertheless, incorrect detection of the joint positions or the loss of human body detection will dramatically effect the action analysis based on skeleton joints. To overcome the drawbacks in either approach, we propose to combine them based on a feature-level fusion scheme. The fusion is accomplished by the random forests method. The objective of combining features from two distinct channels (depth maps & skeleton joints) is to obtain a new representation that can characterize 3D actions better. The random forests method [3] is applied to perform feature fusion, selection, and action classification altogether. It is based on randomly selecting features from either set (STIPs or joints) in each node and construct multiple decision trees to solve the 3D action classification problem. As a result, our approach can combine different, complementary features effectively, to form a new representation of 3D actions and improve the recognition accuracy.

The rest of this paper is organized as follows. In Section 2, our proposed approach is presented. Then the experimental details and experimental results are shown in Section 3. Finally, we draw conclusions.

2. Our Approach

The proposed method has four major steps. First, spatiotemporal features are extracted on depth sequences. Then skeleton joints features are computed from the skeleton joint positions. Thirdly a quantization is performed for the two features respectively, to encode the action sequences with histograms. Finally, a feature-level fusion and action recognition is executed using the random forests method.

2.1. Spatiotemporal Features

Spatiotemporal features are used to capture the complex motion of human actions on depth data. Among the various spatiotemporal interest point detectors and local feature descriptors[15], we have attempted several combinations and select the ones that have a better performance for 3D actions. Because of the space limit, only the best ones in each dataset are presented. The Harris3D detector [8] computes the locations of the interest points according to the response function: $H = det(\mu) - k \cdot trace^3(\mu)$, where $\mu = g(\cdot) \times M$, $g(\cdot)$ is a Gaussian weight function, and M is the second-moment matrix of which the convolution of spatiotemporal gaussian kernel with the video sequence. Similar to the Harris3D detector, [25] proposed the Hessian detector, to use a Hessian Matrix H, and the response function $S = |det(H)|$ to measure the strength of each interest point. HOG/HOF descriptor was proposed in [9], to describe local human motion in RGB videos. It computes the histogram of gradient(HOG) and histogram of optical Flow(HOF) in each local volume. Klaser *et al.*[7] extends the HOG to HOG3D descriptor, which computes the 3D gradient and bins a histogram as the feature vector. ESURF descriptor [25] is an extension of the SURF[2] descriptor that computes the response of Haar-wavelets along three direction and stores the sum as the feature vector. In this work, these spatiotemporal features are used on 3D depth actions other than RGB actions. We found that these features perform differently in different 3D action datasets (see experiments).

2.2. Skeleton Joints Features

The spatiotemporal features are local descriptions of human motions, without considering the spatial information of human body parts which might be important for action encoding. Provided that the human skeleton joints can be estimated fast on 3D data [16], and inspired by Yang *et al.*'s work [28], we use the histogram of the skeleton joints features to complement the spatiotemporal features. Different from [28] where the Naive Bayes classifier is used, we compute the histogram of the joints to combine with the STIP features.

The features from joint locations consist of three parts: (1) current posture: pair-wise joint distances in current posture; (2) motion: joints difference between current posture and the original (in the first frame); and (3) offset: joints differences between current posture and the previous one. A concatenation of the three feature vectors is used to represent the feature for a specific action. The PCA technique is applied for dimensionality reduction.

Specifically, denote p the 3D skeleton joints, for each joint, $p_i = (x_i(t),\ y_i(t),\ z_i(t))$ at frame t. The number of skeleton joints in each frame is denoted as N. So the feature vector can be denoted as:

$$f = [f_{current}\ f_{motion}\ f_{offset}], \tag{1}$$

$$f_{current} = \{p_i - p_j \mid i \neq j\ i, j = 1..N\}, \tag{2}$$

$$f_{motion} = \{p_i(t) - p_i(t-1) \mid i = 1..N\}, \tag{3}$$

$$f_{offset} = \{p_i(t) - p_i(0) \mid i = 1..N\}, \tag{4}$$

where $p(0)$ denotes the original posture in each action sequence. The first frame in each sequence (the neutral posture) is used as $p(0)$ in our experiment. A linear normalization is applied to normalize the feature values to the range of $[-1, 1]$.

2.3. Feature Quantization

Now we have two features: the spatiotemporal features representing local motions at different 3D interest points, and the skeleton joints features representing spatial locations of body parts. To represent each action sequence, we quantize the STIPs features and the skeleton joints feature, respectively, based on K-means clustering. The cluster centers are used as the keywords to construct the histogram bins. Then histogram features are extracted by counting the occurrences of keywords in each depth action sequence. After this step, each action sequence can be represented as two histograms of features: the histogram of STIPs features, and the histogram of skeleton joints features. Then these features are used in the next step for feature-level fusion and action classification.

2.4. Fusion and Classification using the Random Forests

In order to perform the fusion and feature selection of spatiotemporal features and the skeleton joints features, we propose to use the random forests (RFs) method [3]. RFs are usually considered as a classifier using tree predictors in which each tree splits the data depends on the randomly selected features. There are many nice properties to use the random forests: (1) robustness to noise, (2) efficiency for classification, and (3) the improvement of accuracy by growing multiple trees and vote for the most popular class. Here we use the RFs for fusion of distinct features and action classification together.

Let the feature vector be $v \in \mathbb{R}^N$, where the number of the features for each sample is N. A number $n < N$ is specified at each node of the tree, where n features are randomly selected to determine the split of that node. The randomly selected n features contain spatiotemporal features partially and the skeleton joints partially. In this way, the feature fusion is executed randomly and naturally in the tree building process.

At each node, the split function, $f_n(v) : R^n \rightarrow R$, is based on comparing to a threshold $t_n \in R$; the left split is then,

$$I_{left} = \{i \in I_n | f(v_i) < t\}, \tag{5}$$

where I are the training examples falling into that node. The feature vector, v_i, is of length n, and randomly selected from the N features. The best split is determined by the information gain using these features:

$$\Delta Gain = \sum_{i=1}^{2} \left(\frac{|I_i|}{I} \sum_{j=1}^{C} q_{i,j} \cdot \log_2(q_{i,j}) \right), \tag{6}$$

where $|\cdot|$ is the size of the set. I_i are the two splits, I_{left} and I_{right}, $q_{i,j}$ is the proportion of samples in I_i belonging to class j, and C is the number of classes. Several decision trees are growing to generate a forest in this way, and each tree grows until it reaches the maximum tree depth max_{dep}, or the tree node receives the given number of minimum samples min_{node}. In the leaf nodes, the probabilistic distribution for each class is computed.

In testing, each test sample x goes down to one of the leaf nodes in each tree, denoted as $l(t, x)$, which contains the distribution P_n of all the classes. Random forests classifier chooses the most popular class label which gets the most vote over all the trees. The class label is determined by:

$$\hat{c} = \arg\max_j \frac{1}{T} \sum_{t=1}^{T} p^j_{l(t,x)}, \tag{7}$$

where $\hat{c}$ is the predicted class label, T is the total number of trees, $l(t, x)$ is the leaf node of tree t where the test sample x falling into. $p^j_{l(t,x)}$ is the posterior probabilities for class j at leaf node $l(t, x)$, $p^j_n = \frac{|S_j|}{|S|}$, where $|S|$ is the total number of samples in this leaf node and $|S_j|$ is the number of samples of class j in S.

3. Experiments

Our action recognition experiments are conducted on three challenging 3D action databases. We introduce the three databases and the experimental settings, and then present and analyze the experimental results.

3.1. Databases and Experimental Settings

Three different 3D action databases are used in our experiments to test the performance of the proposed approach. The first is the MSRAction3D action dataset [10], capturing human subjects standing at the same place with most of the actions related to upper body movement, e.g. high arm wave, draw circle and boxing. The second is the UTKinect-Action dataset [26] which collects common human actions using the Kinect. The actions are very different from the MSRAction3D dataset, including walk, pickup, and throw. The third is the CAD-60 dataset [17], where 12 human activities are collected in five different locations, using the Kinect installed on a robot. The activities in this dataset are more complex than the other two datasets, such as opening pill container, cooking, and brushing teeth.

MSRAction3D dataset [10] captures 20 human actions using a depth camera similar to the Kinect sensor. Totally 10 subjects were asked to perform the 20 action classes 3 times. Each video clip is of resolution 640×480 at 15fps. We used all of the 557 video clips, along with the skeleton joint locations provided by [10]. In our experiment, we follow the same settings of “cross-subjects” as in [10]. The whole dataset was divided into 3 subsets, half of the subjects are used for training while the other half of subjects are used for testing. The final accuracy on this dataset is the average of the accuracies on the three subsets.

UTKinect-Action dataset [26] contains 10 different action classes performed by 10 subjects, collected by a stationary Kinect sensor. The 10 action classes are: *walk, sit down, stand up, pick up, carry, throw, push, pull, wave hands, clap hands.* Depth sequences are provided with resolution 320×240, and skeleton joint locations are also provided in this dataset. In our experiment, we used the cross-subjects scheme where half of the subjects are used for training while the remaining for testing, which is different from the leave-one-out scheme in [26] where more subjects were used for training in each round.

Cornell Activity Dataset-60 (CAD-60) [17] has 60 RGB-D sequences collected by Kinect sensor , each video is of length about 45s. In this dataset, four different subjects performed 10 different activities in five locations. The five locations are: *office, kitchen, bedroom, bathroom and living room*. To reduce the computational complexity, we first subsampled each video to the length about 500 frames. Then we follow the same procedure of “new person” as in [17] for training and testing.

We attempted different combinations of the STIPs detectors and descriptors, and they perform differently on different datasets, because of the different action categories and application scenarios. We selected the best spatiotemporal feature corresponding to each dataset in order to combine with the skeleton joints features. Specifically, the Harris3D detector and HOG/HOF descriptor are used to represent the local features on MSRAction3D dataset. On the UTKinect-Action dataset, the Harris3D detector and HOG3D descriptor perform the best to extract spatiotemporal features. On CAD-60 dataset, the Hessian detector and ESURF descriptor are adopted. The SVM classifier is applied to find the best spatiotemporal feature for each dataset. After feature extraction, the K-means clustering is applied to quantize the features. Empirically we set $K = 100$ to get the clusters or keywords. For the skeleton joints feature, we perform clustering similarly to obtain the histogram features. Then the random forests method is applied for data fusion, selection and action classification. The number of trees in the random forests is set in the range of $[1,\ 300]$ to explore the performance, and the number of features to select at each tree node is set in the range of $[5,\ 60]$ to observe the differences.

Table 1. Accuracies on three datasets. RFs denotes the random forests method.

MSRAction3D	**Acc.**
STIPs (Harris3D+HOG/HOF)	77.5%
Skeleton Joint Features	90.9%
Combined features with RFs	**94.3%**

UTKinect-Action	**Acc.**
STIPs (Harris3D+HOG3D)	80.8%
Skeleton Joint Features	87.9%
Combined features with RFs	**91.9%**

CAD-60	**Acc.**
STIPs (Hessian+ESURF)	75.0%
Skeleton Joint Features	81.3%
Combined features with RFs	**87.5%**

3.2. Experimental Results

We first evaluate the performance of the proposed approach on the three challenging 3D action datasets. Then we compare our results to the state-of-the-art methods to demonstrate the superiority of the proposed approach.

The experiment results on the MSRAction3D dataset are shown in Table 1. From the results one can see that, the accuracy is 77.5% using only the STIPs features and 90.9% using the skeleton joints features. The skeleton feature has a higher accuracy than the STIPs. It is probably because that the skeleton joint positions are relatively accurate in this dataset and the action classes are more separable based on measuring the body part positions. Using our fusion approach, the recognition rate is improved to 94.3%, higher than either of the two features. This result validates our proposed approach, i.e., combining the distinct, complementary features can improve the 3D action recognition performance. Figure 1 shows the confusion matrices on the three subsets, AS1, AS2, and AS3, respectively.

The experimental results on UTKinect-Action dataset can be found in Table 1. The STIPs features and the skeleton joints features can obtain recognition rates of 80.8% and 87.9%, respectively. A much better performance is obtained by the fusion with an accuracy of 91.9%. In the confusion matrix (see Figure 2), one can see only two action classes with the recognition rate below 0.9. These are dif-

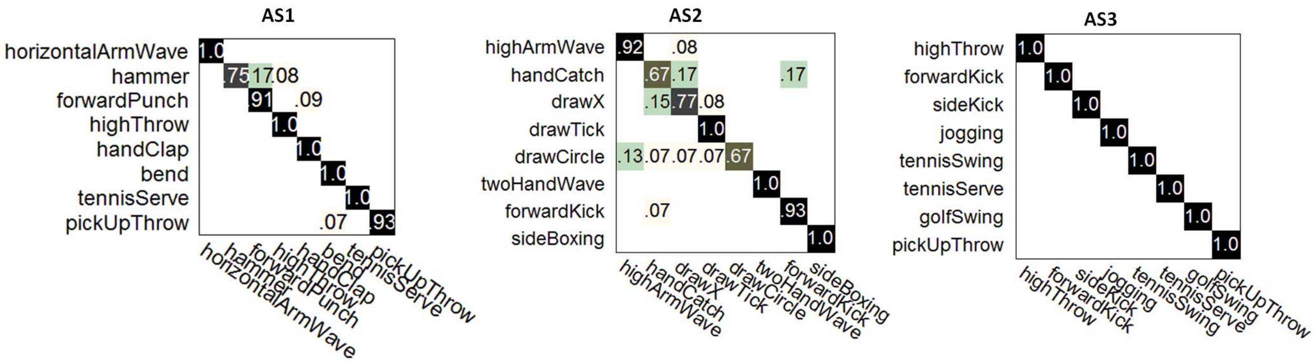

Figure 1. Confusion Matrix of the proposed approach on MSRAction3D dataset. AS1-3 means the action subsets, we follow the same settings as in [10].

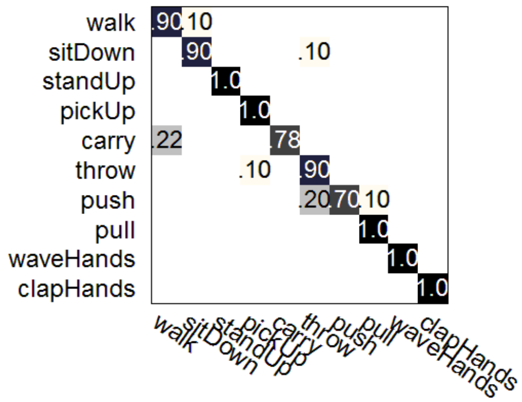

Figure 2. Confusion Matrix of the proposed approach on UTKinect-Action dataset.

ficult cases, such as distinguishing between carry and walk, push and throw, which have similar motions. Even with this, the proposed fusion scheme can still improve the recognition accuracy significantly.

The results on CAD-60 dataset are in Table 1. When only the STIPs feature is used, the recognition rate is 75.0%. When the skeleton joint feature is adopted only, the accuracy is 81.3%. By combining the two features, the accuracy can be increased to 87.5%, which is significantly higher than either feature.

In our experiments, we also tried different number of trees in the random forest. And we observe that the RFs are pretty robust with respect to the tree numbers on all of the three datasets, when the number of trees is 50 or above. The best results shown in the Table 1 are using 225, 150 and 50 trees respectively for MSRAction3D, UTKinect-Action, and CAD-60 datasets.

3.3. Comparison to the State-of-the-Art Results

We further compare our approach to the state-of-the-art approaches for 3D action recognition on the three datasets. In all our experiments, the cross-subjects action recognition is conducted, which is more difficult than using the same subjects for both training and testing [10, 27]. Note that on MSRAction3D dataset, the same settings are used for cross-subjects testing, where half of the subjects for training while the other half for testing. Table 2 shows all reported results that we can find on the MSRAction3D dataset. We can see that our result of 94.3% accuracy is better than all previous results using the same settings. On the UTKinect-Action dataset, our approach has an accuracy of 91.9% which outperforms the HOJ3D feature in [26] (90.9%). Note that in [26], a leave-one-out setting is applied, where more training samples while less test samples are used in their experiment. The cross-subject setting is applied in our experiment, and we still get a better accuracy. Finally, we compare our result with all others on the CAD-60 dataset. Note that the same "new person" setting as [17] is used in our experiment. Since the previous methods are measured with the presicion/recall, we also compute the presicion/recall on this dataset for a direct comparison. The results are shown in Table 3. Our approach obtained a much better accuracy compared to the state-of-the-art works on this dataset. Through the comparison, we can see that our approach outperforms all previous methods on the three challenging action datasets.

4. Conclusions

We have presented a new approach to 3D action recognition. It combines the spatiotemporal features and the skeleton joint features effectively based on the random forests learning method. The spatiotemporal features characterize the local motions while the skeleton joint distance measures depict the spatial distributions of the joints during an action

Table 2. Performance on MSRAction3D dataset using different methods.

Method	Accuracy
High Dimensional Convolutional Network [22]	72.5%
Action Graph on Bag of 3D Points [10]	74.7%
HOJ3D feature [26]	79.0%
Key Pose Learning [11]	80.3%
Eigenjoints [28]	82.3%
STOP feature [20]	84.8%
Random Occupancy Patterns [22]	86.2%
Actionlet [23]	88.2%
HON4D [14]	88.9%
Depth Motion Maps [27]	91.6%
Our approach	**94.3%**

Table 3. Performance on CAD-60 dataset using different methods.

Method	Precision/Recall
J. Sung *et al.* [17]	67.9%/55.5%
X. Yang *et al.* [28]	71.9%/66.6%
Koppula *et al.* [18]	80.8%/71.4%
Our approach	**93.2%/84.6%**

process. These two features can be complementary to each other, and an efficient combination of them can improve the 3D action recognition accuracies. We have conducted experiments on three challenging datasets, and shown that our proposed method can outperform all of the state-of-the-art methods on all three datasets. This demonstrates the good performance of our proposed approach. In future work, we will combine more cues to further improve the accuracy under our proposed feature-level fusion framework.

References

[1] J. Aggarwal and M. S. Ryoo. Human activity analysis: A review. *ACM Computing Surveys (CSUR)*, 43(3):16, 2011.

[2] H. Bay, T. Tuytelaars, and L. Van Gool. Surf: Speeded up robust features. In *Computer Vision–ECCV 2006*, pages 404–417. Springer, 2006.

[3] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[4] L. Chen, H. Wei, and J. M. Ferryman. A survey of human motion analysis using depth imagery. *Pattern Recognition Letters*, 2013.

[5] P. Dollár, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. In *Visual Surveillance and Performance Evaluation of Tracking and Surveillance, 2005. 2nd Joint IEEE International Workshop on*, pages 65–72. IEEE, 2005.

[6] C. Harris and M. Stephens. A combined corner and edge detector. In *Alvey vision conference*, volume 15, page 50. Manchester, UK, 1988.

[7] A. Klaser, M. Marszałek, C. Schmid, et al. A spatio-temporal descriptor based on 3d-gradients. In *British Machine Vision Conference*, 2008.

[8] I. Laptev and T. Lindeberg. Velocity adaptation of space-time interest points. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, volume 1, pages 52–56. IEEE, 2004.

[9] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

[10] W. Li, Z. Zhang, and Z. Liu. Action recognition based on a bag of 3d points. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*, pages 9–14. IEEE, 2010.

[11] L. Miranda, T. Vieira, D. Martinez, T. Lewiner, A. W. Vieira, and M. F. M. Campos. Real-time gesture recognition from depth data through key poses learning and decision forests. *2012 25th SIBGRAPI Conference on Graphics, Patterns and Images*, 0:268–275, 2012.

[12] B. Ni, G. Wang, and P. Moulin. Rgbd-hudaact: A color-depth video database for human daily activity recognition. In *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, pages 1147–1153. IEEE, 2011.

[13] F. Ofli, R. Chaudhry, G. Kurillo, R. Vidal, and R. Bajcsy. Berkeley mhad: A comprehensive multimodal human action database. In *Applications of Computer Vision (WACV), 2013 IEEE Workshop on*, pages 53–60. IEEE, 2013.

[14] O. Oreifej and Z. Liu. Hon4d: Histogram of oriented 4d normals for activity recognition from depth sequences.

[15] A. H. Shabani, D. A. Clausi, and J. S. Zelek. Evaluation of local spatio-temporal salient feature detectors for human action recognition. In *Computer and Robot Vision (CRV), 2012 Ninth Conference on*, pages 468–475. IEEE, 2012.

[16] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from single depth images. *CVPR*, 2:3, 2011.

[17] J. Sung, C. Ponce, B. Selman, and A. Saxena. Unstructured human activity detection from rgbd images. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 842–849. IEEE, 2012.

[18] H. Swetha Koppula, R. Gupta, and A. Saxena. Learning human activities and object affordances from rgb-d videos. 2012.

[19] P. Turaga, R. Chellappa, V. S. Subrahmanian, and O. Udrea. Machine recognition of human activities: A survey. *Circuits and Systems for Video Technology, IEEE Transactions on*, 18(11):1473–1488, 2008.

[20] A. Vieira, E. Nascimento, G. Oliveira, Z. Liu, and M. Campos. Stop: Space-time occupancy patterns for 3d action recognition from depth map sequences. *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, pages 252–259, 2012.

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

[22] J. Wang, Z. Liu, J. Chorowski, Z. Chen, and Y. Wu. Robust 3d action recognition with random occupancy patterns. In *Computer Vision–ECCV 2012*, pages 872–885. Springer, 2012.

[23] J. Wang, Z. Liu, Y. Wu, and J. Yuan. Mining actionlet ensemble for action recognition with depth cameras. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1290–1297. IEEE, 2012.

[24] D. Weinland, R. Ronfard, and E. Boyer. A survey of vision-based methods for action representation, segmentation and recognition. *Computer Vision and Image Understanding*, 115(2):224–241, 2011.

[25] G. Willems, T. Tuytelaars, and L. Van Gool. An efficient dense and scale-invariant spatio-temporal interest point detector. *Computer Vision–ECCV 2008*, pages 650–663, 2008.

[26] L. Xia, C. Chen, and J. Aggarwal. View invariant human action recognition using histograms of 3d joints. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*, pages 20–27. IEEE, 2012.

[27] X. Yang and Y. Tian. Eigenjoints-based action recognition using naive-bayes-nearest-neighbor. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*, pages 14–19. IEEE, 2012.

[28] X. Yang and Y. Tian. Effective 3d action recognition using eigenjoints. *Journal of Visual Communication and Image Representation*, 2013.

[29] X. Yang, C. Zhang, and Y. Tian. Recognizing actions using depth motion maps-based histograms of oriented gradients. In *Proceedings of the 20th ACM international conference on Multimedia*, pages 1057–1060. ACM, 2012.

[30] H. Zhang and L. E. Parker. 4-dimensional local spatio-temporal features for human activity recognition. In *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pages 2044–2049. IEEE, 2011.

[31] Y. Zhao, Z. Liu, L. Yang, and H. Cheng. Combing rgb and depth map features for human activity recognition. In *Signal Information Processing Association Annual Summit and Conference (APSIPA ASC), 2012 Asia-Pacific*, pages 1–4, Dec.