Gradient Local Auto-Correlations and Extreme Learning Machine for Depth-Based Activity Recognition

Chen Chen[1], Zhenjie Hou[2(✉)], Baochang Zhang[3], Junjun Jiang[4], and Yun Yang[3]

[1] Department of Electrical Engineering, University of Texas at Dallas, Richardson, TX, USA
chenchen870713@gmail.com
[2] School of Information Science and Engineering, Changzhou University, Changzhou, China
houzj@cczu.edu.cn
[3] School of Automation Science and Electrical Engineering, Beihang University, Beijing, China
bczhang@buaa.edu.cn, bhu_yunyang@163.com
[4] School of Computer Science, China University of Geosciences, Wuhan, China
junjun0595@163.com

Abstract. This paper presents a new method for human activity recognition using depth sequences. Each depth sequence is represented by three depth motion maps (DMMs) from three projection views (front, side and top) to capture motion cues. A feature extraction method utilizing spatial and orientational auto-correlations of image local gradients is introduced to extract features from DMMs. The gradient local auto-correlations (GLAC) method employs second order statistics (i.e., auto-correlations) to capture richer information from images than the histogram-based methods (e.g., histogram of oriented gradients) which use first order statistics (i.e., histograms). Based on the extreme learning machine, a fusion framework that incorporates feature-level fusion into decision-level fusion is proposed to effectively combine the GLAC features from DMMs. Experiments on the MSRAction3D and MSRGesture3D datasets demonstrate the effectiveness of the proposed activity recognition algorithm.

Keywords: Gradient local auto-correlations · Extreme learning machine · Activity recognition · Depth images · Depth motion map

1 Introduction

Human activity recognition is one of the important areas of computer vision research today. It has a wide range of applications including intelligent video surveillance, video analysis, assistive living, robotics, telemedicine, and human computer interaction (e.g., [1–4]). Research on human activity recognition has initially focused on learning and recognizing activities from video sequences captured by conventional RGB cameras.

Since the recent release of cost-effective 3D depth cameras using structured light or time-of-flight sensors, there has been great interest in solving the problem of human

G. Bebis et al. (Eds.): ISVC 2015, Part I, LNCS 9474, pp. 613–623, 2015.
DOI: 10.1007/978-3-319-27857-5_55

activity recognition by using 3D data. Compared with traditional color images, depth images are insensitive to changes in lighting conditions and provide body shape and structure information for activity recognition. Color and texture are precluded in the depth images, which makes the tasks of human detection and segmentation easier [5]. Moreover, human skeleton information can be estimated from depth images providing additional information for activity recognition [6].

Research on activity recognition has explored various representations (e.g., 3D point cloud [7], projection depth maps [8], spatio-temporal interest points [9], and skeleton joints [10]) of depth sequences. In [7], a bag of 3D points was sampled from depth images to characterize the 3D shapes of salient postures and Gaussian mixture model (GMM) was used to robustly capture the statistical distribution of the points. A filtering method to extract spatio-temporal interest points (STIPs) from depth videos (called DSTIP) was introduced in [9] to localize activity related interest points by effectively suppressing the noise in the depth videos. Depth cuboid similarity feature (DCSF) built around the DSTIPs was proposed to describe the local 3D depth cuboid. Inspired by motion energy images (MEI) of motion history images (MHI) [11], depth images in a depth video sequence were projected onto three orthogonal planes and differences between projected depth maps were stacked to form depth motion maps (DMMs) [8]. Histogram of oriented gradients (HOG) [12] features were then extracted from DMMs as global representations of a depth video. DMMs effectively transform the problem in 3D to 2D. In [13], the procedure of generating DMMs was modified to reduce the computational complexity in order to achieve real-time action recognition. Later in [14], local binary pattern [15] operator was applied to the overlapped blocks in DMMs to enhance the discriminative power for action recognition. Skeleton information has also been explored for activity recognition, for example [10]. Reviews of skeleton based activity recognition methods are referred to [10, 16].

Motivated by the success of DMMs in depth-based activity recognition, our method proceeds along with this direction. Specifically, we introduce the gradient local auto-correlations (GLAC) [17] descriptor and present a new feature extraction method using GLAC and DMMs. A fusion framework based on extreme learning machine (ELM) [18] is proposed to effectively combine the GLAC features from DMMs for activity recognition. The main contributions of this paper are summarized as follows:

1. We introduce a new feature descriptor, GLAC, to extract features from DMMs of depth sequences. The GLAC descriptor, which is based on the second order of statistics of gradients (spatial and orientational auto-correlations of local image gradients), can effectively capture rich information from images.
2. We present a unified fusion framework which incorporates feature-level fusion into decision-level fusion for activity recognition.
3. We demonstrate that ELM has better performance than support vector machine (SVM) in our proposed method for depth-based activity recognition.

The rest of this paper is organized as follows. Section 2 describes the proposed feature extraction method. Section 3 overviews ELM and provides details of the unified fusion framework for activity recognition. Experimental results and discussions are presented in Sect. 4. Finally, Sect. 5 concludes the paper.

2 Feature Extraction from Depth Sequences

2.1 Depth Motion Map

To extract features from depth images, depth motion maps (DMMs) discussed in [13] are used due to their computational efficiency. More specifically, each 3D depth image in a depth video sequence is first projected onto three orthogonal Cartesian planes to generate three 2D projected maps corresponding to front, side, and top views, denoted by map_f, map_s, and map_t, respectively. For a depth video sequence with N frames, the DMMs are obtained as follows:

$$DMM_{\{f,s,t\}} = \sum_{i=1}^{N-1} \left| map^{i+1}_{\{f,s,t\}} - map^{i}_{\{f,s,t\}} \right|, \tag{1}$$

where i represents frame index. A bounding box is considered to extract the foreground in each DMM. Since foreground DMMs of different video sequences may have different sizes, bicubic interpolation is applied to resize all such DMMs to a fixed size and thus to reduce the intra-class variability. Figure 1 shows two example sets of DMMs.

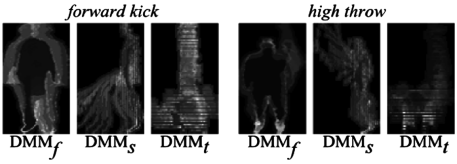

Fig. 1. DMMs for the *forward kick* and *high throw* depth action video sequences.

2.2 Gradient Local Auto-Correlations

GLAC [17] descriptor is an effective tool for extracting shift-invariant image features. Let I be an image region and $\mathbf{r} = (x, y)^t$ be a position vector in I. The magnitude and the orientation angle of the image gradient at each pixel can be represented by $n = \sqrt{\frac{\partial I^2}{\partial x} + \frac{\partial I^2}{\partial y}}$ and $\theta = \arctan\left(\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y}\right)$, respectively. The orientation θ is then coded into D orientation bins by voting weights to the nearest bins to form a gradient orientation vector $\mathbf{f} \in \mathbb{R}^D$. With the gradient orientation vector $\mathbf{f}$ and the gradient magnitude n, the N^{th} order auto-correlation function of local gradients can be expressed as follows:

$$R(d_0, \ldots, d_N, \mathbf{a}_1, \ldots, \mathbf{a}_N) = \int_I \omega[n(\mathbf{r}), n(\mathbf{r}+\mathbf{a}_1), \ldots, n(\mathbf{r}+\mathbf{a}_N)] f_{d_0}(\mathbf{r}) f_{d_1}(\mathbf{r}+\mathbf{a}_1) \cdots f_{d_N}(\mathbf{r}+\mathbf{a}_N) d\mathbf{r}, \tag{2}$$

where $\mathbf{a}_i$ are displacement vectors from the reference point $\mathbf{r}$, f_d is the d^{th} element of $\mathbf{f}$, and $\omega(\cdot)$ indicates a weighting function. In the experiments reported later, $N \in \{0, 1\}$,

$a_{1x,y} \in \{\pm\Delta r, 0\}$, and $\omega(\cdot) \equiv \min(\cdot)$ were considered as suggested in [17], where Δr represents the displacement interval in both horizontal and vertical directions. For $N \in \{0, 1\}$, the formulation of GLAC is given by

$$\begin{aligned} \mathbf{F}_0 : R_{N=0}(d_0) &= \sum_{\mathbf{r}\in I} n(\mathbf{r}) f_{d_0}(\mathbf{r}) \\ \mathbf{F}_1 : R_{N=1}(d_0, d_1, \mathbf{a}_1) &= \sum_{\mathbf{r}\in I} \min[n(\mathbf{r}), n(\mathbf{r}+\mathbf{a}_1)] f_{d_0}(\mathbf{r}) f_{d_1}(\mathbf{r}+\mathbf{a}_1). \end{aligned} \tag{3}$$

The spatial auto-correlation patterns of $(\mathbf{r}, \mathbf{r}+\mathbf{a}_1)$ are shown in Fig. 2.

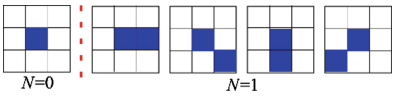

Fig. 2. Configuration patterns of $(\mathbf{r}, \mathbf{r}+\mathbf{a}_1)$.

The dimensionality of the above GLAC features ($\mathbf{F}_0$ and $\mathbf{F}_1$) is $D + 4D^2$. Although the dimensionality of the GLAC features is high, the computational cost is low due to the sparseness of $\mathbf{f}$. It is worth noting that the computational cost is invariant to the number of bins, D, since the sparseness of $\mathbf{f}$ doesn't depend on D.

2.3 DMMs-Based GLAC Features

DMMs generated from a depth sequence are pixel-level features. To enhance the discriminative power and gain a compact representation, we adopt the method in [14] to extract GLAC features from DMMs. Specifically, DMMs are divided into several overlapped blocks and the GLAC descriptor is applied to each block to compute GLAC features (i.e., $\mathbf{F}_0$ and $\mathbf{F}_1$). For each DMM, GLAC features from all the blocks are concatenated to form a single composite feature vector. Therefore, three feature vectors $\mathbf{g}_1$, $\mathbf{g}_2$ and $\mathbf{g}_3$ corresponding to three DMMs are obtained for a depth sequence.

3 Classification Fusion Based on ELM

3.1 Elm

ELM [18] is an efficient learning algorithm for single hidden layer feed-forward neural networks (SLFNs) and has been applied in various applications (e.g., [19, 20]).

Let $\mathbf{y} = [y_1, \ldots, y_k, \ldots, y_C]^T \in \mathbb{R}^C$ be the class to which a sample belongs, where $y_k \in \{1, -1\}$ $(1 \le k \le C)$ 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

$$\sum_{j=1}^{L} \boldsymbol{\beta}_j h\left(\mathbf{w}_j \cdot \mathbf{x}_i + e_j\right) = \mathbf{y}_i, \quad i = 1, \ldots, n, \tag{4}$$

where $h(\cdot)$ is a nonlinear activation function, $\boldsymbol{\beta}_j \in \mathbb{R}^C$ denotes the weight vector connecting the j^{th} hidden node to the output nodes, $\mathbf{w}_j \in \mathbb{R}^M$ denotes the weight vector connecting the j^{th} hidden node to the input nodes, and e_j is the bias of the j^{th} hidden node. (4) can be written compactly as:

$$\mathbf{H}\boldsymbol{\beta} = \mathbf{Y}, \tag{5}$$

where $\boldsymbol{\beta} = [\boldsymbol{\beta}_1^T; \ldots; \boldsymbol{\beta}_L^T] \in \mathbb{R}^{L \times C}$, $\mathbf{Y} = [\mathbf{y}_1^T; \ldots; \mathbf{y}_n^T] \in \mathbb{R}^{n \times C}$, and $\mathbf{H}$ is the hidden layer output matrix. A least-squares solution $\hat{\beta}$ to (5) is

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

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

$$\mathbf{f}_L(\mathbf{x}_i) = \mathbf{h}(\mathbf{x}_i)\boldsymbol{\beta} = \mathbf{h}(\mathbf{x}_i)\mathbf{H}^T \left(\frac{\mathbf{I}}{\rho} + \mathbf{H}\mathbf{H}^T\right)^{-1} \mathbf{Y}, \tag{7}$$

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 Proposed Fusion Framework

In [14], both feature-level fusion and decision-level fusion were examined for action recognition. It was demonstrated that decision-level fusion had better performance than feature-level fusion. To further improve the performance of decision-level fusion, we propose a unified fusion framework that incorporates feature-level fusion into decision-level fusion.

In decision-level fusion, each feature (e.g., $\mathbf{g}_1$, $\mathbf{g}_2$, and $\mathbf{g}_3$) is used individually as input to an ELM classifier. The probability outputs of each individual classifier are merged to generate the final outcome. The posterior probabilities are estimated using the decision function of ELM (i.e., $\mathbf{f}_L$ in (7)) since it estimates the accuracy of the output label. $\mathbf{f}_L$ is normalized to $[0, 1]$ and Platt's empirical analysis [21] using a Sigmoid function is utilized to approximate the posterior probabilities,

$$p(y_k|\mathbf{x}) = \frac{1}{1 + \exp\left(Af_L(\mathbf{x})_k + B\right)}, \tag{8}$$

where $f_L(\mathbf{x})_k$ is the k^{th} output of the decision function $\mathbf{f}_L(\mathbf{x})$. In our experiments, $A = -1$ and $B = 0$. Logarithmic opinion pool (LOGP) [20] is used to estimate a global membership function:

$$\log P(y_k|\mathbf{x}) = \sum_{q=1}^{Q} \alpha_q p_q(y_k|\mathbf{x}), \tag{9}$$

where Q is the number of classifiers and $\{\alpha_q\}_{q=1}^{Q}$ are uniformly distributed classifier weights. The final class label y^* is determined according to

$$y^* = \arg\max_{k=1,\ldots,C} P(y_k|\mathbf{x}). \tag{10}$$

To incorporate feature-level fusion into decision-level fusion, we stack $\mathbf{g}_1$, $\mathbf{g}_2$, and $\mathbf{g}_3$ (feature-level fusion) as the fourth feature $\mathbf{g}_4 = [\mathbf{g}_1, \mathbf{g}_2, \mathbf{g}_3]$. Then these four feature vectors are used individually as inputs to four ELM classifiers. Note that principal component analysis (PCA) is employed for dimensionality reduction of the feature vectors. For a testing sample $\mathbf{x}$, five sets of probability outputs including four sets of probability outputs $\{p_q(y_k|\mathbf{x})\}_{q=1}^{4}$ corresponding to the four classifiers and a set of fusion probability outputs $P(y_k|\mathbf{x})$ by using (9) can be obtained. The class label of the testing sample $\mathbf{x}$ is assigned based on $P(y_k|\mathbf{x})$.

Since each set of probability outputs is able to make a decision on the class label of the testing sample $\mathbf{x}$, we could use the label information (five class labels) from the five sets of probability outputs to reach a more robust classification decision. Here, we employ the majority voting strategy on the five labels. If at least three class labels are the same, we consider there is a major certainty among the five sets of probability outputs and use the majority voted label as the final class label; otherwise, we use the label given by $P(y_k|\mathbf{x})$. By introducing a majority voting step in the decision-level fusion, we not only consider the classification probability but also the classification certainty.

4 Experiments

In this section, we evaluate our proposed activity recognition method on the public MSRAction3D [7] and MSRGesture3D [22] datasets which consist of depth sequences captured by RGBD cameras. Some example depth images from these two datasets are presented in Fig. 3. Our method is then compared with the existing methods. The source code of our method will be available on our website.

4.1 MSRAction3D Dataset

The MSRAction3D dataset [7] includes 20 actions performed by 10 subjects. Each subject performs each action 2 or 3 times. To facilitate a fair comparison, the same

(a) MSRAction3D

(b) MSRGesture3D

Fig. 3. Sample depth images of different actions/gestures.

experimental setup in [23] 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 rest subjects are used for testing.

First of all, we estimate the optimal parameter set $(D, \Delta r)$ for the GLAC descriptor using the training data via five-fold cross validation. The recognition results with various parameter sets for the MSRAction3D dataset are shown in Table 1. Therefore, $D = 10$ and $\Delta r = 8$ are chosen in terms of the activity recognition accuracy.

Table 1. Recognition accuracy (%) of GLAC with different parameter sets $(D, \Delta r)$ for the MSRAcion3D dataset using training data

Δr \ D	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
1	79.0	84.5	87.8	90.7	90.4	90.0	90.4	91.5	92.2	92.2	92.6	91.8
2	78.3	84.2	87.1	89.6	91.1	90.7	91.5	91.1	91.1	90.7	91.8	91.5
3	79.4	82.7	86.7	89.6	90.7	90.4	91.8	91.5	91.5	91.8	91.1	91.8
4	79.8	82.0	84.9	88.9	88.9	88.5	90.7	91.1	91.1	90.0	91.5	91.1
5	81.2	84.5	85.6	90.4	90.4	91.8	92.2	92.2	92.9	92.9	92.6	92.6
6	83.1	86.0	86.3	90.7	91.5	91.1	92.6	91.8	92.9	92.9	92.9	92.9
7	83.8	86.7	86.3	90.4	90.4	92.6	91.8	92.9	92.9	93.7	92.9	92.6
8	84.2	86.0	86.3	90.4	91.5	92.6	92.9	93.3	93.7	93.7	93.7	92.9
9	84.2	83.8	85.6	89.6	90.0	90.7	91.8	91.8	92.6	92.9	92.9	92.9
10	82.3	85.3	85.6	89.3	90.0	90.7	91.5	91.8	92.6	92.6	92.6	92.2

A comparison of our method with the existing methods is carried out. The outcome of the comparison is listed in Table 2. We also report the results of feature-level fusion and decision-level fusion using only $\mathbf{g}_1$, $\mathbf{g}_2$, and $\mathbf{g}_3$. These two methods are denoted by DMM-GLAC-DF and DMM-GLAC-FF. As we can see that our method achieves an accuracy of 92.31 %, only 0.78 % inferior to the state-of-the-art accuracy (93.09 %) of SNV [5]. Moreover, the proposed unified fusion method outperforms DMM-GLAC-FF by 1.83 %, which demonstrates the benefit of incorporating the concatenated feature $\mathbf{g}_4$ into decision-level fusion. The confusion matrix of our method for the MSRAction3D dataset is shown in Fig. 4(a). The recognition errors concentrate on similar actions, e.g., *draw circle* and *draw tick*. This is mainly because the DMMs of these actions are similar.

Table 2. Comparison of recognition accuracy on the MSRAction3D dataset

Method	Accuracy
Bag of 3D points [7]	74.70 %
EigenJoints [10]	82.30 %
STOP [24]	84.80 %
Random Occupancy Pattern [23]	86.50 %
Actionlet Ensemble [25]	88.20 %
DMM-HOG [8]	88.73 %
Histograms of Depth Gradients [27]	88.80 %
HON4D [26]	88.89 %
DSTIP [9]	89.30 %
DMM-LBP-FF [14]	91.90 %
DMM-LBP-DF [14]	93.00 %
SNV [5]	**93.09 %**
DMM-GLAC-FF	89.38 %
Proposed	90.48 %
DMM-GLAC-DF	92.31 %

4.2 MSRGesture3D Dataset

The MSRGesture3D dataset [22] consists of 12 gestures defined by American Sign Language (ASL). It contains 333 depth sequences. The leave one subject out cross-validation test [23] is utilized in our evaluation. $D = 12$ and $\Delta r = 1$ are selected for GLAC based on the parameter tuning experiment for this dataset. Our method obtains the state-of-the-art accuracy of 95.5 % which outperforms all previous methods as shown in Table 3. The confusion matrix of our method for the MSRGesture3D dataset is demonstrated in Fig. 4(b).

Table 3. Comparison of recognition accuracy on the MSRGesture3D dataset

Method	Accuracy
Random Occupancy Pattern [23]	88.50 %
DMM-HOG [8]	89.20 %
Histograms of Depth Gradients [27]	93.60 %
HON4D [26]	92.45 %
Action Graph on Silhouett [22]	87.70 %
DMM-LBP-FF [14]	93.40 %
DMM-LBP-DF [14]	94.60 %
SNV [5]	94.74 %
Proposed	**95.50 %**

4.3 ELM vs. SVM

We also conduct comparison between ELM and SVM for activity recognition. For our method using SVM as the classifier, LIBSVM [28] toolbox is utilized to provide

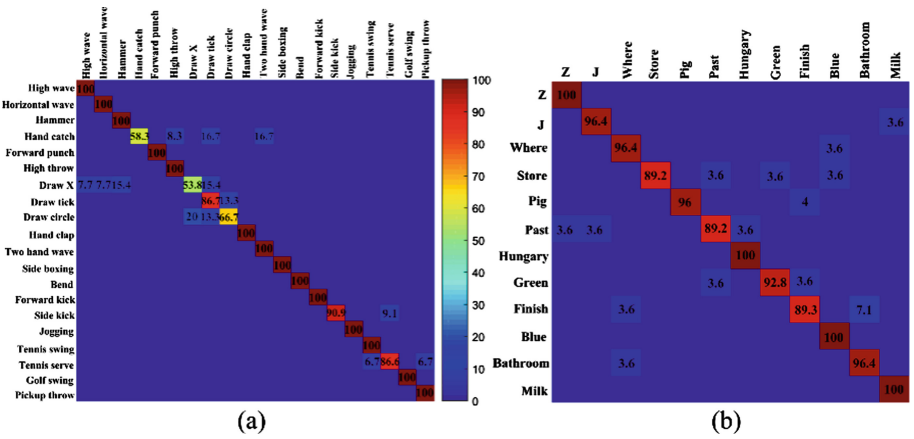

Fig. 4. Confusion matrices of our method on (a) the MSRAction3D dataset and (b) the MSRGesture3D dataset. This figure is best seen on screen.

probability estimates for multi-class classification. The comparison results in terms of recognition accuracy are presented in Table 4. It is easy to see that ELM has superior performance over SVM for both datasets. The standardized McNemar's test [20] is employed to verify the statistical significance in accuracy improvement of our ELM-based method. A value of $|Z| > 1.96$ in the McNemar's test indicates there is a significant difference in accuracy between two classification methods. The sign of Z indicates whether classifier 1 statistically outperforms classifier 2 ($Z > 0$) or vice versa. As we can see that the ELM-based method statistically outperforms SVM-based method.

Table 4. Performance comparison between ELM and SVM in our method

MSRAction3D		MSRGesture3D	
Accuracy		Accuracy	
ELM	92.31 %	ELM	95.50 %
SVM	86.80 %	SVM	92.80 %
Z/significant?		Z/significant?	
ELM (classifier 1) *vs*. SVM (classifier 2)		ELM (classifier 1) *vs*. SVM (classifier 2)	
3.13/yes		2.04/yes	

5 Conclusions

We have presented a novel framework for activity recognition from depth sequences. The gradient local auto-correlations (GLAC) features utilize spatial and orientational auto-collections of local gradients to describe the rich texture information of the depth motion maps generated from a depth sequence. A unified fusion scheme that combines

feature-level fusion and decision-level fusion is proposed based on extreme learning machine for activity recognition. Our method is evaluated on two public benchmark datasets and the experimental results demonstrate that the proposed method can achieve competitive or better performance compared to a number of state-of-the-art methods.

Acknowledgement. We acknowledge the support of the Industry, Teaching and Research Prospective Project of Jiangsu Province (grant No. BY2015027-12), the Natural Science Foundation of China, under contracts 61063021, 61272052 and 61473086, and the Program for New Century Excellent Talents of the University of Ministry of Education of China.

References

1. Chen, C., Jafari, R., Kehtarnavaz, N.: Improving human action recognition using fusion of depth camera and inertial sensors. IEEE Trans. Hum.-Mach. Syst. **45**(1), 51–61 (2015)
2. Chen, C., Liu, K., Jafari, R., Kehtarnavaz, N.: Home-based senior fitness test measurement system using collaborative inertial and depth sensors. In: EMBC, pp. 4135–4138 (2014)
3. Theodoridis, T., Agapitos, A., Hu, H., Lucas, S.M.: Ubiquitous robotics in physical human action recognition: a comparison between dynamic ANNs and GP. In: ICRA, pp. 3064–3069 (2008)
4. Chen, C., Kehtarnavaz, N., Jafari, R.: A medication adherence monitoring system for pill bottles based on a wearable inertial sensor. In: EMBC, pp. 4983–4986 (2014)
5. Yang, X., Tian, Y.: Super normal vector for activity recognition using depth sequences. In: CVPR, pp. 804–811 (2014)
6. Shotton, J., Fitzgibbon, A., Cook, M., Sharp, T., Finocchio, M., Moore, R., Blake, A.: Real-time human pose recognition in parts from single depth images. In: CVPR, pp. 1297–1304 (2011)
7. Li, W., Zhang, Z., Liu, Z.: Action recognition based on a bag of 3D points. In: CVPRW, pp. 9–14 (2010)
8. Yang, X., Zhang, C., Tian, Y.: Recognizing actions using depth motion maps-based histograms of oriented gradients. In: ACM Multimedia, pp. 1057–1060 (2012)
9. Xia, L., Aggarwal, J.K.: Spatio-temporal depth cuboid similarity feature for activity recognition using depth camera. In: CVPR, pp. 2834–2841 (2013)
10. Yang, X., Tian, Y.: Effective 3d action recognition using eigenjoints. J. Vis. Commun. Image Represent. **25**(1), 2–11 (2014)
11. Bobick, A.F., Davis, J.W.: The recognition of human movement using temporal templates. IEEE Trans. Pattern Anal. Mach. Intell. **23**(3), 257–267 (2001)
12. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR, pp. 886–893 (2005)
13. Chen, C., Liu, K., Kehtarnavaz, N.: Real-time human action recognition based on depth motion maps. J. Real-Time Image Process., 1–9 (2013). doi:10.1007/s11554-013-0370-1
14. Chen, C., Jafari, R., Kehtarnavaz, N.: Action recognition from depth sequences using depth motion maps-based local binary patterns. In: WACV, pp. 1092–1099 (2015)
15. Ojala, T., Pietikäinen, M., Mäenpää, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Trans. Pattern Anal. Mach. Intell. **24** (7), 971–987 (2002)
16. Vemulapalli, R., Arrate, F., Chellappa, R.: Human action recognition by representing 3D skeletons as points in a lie group. In: CVPR, pp. 588–595 (2014)

17. Kobayashi, T., Otsu, N.: Image feature extraction using gradient local auto-correlations. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part I. LNCS, vol. 5302, pp. 346–358. Springer, Heidelberg (2008)
18. Huang, G.B., Zhu, Q.Y., Siew, C.K.: Extreme learning machine: theory and applications. Neurocomputing **70**(1), 489–501 (2006)
19. Chen, C., Zhou, L., Guo, J., Li, W., Su, H., Guo, F.: Gabor-filtering-based completed local binary patterns for land-use scene classification. In: 2015 IEEE International Conference on Multimedia Big Data, pp. 324–329 (2015)
20. Li, W., Chen, C., Su, H., Du, Q.: Local binary patterns and extreme learning machine for hyperspectral imagery classification. IEEE Trans. Geosci. Remote Sens. **53**(7), 3681–3693 (2015)
21. Platt, J.: Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. Adv. Large Margin Classifiers **10**(3), 61–74 (1999)
22. Kurakin, A., Zhang, Z., Liu, Z.: A real time system for dynamic hand gesture recognition with a depth sensor. In: EUSIPCO, pp. 1975–1979 (2012)
23. Wang, J., Liu, Z., Chorowski, J., Chen, Z., Wu, Y.: Robust 3D action recognition with random occupancy patterns. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part II. LNCS, vol. 7573, pp. 872–885. Springer, Heidelberg (2012)
24. Vieira, A.W., Nascimento, E.R., Oliveira, G.L., Liu, Z., Campos, M.F.: STOP: space-time occupancy patterns for 3D action recognition from depth map sequences. In: Alvarez, L., Mejail, M., Gomez, L., Jacobo, J. (eds.) CIARP 2012. LNCS, vol. 7441, pp. 252–259. Springer, Heidelberg (2012)
25. Wang, J., Liu, Z., Wu, Y., Yuan, J.: Mining actionlet ensemble for action recognition with depth cameras. In: CVPR, pp. 1290–1297 (2012)
26. Oreifej, O., Liu, Z.: HON4D: Histogram of oriented 4D normals for activity recognition from depth sequences. In: CVPR, pp. 716–723 (2013)
27. Rahmani, H., Mahmood, A., Huynh, D.Q., Mian, A.: Real time action recognition using histograms of depth gradients and random decision forests. In: WACV, pp. 626–633 (2014)
28. Chang, C.C., Lin, C.J.: LIBSVM: a library for support vector machines. ACM Trans. Intell. Syst. Technol. **2**(3), 1–27 (2011)