Action Recognition Using Completed Local Binary Patterns and Multiple-class Boosting Classifier

Yun Yang, Baochang Zhang, Linlin Yang
School of Automation Science and Electrical Engineering
Beihang University
Beijing, China
{yangyun,bczhang,yanglin}@buaa.edu.cn

Chen Chen
Department of Electrical Engineering
The University of Texas at Dallas
Richardson, TX, USA
chenchen870713@gmail.com

Wankou Yang
School of Automation

Southeast University
Nanjing, China
wkyang@seu.edu.cn

Abstract

This paper, for the first time, introduces a multiple-class boosting scheme (MBS) to combine depth motion maps (DMMs) and completed local binary patterns (CLBP) for action recognition. DMMs derive from projecting depth frames onto three orthogonal Cartesian planes (front, side and top) and characterize the motion energy of an action, on which the CLBP features are further extracted. And then a new multi-class boosting method is used and leads to an effective decision-level classifier. Extensive experiments on the MSRAction3D and MSRGesture3D datasets indicate that the proposed MBS method achieves new state-of-the-art results.

1. Introduction

Human action recognition has been an active research topic in computer vision. It has a wide range of applications including human computer interaction, motion sensing based gaming, intelligent surveillance and assisted living. Early research focuses on video sequences captured by RGB video cameras [1-4]. In [1], binary motion-energy images (MEI) and motion-history images (MHI) are used to represent where motion has occurred and characterize human actions. In [2], a low computational-cost volumetric action representation from different view angles is utilized to obtain high recognition rates. In [3], the notion of spatial interest points is extended to the spatio-temporal domain based on the idea of the Harris interest point operator. The results show robustness to occlusion and noise. In [4], motion descriptor based on optical flow measurements in a spatio-temporal volume is introduced to deal with low resolution images.

Despite researchers have made great progress in the past decades, robust action recognition in various real world conditions is still a challenging task. With the development of RGBD cameras, especially Microsoft Kinect, more recent research works on human action recognition have been carried out using depth images [17-19]. Compared with the conventional RGB cameras, depth cameras provide depth information which is robust to changes in lighting conditions. In [5], a bag of 3D points corresponding to the nodes in an action graph are used to recognize human action from depth sequences. An actionlet ensemble model is proposed in [6] and the developed local occupancy patterns features are shown to be robust to noise and invariant to translational and temporal misalignments. In [7], Histograms of Oriented Gradients (HOG) computed from Depth Motion Maps (DMMs) are used to capture body shape and motion information from depth images. In [17], Chen *et al*. combines Local binary pattern (LBP) and the extreme learning machine (ELM), which achieves the best performance on their own datasets. From the literature review, we know that the discriminative feature extraction and classifier design play key roles in the performance improvement. To enhance the robustness of the existing features, in this paper, DMMs and completed local binary patterns (CLBP) [10] are first used to extract discriminative features from depth images. And then a multi-class boosting scheme is proposed to combine the ELM base classifiers. More specifically, two main contributions of our work are summarized as follows:

1. The combination of CLBP features and DMMs is, for the first time, employed to represent human action. CLBP is a completed model based on LBP and achieves significant improvement for rotation invariant texture classification. Different from previous work focusing on feature-level combination, we pay attention to its application to action recognition based on the decision-level combination.

2. We propose a multi-class boosting algorithm to improve the performances of the ELM classifier [11]. Three kinds of ELM base classifiers on the CLBP features are obtained, which is further combined together to improve the classification performance. Inspired from the multi-class SVM [14] and AdaBoost methods [12], we propose a multiple-class boosting scheme (MBS) by adding an inequality constraint derived from [14] into the boosting objective. The extensive experimental results validate the proposed method.

The rest of the paper is organized as follows. Section 2 describes the details of DMMs-based CLBP features. Section 3 introduces the ELM classifier and the proposed multi-class boosting method. Experimental results are given in Section 4. Some concluding remarks are in Section 5.

2. Feature extraction from depth images

2.1. Depth motion maps

In a depth image, the pixel values indicate the distances between the surface of an object and a depth camera, therefore providing 3D structure information of a scene. Yang *et al*. [7] proposes to project each depth frame in a depth sequence onto three orthogonal Cartesian planes to make use of the 3D structure and shape information.

Chen *et al*. [8] modifies the procedure of generating DMMs to reduce the computational complexity. More specifically, each 3D depth frame generates three 2D projected maps according to front (f), side (s), and top (t) views, i.e. map_f, map_s, and map_t. Then the depth motion maps are calculated according to

$$DMM_{\{f,s,t\}} = \sum_{i=1}^{N-1} \left| map_{\{f,s,t\}}^{i+1} - map_{\{f,s,t\}}^{i} \right|, \qquad (1)$$

where i represents frame index and N is the total number of frames in a depth sequence. In this paper, we use the method in [8] to generate DMMs due to its computational efficiency.

2.2. DMMs-based CLBP features

LBP [9] is an effective texture descriptor has been used in various image processing and computer vision applications [20]. Given a center pixel t_c, its neighboring pixels are equally spaced on a circle of radius r ($r > 0$) with the center at t_c. If the coordinates of t_c are $(0,0)$ and m neighbors $\{t_i\}_{i=0}^{m-1}$ are considered, the coordinates of t_i are $(-r\sin(2\pi i/m), r\cos(2\pi i/m))$. The LBP is computed by thresholding the neighbors $\{t_i\}_{i=0}^{m-1}$ with the center pixel t_c to generate an m-bit binary number. The resulting LBP for t_c can be expressed in decimal form as follows:

$$LBP_{m,r}(t_c) = \sum_{i=0}^{m-1} s(t_i - t_c)2^i = \sum_{i=0}^{m-1} s(d_i)2^i, \qquad (2)$$

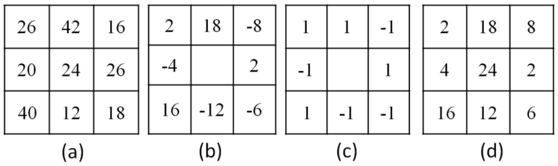
where $d_i = (t_i - t_c)$ is the difference between the center pixel and each neighbor, $s(d_i) = 1$ if $d_i \geq 0$ and $s(d_i) = 0$ if $d_i < 0$. The LBP only uses the sign information of d_i while ignoring the magnitude information. However, the sign and magnitude are complementary and they can be used to exactly reconstruct the difference d_i. In the CLBP scheme [10], the image local differences are decomposed into two complementary components: the signs and magnitudes (absolute values of d_i, i.e., $|d_i|$). Figure 1 shows an example of the sign and magnitude components of the CLBP extracted from a sample block. Note that "0" is coded as "-1" in CLBP [see Figure 1 (c)]. Two operators, namely CLBP-Sign (CLBP_S) and CLBP-Magnitude (CLBP_M), are used to code these two components. CLBP_S is equivalent to the traditional LBP operator. The CLBP_M operator is defined as follows:

$$CLBP\_M_{m,r} = \sum_{i=0}^{m-1} p(|d_i|, c)2^i, \quad p(u,c) = \begin{cases} 1, & u \geq c \\ 0, & u < c \end{cases} \qquad (3)$$

where c is a threshold that is set to the mean value of $|d_i|$ from the whole image. The CLBP-Center part which codes the values of the center pixels also has discriminant information. It is coded as:

$$CLBP\_C_{m,r} = p(t_c, c_1), \qquad (4)$$

where p is defined in (3) and the threshold c_1 is set as the average gray level of the whole image.



26	42	16
20	24	26
40	12	18

(a)

2	18	-8
-4		2
16	-12	-6

(b)

1	1	-1
-1		1
1	-1	-1

(c)

2	18	8
4	24	2
16	12	6

(d)

Figure 1: (a) 3×3 sample block; (b) the local differences; (c) the sign component of CLBP; and (d) the magnitude component of CLBP.

In our feature extraction method, DMMs are first generated from a depth sequence, and then the DMMs are divided into several overlapped blocks. Each component (CLBP_S, CLBP_M, and CLBP_C) of the CLBP operator is applied to each block from the three DMMs and histograms of all blocks are concatenated to form a single composite feature vector. Therefore, three CLBP histogram feature vectors denoted by DMM-CLBP_S, DMM-CLBP_M, and DMM-CLBP_C are obtained.

3. Decision-level classifier fusion based on a multiple-class boosting scheme

3.1. Extreme learning machine

ELM [11] is an efficient learning algorithm for single-hidden-layer feed-forward neural networks (SLFNs). Let $\mathbf{y} = [y_1, ..., y_k, ..., y_C]^T \in \mathbb{R}^C$ be the class to which a sample belongs, where $y_k \in \{1, -1\}$ ($1 \leq k \leq C$) and C is the number of classes. Given n training samples $\{\mathbf{x}_i, \mathbf{y}_i\}_{i=1}^{n}$, where $\mathbf{x}_i \in \mathbb{R}^M$ and $\mathbf{y}_i \in \mathbb{R}^C$, a single hidden layer neural network having L hidden nodes can be expressed as

$$\sum_{j=1}^{L} \boldsymbol{\beta}_j h(\mathbf{w}_j \cdot \mathbf{x}_i + e_j) = \mathbf{y}_i,\ i = 1,...,n, \tag{5}$$

where $h(\cdot)$ is a nonlinear activation function (e.g., Sigmoid function), $\boldsymbol{\beta}_j \in \mathbb{R}^C$ denotes the weight vector connecting the jth hidden node to the output nodes, $\mathbf{w}_j \in \mathbb{R}^M$ denotes the weight vector connecting the jth hidden node to the input nodes, and e_j is the bias of the jth hidden node. The above n equations can be written compactly as:

$$\mathbf{H}\boldsymbol{\beta} = \mathbf{Y}, \tag{6}$$

where $\boldsymbol{\beta} = [\boldsymbol{\beta}_1^T;...;\boldsymbol{\beta}_L^T] \in \mathbb{R}^{L\times C}$, $\mathbf{Y} = [\mathbf{y}_1^T;...;\mathbf{y}_n^T] \in \mathbb{R}^{n\times C}$, and $\mathbf{H}$ is the hidden layer output matrix. A least-squares solution $\hat{\boldsymbol{\beta}}$ of (6) is found to be

$$\hat{\boldsymbol{\beta}} = \mathbf{H}^{\dagger}\mathbf{Y}, \tag{7}$$

where $\mathbf{H}^{\dagger}$ is the *Moore-Penrose generalized inverse* of matrix $\mathbf{H}$. The output function of the ELM classifier is

$$\mathbf{f}_L(\mathbf{x}_i) = \mathbf{h}(\mathbf{x}_i)\boldsymbol{\beta} = \mathbf{h}(\mathbf{x}_i)\mathbf{H}^T\left(\frac{\mathbf{I}}{\rho} + \mathbf{H}\mathbf{H}^T\right)^{-1}\mathbf{Y}, \tag{8}$$

where $1/\rho$ is a regularization term. The label of a test sample is assigned to the index of the output nodes with the largest value. In our experiments, we use a kernel-based ELM (KELM) with a radial basis function (RBF) kernel.

3.2. Multiple-class boosting scheme

Ensemble learning is an effective approach to obtain high classification performance. Ensembles have more flexibility and can reduce problems related to over-fitting of the training data. As one of the outstanding types of ensembles, Boosting involves incrementally building an ensemble by training each new model instance to emphasize the training instances that previous models misclassified. In this paper, we concentrate on AdaBoost [12] which is the first practical boosting algorithm and introduce a new multi-class boosting method.

Let $h_i(x)$ denotes the ith base classifier, a boosting algorithm seeks for a convex linear combination

$$G(x) = \sum_{i=1}^{m} \omega_i h_i(x), \tag{9}$$

where ω_i is weight coefficient of the ith weak classifier. AdaBoost has proved to be equivalent to minimize the exponential loss function [13]:

$$\min_{\omega} \sum_{i=1}^{N} \exp(-y_i G(x_i)),\quad s.t.\quad \omega \geq 0\,. \tag{10}$$

The logarithmic function $\log(\cdot)$ is a strictly monotonically increasing function and it is easy to calculate the minimum value of a non-exponential function. AdaBoost is equal to solve [13]:

$$\min_{\omega} \log(\sum_{i=1}^{N} \exp(-y_i G(x_i))),\ \ s.t.\ \ \omega \geq 0, 1^T\omega = 1/T\,. \tag{11}$$

The constraint $1^T\omega = 1/T$ avoids enlarging the solution ω by an arbitrary large factor to make the cost function approach zero in the case of separable training data. In [14], Crammer and Singer propose to construct multiclass predictors with piecewise linear bound. Considering simplicity and efficiency of linear function, we use the following rule for multiple-class classification,

$$C(x) = \arg\max_{j=1}^{n}\{\omega_j{}^T \cdot x\}\,. \tag{12}$$

And then we heuristically propose the following linear objective function as:

$$\max_{j}(\omega_j^T \cdot x - \omega_m^T \cdot x), \tag{13}$$

Where $m \neq j$. Finally, we propose a multi-class boosting method as:

$$f(\omega_j^T, \omega_m) = \omega_j^T \cdot x - \omega_m^T \cdot x + \log(\sum_{i} \exp(z_i)) + \lambda \times \|\omega\|_2\,. \tag{14}$$

The method mentioned above is the proposed MBS approach. Once we obtain the probability output of three ELM classifiers utilizing DMM-CLBP_S, DMM-CLBP_M, and DMM-CLBP_C feature vectors, we conduct the MBS method to obtain the weight coefficients. Our method utilizes the information derived from DMMs and CLBP operators, to improve the performance of the ELM base classifiers. We solve our objective based on the MATLAB toolbox.

4. Experimental results

We evaluate our action recognition method on the MSRAction3D [5] and MSRGesture3D [15] datasets which consist of depth sequences captured by RGBD cameras. Our method is then compared with the existing methods.

4.1. MSRAction3D dataset

The MSRAction3D dataset [5] includes 20 actions performed by 10 subjects. Each subject performs each action 2 or 3 times. The resolution of each depth image is 240×320. Two different experimental settings are used to evaluate our method.

Setting 1 - The same experimental setting reported in [5] is followed. Specifically, the actions are divided into three subsets as listed in Table 1. For each subset, three different tests are performed. In test one, 1/3 of the samples are used for training and the rest for testing; in test two, 2/3 of the samples are used for training and the rest for testing; in the cross subject test, one half of the subjects (1, 3, 5, 7, 9) are used for training and the rest for testing.

Action set 1 (AS1)	**Action set 2 (AS2)**	**Action set 3 (AS3)**
Horizontal wave (2)	High wave (1)	High throw (6)
Hammer (3)	Hand catch (4)	Forward kick (14)
Forward punch (5)	Draw x (7)	Side kick (15)
High throw (6)	Draw tick (8)	Jogging (16)
Hand clap (10)	Draw circle (9)	Tennis swing (17)
Bend (13)	Two hand wave (11)	Tennis serve (18)
Tennis serve (18)	Forward kick (14)	Golf swing (19)
Pickup throw (20)	Side boxing (12)	Pickup throw (20)

Table 1: Three subsets of actions for the MSR-Action3D dataset.

Setting 2 - The same experimental setup in [24] is used. A total of 20 actions are employed and one half of the subjects (1, 3, 5, 7, 9) are used for training and the remaining subjects are used for testing.

To facilitate a fair comparison, we set the same parameters of DMMs and blocks as noted in [17]. We fix m=4 and r=1 for the CLBP operator in terms of classification performance and computational complexity.

The results shown in Table 2 clearly demostrate the effectiveness of our method. In test one, our method achieves 100% recognition accuracy in AS3 and comparable results to the highset accuracies in AS1 and AS2. In test two, our method reaches 100% recognition accuracy for all the three subsets. In the cross subject test, which is a challenging setting due to the large inter-class variations of different training and testing subjects, the MBS method achieves the highest average recognition accuracy.

The comparison results of setting 2 are illustrated in Table 3. Our approach achieves the highest recognition accuracy. It should be noted that DMM-LBP-DF is also a decision-level fusion method, the recognition accuracy of our method is more than 2% higher over DMM-LBP-DF which demonstrates the effectiveness of our proposed multi-class boosting fusion scheme.

Method	Accuracy (%)
DMM-HOG [7]	85.5
Random Occupancy Patterns [16]	86.5
DMM-LBP-FF [17]	91.9
DMM-LBP-DF [17]	93.0
HON4D [25]	88.9
Actionlet Ensemble [6]	88.2
Depth Cuboid [26]	89.3
Rahmani *et al.* [27]	88.8
Vemulapalli *et al.* [23]	89.5
MBS (Ours)	**95.2**

Table 3: Recognition accuracy (%) compared with previous methods on the MSRAction3D dataset.

4.2. MSRGesture3D dataset

The MSRGesture3D dataset [15] consists of 12 gestures defined by American Sign Language (ASL). It contains 333 depth sequences. The same parameters reported in [17] are used here for the sizes of DMMs and blocks.

Table 4 shows the recognition results of our method as well as other existing methods on the MSRGesture3D dataset. As can be seen from Table 4, decision level fusion approach (DMM-LBP-DF) achieves the closest recognition accuracy to our method. This indicates that the DMMs model and texture features contribute more effect on gesture recognition than the decision-level fusion. However, both our method and DMM-LBP-DF outperform other methods considerably.

Method	Accuracy (%)
Random Occupancy Patterns [16]	88.5
HON4D [25]	92.5
Rahmani *et al.* [27]	93.6
DMM-HOG [7]	89.2
DMM-LBP-FF [17]	93.4
DMM-LBP-DF [17]	94.6
Edge Enhanced DMM [28]	90.5
Kurakin *et al.* [15]	87.7
MBS (Ours)	**94.7**

Table 4: Recognition accuracy (%) compared with previous methods on the MSRGesture3D dataset.

Method	Test one				Test two				Cross subject			
	AS1	AS2	AS3	Average	AS1	AS2	AS3	Average	AS1	AS2	AS3	Average
Li *et al.* [5]	89.5	89.0	96.3	91.6	93.4	92.9	96.3	94.2	72.9	71.9	79.2	74.7
DMM-HOG [7]	97.3	92.2	98.0	95.8	98.7	94.	98.7	97.4	96.2	84.1	94.6	91.6
HOJ3D [21]	**98.5**	96.7	93.5	96.2	98.6	97.2	94.9	97.2	88.0	85.5	63.6	79.0
Chaaraoui *et al.* [22]	-	-	-	-	-	-	-	-	91.6	90.8	97.3	93.2
Vemulapalli *et al.* [23]	-	-	-	-	-	-	-	-	95.3	83.9	98.2	92.5
DMM-LBP-FF [17]	96.7	**100**	99.3	98.7	**100**	**100**	**100**	**100**	98.1	92.0	94.6	94.9
DMM-LBP-DF [17]	98.0	97.4	99.3	98.2	**100**	**100**	**100**	**100**	**99.1**	92.9	92.0	94.7
Occupancy Patterns [24]	98.2	94.8	97.4	96.8	99.1	97.0	98.7	98.3	91.7	72.2	**98.6**	87.5
MBS (Ours)	98.0	98.7	**100**	**98.9**	**100**	**100**	**100**	**100**	**99.1**	**93.8**	98.2	**97.0**

Table 2. Comparison of recognition accuracies (%) of our method and previous methods on the MSRAction3D dataset using setting 1.

5. Conclusion

In this paper, we propose an effective feature descriptor and a novel decision-level fusion method for action recognition. This feature descriptor combines depth motion maps (DMMs) and completed local binary patterns (CLBP) for an effective action representation of a depth video sequence. In decision-level fusion, we add the inequality constraints derived from multi-class Support Vector Machine to modify the general AdaBoost optimization function. Kernel-based extreme learning machine (KELM) classifiers are utilized as the basic classifiers of AdaBoost. The experimental results on two benchmark datasets (MSRAction3D and MSRGesture3D) demonstrate that our method outperforms the state-of-art methods.

Acknowledgement

We acknowledged the support of the Natural Science Foundation of China, under Contracts 61272052 and 61473086, and the Program for New Century Excellent Talents of the University of Ministry of Education of China.

References

[1] A. Bobick, and J. Davis. The recognition of human movement using temporal templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(3):257–267, 2001.

[2] A. Iosifidis, A. Tefas, and I. Pitas. Multi-view action recognition based on action volumes, fuzzy distances and cluster discriminant analysis. *Journal of Signal Processing*, 93(6):1445–1457, 2013.

[3] I. Laptev. On Space-Time Interest Points. *Journal of Computer Vision*, 64(2):432–439, 2005.

[4] A. A. Efros, E. C. Berg, G. Mori, and J. Malik. Recognizing action at a distance. In *ICCV*, pages 726–733, 2003.

[5] W. Li, Z. Zhang, and Z. Liu. Action recognition based on a bag of 3D points. In *CVPR Workshops*, pages 9–14, 2010.

[6] J. Wang, Z. Liu, Y. Wu, and J. Yuan. Mining actionlet ensemble for action recognition with depth cameras. In *CVPR*, pages 1290–1297, 2012.

[7] X. Yang, C. Zhang, and Y. Tian. Recognizing actions using depth motion maps-based histograms of oriented gradients. In *ACM Multimedia*, pages 1057–1060, 2012.

[8] C. Chen, K. Liu, and N. Kehtarnavaz. Real time human action recognition based on depth motion maps. *Journal of Real-Time Image Processing*, 2013.

[9] T. Ojala, M. Pietikäinen, and T. Mäenpää. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, 2002.

[10] Z.-H. Guo, L. Zhang, and D. Zhang. A completed modeling of local binary pattern operator for texture classification. *IEEE Transactions on Image Processing*, 19(6):1657–1663, 2010.

[11] G.-B. Huang, H. Zhou, X. Ding, and R. Zhang. Extreme learning machine for regression and multiclass classification. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 42(2):513–529, 2012.

[12] Y. Freund, and R. E. Schapire. Experiments with a New Boosting Algorithm. *International Conference on Machine Learning*, pages 148–156, 1996.

[13] M. Collins, R.E. Schapire, and Y. Singer. Logistic Regression, AdaBoost and Bregman Distances. *Machine Learning*, 48(1):253-285, 2002.

[14] K. Crammer, and Y. Singer. On the algorithmic implementation of multiclass kernel-based vector machines. *Journal of Machine Learning Research*, 2(2), 2001.

[15] A. Kurakin, Z. Zhang, and Z. Liu. A real time system for dynamic hand gesture recognition with a depth sensor. In *EUSIPCO*, pages 1975–1979, 2012.

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

[17] C. Chen, R. Jafari, and N. Kehtarnavaz. Action Recognition from Depth Sequences Using Depth Motion Maps-Based Local Binary Patterns. In *WACV*, pages 1092-1099, 2015.

[18] C. Chen, R. Jafari, and N. Kehtarnavaz. Improving Human Action Recognition Using Fusion of Depth Camera and Inertial Sensors, *IEEE Transactions on Human-Machine Systems*, 45(1):51–61, 2015.

[19] C. Chen, N. Kehtarnavaz, and R. Jafari. A Medication Adherence Monitoring System for Pill Bottles Based on a Wearable Inertial Sensor. *IEEE Engineering in Medicine and Biology Society*, pages 4983-4986, 2014.

[20] W. Li, C. Chen, H. Su, and Q. Du. Local Binary Patterns for Spatial-Spectral Classification of Hyperspectral Imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 53(7):3681–3693, 2015.

[21] L. Xia, C.-C. Chen, and J. Aggarwal. View invariant human action recognition using histograms of 3d joints. In *CVPR Workshops*, pages 20–27, 2012.

[22] A. A. Chaaraoui, J. R. Padilla-López, P. Climent-Pérez, and F. Flórez-Revuelta. Evolutionary joint selection to improve human action recognition with rgb-d devices. *Expert Systems with Applications*, 41(3):786–794, 2014.

[23] R. Vemulapalli, F. Arrate, and R. Chellappa. Human action recognition by representing 3d human skeletons as points in a lie group. In *CVPR*, 2014.

[24] A. W. Vieira, E. R. Nascimento, G. L. Oliveira, Z. Liu, and M. F. Campos. On the improvement of human action recognition from depth map sequences using space-time occupancy patterns. *Pattern Recognition Letters*, 36:221–227, 2014.

[25] O. Oreifej and Z. Liu. Hon4d: Histogram of oriented 4d normals for activity recognition from depth sequences. In *CVPR*, pages 716–723, 2013.

[26] L. Xia and J. Aggarwal. Spatio-temporal depth cuboid similarity feature for activity recognition using depth camera. In *CVPR*, pages 2834–2841, 2013.

[27] H. Rahmani, A. Mahmood, D. Q. Huynh, and A. Mian. Real time action recognition using histograms of depth gradients and random decision forests. In *WACV*, pages 626–633, 2014.

[28] C. Zhang and Y. Tian. Edge enhanced depth motion map for dynamic hand gesture recognition. In *CVPR Workshops*, pages 500–505, 2013.