3DV: 3D Dynamic Voxel for Action Recognition in Depth Video

Yancheng Wang[1], Yang Xiao[1†], Fu Xiong[2], Wenxiang Jiang[1], Zhiguo Cao[1], Joey Tianyi Zhou[3], and
Junsong Yuan[4]

[1] National Key Laboratory of Science and Technology on Multispectral Information Processing, School
of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan
430074, China   [2]Megvii Research Nanjing, Megvii Technology, China
[3] IHPC, A*STAR, Singapore   [4]CSE Department, State University of New York at Buffalo

yancheng_wang, Yang_Xiao@hust.edu.cn, xiongfu@megvii.com, wenx_jiang, zgcao@hust.edu.cn,
zhouty@ihpc.a-star.edu.sg, jsyuan@buffalo.edu

Abstract

To facilitate depth-based 3D action recognition, 3D dynamic voxel (3DV) is proposed as a novel 3D motion representation. With 3D space voxelization, the key idea of 3DV is to encode 3D motion information within depth video into a regular voxel set (i.e., 3DV) compactly, via temporal rank pooling. Each available 3DV voxel intrinsically involves 3D spatial and motion feature jointly. 3DV is then abstracted as a point set and input into PointNet++ for 3D action recognition, in the end-to-end learning way. The intuition for transferring 3DV into the point set form is that, PointNet++ is lightweight and effective for deep feature learning towards point set. Since 3DV may lose appearance clue, a multi-stream 3D action recognition manner is also proposed to learn motion and appearance feature jointly. To extract richer temporal order information of actions, we also divide the depth video into temporal splits and encode this procedure in 3DV integrally. The extensive experiments on 4 well-established benchmark datasets demonstrate the superiority of our proposition. Impressively, we acquire the accuracy of 82.4% and 93.5% on NTU RGB+D 120 [13] with the cross-subject and cross-setup test setting respectively. 3DV's code is available at https://github.com/3huo/3DV-Action.

1. Introduction

During the past decade, due to the emergence of low-cost depth camera (e.g., Microsoft Kinect [52]) 3D action recognition becomes an active research topic, with the wide-range application scenarios of video surveillance, human-machine interaction, etc [45, 46]. The state-of-the-art 3D action recognition approaches can be generally categorized

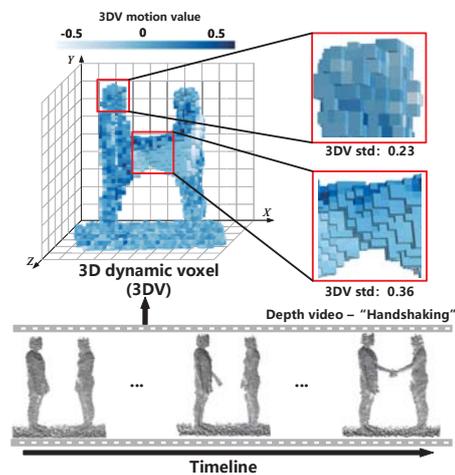†Yang Xiao is corresponding author (Yang_Xiao@hust.edu.cn).

Figure 1. A live "Handshaking" 3DV example from NTU RGB+D 60 dataset [33]. 3DV motion value reveals the temporal order of 3D motion component. The later motion component is of higher value, and vice verse. And, the local region of richer motion information holds higher standard deviation on 3DV motion value.

into the depth-based [28, 17, 16, 36, 11, 51, 35, 34] and skeleton-based groups [32, 48, 22, 10, 42, 46]. Since accurate and robust 3D human pose estimation is still challenging [47, 21], we focus on depth-based avenue in this work.

Since human conducts actions in 3D space, capturing 3D motion pattern effectively and efficiently is crucial for depth-based 3D action recognition. An intuitive way is to calculate dense scene flow [1]. However this can be time consuming [1], which may not be preferred by the practical applications. Recently, dynamic image [3, 2] able to represent the motion information within RGB video compactly has been introduced to depth domain for 3D action characterization [42, 46]. It can compress RGB video into a single image, while still maintaining the motion characteristics well via temporal rank pooling [6, 5]. Thus dynamic image

can fit deep CNN model [8] well for action categorization, which is leveraged by CNN's strong pattern representation capacity. Nevertheless we argue that the ways of applying dynamic image to 3D field in [42, 46] have not fully exploited 3D descriptive clue within depth video, although normal vector [42] or multi-view projection [46] is applied. The insight is that, both methods in [42, 46] finally encode 3D motion information onto the 2D image plane to fit CNN. Thus, they cannot well answer the question *"Where does the certain 3D motion pattern within human action appear in 3D space? "* crucial for effective 3D action characterization due to the fact that human actions actually consists of both motion patterns and compact spatial structure [29].

To address the concern above, we propose 3D dynamic voxel (3DV) as a novel 3D motion representation for 3D action representation. To extract 3DV, 3D space voxelization is first executed. Each depth frame will be transformed into a regular voxel set. And the appearance content within it can be encoded by observing whether the yielded voxels have been occupied or not [40], in a binary way. Then, temporal rank pooling [6, 5] is executed towards all the binary voxel sets to compress them into one single voxel set termed 3DV particularly. Thus, *3D motion and spatial characteristics of 3D action can be encoded into 3DV jointly.* To reveal this, a live "Handshaking" 3DV example is provided in Fig. 1. As shown, each available 3DV voxel possesses a motion value able to reflect the temporal order of its corresponding 3D motion component. Specifically, the later motion component is of higher value, and vice verse. Meanwhile, the local region of richer 3D motion information possesses higher standard deviation on 3DV motion value (e.g., hand region vs. head region). Meanwhile, 3DV voxel's location reveals the 3D position of its 3D motion component. Thus, 3DV's spatial-motion representative ability can essentially leverage 3D action characterization. To involve richer temporal order information, we further divide depth video into finer temporal splits. This is encoded in 3DV integrally by fusing the motion values from all the temporal splits.

With 3DV, the upcoming question is how to choose the adaptive deep learning model to conduct 3D action recognition particularly. Towards voxel set, 3D CNN [20, 7, 21] is often used for 3D visual pattern understanding, and also applicable to 3DV. However, it is difficult to train due to the large number of convolutional parameters. Inspired by the recent success of the lightweighted deep learning models on point set (e.g., PointNet++ [25]), we propose to transfer 3D-V into the point set form as the input of PointNet++ to conduct 3D action recognition in end-to-end learning manner. That is, each 3DV voxel will be abstract as a point characterized by its 3D location index and motion value. Our intuition is to alleviate the training difficulty and burden.

Although 3DV can reveal 3D motion information, it still may lose appearance details as in Fig. 1. Since appearance

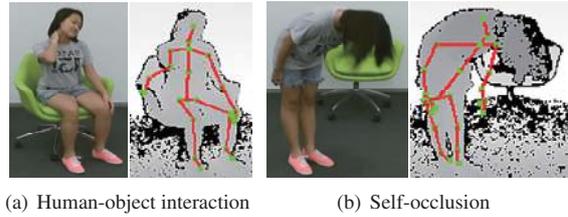(a) Human-object interaction　　(b) Self-occlusion

Figure 2. 3D skeleton extraction failure cases in NTU RGB+D 60 dataset [13], due to human-object interaction and self-occlusion. The depth frame and its RGB counterpart are shown jointly.

also plays vital role for action recognition [23, 37], only using 3DV may weaken performance. To alleviate this, a multi-stream deep learning model using PointNet++ is also proposed to learn 3D motion and appearance feature jointly. In particular, it consists of one motion stream and multiple appearance streams. The input of motion stream is 3DV. And, the inputs of appearance streams are the depth frames sampled from the different temporal splits. They will also be transformed into the point set form to fit PointNet++.

The experiments on 2 large-scale 3D action recognition datasets (i.e., NTU RGB+D 120 [13] and 60 [33]), and 2 small-scale ones (i.e., N-UCLA [41] and UWA3DII [26]) verify 3DV's superiority over the state-of-the-art manners.

The main contributions of this paper include:

• 3DV: a novel and compact 3D motion representative manner for 3D action characterization;

• PointNet++ is applied to 3DV for 3D action recognition in end-to-end learning way, from point set perspective;

• A multi-stream deep learning model is proposed to learn 3D motion and appearance feature jointly.

2. Related Works

3D action recognition. The existing 3D action recognition approaches generally falls into the depth-based [23, 23, 48, 22, 10, 42, 46] and skeleton-based groups [15, 17, 16, 36, 11, 51, 35, 34]. Recently the skeleton-based approaches with RNN [15] and GCN [35] has drawn more attention, since using 3D skeleton can help to resist the impact of variations on scene, human attribute, imaging viewpoint, etc. However, one critical issue should not be ignored. That is, accurate and robust 3D human pose estimation is still not trivial [47, 21]. To reveal this, we have checked the 3D skeletons within NTU RGB+D 60 [13] carefully. Actually, even under the constrained condition 3D skeleton extraction still may fail to work as in Fig. 2. Thus, currently for the practical applications depth-based manner seems more preferred and is what we concern.

Most of the paid efforts focus on proposing 3D action representation manner to capture 3D spatial-temporal appearance or motion pattern. At the early stage, the hand-crafted descriptions of bag of 3D points [12], depth mo-

tion map (DMM) [49], Histogram of Oriented 4D Normals (HON4D) [23], Super Normal Vector (SNV) [19] and binary range sample feature [19] are proposed from the different research perspectives. Recently CNN [8, 50] has been introduced to this field [43, 44, 42, 46], and enhanced performance remarkably. Under this paradigm, the depth video will be compressed into one image using DMM [49] or dynamic image [3, 2] to fit CNN. To better exploit 3D descriptive clue, normal vector or multi-view projection is applied additionally. However, they generally suffer from 2 main defects. First, as aforementioned DMM or dynamic image cannot fully reveal 3D motion characteristics well. Secondly, they tend to ignore appearance information.

Temporal rank pooling. To represent action, temporal rank pooling [6, 5] is proposed to capture the frame-level evolution characteristics within video. Its key idea is to train a linear ranking machine towards the frames to arrange them in chronological order. Then, the parameters of the ranking machine can be used as the action representation. By applying temporal rank pooling to the raw frame pixels, dynamic image [3, 2] is proposed with strong motion representative ability and adaptive to CNN. As aforementioned, temporal rank pooling has recently been applied to 3D action recognition [42, 46]. However, how to use it to fully reveal 3D motion property still has not been deeply studied.

Deep learning on point set. Due to the irregularity of 3D point set, typical convolutional architectures (e.g., CNN [8]) cannot handle it well. To address this, deep learning on point set draws the increasing attention. Among the paid efforts, PointNet++ [25] is the representative one. It contributes to ensure the permutation invariance of point sets, and capture 3D local geometric clue. However, it has not been applied to 3D action recognition yet.

Accordingly, 3DV is proposed to characterize 3D motion compactly, via temporal rank pooling. The adaptive multi-stream deep learning model using PointNet++ is also proposed to learn 3D motion and appearance feature jointly.

3. 3DV: A Novel Voxel Set based Compact 3D Motion Representative Manner

Our research motivation on 3DV is to seek a compact 3D motion representative manner to characterize 3D action. Accordingly, deep feature learning can be easily conducted on it. The proposition of 3DV can be regarded as the essential effort for extending temporal rank pooling [6, 5] originally for 2D video to 3D domain, to capture 3D motion pattern and spatial clue jointly. The main idea for 3DV extraction is in Fig. 3. The depth frames will be first map into point clouds to better reveal 3D characteristics. Then, 3D voxelization is executed to further transform the disordered point clouds into the regular voxel sets. Consequently, 3D action appearance clue within the certain depth frame can be described by judging whether the voxels have been oc-
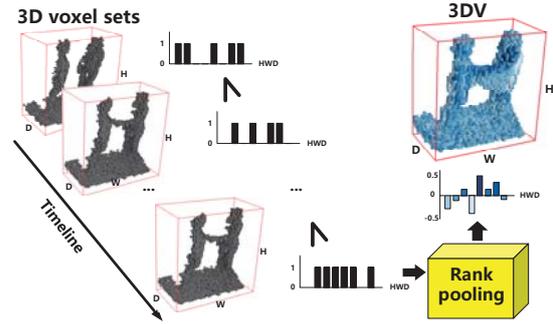

Figure 3. The main idea for 3DV extraction via temporal rank pooling, towards the 3D voxel sets transformed from depth frames.

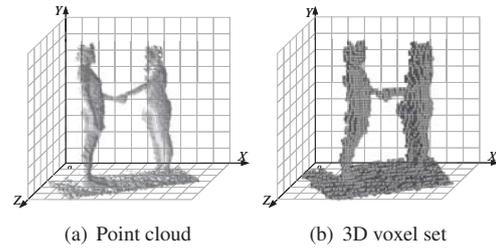
(a) Point cloud   (b) 3D voxel set

Figure 4. The point cloud and its corresponding 3D voxel set sampled from "Handshaking".

cupied or not. Then temporal rank pooling is executed to the yielded binary voxel sets to compress them into one voxel set (i.e., 3DV), to reveal the 3D appearance evolution within actions compactly. The resulting ranking machine parameters can actually characterize 3D motion pattern of the corresponding 3DV voxels. In particular, each 3DV voxel possesses a motion value (i.e., ranking machine parameter). And, its 3D position can encode the spatial property of the corresponding 3D motion pattern. Action proposal will also be conducted to resist background.

3.1. Voxel-based 3D appearance representation

Projecting 3D data to 2D depth frame actually distorts the real 3D shape [21]. To better represent 3D appearance clue, we map the depth frame into point cloud. Nevertheless, one critical problem emerges. That is, temporal rank pooling cannot be applied to the yielded point clouds directly, due to their disordered property [25] as in Fig. 4(a). To address this, we propose to execute 3D voxelization towards the point clouds. Then the 3D appearance information can be described by observing whether the voxels have been occupied or not, disregarding the involved point number as

$$V_t(x, y, z) = \begin{cases} 1, & \text{if } V_t(x, y, z) \text{ is occupied} \\ 0, & \text{otherwise} \end{cases}, \quad (1)$$

where $V_t(x, y, z)$ indicates one certain voxel at the t-th frame; (x, y, z) is the regular 3D position index. This actually holds 2 main profits. First, the yielded binary 3D voxel sets are regular as in Fig. 4(b). Thus, temporal rank pooling

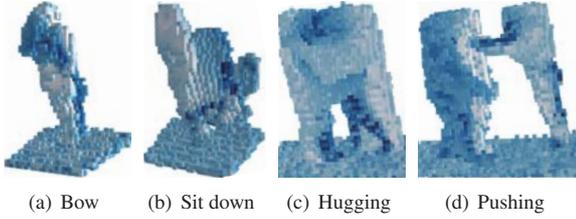(a) Bow   (b) Sit down   (c) Hugging   (d) Pushing

Figure 5. The 3DV examples from NTU RGB+D 60 dataset [33].

can be applied to them for 3DV extraction. Meanwhile the binary voxel-wise representation manner is of higher tolerance towards the intrinsic sparsity and density variability problem [25] within point clouds, which essentially helps to leverage generalization power.

3.2. 3DV extraction using temporal rank pooling

With the binary 3D appearance voxel sets above, temporal rank pooling is executed to generate 3DV. A linear temporal ranking score function will be defined for compressing the voxel sets into one voxel set (i.e., 3DV).

Particularly, suppose $V_i, \ldots, V_T$ indicate the binary 3D appearance voxel sets, and $\overline{V_i} = \frac{1}{t} \times \sum_i^t V_i$ is their average till time t. The ranking score function at time t is given by

$$S(t|\mathbf{w}) = \langle \mathbf{w}, \overline{V_i} \rangle, \qquad (2)$$

where $\mathbf{w} \in \mathbb{R}^d$ is the ranking parameter vector. $\mathbf{w}$ is learned from the depth video to reflect the ranking relationship among the frames. The criteria is that, the later frames are of larger ranking scores as

$$q > t \Rightarrow S(q|\mathbf{w}) > S(t|\mathbf{w}). \qquad (3)$$

The learning procedure of $\mathbf{w}$ is formulated as a convex optimization problem using RankSVM [38] as

$$\mathbf{w}^* = \underset{\mathbf{w}}{\operatorname{argmin}} \frac{\lambda}{2} \| \mathbf{w} \|^2 +$$
$$\frac{2}{T(T-2)} \times \sum_{q>t} \max\{0, 1 - S(q|\mathbf{w}) + S(t|\mathbf{w})\}. \qquad (4)$$

Specifically, the first term is the often used regularizer for SVM. And, the second is the hinge-loss for soft-counting how many pairs $q > t$ are incorrectly ranked, which does not obey $S(q|\mathbf{w}) > S(t|\mathbf{w}) + 1$. Optimizing Eqn. 4 can map the 3D appearance voxel sets $V_i, \cdots, V_T$ to a single vector $\mathbf{w}^*$. Actually, $\mathbf{w}^*$ encodes the dynamic evolution information from all the frames. Spatially reordering $\mathbf{w}^*$ from 1D to 3D in voxel form can construct 3DV for 3D action characterization. Thus, each 3DV voxel can be jointly encoded by the corresponding $\mathbf{w}^*$ item as motion feature and its regular 3D position index (x, y, z) as spatial feature. Some more 3DV examples are shown in Fig. 5. We can intuitively observe that, 3DV can actually distinguish the different actions from motion perspective even human-object or human-human interaction happens. Meanwhile to accelerate 3DV
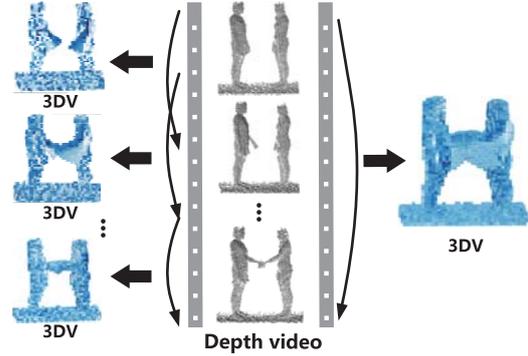

Figure 6. Temporal split for 3DV extraction.

extraction for application, the approximated temporal rank pooling [2] is used by us during implementation.

3.3. Temporal split

Applying temporal rank pooling to whole depth video may vanish some fine temporal order information. To better maintain motion details, we propose to execute temporal split for 3DV. The depth video will be divided into T_1 temporal splits with the overlap ratio of 0.5, which is the same as [46]. 3DV will extract from all the temporal splits and the whole depth video simultaneously as in Fig. 6, to involve the global and partial temporal 3D motion clues jointly.

3.4. Action proposal

Since background is generally not helpful for 3D action characterization, action proposal is also conducted by us, following [46] but with some minor modifications. First YOLOv3-Tiny [30] is used for human detection instead of Faster R-CNN [31], concerning running speed. Meanwhile, human and background are separated by depth thresholding. Particularly, depth value histogram is first extracted with the discretization interval of 100 mm. The interval of highest occurrence probability is then found. The threshold is empirically set as its mediate value plus 200 mm. Then, 3DV will be extracted only from action proposal's 3D space.

4. Deep learning network on 3DV

After acquiring 3DV, the upcoming problem is how to conduct deep learning on it to conduct feature learning and 3D action type decision jointly. Since 3DV appears in 3D voxel form, an intuitive way is to apply 3D CNN to it as many 3D visual recognition methods [20, 7, 21] does. Nevertheless 3D CNN is generally hard to train, mainly due to its relatively large number of model parameters. Deep learning on point set (e.g., PointNet [24] and PointNet++ [25]) is the recently emerged research avenue to address the disordered characteristics of point set, with promising performance and lightweight model size. Inspired by this, we propose to apply PointNet++ to conduct
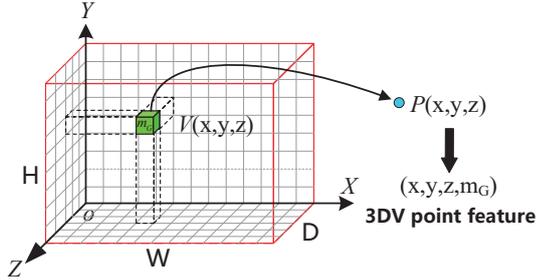
Figure 7. The procedure of abstracting 3DV voxel $V\{x, y, z\}$ into 3DV point $P\{x, y, z\}$.

deep learning on 3DV instead of 3D CNN concerning effectiveness and efficiency jointly. To this end, 3DV will be abstracted into point set form. To our knowledge, using PointNet++ to deal with voxel data has not been well studied before. Meanwhile since 3DV tends to loose some appearance information as shown Fig. 4, a multi-stream deep learning model based on PointNet++ is also proposed to learn appearance and motion feature for 3D action characterization.

4.1. Review on PointNet++

PointNet++ [25] is derived from PointNet [24], the pioneer in deep learning on point set. PointNet is proposed mainly to address the disordered problem within point clouds. However, it cannot capture the local fine-grained pattern well. PointNet++ alleviates this in a local-to-global hierarchical learning manner. It declares 2 main contributions. First, it proposes to partition the set of points into overlap local regions to better maintain local fine 3D visual clue. Secondly, it uses PointNet recursively as the local feature learner. And, the local features will be further grouped into larger units to reveal the global shape characteristics. In summary, PointNet++ generally inherits the merits of PointNet but with stronger local fine-grained pattern descriptive power. Compared with 3D CNN, PointNet++ is generally of more light-weight model size and higher running speed. Meanwhile, it tends to be easier to train.

The main intuitions for why we apply PointNet++ to 3D-V lie into 3 folders. First, we do not want to trap in the training challenges of 3D CNN. Secondly, PointNet++ is good at capturing local 3D visual pattern, which is beneficial for 3D action recognition. That is, local 3D motion pattern actually plays vita role for good 3D action characterization, as the hand region shown in Fig. 1 towards "Handshaking". Last, applying PointNet++ to 3DV is not a difficult task. What we need to do is to abstract 3DV into the point set form, which will be illustrated next.

4.2. Abstract 3DV into point set

Suppose the acquired 3DV for a depth video without temporal split is of size $H \times W \times D$, each 3DV voxel $V(x, y, z)$ will possesses a global motion value m_G given
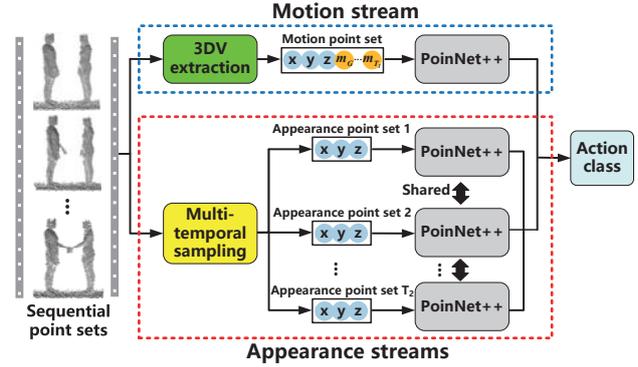

Figure 8. PointNet++ based multi-stream network for 3DV to learn motion and appearance feature jointly.

by temporal rank pooling as in Fig. 7 where (x, y, z) indicates the 3D position index of $V(x, y, z)$ within 3DV. To fit PointNet++, $V(x, y, z)$ is then abstracted as a 3DV point $P(x, y, z)$ with the descriptive feature of (x, y, z, m_G). Particularly, (x, y, z) denotes the 3D spatial feature and m is the motion feature. Thus, the yielded $P(x, y, z)$ is able to represent the 3D motion pattern and corresponding spatial information integrally. Since (x, y, z) and m are multi-modular feature, feature normalization is executed to balance their effect towards PointNet++ training. Specifically, m is linearly normalized into the range of $[-0.5, 0.5]$. Towards the spatial feature, y is first linearly normalized into the range of $[-0.5, 0.5]$. Then x and z are re-scaled respectively, according to their size ratio towards y. In this way, the 3D geometric characteristics can be well maintained to alleviate distortion. As illustrated in Sec. 3.3, temporal split is executed in 3DV to involve multi-temporal motion information. Thus, each 3DV point $P(x, y, z)$ will correspond to multiple global and local motion values. We propose to concatenate all the motion values to describe 3D motion pattern integrally. $P(x, y, z)$ will be finally characterized by the spatial-motion feature as

$$F_{P(x,y,z)} = (\overbrace{x, y, z}^{Spatial} \overbrace{m_G, m_1, ...m_{T_1}}^{Motion}), \qquad (5)$$

where m_G is the motion feature extracted from whole video; m_i denotes motion feature from the i-th temporal split; and T_1 is the number of temporal splits as in Sec. 3.3.

4.3. Multi-stream network

Since 3DV may lose fine appearance clue, a multi-stream network using PointNet++ is proposed to learn motion and appearance feature jointly, following the idea in [37] for RGB video. As in Fig. 8, it consists of 1 motion stream and multiple appearance streams. The input of motion stream is the single 3DV point set from Sec. 4.2. For motion PointNet++ the 3DV points with all the motion features of 0 will not be sampled. And, the inputs of appearance streams are

the raw depth point sets sampled from T_2 temporal splits with action proposal. Particularly, they share the same appearance PointNet++. Motion and appearance feature is late fused via concatenation at fully-connected layer.

5. Implementation details

3DV voxel is set of size $35mm \times 35mm \times 35mm$. T_1 and T_2 is set to 4 and 3 respectively, for multi-temporal motion and appearance feature extraction. For PointNet++, farthest point sampling is used on the centroids of local regions. The sampled points are grouped with ball query. The group radius at the first and second level is set to 0.1 and 0.2 respectively. Adam [9] is applied as the optimizer with batch size of 32. Leaning rate begins with 0.001, and decays with a rate of 0.5 every 10 epochs. Training will end when reaching 70 epochs. During training, we perform data augmentation for 3DV points and raw depth points including random rotation around Y and X axis, jittering and random points dropout. Multi-stream network is implemented using PyTorch. Within each stream, PointNet++ will sample 2048 points for both of motion and appearance feature learning.

6. Experiments

6.1. Experimental setting

Dataset: NTU RGB+D 120 [13]. It is the most recently emerged challenging 3D action recognition dataset, and also of the largest size. Particularly, 114,480 RGB-D action samples of 120 categories captured using Microsoft Kinect v2 are involved in this dataset. These involved action samples are of large variation on subject, imaging viewpoint and background. This imposes essential challenges to 3D action recognition. The accuracy of the state-of-the-art approaches is not satisfactory (i.e., below 70%) both under the cross-subject and cross-setup evaluation criteria.

Dataset: NTU RGB+D 60 [33]. It is the preliminary version of NTU RGB+D 120. That is, 56,880 RGB-D action samples of 60 categories captured using Microsoft Kinect v2 are involved in this dataset. Before NTU RGB+D 120, it is the largest 3D action recognition dataset. Cross-subject and cross-view evaluation criteria is used for test.

Dataset: N-UCLA [41]. Compared with NTU RGB+D 120 and NTU RGB+D 120, this is a relatively small-scale 3D action recognition dataset. It only contains 1475 action samples of 10 action categories. These samples are captured using Microsoft Kinect v1 from 3 different viewpoints, with relatively higher imaging noise. Cross-view evaluation criteria is used for test.

Dataset: UWA3DII [26]. This is also a small-scale 3D action recognition dataset with only 1075 video samples from 30 categories. One essential challenge of this dataset is the limited number of training samples per action category. And, the samples are captured using Microsoft Kinect

Table 1. Performance comparison on action recognition accuracy (%) among different methods on NTU RGB+D 120 dataset.

Methods	Cross-subject	Cross-setup
Input: 3D Skeleton		
NTU RGB+D 120 baseline [13]	55.7	57.9
GCA-LSTM [17]	58.3	59.3
FSNet [14]	59.9	62.4
Two stream attention LSTM [16]	61.2	63.3
Body Pose Evolution Map [18]	64.6	66.9
SkeleMotion [4]	67.7	66.9
Input: Depth maps		
NTU RGB+D 120 baseline [13]	48.7	40.1
3DV-PointNet++ (ours)	**82.4**	**93.5**

Table 2. Performance comparison on action recognition accuracy (%) among different methods on NTU RGB+D 60 dataset.

Methods	Cross-subject	Cross-view
Input: 3D Skeleton		
SkeleMotion [4]	69.6	80.1
GCA-LSTM [17]	74.4	82.8
Two stream attention LSTM [16]	77.1	85.1
AGC-LSTM [36]	89.2	95.0
AS-GCN [11]	86.8	94.2
VA-fusion [51]	89.4	95.0
2s-AGCN [35]	88.5	95.1
DGNN [34]	**89.9**	96.1
Input: Depth maps		
HON4D [23]	30.6	7.3
SNV [48]	31.8	13.6
HOG^2 [22]	32.2	22.3
Li. [10]	68.1	83.4
Wang. [42]	87.1	84.2
MVDI [46]	84.6	87.3
3DV-PointNet++ (ours)	88.8	**96.3**

v1 with relatively high imaging noise.

Input data modality and evaluation metric. During experiments, the input data of our proposed 3DV based 3D action recognition method is only depth maps. We will not use any other auxiliary information, such as skeleton, RGB image, human mask, etc. The training / test sample splits and testing setups on all the 4 datasets are strictly followed for fair comparison. Classification accuracy on all the action samples is reported for performance evaluation.

6.2. Comparison with state-of-the-art methods

NTU RGB+D 120: Our 3DV based approach is compared with the state-of-the-art skeleton-based and depth-based 3D action recognition methods [13, 17, 16, 18, 4] on this dataset. The performance comparison is listed in Table 1. We can observe observed that:

• It is indeed impressive that, our proposition achieves the breaking-through results on this large-scale challenging dataset both towards the cross-subject and cross-setup test settings. Particularly we achieve 82.4% and 93.5% on these 2 settings respectively, which outperforms the state-of-the-art manners by large margins (i.e., 14.7% at least on cross-subject, and 26.6% at least on cross-setup). This essentially verifies the superiority of our proposition;

• The performance of the other methods is poor. This reveals the great challenges of NTU RGB+D 120 dataset;

Table 3. Performance comparison on action recognition accuracy (%) among different depth-based methods on N-UCLA dataset.

Methods	Accuracy
HON4D [23]	39.9
SNV [48]	42.8
AOG [41]	53.6
HOPC [27]	80.0
MVDI [46]	84.2
3DV-PointNet++ (ours)	**95.3**

Table 4. Performance comparison on action recognition accuracy (%) among different depth-based methods on UWA3DII dataset.

Methods	Mean accuracy
HON4D [23]	28.9
SNV [48]	29.9
AOG [41]	26.7
HOPC [27]	52.2
MVDI [46]	68.1
3DV-PointNet++ (ours)	**73.2**

Table 5. Effectiveness of 3DV motion feature on NTU RGB+D 120 dataset. Appearance stream is not used.

T_1	3DV point feature	Cross-subject	Cross-setup
1	(x, y, z)	61.4	68.9
1	(x, y, z, m_G)	75.1	87.4

Table 6. Effectiveness of temporal split for 3DV extraction on N-TU RGB+D 120 dataset. Appearance stream is not used.

T_1	3DV point feature	Cross-subject	Cross-setup
1	(x, y, z, m_G)	75.1	87.4
2	(x, y, z, m_G, m_1, m_2)	75.8	89.6
4	$(x, y, z, m_G, m_1, \ldots, m_4)$	76.9	92.5

Table 7. Effectiveness of appearance stream on NTU RGB+D 60 and 120 dataset.

Dataset	Input stream	Cross-subject	Cross-setup
		Cross-subject	Cross-setup
NTU 120	3DV	76.9	92.5
	Appearance	72.1	79.4
	3DV+appearance	**82.4**	**93.5**
		Cross-subject	Cross-view
NTU 60	3DV	84.5	95.4
	Appearance	80.1	85.1
	3DV+appearance	**88.8**	**96.3**

Table 8. Effectiveness of action proposal on N-UCLA dataset.

Action proposal	Accuracy
W/O	92.9
With	**95.3**

• Our method achieves better performance on cross-setup case than cross-subject. This implies that, 3DV is more sensitive to subject variation.

NTU RGB+D 60: The proposed method is compared with the state-of-the-art approaches [17, 16, 36, 11, 51, 35, 34, 23, 48, 22, 10, 42, 46] on this dataset. The performance comparison is listed in Table 2. We can see that:

• Our proposition still significantly outperforms all the depth-based manners, both on the cross-subject and cross-view test settings.

• On cross-view setting, the proposed method is also superior to all the skeleton-based manners. And, it is only slightly inferior to DGNN [34] on cross-subject setting; This reveals that, only using depth maps can still achieve the promising performance.

• By comparing Table 1 and 2, we can find that the performance of some methods (i.e., GCA-LSTM [17], Two stream attention LSTM [16],) significantly drops. Concerning the shared cross-subject setting, GCA-LSTM drops 16.1% and Two stream attention LSTM drops 15.9%. However, our manner only drops 6.4%. This demonstrates 3D-V's strong adaptability and robustness.

N-UCLA and UWA3DII: We compared the proposed manner with the state-of-the-art depth-based approaches [23, 48, 41, 27, 46] on these 2 small-scale datasets. The performance comparison is given in Table 3 and 4 respectively. To save space, the average accuracy of the different viewpoint combinations is reported on UWA3DII. It can be summarized that:

• On these 2 small-scale datasets, the proposed approach still consistently outperforms the other depth-based manners. This demonstrates that, our proposition takes advantages over both of the large-scale and small-scale test cases;

• 3DV does not perform well on UWA3DII, with the accuracy of 73.2%. In our opinion, this may be caused by the fact that the training sample amount per class is limited on this dataset. Thus, deep learning cannot be well conducted.

6.3. Ablation study

Effectiveness of 3DV motion feature: To verify this, we choose to remove 3DV motion feature from the sampled 3DV points within PointNet++ to observe the performance change. The comparison results on NTU RGB+D 120 dataset are given in Table 5. We can see that, without the motion feature 3DV's performance will significantly drop (i.e., 18.5% at most).

Effectiveness of temporal split for 3DV extraction: Towards this, the temporal split number T_1 is set to 1, 2, and 4 respectively on NTU RGB+D 120 dataset. The comparison results are listed in Table 6. Obviously, temporal split can essentially leverage the performance in all test cases.

Effectiveness of appearance stream: This is verified on NTU RGB+D 60 and 120 dataset simultaneously, as listed in Table 7. We can observe that:

• The introduction of appearance stream can consistently enhance the performance of 3DV on these 2 datasets, towards all the 3 test settings;

• 3DV stream significantly outperforms the appearance stream consistently, especially on cross-setup and cross-view settings. This verifies 3DV's strong discriminative power for 3D action characterization in motion way.

Effectiveness of Action proposal: The performance comparison of our method with and without action proposal on N-UCLA dataset is listed in Table 8. Actually, action proposal can help to enhance performance.

PointNet++ vs. 3D CNN: To verify the superiority of PointNet++ for deep learning on 3DV, we compare it with

Table 9. Comparison on performance and complexity between PointNet++ and C3D on N-UCLA and NTU RGB+D 60 dataset.

Method	Parameters	FLOPs	N-UCLA	NTU RGB+D 60
C3D	29.2M	10.99G	64.5	85.0
PointNet++	**1.24M**	**1.24G**	**71.3**	**90.0**

Table 10. Performance comparison among the different sampling point numbers for 3DV, on NTU RGB+D 120 dataset.

Sampling point number	Cross-subject	Cross-view
512	74.9	89.0
1024	75.7	90.9
2048	**76.9**	**92.5**

Table 11. Performance comparison among the different 3DV voxel sizes, on NTU RGB+D 120 dataset.

Voxel size (mm)	Cross-subject	Cross-view
$25 \times 25 \times 25$	75.9	92.0
$35 \times 35 \times 35$	**76.9**	**92.5**
$50 \times 50 \times 50$	76.0	91.6
$75 \times 75 \times 75$	74.1	90.4

3D CNN. particularly, the well-established 3D CNN model (i.e., C3D [39]) for video classification is used with some modification. That is, the number of C3D's input channels is reduced from 3 to 1. And, 3DV is extracted with the fixed size of $50 \times 50 \times 50$ grids as the input of C3D. Without data augmentation and temporal split, the performance and model complexity comparison on N-UCLA and NTU RGB+D 60 dataset (cross-view setting) is given in Table 9. We can see that, PointNet++ essentially takes advantage both on effectiveness and efficiency.

6.4. Parameter analysis

Sampling point number on 3DV: Before inputting 3DV point set into PointNet++, farthest point sampling is executed first. To investigate the choice of sampling point number, we compare the performance of 3DV stream with the different sampling point number values on NTU RGB+D 120 dataset. The results are listed in Table 10. That is, 2048 can achieve the best performance on 3DV.

3DV voxel size: To investigate the choice of 3D voxel size, we compare the performance of 3DV stream with the different 3DV voxel sizes on NTU RGB+D 120 dataset. The results are listed in Table 11. Particularly, $35mm \times 35mm \times 35mm$ is the optimal 3DV voxel size.

6.5. Other issues

Running time: On the platform with CPU: Intel(R) Xeon(R) CPU E5-2690 v3 @ 2.6GHz (only using 1 core), and GPU: 1 Nvidia RTX 2080Ti, 3DV's overall online running time is 2.768s/video as detailed in Table 12. Particularly, 100 samples with the average length of 97.6 frames are randomly selected from NTU RGB+D 60 dataset for test.

Approximated temporal rank pooling: In our implementation, the approximated temporal rank pooling is used for 3DV extraction due to its high running efficiency. We compare it with the original one on N-UCLA data, using

Table 12. Item-wise time consumption of 3DV per video.

Unit	Item	Time (ms)	Unit	Item	Time (ms)
GPU	Human detection	231ms	CPU	3DV Pointlization	88ms
CPU	Point cloud voxelization	2107ms	GPU	PointNet++ forward	242ms
CPU	Temporal rank pooling	100ms		**Overall**	2768ms

Table 13. Performance comparison between the original and approximated temporal rank pooling for 3DV on N-UCLA dataset.

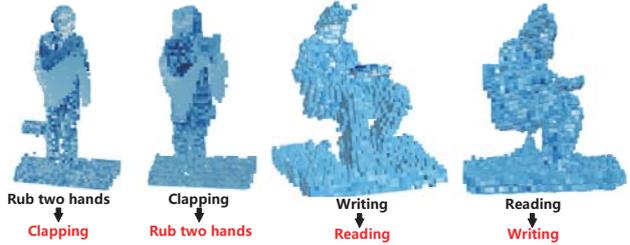| Temporal rank pooling methods	Accuracy	Time per sample
Original	**95.8**	1.12s
Approximated	95.3	**0.10s**

Figure 9. Some classification failure cases of 3DV. Ground-truth action label is shown in black, and the prediction is in red.

CPU. As shown in Table 13, the approximated temporal rank pooling runs much faster than the original one with the similar performance.

3DV failure cases: Some classification failure cases of 3DV are shown in Fig. 9. We find that, the failures tend to be caused by the tiny motion difference between the actions.

7. Conclusions

In this paper, 3DV is proposed as a novel and compact 3D motion representation for 3D action recognition. PointNet++ is applied to 3DV to conduct end-to-end feature learning. Accordingly, a multi-stream PointNet++ based network is also proposed to learn the 3D motion and depth appearance feature jointly to better characterize 3D actions. The experiments on 4 challenging datasets demonstrate the superiority of our proposition both for the large-scale and small-scale test cases. How to further enhance 3DV's discriminative power is what we mainly concern about in future, especially towards the tiny motion patterns.

References

[1] Tali Basha, Yael Moses, and Nahum Kiryati. Multi-view scene flow estimation: A view centered variational approach. *International Journal of Computer Vision*, 101(1):6–21, 2013. 1

[2] Hakan Bilen, Basura Fernando, Efstratios Gavves, and Andrea Vedaldi. Action recognition with dynamic image networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12):2799–2813, 2017. 1, 3, 4

[3] Hakan Bilen, Basura Fernando, Efstratios Gavves, Andrea Vedaldi, and Stephen Gould. Dynamic image networks for action recognition. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3034–3042, 2016. 1, 3

[4] Carlos Caetano, Jessica Sena, François Brémond, Jefersson A dos Santos, and William Robson Schwartz. Skelemotion: A new representation of skeleton joint dequences based on motion information for 3d action recognition. *arXiv preprint arXiv:1907.13025*, 2019. 6

[5] Basura Fernando, Efstratios Gavves, José Oramas, Amir Ghodrati, and Tinne Tuytelaars. Rank pooling for action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(4):773–787, 2016. 1, 2, 3

[6] Basura Fernando, Efstratios Gavves, Jose M Oramas, Amir Ghodrati, and Tinne Tuytelaars. Modeling video evolution for action recognition. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5378–5387, 2015. 1, 2, 3

[7] Liuhao Ge, Hui Liang, Junsong Yuan, and Daniel Thalmann. 3d convolutional neural networks for efficient and robust hand pose estimation from single depth images. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1991–2000, 2017. 2, 4

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. 2, 3

[9] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[10] Junnan Li, Yongkang Wong, Qi Zhao, and Mohan Kankanhalli. Unsupervised learning of view-invariant action representations. In *Porc. Advances in Neural Information Processing Systems (NIPS)*, pages 1262–1272, 2018. 1, 2, 6, 7

[11] Maosen Li, Siheng Chen, Xu Chen, Ya Zhang, Yanfeng Wang, and Qi Tian. Actional-structural graph convolutional networks for skeleton-based action recognition. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3595–3603, 2019. 1, 2, 6, 7

[12] Wanqing Li, Zhengyou Zhang, and Zicheng Liu. Expandable data-driven graphical modeling of human actions based on salient postures. *IEEE Transactions on Circuits and Systems for Video Technology*, 18(11):1499–1510, 2008. 2

[13] Jun Liu, Amir Shahroudy, Mauricio Lisboa Perez, Gang Wang, Ling-Yu Duan, and Alex Kot Chichung. NTU RGB+D 120: A large-scale benchmark for 3d human activity understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019. 1, 2, 6

[14] Jun Liu, Amir Shahroudy, Gang Wang, Ling-Yu Duan, and Alex Kot Chichung. Skeleton-based online action prediction using scale selection network. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019. 6

[15] Jun Liu, Amir Shahroudy, Dong Xu, and Gang Wang. Spatio-temporal LSTM with trust gates for 3d human action recognition. In *Proc. European Conference on Computer Vision (ECCV)*, pages 816–833, 2016. 2

[16] Jun Liu, Gang Wang, Ling-Yu Duan, Kamila Abdiyeva, and Alex C Kot. Skeleton-based human action recognition with global context-aware attention lstm networks. *IEEE Transactions on Image Processing*, 27(4):1586–1599, 2017. 1, 2, 6, 7

[17] Jun Liu, Gang Wang, Ping Hu, Ling-Yu Duan, and Alex C Kot. Global context-aware attention lstm networks for 3d action recognition. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1647–1656, 2017. 1, 2, 6, 7

[18] Mengyuan Liu and Junsong Yuan. Recognizing human actions as the evolution of pose estimation maps. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1159–1168, 2018. 6

[19] Cewu Lu, Jiaya Jia, and Chi-Keung Tang. Range-sample depth feature for action recognition. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 772–779, 2014. 3

[20] Daniel Maturana and Sebastian Scherer. Voxnet: A 3d convolutional neural network for real-time object recognition. In *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 922–928, 2015. 2, 4

[21] Gyeongsik Moon, Ju Yong Chang, and Kyoung Mu Lee. V2V-PoseNet: Voxel-to-voxel prediction network for accurate 3d hand and human pose estimation from a single depth map. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5079–5088, 2018. 1, 2, 3, 4

[22] Eshed Ohn-Bar and Mohan Trivedi. Joint angles similarities and hog2 for action recognition. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPR)*, pages 465–470, 2013. 1, 2, 6, 7

[23] Omar Oreifej and Zicheng Liu. HON4D: Histogram of oriented 4d normals for activity recognition from depth sequences. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 716–723, 2013. 2, 3, 6, 7

[24] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. PointNet: Deep learning on point sets for 3d classification and segmentation. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 652–660, 2017. 4, 5

[25] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. PointNet++: Deep hierarchical feature learning on point sets in a metric space. In *Proc. Advances in Neural Information Processing Systems (NIPS)*, pages 5099–5108, 2017. 2, 3, 4, 5

[26] Hossein Rahmani, Arif Mahmood, Du Huynh, and Ajmal Mian. Histogram of oriented principal components for cross-view action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(12):2430–2443, 2016. 2, 6

[27] Hossein Rahmani, Arif Mahmood, Du Q Huynh, and Ajmal Mian. Hopc: Histogram of oriented principal components of 3d pointclouds for action recognition. In *Proc. European Conference on Computer Vision (ECCV)*, pages 742–757. Springer, 2014. 7

[28] Hossein Rahmani and Ajmal Mian. 3d action recognition from novel viewpoints. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 1506–1515, 2016. 1

[29] Cen Rao, Alper Yilmaz, and Mubarak Shah. View-invariant representation and recognition of actions. *International Journal of Computer Vision*, 50(2):203–226, 2002. 2

[30] Joseph Redmon and Ali Farhadi. YOLOv3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018. 4

[31] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Proc. Advances in Neural Information Processing Systems (NIPS)*, pages 91–99, 2015. 4

[32] Zhou Ren, Junsong Yuan, Jingjing Meng, and Zhengyou Zhang. Robust part-based hand gesture recognition using kinect sensor. *IEEE Transactions on Multimedia*, 15(5):1110–1120, 2013. 1

[33] Amir Shahroudy, Jun Liu, Tian-Tsong Ng, and Gang Wang. NTU RGB+D: A large scale dataset for 3d human activity analysis. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1010–1019, 2016. 1, 2, 4, 6

[34] Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. Skeleton-based action recognition with directed graph neural networks. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7912–7921, 2019. 1, 2, 6, 7

[35] Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. Two-stream adaptive graph convolutional networks for skeleton-based action recognition. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12026–12035, 2019. 1, 2, 6, 7

[36] Chenyang Si, Wentao Chen, Wei Wang, Liang Wang, and Tieniu Tan. An attention enhanced graph convolutional lstm network for skeleton-based action recognition. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1227–1236, 2019. 1, 2, 6, 7

[37] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *Proc. Advances in Neural Information Processing Systems (NIPS)*, pages 568–576, 2014. 2, 5

[38] Alex J Smola and Bernhard Schölkopf. A tutorial on support vector regression. *Statistics and Computing*, 14(3):199–222, 2004. 4

[39] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, pages 4489–4497, 2015. 8

[40] Jiang Wang, Zicheng Liu, Jan Chorowski, Zhuoyuan Chen, and Ying Wu. Robust 3d action recognition with random occupancy patterns. In *Proc. European Conference on Computer Vision (ECCV)*, pages 872–885, 2012. 2

[41] Jiang Wang, Xiaohan Nie, Yin Xia, Ying Wu, and Song-Chun Zhu. Cross-view action modeling, learning and recognition. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2649–2656, 2014. 2, 6, 7

[42] Pichao Wang, Wanqing Li, Zhimin Gao, Chang Tang, and Philip O. Ogunbona. Depth pooling based large-scale 3-d action recognition with convolutional neural networks. *IEEE Transactions on Multimedia*, 20(5):1–1, 2018. 1, 2, 3, 6, 7

[43] Pichao Wang, Wanqing Li, Zhimin Gao, Chang Tang, Jing Zhang, and Philip Ogunbona. Convnets-based action recognition from depth maps through virtual cameras and pseudo-coloring. In *Proc. ACM International Conference on Multimedia (ACM MM)*, pages 1119–1122, 2015. 3

[44] Pichao Wang, Wanqing Li, Zhimin Gao, Jing Zhang, Chang Tang, and Philip O Ogunbona. Action recognition from depth maps using deep convolutional neural networks. *IEEE Transactions on Human-Machine Systems*, 46(4):498–509, 2015. 3

[45] Pichao Wang, Wanqing Li, Philip Ogunbona, Jun Wan, and Sergio Escalera. Rgb-d-based human motion recognition with deep learning: A survey. *Computer Vision and Image Understanding*, 171:118–139, 2018. 1

[46] Yang Xiao, Jun Chen, Yancheng Wang, Zhiguo Cao, Joey Tianyi Zhou, and Xiang Bai. Action recognition for depth video using multi-view dynamic images. *Information Sciences*, 480:287–304, 2019. 1, 2, 3, 4, 6, 7

[47] Fu Xiong, Boshen Zhang, Yang Xiao, Zhiguo Cao, Taidong Yu, Joey Tianyi Zhou, and Junsong Yuan. A2J: Anchor-to-joint regression network for 3d articulated pose estimation from a single depth image. In *Proc. IEEE Internation Conference on Computer Vision (ICCV)*, 2019. 1, 2

[48] Xiaodong Yang and YingLi Tian. Super normal vector for activity recognition using depth sequences. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 804–811, 2014. 1, 2, 6, 7

[49] Xiaodong Yang, Chenyang Zhang, and YingLi Tian. Recognizing actions using depth motion maps-based histograms of oriented gradients. In *Proc. ACM International Conference on Multimedia (ACM MM)*, pages 1057–1060, 2012. 3

[50] Le Zhang, Zenglin Shi, Ming-Ming Cheng, Yun Liu, Jia-Wang Bian, Joey Tianyi Zhou, Guoyan Zheng, and Zeng Zeng. Nonlinear regression via deep negative correlation learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019. 3

[51] Pengfei Zhang, Cuiling Lan, Junliang Xing, Wenjun Zeng, Jianru Xue, and Nanning Zheng. View adaptive neural networks for high performance skeleton-based human action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019. 1, 2, 6, 7

[52] Zhengyou Zhang. Microsoft kinect sensor and its effect. *IEEE Multimedia*, 19(2):4–10, 2012. 1