Body Surface Context: A New Robust Feature for Action Recognition From Depth Videos

Yan Song, Jinhui Tang, *Member, IEEE,* Fan Liu, and Shuicheng Yan, *Senior Member, IEEE*

***Abstract*—Human action recognition in videos is useful for many applications. However, there still exist huge challenges in real applications due to the variations in the appearance, lighting condition and viewing angle, of the subjects. In this consideration, depth data have advantages over red, green, blue (RGB) data because of their spatial information about the distance between object and viewpoint. Unlike existing works, we utilize the 3-D point cloud, which contains points in the 3-D real-world coordinate system to represent the external surface of human body. Specifically, we propose a new robust feature, the body surface context (BSC), by describing the distribution of relative locations of the neighbors for a reference point in the point cloud in a compact and descriptive way. The BSC encodes the cylindrical angular of the difference vector based on the characteristics of human body, which increases the descriptiveness and discriminability of the feature. As the BSC is an approximate object-centered feature, it is robust to transformations including translations and rotations, which are very common in real applications. Furthermore, we propose three schemes to represent human actions based on the new feature, including the skeleton-based scheme, the random-reference-point scheme, and the spatial–temporal scheme. In addition, to evaluate the proposed feature, we construct a human action dataset by a depth camera. Experiments on three datasets demonstrate that the proposed feature outperforms RGB-based features and other existing depth-based features, which validates that the BSC feature is promising in the field of human action recognition.**

***Index Terms*—Depth video, feature, human action recognition, point cloud.**

I. Introduction

RECOGNIZING human actions in videos have attracted much attention in many research areas, like multimedia content analysis, computer vision, and pattern recognition, due to its wide applications, such as event detection in surveillance and human–computer interfaces. Although some existing works [1], [2], [17]–[20] have achieved satisfactory performance in research environments, there still exist huge challenge in applying the technology to real applications. As opposed to gait recognition, the aim of human action recognition is to generalize the variations for the same action category [2], including variations in the subjects' appearance, lighting condition, and viewing angle. Unfortunately, most existing works in this area [1], [2], [17]–[20] focus on red, green, blue (RGB) videos, in which these variations tend to bring in large intraclass variety. In addition, the imaging process of general RGB cameras is to project 3-D scenes onto a 2-D plane, during which important spatial information is lost. Although combining multiple cameras from different viewing angles [3]–[7] can remedy this defect, assumptions and high computation cost are needed, which make it inappropriate for real-time applications. Besides, background complexity also throws a very big challenge to the RGB-based methods.

Depth image contains spatial information which has an advantage over RGB image. Unfortunately, depth-image-capture devices [8]–[10] are seldom adopted in action recognition due to low speed, low resolution, inconvenience, and high cost. With the recent invention of the low-cost Microsoft Kinect sensor, high-resolution depth images can be provided with synchronized RGB images in real time. Meanwhile, publically available software development kits (SDKs) provide real-time skeleton trackers and body segmentation. These simplify the difficult issues in real-complicated situations, such as human detection and body extraction. However, many existing works simply extended the schemes used in RGB-based approaches to depth data [12], [13], which neglect the characteristics and advantages of depth information, or rely on the skeleton data [14]–[16] that became unreliable for postures with self occlusion (see details in Section II-B).

Based on the above observations, this paper focuses on utilizing the characteristics of depth information for the representation of human actions in videos, which is an important issue and can significantly influence the performance of human action recognition. Most exiting works extracted features directly from depth videos [12]–[16]. We argue that the 3-D surface of human body contains essential information about human actions. Therefore, unlike existing works, we utilize the 3-D point cloud, which contains points in the 3-D real-world coordinate system to represent the external surface of the human body. Another advantage is that the 3-D point cloud

Manuscript received June 26, 2013; revised September 25, 2013 and December 11, 2013; accepted January 7, 2014. Date of publication January 24, 2014; date of current version June 3, 2014. This work was supported in part by the 973 Program under Grant 2014CB347600; in part by the National Nature Science Foundation of China under Grants 61202133, 61103059, 61173014, and 61301106; in part by the Program for New Century Excellent Talents in University under Grant NCET-12-0623; and in part by the Research Fund for the Doctoral Program of Higher Education of China under Grants 20123219120027, 20113219120022, and 20123219120024. This paper was recommended by Associate Editor R. Hamzaoui.

Y. Song, J. Tang, and F. Liu are with the School of Computer Science and Technology, Nanjing University of Science and Technology, Nanjing 210094, China (e-mail: songyan@mail.njust.edu.cn; jinhuitang@mail.njust.edu.cn; fanliu.njust@gmail.com).

S. Yan is with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore 117576 (e-mail: eleyans@nus.edu.sg).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2014.2302558

of human body surface can avoid the perspective distortion in depth images. Specifically, we propose a new basic feature named the body surface context (BSC) for describing the surface of human body. Then, we further explore the use of this new feature for human action representation by several schemes both statically and spatial–temporally.

The BSC feature describes the distribution of relative locations of the neighbors for a reference point in the point cloud of human body surface. We observe that the states of human bodies are restricted for the same action under normal conditions. Thus, the cylindrical angular of the difference vector is more important for the description of human body compared with other free-form objects. Therefore, the BSC encodes the cylindrical angular which increases its descriptiveness and discriminability. However, the cylindrical angular cannot be defined unambiguously here. We utilize the characteristics of human body to simplify this issue, and it also makes the feature very suitable for human action description. Consequently, the feature is approximately object-centered, which makes it robust to variations including translations and rotations which are very common in real situations. Then, three different schemes for human action representation and comparison based on the new feature are investigated: 1) the skeleton-based scheme; 2) the random-reference-point scheme; and 3) the spatial–temporal scheme. To verify the efficiency of the proposed method, we also construct a dataset named the NJUST RGB-D action dataset including RGB data, depth data, skeleton data, and body segmentation map.

The main contributions include: 1) we propose a new robust feature for describing human body surface based on the point cloud rendered from depth data; 2) we investigate three schemes of human action representation and comparison based on the new feature; and 3) we provide a new human action dataset which will be a benchmark for action recognition based on depth data.

The rest of the paper is organized as follows. We first review the related work in Section II and elaborate on the details of the proposed feature in Section III. Then, we present three schemes for action representation and comparison in Section IV. Experimental results are described in Section V. Finally, we conclude this paper along with discussion of future work in Section VI.

II. Related Work

In this section, we will review related works from four aspects including action recognition based on RGB and RGB-D data, respectively, features for 3-D surface shapes and RGB-D human action datasets.

A. Human Action Recognition Based on RGB Data

Human action recognition has been a popular research topic for years. It includes two main issues: 1) the representation of actions and 2) the classification algorithm. As this paper focuses on action representation, we mainly address the first issue. Earlier works mainly adopted holistic features based on silhouette or shape [21], and human body models [22]. However, these methods were highly dependent on the performance of segmentation and tracking, which may not deliver satisfactory results in real situations due to possible occlusions and cluttered backgrounds. Inspired by the success of local features, such as SIFT [23], in object recognition, local spatial–temporal features (LSTF) for action recognition have attracted lots of attention, since they are more robust to occlusions than holistic features. The most popular LSTFs include cuboids feature proposed by Dollar *et al.* [19], as well as HoF and HoG proposed by Laptev *et al.* [24]. To remedy the loss of location information of LSTFs, many works also encoded structure information in action representation [25]. Besides, shape and motion information were usually adopted in action representation [42]. However, as RGB data only contain appearance information, these methods still have difficulties in representing human bodies especially in complex and cluttered scenes.

B. Joints Localization, Gesture Recognition and Action Recognition Based on RGB-Depth Data

Many researchers adopted depth images or RGB-D data for joints localization, gesture recognition, and activity analysis. Earlier works adopted range images rendered from devices such as the time of flight (ToF) cameras [9]. With the launch of the Kinect sensor, many researchers tend to adopt it for RGB-D data rendering.

Depth images provide important cues for joints localization and gesture recognition. Shotton *et al.* [26] proposed to predict 3-D positions of body joints from a single depth image by employing comparison features and mapping pose estimation problem to per-pixel classification problem. It formed a core component of the Kinect platform. Hernandez-Vela *et al.* [27] used random forest and graph cuts for segmenting human limbs while Charles and Everingham [28] learned a model of limb shape to estimate human pose. For gesture recognition, joint location-based features were the most popular. Reyes *et al.* [29] adopted the relative coordinates of joints as a feature and Raptis *et al.* [30] used an angular representation of the skeleton for dance gesture classification.

For human action recognition, there are two kinds of schemes to utilize depth information. One is to extend the schemes used in RGB-based methods to fuse the information from RGB and depth channels. For example, Zhang and Parker [12] extended the cuboids feature to a 4-D LSTF by adding depth information. Ni *et al.* [13] introduced a depth-layered multichannel spatial–temporal feature and 3-D motion history images for fusing color and depth information straightforwardly. The methods designed for RGB data cannot utilize the characteristics and advantages of depth information. The other one is to extract new features from depth images which can be further divided into two types: skeleton-based features and depth-image-based features. Yang and Tian [14] proposed a feature called EigenJoints based on position differences of joints. Sung *et al.* [15] computed features based on 3-D Euclidean coordinates and the orientation matrix of each joint. Similarly, Wang *et al.* [16] proposed invariant features based on 3-D joint position. They also computed local occupancy information based on the local 3-D point cloud around a joint. These skeleton-based features depended

highly on the skeleton localization which may be unreliable for postures with self-occlusion. Depth-image-based features are extracted directly from depth images. Xia and Aggarwal [44] extracted STIPs from depth videos and built a depth cuboid similarity feature. Jalal *et al.* [43] obtained invariant features via R transformation on depth contours. Li *et al.* [31] sampled a specified number of points along the contours of the projections to form a bag of 3-D points as representation. However, their bag of 3-D points is different from our feature. First, they only used depth images, while we restore 3-D point clouds which contain points in the 3-D real-world coordinate system. Second, they used the locations of the sampled points while we use the context of body surfaces corresponding to reference points. Besides, the bag of 3-D points feature is not invariant to rotations.

C. Features for 3-D Surface Shapes

There are a huge body of researches in 3-D surface representation for 3-D object recognition and retrieval. Surface shapes are represented by dense point clouds rendered by 3-D sensors. Spin image [32] describes a surface by utilizing a descriptive image to represent each surface point which encodes global properties of the surface using an object-centered coordinate system. However, it is designed for general free-form objects. The invariance against many transformations actually leads to the decrease of discriminability. Surfaces were represented by describing different aspects of local shapes precisely by histograms, such as pixel depth, surface normal and curvature [33], [34]. These descriptors are inappropriate for our data, since the Kinect sensor is not a precise 3-D sensor and the depth images are noisy. 2-D shape descriptors, such as the shape context [35], were also extended to 3-D domain [36], [37], [41]. Kokkinos *et al.* [36] charted the surface by shooting geodesic based on a triangular mesh to generalize the polar sampling of the image domain to surfaces. Unfortunately, the triangulation process may bring in noises to our data. By stacking body contours along the time axis [37], [38], surface descriptors were adopted in human action recognition. Frome *et al.* [41] extended the shape context to 3-D by segmenting the 3-D spherical support region into bins in a different way from ours. It should be noted that there is a difference in handling the freedom in the azimuth direction. As they proposed the feature for a retrieval task, they chose some direction as an initial direction and then rotated it about its north pole into L positions to get L features. For the query data, only one position was included. Obviously, this is infeasible for our recognition task. Therefor, we choose the vector vertical to the horizontal plane as the reference vector to simplify this issue. This makes our feature special for action recognition. In addition, their work did not include action representation steps. Therefore, our feature is more appropriate for action recognition task in depth videos.

D. Kinect Human Action Dataset

The MSR-Action3D dataset [31] was comprised of 20 simple and basic actions captured by a depth sensor similar to the Kinect device. It only contained depth images and skeleton data. The Cornell Activity Dataset [15] comprised of 12 daily-life actions in five indoor environments. It contained RGB-D data and skeleton data. RGBD-HuDaAct [13] aimed for the application of assisted living in health care. It included 12 human daily activities. Only RGB-D data were provided. The MSRDailyActivity3D Dataset [16] was a daily activity dataset captured by a Kinect device. It provided RGB-D data, skeleton data and body segmentation map, which is similar to ours. However, most of their categories do not overlap with ours. Different from the existing datasets, our dataset aims for the applications of surveillance in public places. Besides, our dataset contains data for view variation evaluation.

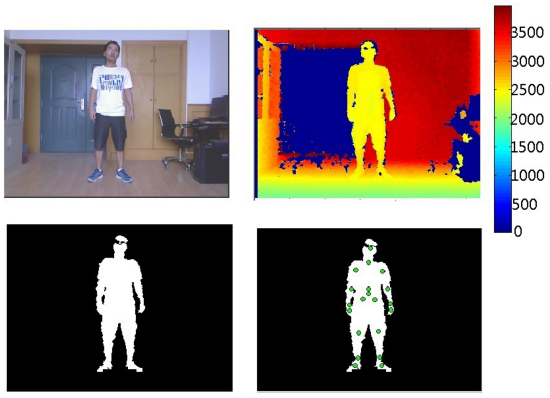


Fig. 1. Data output of the Kinect device. Top left: RGB frame. Top right: depth frame. Bottom left: human body segmentation map. Bottom right: 20 skeleton joints on the body segmentation map.

III. Body Surface Context

A. Point Cloud of Human Body Surface

The latest SDK for Kinect provides RGB and depth data. Each pixel in the depth image contains the distance, in millimeters, from the camera plane to the nearest object. The skeleton data are generated by processing the depth data, which contain 3-D coordinates expressed in meters for 20 human skeleton joints. They can be converted to depth image coordinates. Besides, it can identify up to six human bodies in a segmentation map by processing the RGB and depth data. Fig. 1 shows one frame of RGB data, depth data, human body segmentation map, and skeleton data on body segmentation map, respectively.

Although we can extract features based on the skeleton or body contour, there are some defects and unreliable factors. The skeleton information is essentially the 20 points in the 3-D space, which can represent the sketchy pose but lack the details. More importantly, skeleton data become unreliable when self-occlusion occurs or the subject moves fast, as shown in Fig. 2. For features based on body contours, the details inside the contours are lost.

On the contrary, the surface of human body contains rich information which characterizes human actions. Thus, we use the point cloud computed by the depth image and the segmentation map to represent the surface of human body. A point cloud is a set of points in a 3-D coordinate system that typically represents the external surface of an object. A pixel

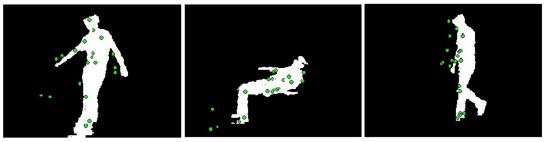


Fig. 2. Examples of skeleton tracking errors.

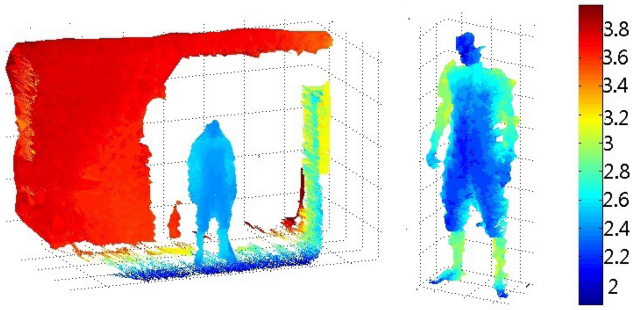


Fig. 3. Left: triangular mesh of the point cloud of the scene in the frame shown in Fig. 1. Right: triangular mesh of the point cloud of the human body surface.

$(x_d,\ y_d)$ in the depth image can be converted to the 3-D real-world coordinates $(x_r,\ y_r,\ z_r)$, whose unit is meter, by estimating some calibration parameters [11]. Then, we adopt the empirical converting equations used in [39]

$$\begin{aligned} x_r &= \frac{(x_d - h/2) \times \mathrm{DepImg}(x_d, y_d)}{c \times h/480 \times \mathrm{MM\_PER\_M}} \\ y_r &= \frac{(y_d - w/2) \times \mathrm{DepImg}(x_d, y_d)}{c \times w/640 \times \mathrm{MM\_PER\_M}} \\ z_r &= \frac{\mathrm{DepImg}(x_d, y_d)}{\mathrm{MM\_PER\_M}} \end{aligned} \tag{1}$$

where w and h are the width and the height of the depth image, c is a constant which is equal to 570.3, and MM_PER_M is 1000. $DepImg(x_d,\ y_d)$ is the pixel value of $(x_d,\ y_d)$ in the depth image. As shown in Fig. 3, the left is the triangular mesh of the point cloud of the whole scene in the frame shown in Fig. 1 and the right is the triangular mesh of the point cloud of the human body. The triangulation here is only for visualization clarity and is not used in the following algorithm.

B. BSC

It is a very effective approach to represent a shape or a surface by describing the context structures for reference points [33], [34]. Thus, we adopt the scheme for feature generation. Suppose that a reference point in the point cloud of a human body surface is denoted by P_o=$(x_o,\ y_o,\ z_o)$ and a target point is denoted by P_t=$(x_t,\ y_t,\ z_t)$. The relative position of the target point to the reference point is represented by the difference vector from the latter to the former

$$V_d = (\delta x, \delta y, \delta z) = (x_t - x_o, y_t - y_o, z_t - z_o). \tag{2}$$

Since all views of the surface can be represented by a single feature generated in an object-centered coordinate system, we aim to represent the difference vector V_d in an object-centered coordinate system rather than a view-dependent coordinate system. One simple way to generate a local object-centered coordinate system is to use an oriented point, i.e., a point with surface normal. Thus, we firstly adopt spin image [32] for a preliminary descriptor.

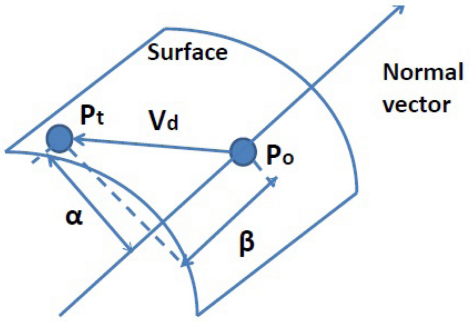


Fig. 4. Parameter α and β for the relative position of a target point P_t to an oriented point P_o.

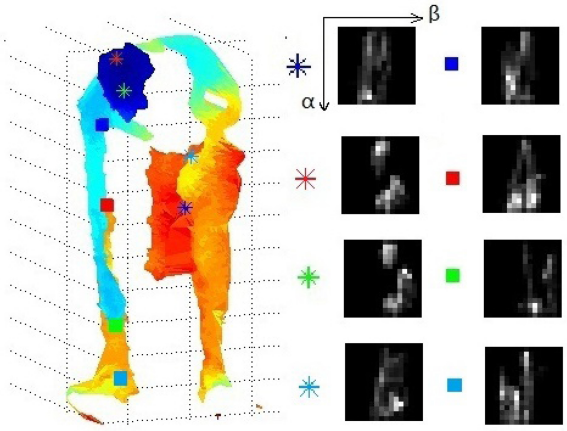


Fig. 5. 2-D histogram features for eight reference points on a bending human body surface.

Spin image [32] is an invariant 2-D image by using two parameters to describe the relative positions to the reference point. Given an oriented point P_o, a cylindrical coordinate system can be decided. The difference vector V_d from a target point P_t to P_o can be partially represented by two parameters: the radial coordinates α and the elevation coordinates β. As shown in Fig. 4, the radial coordinates α is the distance from P_t to the normal vector of P_o which is always a positive value. The elevation coordinates β is the signed distance from P_t to the tangential plane of P_o. A 2-D histogram indexed by normalized α and β is created by accumulating the relative locations of the neighborhood for P_o defined as

$$\{P | d(P_o, P) < \sqrt{2} \times d_n\} \tag{3}$$

where $d(P_o,\ P)$ is the Euclidean distance between P_o and P, and d_n is a threshold to define the neighborhood of P_o. The numbers of bins of the histogram are denoted by H_w and H_h. Fig. 5 shows a surface of a bending human body. 2-D histogram features for eight reference points are shown

on the right. H_w and H_h are both set to 16, and d_n is set to 0.4 here.

By introducing the spin image to human body representation, we can describe a local human body surface context based on a reference point by a 2-D histogram. We notice that only two parameters in the object-centered cylindrical coordinate system are considered. This leads to the invariance to rotation transformation around the normal vector which is useful in free-form objects retrieval and recognition. However, this issue should be reconsidered in our case from two aspects. First, the omission of the third parameter, the cylindrical angular, leads to the decrease of descriptiveness. Second, the invariance to the rotation transformation around the normal vector is unnecessary in our case. Under normal conditions, the states of human bodies are restricted. For example, an action is seldom executed upside down. Therefore, we extend the preliminary descriptor to the BSC feature based on the above considerations. Specifically, we take the third parameter, the cylindrical angular, into consideration. Unfortunately, it cannot be defined unambiguously, which is also the reason why the spin image omits it. In other words, the starting orientation of the cylindrical angular cannot be determined in the local cylindrical coordinate system. In our case, this problem can be solved based on the characteristics of human bodies. As mentioned above, different from free-form objects, the state of human body cannot be arbitrary for the same action under normal conditions. Therefore, the vector vertical to the horizontal plane can be a reasonable reference vector for human bodies.

Then, the cylindrical angular is decided as follows. As shown in Fig. 6(a), the reference vector, i.e., the vertical vector, is denoted by V_r. It is the vector of (0, 1, 0) in the real world coordinate system. The projection of V_d on the tangential plane of P_o is denoted by V_{pt}. Then, the cylindrical angular can be defined by the phase of V_{pt}. To achieve this, we choose V_{pr} as the starting orientation, which is the projection of V_r on the tangential plane. Then, the cylindrical angular is decided as the rotation angle θ_r from V_{pr} to V_{pt}. For convenience, we compute another equivalent rotation angle $\theta_{r'}$ instead, which is the rotation angle from the vector $V_{pr'}$ to the vector $V_{pt'}$, as shown in Fig. 6(b). Let V_N denote the normal vector of P_o. $V_{pr'}$ is the cross product of V_r and V_N. Similarly, $V_{pt'}$ is the cross product of V_d and V_N

$$\begin{aligned} V_{pr'} &= V_r \times V_N \\ V_{pt'} &= V_d \times V_N. \end{aligned} \tag{4}$$

Obviously, $V_{pr'}$ is perpendicular to the plane decided by V_r and V_N. Thus, it is perpendicular to V_{pr} since V_{pr} is on that plane. Similarly, $V_{pt'}$ is perpendicular to V_{pt}. $V_{pr'}$ and $V_{pt'}$ are both on the tangential plane. Therefore, $\theta_{r'}$ is equal to θ_r.

However, we cannot compute directly the rotation angle between two 3-D vectors. As $V_{pr'}$ and $V_{pt'}$ are both on the tangential plane, we convert the computation of 3-D rotation angle to the computation of 2-D rotation angle by coordinate rotation. Specifically, we compute the coordinates of the two vectors in the coordinate system where V_N is the z-axis and $V_{pr'}$ is the y-axis as shown in Fig. 7. According to the

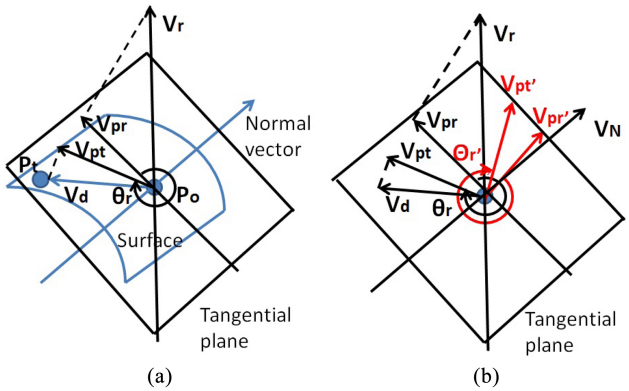


Fig. 6. Computation process of the cylindrical angular. (a) Rotation angle θ_r. (b) Rotation angle $\theta_{r'}$.

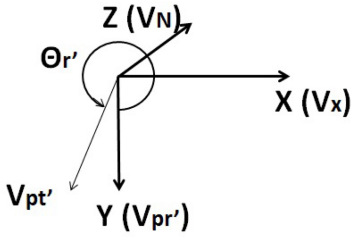


Fig. 7. New coordinate system.

coordinate system rotation in computer vision, the rotation matrix is

$$R = (V_x \quad V_{pr'} \quad V_N) = (V_{pr'} \times V_N \quad V_{pr'} \quad V_N) \tag{5}$$

where V_x is the cross product of $V_{pr'}$ and V_N. Then, the coordinates of $V_{pr'}$ and $V_{pt'}$ in the new coordinate system, denoted as $V^n_{pr'}$ and $V^n_{pt'}$, are computed as

$$\begin{aligned} V^n_{pr'} &= R^{-1} \times V_{pr'} \\ V^n_{pt'} &= R^{-1} \times V_{pt'}. \end{aligned} \tag{6}$$

Now, the two vectors $V^n_{pr'}$ and $V^n_{pt'}$ are on the same plane where z=0. To decide whether a rotation angle between two vectors is in [0, π] or [π, 2π], without loss of generality, we first define the anticlockwise around the normal vector as the positive rotation direction. The range of the rotation angle between two 2-D vectors can be decided by the sign of the cross product of them. Thus, the rotation angle $\theta_{r'}$ is computed as

$$\theta_{r'} = \begin{cases} \arccos \dfrac{V^n_{pr'} \cdot V^n_{pt'}}{|V^n_{pr'}||V^n_{pt'}|} & V^n_{pr'} \times V^n_{pt'} < 0 \\ 2\pi - \arccos \dfrac{V^n_{pr'} \cdot V^n_{pt'}}{|V^n_{pr'}||V^n_{pt'}|} & V^n_{pr'} \times V^n_{pt'} \geq 0. \end{cases} \tag{7}$$

Assuming that $V^n_{pr'}$ and $V^n_{pt'}$ are $(x_1\mathbf{i}, y_1\mathbf{j})$ and $(x_2\mathbf{i}, y_2\mathbf{j})$, respectively, then

$$V^n_{pr'} \times V^n_{pt'} = (x_1 y_2 - y_1 x_2)\mathbf{k} \tag{8}$$

where $\mathbf{i}$, $\mathbf{j}$, $\mathbf{k}$ are the unit vectors of the x-, y-, and z-axes of the new coordinate system, respectively. The sign of the cross product is actually the sign of (x_1y_2-y_1x_2).

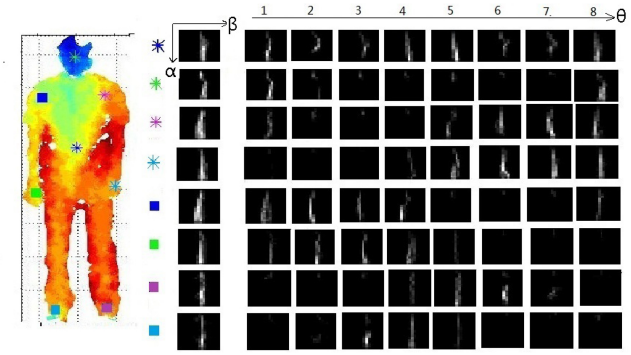


Fig. 8. 2-D histograms expanded along the third dimension for the BSC features of eight reference points.

Similar as the other two parameters, the cylindrical angular is quantized to H_a bins. The BSC feature for a reference point is a 3-D histogram. Fig. 8 shows the 2-D histograms expanded along the third dimension for the BSC features of eight reference points. Here, H_a is set to eight. Eight skeleton joints labeled by different markers on a standing human body surface are shown on the left. The first column of histograms is the spin image. The eight columns on its right are the eight 2-D histograms of the BSC feature. Actually, the sum of the eight columns is equal to the first column. Although the eight points locate at different salient positions of the human body, we observe that the spin images are quite similar. This is due to the discriminability deficiency of the spin image. Comparatively, the BSC is able to provide abundant information to discriminate these reference points.

C. Discussion

As the vertical vector is involved in the feature generation, the BSC is not an absolute object-centered feature. However, it does not decrease the invariance to the rotation transformations. The reference vector has definite physical meaning and is invariant to action types. Due to the particularity of human actions, the relative angular of human body to the vertical vector does not vary largely for the same action under normal conditions. We can approximately consider the vertical vector as intrinsic. Thus, our feature is an approximately object-centered feature. In practice, when the camera is set upright it is equal to the y-axis of the camera coordinate system. When the x-rotation or the z-rotation occurs, the vertical vector can be decided by finding the ground first, which is a feasible task for depth videos. Consequently, the BSC feature is invariant to translation and rotation transformations.

IV. Human Action Representation and Comparison

The BSC feature is a compact representation for a surface corresponding to a reference point. In this section, we explore three schemes to incorporate the new feature into a metric for action comparison. First, as the skeleton contains important information of poses, it is natural to choose skeleton joints as reference points. Second, as mentioned above, since skeleton data may be unreliable, we also sample random points on body surfaces as reference points. Finally, an action is considered in a spatial–temporal way. Spatial–temporal interest points are detected as reference points, and then the BSC feature is extended to the temporal domain.

For Scheme 1 and 2, actions are considered as sequences of postures. For both the schemes, first, a set of key postures is selected to represent each action/video by a clustering algorithm with a distance metric defined based on the BSC features of skeleton joints. In this paper, the concept of key postures for actions is similar to the concept of key frames for videos. Then, Scheme 1 defines action distance by comparing the BSC features of skeleton joints in key postures, while Scheme 2 randomly samples a few reference points from the key postures and then adopts the bag-of-words (BOW) model to represent an action/video. For Scheme 3, actions are considered as spatial–temporal cubes and reference points are generated like spatial–temporal interest points. The BOW model is also adopted to represent an action/video. For action recognition, each action representation scheme is tested individually. We adopt the k-Nearest Neighbor (kNN) algorithm and the support vector machine (SVM) as the multiclass classifiers. Details will be presented in the experiments.

A. Scheme 1: Skeleton-Based Scheme

We have 20 skeleton joints provided by the Kinect SDK for a human body. The 20 BSC features of these skeleton joints are denoted by $(bsc^1, bsc^2, \ldots, bsc^{20})$. The distance between two postures P_1 and P_2 is defined as the average χ^2 distance of their 20 BSC features

$$D_{\mathrm{P}}(P_1, P_2) = \frac{1}{20}\sum_{i=1}^{20} Dist_{\chi^2}(bsc_1^i, bsc_2^i). \tag{9}$$

For each action, N key postures are selected by the clustering algorithm. Suppose two actions A_1 and A_2 are represented by key posture sets: $\{P_i^1 |\ i{=}1, 2, \ldots, N_1\}$ and $\{P_i^2 |\ i{=}1, 2, \ldots, N_2\}$. Then, the distance between the two actions is defined by

$$D_A(A_1, A_2) = \frac{\sum_{i=1}^{N_1} min\{D_{\mathrm{P}}(P_i^1, P_{i'}^2)|i'\}}{N_1 + N_2} + \frac{\sum_{j=1}^{N_2} min\{D_P(P_j^2, P_{j'}^1)|j'\}}{N_1 + N_2}. \tag{10}$$

B. Scheme 2: Random-Reference-Point-Based Scheme

Although skeletons have semantic meanings, they are unavailable for some postures. Here, we aim to sample an appropriate number of points to cover the body. For the purpose of sampling points fairly regardless of their postions, we design a random sampling scheme. As the result of a random sampling, the points may be close to each other. To avoid the redundancy, we remove the points that are close to others. To sample enough points, we repeat this procedure for several times. During each sampling, the points close to the previously sampled points are also removed. The procedure is summarized in Table I. S_p is the point cloud of human body. S_r is the final sampled point set. S_s is the sampled

TABLE I
ALGORITHM 1

Random sampling for reference points

Input: S_p ={$P_i|i$=1,2,...,N_p}, N_s, N_i, d_t
Output: S_r={$P_j|j$=1,2,...,N_r}
1.S_r={}, S_s={}
2.**for** i=1 to N_i
  a. Randomly sample N_s points from S_p to S_s
  b. Remove the self-redundancy
    1) d_0=0, compute the distance matrix D_1 of S_s
    2) **While** $d_0 < d_t$
      i. Find the minimum element d_1 of D_1,
      d_0=d_1, x=the column index of d_1
      ii. Remove the x^{th} column and the x^{th} row
      from D_1, remove the x^{th} point from S_s
    end
  c. Remove the redundancy considering the previous samplings
    1) Compute the distance matrix D_2 from S_s to S_r
    2) Find the elements of D_2 which are smaller than d_t
    3) Remove these elements from S_s
  d. S_r=$S_r \cup S_s$, S_s={}
end

point set in the current iteration. N_p is the number of all the points. N_r is the number of sampled points. N_s is the number of sampled points in the current iteration. N_i is the iteration time. d_t is the threshold to control the redundancy. After the random sampling procedure, we have a few reference points for each posture. For each action, we have N key postures as mentioned in the previous subsection. To make a richer sampling, we sample points from M most similar postures of each key posture based on the distance defined in (9). Finally, an action is represented by the BSC features of random reference points sampled from $N \times M$ postures, and then the BOW model is adopted.

C. Scheme 3: Spatial–Temporal Scheme

In recent years, it has become popular to consider a video as a 3-D spatial–temporal cube instead of a sequence of frames in human action recognition. Local spatial–temporal features are adopted in human action recognition [19], [24] by extending interest point-based features to the 3-D space. Here, we follow this scheme to extend our feature to the spatial–temporal domain.

First, spatial–temporal interest points are detected at the most informative locations by the algorithm proposed by Dollar *et al.* [19]. Different from the original method, we apply the interest point detection on depth videos instead of RGB videos. The response function is $R = [I \times g_\sigma \times h_{ev}]^2 + [I \times g_\sigma \times h_{od}]^2$, where g_σ is a Gaussian filter applied on the spatial dimensions, and h_{ev} and h_{od} are a quadrature pair of Gabor filters applied on the temporal dimension. Then, these interest points are considered as reference points for computing spatial–temporal BSC features. For each reference point, we consider not only the surface in the current frame, but also the frames in the temporal neighborhood, as shown in Fig. 9. The bold red

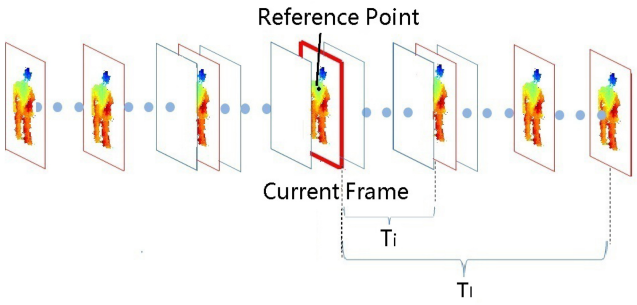


Fig. 9. Temporal neighborhood of a reference point.

parallelogram is the current frame with the reference point. The red parallelograms are frames in the temporal neighborhood which are considered in the feature generation. T_i is the interval between the current frame and the adjacent considered frame. T_l is the length of the temporal neighborhood which is a multiple of T_i. In total, $(2 \times T_l / T_i + 1)$ surfaces are considered for a reference point. The BSC features generated from these frames are concatenated to form the spatial–temporal BSC feature. This process is shown in Fig. 10. As shown in Fig. 10(a), the middle posture is the current frame and P is the detected reference point, whose coordinate in the point cloud is (x, y, z). The left and the right are the past and the future neighboring frames, respectively. P in the past and future postures are both at the same location (x, y, z). The regions in the green circles are the spatial neighborhoods of P. As we only consider human body surface, only the yellow regions are considered in the BSCs generation. Note that the reference point is fixed at location (x, y, z) in the considered frames, no matter it is on the body or not. Fig. 10(b) shows the point clouds of the three postures. In the right image, the black region in the green circle is the considered body surface in the BSC generation of the past posture, the gray one is considered for the current posture, and the purple one is considered for the future posture. Then, the BSCs are concatenated to make the final spatial–temporal BSC. The rationale is to encode changes of body surface in the spatial neighborhood for a fixed point, which contains important motion information. Therefore, the feature dimension is $H_w \times H_h \times H_a \times (2 \times T_l / T_i + 1)$. Note that we do not track the reference point in the temporal neighbors, but use the original reference point to compute all the BSC features. Thus, the reference point is possibly not on the body surface in neighboring frames. However, the relative displacement of the neighboring surfaces from the reference point actually contains important motion information which is encoded in the spatial–temporal BSC feature. For computation efficiency, primary component analysis (PCA) is adopted to reduce the feature dimension. Similar to Scheme 2, the BOW model is adopted for action representation.

V. EXPERIMENTAL RESULT

In this section, we evaluate the effectiveness of the new feature. We first explore the impact of a parameter in the feature generation. Second, we verify the three action representation and comparison schemes. Third, we test the robustness of the

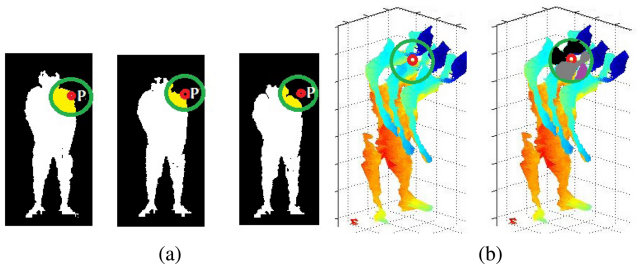


Fig. 10. Generation of the spatial–temporal BSC. (a) A reference point and its spatial neighborhoods in three successive considered frames. (b) Left: A reference point and its spatial neighborhood in the point clouds of three successive postures. Right: Surface regions of these postures that are considered in the spatial-temporal BSC feature extraction.

Fig. 11. Samples of RGB frame in our Kinect human action dataset. Top row: *Bending*, *Bending-side*, *Boxing*, *CheckingTime*, and *Dringking*. Middle row: *Kicking*, *LyingDown*, *OpeningCloset*, *Squatting*, and *SittingDown*. Bottom row: *DroppingBag*, *TakingPhoto*, *Drinking-30D*, *Squatting-30D*, and *SittingDown-30D*.

feature against view variation. Finally, the feature is compared with other state-of-the-art features for action recognition. Considering that our feature is based on the surface of human body, body segmentation map is indispensable. Thus, we construct a new action dataset by the Kinect camera. We also test our methods on two public datasets including the MSR Action3D Dataset [31] and the MSRDailyActivity3D Dataset [16].

A. Experiments on Our Kinect Human Action Dataset

1) *Dataset:* To evaluate our method, we construct a human action dataset by the Kinect device. The videos are collected in lab environments. The distance from the camera to the subject is about three meters. It contains 19 action categories: *Bending*, *Bending-side*, *Boxing*, *CheckingTime*, *Drinking*, *DroppingBag*, *Kicking*, *LyingDown*, *OpeningCloset*, *PickingUp*, *PullingOut*, *SittingDown*, *Squatting*, *StandingUp*, *TakingPhoto*, *Telephoning*, *Tossing*, *Walking*, and *Waving*. Besides, for view variation evaluation, six actions are performed with view variation of 30 degree: *Bending-30D*, *Boxing-30D*, *Drinkin-30D*, *SittingDown-30D*, *Squatting-30D* and *StandingUp-30D*. Each action is performed by ten subjects in two scenes. Altogether, there are 500 action samples and about 42 700 frames. For each sample, RGB frames, depth frames, skeleton data, and body segmentation are provided. The frame rate is 30 frames per second and the resolution is 320×240 pixels. Several samples are shown in Fig. 11. We have made the data available at: http://137.132.145.151/public/other/NJUST_RGBD_ActionDataset.zip

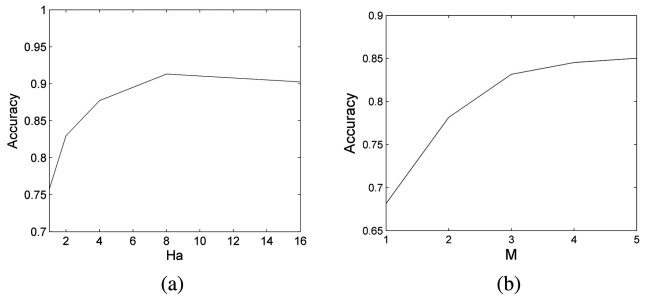


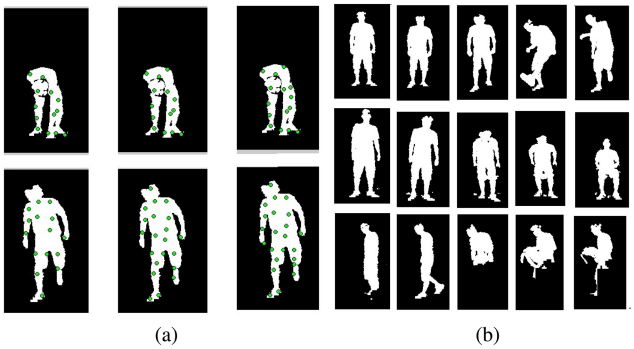
Fig. 12. (a) Accuracy against variation of H_a. (b) Accuracy against variation of M.

2) *Experimental setting and evaluation criterion:* In the rest of the paper, S1 denoted Scheme 1 and so on. For the BSC generation, d_n is set to 0.4. H_w and H_h are both set to 16 in S1 and S2. In S2, N_s is set to 50, N_i is set to 3 and d_t is set to 0.02. For key postures selection, K-means clustering is adopted and N and M are both set to 5. In the interest point detection, the spatial and temporal scales are set to 2 and 3, respectively. T_i is set to 2 and T_l is set to 4. For spatial–temporal BSC, H_w and H_h are both set to 8 and feature dimension is reduced to 300 by PCA. For the BoW scheme, the dictionary size is set to 500.

We adopt the kNN algorithm and extend SVM in a one-versus-all setting as the multiclass classifiers. k is set to 10 for kNN and the regularization parameter for SVM is set to 100. We adopt the leave-one-subject-out (LOSO) scheme for evaluation. For each run, action samples of one subject are chosen as testing data, and all the remaining samples of other subjects are used as training data. We repeat this process by permuting the test actor. Then, the overall performance is the average of all the runs. Recognition accuracy and confusion matrix are used as the evaluation criterion.

3) *Impact of the parameters:* Here, kNN is adopted as the classifier. First, we explore the impact of the parameter H_a, i.e., the bin number of the rotation angle θ_r. Here, we adopt S3 for action representation. Fig. 12(a) shows the accuracy against the variation of H_a. It is observed that the performance increases as H_a increases when it is smaller than 8, and then the performance decreases a little. We believe that increasing the bin number will increase the descriptiveness of the BSC, but too many bins will also decrease its generalization ability. Thus H_a is set to 8 in the following experiments. Then, we also explore the impact of the parameter M, i.e., the number of similar postures for each key postures in S2. Fig. 12(b) shows the accuracy against the variation of M. The performance increases as M increases from 1 to 5, but the rate decrease when it approaches to 5. This verifies that sampling points in similar postures will bring in additional information. M is set to 5 in the following experiments.

4) *Comparison of three action representation schemes:* kNN is adopted as the classifier. Two examples of random reference points are shown in Fig. 13(a). The first column is the result of the first sampling and so on. The average number of reference points after the first sampling of all frames in the dataset is 15.29. The average number after the second and



(a) (b)

Fig. 13. (a) Random reference points after three times of selection. (b) Key postures for actions of *Kicking*, *SittingDown*, and *PickingUp*.

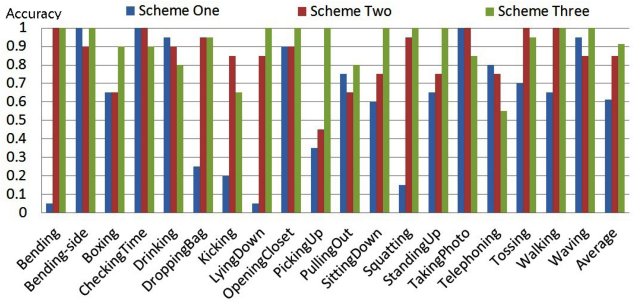


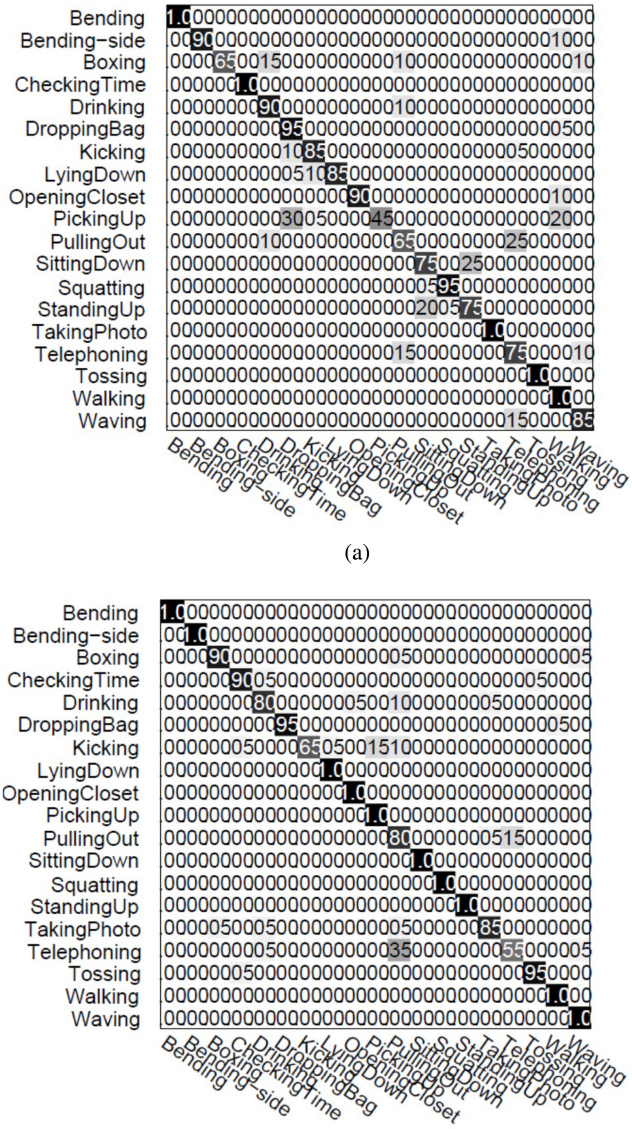
Fig. 14. Recognition accuracies of three representation schemes.

the third sampling are 18.92 and 21.05 respectively. Fig. 13(b) shows examples of key postures chosen by K-means clustering for *Kicking*, *SittingDown* and *PickingUp*.

The comparison result is shown in Fig. 14. For 13 categories out of 19, S3 performs the best. Overall, S3 achieves an average accuracy of 0.9132, which outperforms S1 by 0.3 and outperforms S2 by 0.063. It turns out that S2 outperforms S1. It demonstrates that the skeleton is not reliable in this dataset. There may be errors in skeleton joint localization because of gesture deformation, such as *Bending*, *LyingDown*, and *Squatting*. Also, occlusions can bring in errors, such as in *DroppingBag* and *PickingUp*.

The confusion matrices of S2 and S3 are shown in Fig. 15. We observe that S3 performs better for action pairs such as *SittingDown* and *StandingUp*, *PickingUp* and *DroppingBag*, which are temporally inverse. This is because the key postures of these action pairs, used in S2, are similar to each other. Compared with S2, S3 contains temporal information which is very important for differentiating these action pairs, as proved by the experimental result.

5) *Comparison with other features:* In this section, we compare our feature with several existing features including RGB features and depth features. For RGB features, we compare two popular LSTFs with our feature, including the cuboids feature [19] and the HogHof [45]. For depth features, we compare the depth-layered multichannel STIPs (DLMC-STIPs) [13], the bag of 3-D points [31] and the 3-D shape context [41] with our feature. The spin image feature [32] is compared with the BSC feature to testify the effectiveness of the cylindrical angular. The DLMC-STIPs is generated



(a)

(b)

Fig. 15. Confusion matrices of (a) S2 and (b) S3.

by dividing the spatio-temporal interest points into several depth-layered channels, and then pooling independently the STIPs within different channels. For the two RGB LSTFs and the DLMC-STIP, the BoW scheme is adopted for action comparison and the codebook size is set to 512. For the DLMC-STIPs, spatial pyramid matching kernel is used for representations with $l = 3$ depth spatial levels as in [13]. In [31], action graph was used to model the dynamics of human motion. Here, we only focus on their bag-of-3-D-points feature, which is a specified number of sampled points along the contours of the projections. Since it is a posture-level representation, we combine it with S1 for action representation and comparison. But the non-Euclidean relational fuzzy C-means is adopted for key postures generation and Hausdorf distance is adopted as distance metric for postures as in [31]. The 3-D shape context segments the 3-D support region into bins by extending the shape context. As the method for the initial

TABLE II
COMPARISON OF BSC FEATURE WITH SEVERAL RGB AND DEPTH FEATURES ON OUR DATASET

Feature	Accuracy(kNN)	Accuracy(SVM)
Cuboids Feature [19]	0.6658	0.7474
HogHof [45]	0.7047	0.8105
DLMC-STIPs [13]	0.7379	0.8579
Bag of 3D Points [31]+S1	0.6074	0.6421
3D Shape Context [41]+reference vetor+S3	0.9053	0.9447
Spin Image [32]+S3	0.7579	0.7842
BSC+S1	0.6132	0.6745
BSC+S2	0.8502	0.8911
BSC+S3	**0.9132**	**0.9474**
BSC+RGB LSTF [19]+S3	0.8929	0.9368

direction selection in [41] is infeasible for recognition tasks, we combined the 3-D shape context with our method by using the reference vector. In addition, we combine it with S3 since action representation was not included in [41]. Both kNN and SVM are adopted as the classifiers here.

The comparison results are shown in Table II. It is noted that although the depth information is encoded naively as the DLMC-STIPs, the performance increases obviously compared with the RGB features. The BSC feature outperforms the DLMC-STIPs by 0.0332 and 0.0895 when it is combined with S2 and S3, respectively. We notice that S1 is a relatively weak scheme for action representation, thus the performance of the bag of 3-D points is not very satisfactory. However, the BSC feature surpasses it by 0.0324 when they are both combined with S1. Actually, for each posture, there are 80 points in the bag of 3-D points feature while there are 20 reference points in the BSC feature. It proves that the BSC feature encodes more spatial information for action recognition. The BSC feature performs slightly better than the 3-D shape context. It demonstrates that the support region segmentation does not impact the recognition performance significantly. The BSC feature outperforms the spin image by 0.1632. We attribute this improvement to the fact that the encoding of cylindrical angular increases the descriptiveness and discriminability of the feature. We also report the result of the combination of the BSC feature and the RGB LSFT feature [19] by catenating the two kinds of features. Note that it does not improve the performance compared with only adopting the depth feature. We argue that this is because the BSC encodes sufficient information for action recognition.

6) *Evaluation of the robustness against view variation:* To evaluate the robustness against view variation, we compare the performance of the BSC with the LSTF [19] on the view variation dataset. S3 and kNN are adopted here. The dataset consists of six actions including *Bending*, *Boxing*, *Drinking*, *SittingDown*, *Squatting*, and *StandingUp*. Each action contains two views including the frontal view (View 1) and the view with 30-degree rotation around the vertical axis (View 2). We use the cross-view learning scheme, which uses one view as the training set and the other view as the testing set. Thus we have four runs:

TABLE III
ACCURACIES OF FOUR RUNS FOR VIEW VARIATION EVALUATION

Run	*Run-1*	*Run-2*	*Run-3*	*Run-4*
Average accuracy	0.7833	0.5833	0.8667	0.8667

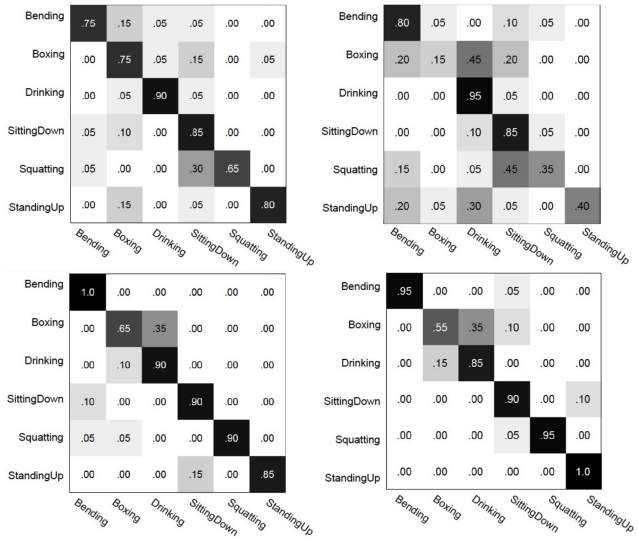


Fig. 16. Confusion matrices of four runs for view variation evaluation. Top left, top right, bottom left, and bottom right are run 1 to run 4 successively.

Run-1: LSTF training on View 1 and testing on View 2;
Run-2: LSTF training on View 2 and testing on View 1;
Run-3: BSC training on View 1 and testing on View 2;
Run-4: BSC training on View 2 and testing on View 1.

The results are shown in Table III. The BSC outperforms the LSTF on two training-testing schemes by 0.083 and 0.283, respectively. The second training-testing scheme is more challenging, and therefore the performance of the LSTF deteriorates significantly. Comparatively, the BSC performs more robustly against view variation. The confusion matrices are shown in Fig. 16. For the BSC, the main confusion comes from *Boxing* and *Drinking*, which are very similar in view variation. However, confusions are more common for the LSTF.

B. Experiments on MSR Action3D

The MSR Action3D dataset [31] was recorded with a depth sensor similar to the Kinect device. It includes 20 action types: *High arm wave*, *Horizontal arm wave*, *Hammer*, *Hand catch*, *Forward punch*, *High throw*, *Draw x*, *Draw tick*, *Draw circle*, *Hand clap*, *Two hand wave*, *Sideboxing*, *Bend*, *Forward kick*, *Side kick*, *Jogging*, *Tennis swing*, *Tennis serve*, *Golf swing*, and *Pick up & throw*. There are 567 depth map sequences in total. The background of the this dataset is clean, so we can obtain the segmentation map of human body. This dataset is challenging since many actions are very similar to each other, especially the arm actions.

The parameters of the BSC feature are set the same as on our dataset and S3 is adopted for action representation. The spatial and temporal scales are both set to 1.5 in the interest point

TABLE IV
ACCURACIES OF BSC AND PREVIOUS FEATURES ON MSR ACTION3D DATASET

Method	Average accuracy
Bag of 3D points [31]	0.747
Joint Position features+ LOP [16]	0.882
HON4D [46]	0.8836
Cuboid Similarity feature [44]	0.893
BSC + S3	0.8745
BSC + S3 +structure information	0.9036

detection. We notice that many spatio-temporal interest points (reference points) are on the background. As the background is clean in this dataset, the coordinates of those reference points in the point cloud are all (0, 0, 0). This will bring in significant error. Therefore, we shift those reference points to the adjacent frames, so that they are all on the human bodies. In addition, we also encode structure information as in [46] by dividing each human body into 4×3 cells and concatenating the histograms to form the final feature.

We compare our feature with the state-of-the-art features on the cross-subject test setting as in [31] and [16]. Samples of half of the subjects are adopted as training data and the rest are adopted as test data. The comparative experiment results are listed in Table IV. Wang *et al.* [16] proposed two features including the local occupancy patterns and the joint position feature. The local occupancy patterns (LOP) feature computes the local occupancy information based on the 3-D point cloud around a particular joint. The joint position features compute the difference between the positions of two joints. HON4D [46] described depth sequences using a histogram describing the distribution of surface normal orientations in the 4-D space of time, depth, and spatial coordinates. Cuboid similarity feature [44] described the local 3-D depth cuboid around the spatio-temporal interest points with an adaptable supporting size. Using the BSC feature, we obtain an average accuracy of 0.8745. When the BSC is combined with structure information, the average accuracy achieves 0.9036 which outperforms other state-of-the-art features.

C. Experiments on MSRDailyActivity3D

The MSRDailyActivity3D dataset [16] is an activity dataset collected by a Kinect device. There are 16 activity types: *Call-cellphone*, *Cheerup*, *Drink*, *Eat*, *Laydownonsofa*, *Playgame*, *Playguitar*, *Readbook*, *Sitdown*, *Sitstill*, *Standup*, *Tosspaper*, *Uselaptop*, *Usevacuumcleaner*, *Walk*, and *Writeonapaper*. Each action is performed by ten subjects. Some actions are performed in two poses: sitting and standing. Altogether, there are 320 action samples. Similar as our dataset, many activities involve humans-object interactions.

The parameters of the BSC feature are set the same as on our dataset. We adopt SVM as the classifier as in [16]. Using the BSC feature, we obtain an average accuracy of 0.778. Also, we compare our feature with several existing features based on joint positions. Muller and Rodder [40] modeled actions by dynamic temporal warping (DTW) and matched the 3-D joint positions to a template. We note that [16] proposes additional steps of discriminative actionlets mining to improve the accuracy. We do not include these steps here since our aim is to compare our basic feature with theirs. Table V summarizes the average accuracies for these features. The BSC outperforms the spin image, the DTW, the LOP, and the joint position feature by 0.172, 0.234, 0.349, and 0.094, respectively. Note that the Fourier temporal pyramid features encode temporal dynamics of actions which is not included in our method. We observe that human body segmentation is relatively unsatisfactory because the sofa which the subjects sit on tends to be considered as part of human body. Samples of unsatisfactory body segmentation are shown in Fig. 17. As our feature is dependent on body segmentation, we believe that this causes the performance decrease of the BSC. How to solve this problem will be the key of our future work. Even though, the BSC is still comparable to the Fourier temporal pyramid features.

TABLE V
ACCURACIES OF THE BSC AND PREVIOUS FEATURES ON THE MSR DAILY ACTIVITY3D DATASET

Method	Average accuracy
Dynamic Temporal Warping [40]	0.54
LOP feature [16]	0.425
Joint Position features [16]	0.68
Joint Position features + LOP + Pyramid [16]	0.78
Spin Image [32] + S3	0.606
BSC + S3	0.778

Fig. 17. Unsatisfactory body segmentation on MSRDailyActivity3D.

VI. CONCLUSION

In this paper, we proposed a new feature based on depth data derived from depth camera. Unlike existing methods, we compute the feature on the point cloud of human body surfaces. The feature is computed by the distribution of relative locations of local neighbors for a reference point. The new feature encodes the cylindrical angular of the difference vector. In addition, it is an approximate object-centered feature which makes it more robust to many variations. Furthermore, we proposed three schemes based on the feature to represent actions. Experiments show that our feature achieves superior performance over RGB-based feature and spin image. Besides, it is proved that our feature performs robustly to view variations. Experiments on the MSR Action3D and the MSRDailyActivity3D datasets show that our feature outperforms or is comparable to existing skeleton based features. Our future work includes studying the more robust segmentation methods. Besides, we also aim to exploit the feature for more challenging actions including multiple-person interactions.

REFERENCES

[1] M. Bregon, S. G. Gong, and T. Xiang, "Recognizing action as clouds of space-time interest points," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2009, pp. 1948–1955.

[2] R. Poppe, "A survey on vision-based human action recognition," *Image Vision Comput.*, vol. 28, pp. 976–990, Jun. 2010.

[3] M. Ahmad and S.-W. Lee, "Human action recognition using shape and CLG-motion flow from multi-view image sequences," *Pattern Recognit.*, vol. 41, pp. 2237–2252, Jul. 2008.

[4] P. Natarajan, V. K. Singh, and R. Nevatia, "Learning 3D action models from a few 2D videos for view invariant action recognition," in *Proc. IEEE Int. Conf. Comput. Vision Pattern Recognit.*, Jun. 2010, pp. 2006–2013.

[5] D. Weinland, E. Boyer, and R. Ronfard, "Action recognition from arbitrary views using 3D exemplars," in *Proc. IEEE Int. Conf. Comput. Vision*, Oct. 2007, pp. 1–7.

[6] D. Weinland, R. Ronfard, and E. Boyer, "Free viewpoint action recognition using motion history volumes," *Comput. Vision Image Understand.*, vol. 104, pp. 249–257, Nov./Dec. 2006.

[7] M. B. Holte, C. Tran, M. M. Trivedi, and T. B. Moeslund, "Human action recognition using multiple views: A comparative perspective on recent deveploments," in *Proc. Joint ACM Workshop Human Gesture Behav. Understand.*, 2011, pp. 47–52.

[8] L. A. Schwarz, A. Mkhitaryan, D. Mateus, and N. Navab, "Human skeleton tracking from depth data using geodesic distances and optical flow," *Image Vision Comput.*, vol. 30, pp. 217–226, Mar. 2012.

[9] F. Wientapper, K. Ahrens, H. Wuest, and U. Bockholt, "Linear-projection-based classification of human postures in time-of-flight data," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, Oct. 2009, pp. 559–564.

[10] F. Tsalakanidou and S. Malassiotis, "Robust facial action recognition from real-time 3D streams," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. Workshops*, Jun. 2009, pp. 4–11.

[11] K. Khoshelham, "Accuracy analysis of Kinect depth data," in *Proc. Int. Arch. Photogrammetry, Remote Sens. Spatial Inf. Sci. Workshop Laser Scan.*, 2011, pp. 133–138.

[12] H. Zhang and L. E. Parker, "4-dimensional local spatio-temporal features for human activity recognition," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2011, pp. 2044–2049.

[13] B. B. Ni, G. Wang, and P. Moulin, "RGBD-HuDaAct: A color-depth video database for human daily activity recognition," in *Proc. IEEE Int. Conf. Comput. Vision Workshops*, Nov. 2011, pp. 1147–1153.

[14] X. D. Yang and Y. L. Tian, "EigenJoints-based action recognition using Naïve-Bayes-Nearest-Neighbor," in *Proc. IEEE Int. Conf. Comput. Vision Workshop*, Jun. 2012, pp. 14–19.

[15] J. Sung, C. Ponce, B. Selman, and A. Saxena, "Unstructured human activity detection from RGBD images," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2012, pp. 842–849.

[16] J. Wang, Z. C. Liu, Y. Wu, and J. S. Yuan, "Mining actionlet ensemble for action recognition with depth cameras," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2012, pp. 1290–1297.

[17] J. C. Niebles, H. C. Wang, and F. F. Li, "Unsupervised learning of human action categories using spatial-temporal words," *Int. J. Comp. Vis.*, vol. 79, no. 3, pp. 299–318, Sep. 2008.

[18] C. Schuldt, I. Laptev, and B. Caputo, "Recognizing human actions: A local SVM approach," in *Proc. IEEE Int. Conf. Pattern Recognit.*, Aug. 2004, pp. 32–36.

[19] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie, "Behavior recognition via sparse spatio-temporal features," in *Proc. 14th Int. Conf. Comput. Commun. Netw.*, Oct. 2005, pp. 65–72.

[20] Y. Song, Y.-T. Zheng, S. Tang, X. Zhou, Y. Zhang, S. Lin, *et al.*, "Localized multiple kernel learning for realistic human action recognition in videos," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 9, pp. 1193–1202, Sep. 2011.

[21] A. Veeraraghavan, A. R. Chowdhury, and R. Chellappa, "Matching shape sequences in video with application in human movement analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 12, pp. 1896–1909, Dec. 2005.

[22] G. Mori, X. Ren, A. A. Efros, and J. Malik, "Recovering human body configurations: Combining segmentation and recognition," in *Proc. IEEE Int. Conf. Comput. Vision Pattern Recognit.*, Jun./Jul. 2004, pp. 326–333.

[23] D. Lowe, "Distinctive image features form scale-invariant keypoints," *Int. J. Comp. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[24] I. Laptev, M. Marszaek, C. Schmid, and B. Rozenfeld, "Learning realistic human actions from movies," in *Proc. IEEE Int. Conf. Comput. Vision Pattern Recognit.*, Jun. 2008, pp. 1–8.

[25] A. Kovashka and K. Grauman, "Learning a hierarchy of discriminant space-time neighborhood features for human action recognition," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2010, pp. 2046–2053.

[26] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore *et al.*, "Real-time human pose recognition in parts from single depth images," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2011, pp. 1297–1304.

[27] A. Hernandez-Vela, N. Zlateva, A. Marinov, M. Reyes, P. Radeva, D. Dimov, *et al.*, "Graph cuts optimization for multi-limb human segmentation in depth maps," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2012, pp. 726–732.

[28] J. Charles and M. Everingham, "Learning shape models for monocular human pose estimation from the Microsoft Xbox Kinect," in *Proc. IEEE Int. Conf. Comput. Vision Workshop*, Nov. 2011, pp. 1202–1208.

[29] M. Reyes, G. Dominguez, and S. Escalera, "Feature weighting in dynamic time warping for gesture recognition in depth data," in *Proc. IEEE Int. Conf. Comput. Vision Workshop*, Nov. 2011, pp. 1182–1188.

[30] M. Raptis, D. Kirovski, and H. Hoppe, "Real-time classification of dance gestures from skeleton animation," in *Proc. ACM SIGGRAPH/Eurograph. Symp. Comput. Animation*, 2011, pp. 147–156.

[31] W. Q. Li, Z. Y. Zhang, and Z. C. Liu, "Action recognition based on a bag of 3D points," in *Proc. IEEE Int. Conf. Comput. Vision Worshops*, Jun. 2010, pp. 9–14.

[32] A. E. Johnson and M. Hebert, "Using spin images for efficient object recognition in cluttered 3D scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 5, pp. 433–449, May 1999.

[33] G. Hetzel, B. Leibe, P. Levi, and B. Schiele, "3D object recognition from range images using local feature histograms," in *Proc. IEEE Int. Conf. Comput. Vision*, 2001, pp. 394–399.

[34] H. Chen and B. Bhanu, "3D free-form object recognition in range images using local surface patches," *Pattern Recognit. Lett.*, vol. 28, pp. 1252–1262, Jul. 2007.

[35] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 4, pp. 509–522, Apr. 2002.

[36] I. Kokkinos, M. M. Bronstein, R. Litman, and A. M. Bronstein, "Intrinsic shape context descriptors for deformable shapes," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2012, pp. 159–166.

[37] M. Grundmann, F. Meier, and I. Essa, "3D shape context and distance transform for action recognition," in *Proc. IEEE Conf. Pattern Recognit.*, Dec. 2008, pp. 1–4.

[38] R. B. Rusu, J. Bandouch, Z. C. Marton, N. Blodow, and M. Beetz, "Action recognition in intelligent environments using point cloud features extracted from silhouette sequences," in *Proc. IEEE Int. Symp. Robot Human Interactive Commun.*, Aug. 2008, pp. 267–272.

[39] K. Lai, L. F. Bo, X. F. Ren, and D. Fox, "Sparse distance learning for object recognition combining RGB and depth information," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2011, pp. 4007–4013.

[40] M. Muller and T. Roder, "Motion templates for automatic classification and retrieval of motion capture data," in *Proc. ACM SIGGRAPH/Eurograph. Symp. Comput. Animation*, 2006, pp. 137–146.

[41] A. Frome, D. Huber, R. Kolluri, T. Bulow, and J. Malik, "Recognizing objects in range data using regional point descriptors," in *Proc. 8th Eur. Conf. Comput. Vision*, 2004, pp. 224–237.

[42] W. Q. Li, Z. Y. Zhang, and Z. C. Liu, "Expandable data-driven graphical modeling of human actions based on salient postures," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 11, pp. 1499–1510, Nov. 2008.

[43] A. Jalal, M. Z. Uddin, and T.-S. Kim, "Depth video-based human activity recognition system using translation and scaling invariant features for life logging at smart home," *IEEE Trans. Consum. Electron.*, vol. 58, no. 3, pp. 863–871, Aug. 2012.

[44] L. Xia and J. K. Aggarwal, "Spatio-temporal depth cuboid similarity feature for activity recognition using depth camera," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2013, pp. 2834–2841.

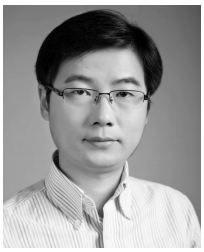
[45] I. Laptev and T. Lindeberg, "Space-time interest points," in *Proc. IEEE Conf. Comput. Vision*, Oct. 2003. pp. 432–439.

[46] O. Oreifej and Z. C. Liu, "HON4D: Histogram of oriented 4D normals for activity recognition from depth sequences," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2013. pp. 716–723.

Yan Song received the Ph.D. degree in computer application technology from Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2011.

She is currently a Lecturer with Nanjing University of Science and Technology, Nanjing, China. She was an intern with School of Computing, National University of Singapore, Singapore, from 2009 to 2010. Her research interests include multimedia information processing, in particular, video content analysis and understanding. The main focus is on human action recognition and event detection of videos.

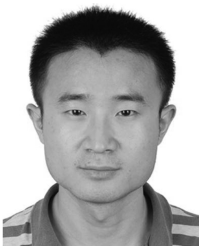
Dr. Song has served as a reviewer for *Journal of Visual Communication and Image Representation*, *Neurocomputing*. She received the 2011 ZhuLiYueHua Award of Chinese Academy of Science. She is a member of ACM and China Computer Federation.



Jinhui Tang (M'13) received the B.E. and Ph.D. degrees from University of Science and Technology of China, Beijing, China, in 2003 and 2008, respectively.

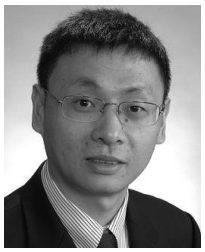
He is currently a Professor with Nanjing University of Science and Technology, Nanjing, China. From 2008 to 2010, he was a Research Fellow with the School of Computing, National University of Singapore, Singapore. During that period, he visited the School of Information and Computer Science, University of California, Irvine, Irvine, CA, USA, from January 2010 to April 2010, as a Visiting Research Scientist. From 2011 to 2012, he was a Visiting Researcher with Microsoft Research Asia. He has authored over 70 journal and conference papers in these areas, including IEEE TRANSACTION ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, IEEE TRANSACTION ON IMAGE PROCESSING, IEEE TRANSACTION ON MULTIMEDIA, IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, *ACM Transactions on Multimedia Computing*, *ACM Multimedia*, and *ACM Computer Vision and Pattern Recognition*. His research interests include large-scale multimedia search, social media mining, and computer vision.

Dr. Tang is Editorial Board Member of *Pattern Analysis and Applications*, *Multimedia Tools and Applications*, *Information Sciences*, *Neurocomputing*, and is a Guest Editor of IEEE TRANSACTION ON MULTIMEDIA, *ACM TIST*, *MMSJ*, *JVCI*, and *Neurocomputing*, a Technical Committee Member for over 30 international conferences, and a Reviewer for over 30 prestigious international journals. He is a co-recipient of the Best Paper Award in ACM Multimedia 2007, PCM 2011, and ICIMCS 2011. He is a member of ACM.



Fan Liu received the bachelor's degree in network engineering from Nanjing University of Science and Technology, Nanjing, China, 2005, where he is currently pursuing the Ph.D. degree in computer application technology.

His research interests include computer vision and pattern recognition.



Shuicheng Yan (SM'13) received the Ph.D. degree from Peking University, China, in 2004. Currently an Associate Professor with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore, and the Founding Lead of the Learning and Vision Research Group (http://www.lv-nus.org). He has authored or co-authored over 300 technical papers over a wide range of research topics, with Google Scholar citation H-index-42. His research interests include computer vision, multimedia, and machine learning.

He is an Associate Editor of IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY and *ACM Transactions on Intelligent Systems and Technology*, and served as the Guest Editor for the special issues for TMM and CVIU. He received the Best Paper Award from ACM MM12, PCM'11, ACM MM10, ICME10, and ICIMCS'09. He was also the winner of the classification task in PASCAL VOC 2010–2012, the segmentation task in PASCAL VOC 2012, the honorable mention prize of the detection task in PASCAL VOC'10, 2010 TCSVT Best Associate Editor (BAE) Award, 2010 Young Faculty Research Award, 2011 Singapore Young Scientist Award, and 2012 NUS Young Researcher Award.