Depth Pooling Based Large-Scale 3-D Action Recognition with Convolutional Neural Networks

Pichao Wang
University of Wollongong, pw212@uowmail.edu.au

Wanqing Li
University of Wollongong, wanqing@uow.edu.au

Zhimin Gao
University of Wollongong, zg126@uowmail.edu.au

Chang Tang
Tianjin University, University of Wollongong

Philip O. Ogunbona
University of Wollongong, philipo@uow.edu.au

Recommended Citation

Depth Pooling Based Large-Scale 3-D Action Recognition with Convolutional Neural Networks

Abstract

1999-2012 IEEE. This paper proposes three simple, compact yet effective representations of depth sequences, referred to respectively as dynamic depth images (DDI), dynamic depth normal images (DDNI), and dynamic depth motion normal images (DDMNI), for both isolated and continuous action recognition. These dynamic images are constructed from a segmented sequence of depth maps using hierarchical bidirectional rank pooling to effectively capture the spatial-temporal information. Specifically, DDI exploits the dynamics of postures over time, and DDNI and DDMNI exploit the 3-D structural information captured by depth maps. Upon the proposed representations, a convolutional neural network (ConvNet)-based method is developed for action recognition. The image-based representations enable us to fine-tune the existing ConvNet models trained on image data without training a large number of parameters from scratch. The proposed method achieved the state-of-art results on three large datasets, namely, the large-scale continuous gesture recognition dataset (means the Jaccard index 0.4109), the large-scale isolated gesture recognition dataset (59.21%), and the NTU RGB+D dataset (87.08% cross-subject and 84.22% cross-view) even though only the depth modality was used.

Disciplines

Engineering | Science and Technology Studies

Depth Pooling Based Large-scale 3D Action Recognition with Convolutional Neural Networks

Pichao Wang, *Member, IEEE,* Wanqing Li, *Senior Member, IEEE,*
Zhimin Gao, Chang Tang, and Philip Ogunbona, *Senior Member, IEEE,*

Abstract—This paper proposes three simple, compact yet effective representations of depth sequences, referred to respectively as Dynamic Depth Images (DDI), Dynamic Depth Normal Images (DDNI) and Dynamic Depth Motion Normal Images (DDMNI), for both isolated and continuous action recognition. These dynamic images are constructed from a segmented sequence of depth maps using hierarchical bidirectional rank pooling to effectively capture the spatial-temporal information. Specifically, DDI exploits the dynamics of postures over time and DDNI and DDMNI exploit the 3D structural information captured by depth maps. Upon the proposed representations, a ConvNet based method is developed for action recognition. The image-based representations enable us to fine-tune the existing Convolutional Neural Network (ConvNet) models trained on image data without training a large number of parameters from scratch. The proposed method achieved the state-of-art results on three large datasets, namely, the Large-scale Continuous Gesture Recognition Dataset (means Jaccard index 0.4109), the Large-scale Isolated Gesture Recognition Dataset (59.21%), and the NTU RGB+D Dataset (87.08% cross-subject and 84.22% cross-view) even though only the depth modality was used.

Index Terms—Large-scale, Depth, Action Recognition, Convolutional Neural Networks.

I. INTRODUCTION

MICROSOFT Kinect Sensors provide an affordable technology to capture depth maps and RGB images in real-time. Recognition of human actions from depth data is one of the most active research topics in multimedia signal processing. Compared to traditional images, depth maps offer better geometric cues and less sensitivity to illumination changes for action recognition. Since the first work of such a type [1] reported in 2010, many methods [2]–[4] have been proposed based on specific hand-crafted feature descriptors extracted from depth. Most previous action recognition methods primarily focus on modeling spatial configuration based on hand-designed features and adopt dynamic time warpings (DTWs), Fourier temporal pyramid (FTP) or hidden Markov models (HMMs) to represent temporal information. However, these hand-crafted features are always shallow and dataset-dependent. Recently, recurrent neural networks (RNNs) [5]–[7] have also been adopted for action recognition. RNNs tend to overemphasize the temporal information especially when

there is insufficient training data, leading to overfitting. Up to now, it remains unclear how video could be effectively represented and fed to deep neural networks for classification. For instance, a video can be considered as a sequence of still images [8] with some form of temporal smoothness [9], [10] and fed into a convolutional network (ConvNet). It is also possible to extend the ConvNet to a third temporal dimension [11], [12] by replacing 2D filters with their 3D equivalent. Another possibility is to regard the video as the output of a neural network encoder [13]. Lastly, one can treat the video as a sequence of images that can be fed to a RNN [6], [14], [15].

Inspired by the promising performance of recently introduced rank pooling machine [16]–[18] on RGB videos, we propose to extend the rank pooling method to encode depth map sequences into dynamic images. However, due to the insensitivity of depth to motion, the generated dynamic images by directly applying rank pooling method [17] are not enough to exploit rich spatial-temporal and structural information. In this paper, to take full use of depth advantages, namely, being insensitive to illumination changes and providing the 3D surface information of the objects, three image-based depth representations, referred to as respectively Dynamic Depth Image (DDI), Dynamic Depth Normal Image (DDNI) and Dynamic Depth Motion Normal Image (DDMNI), are proposed and they are constructed by applying rank pooling hierarchically and bidirectionally to a segmented sequence of depth maps. The hierarchical bidirectional rank pooling overcomes the drawbacks of the conventional ranking method [17], that is, it eliminates the bias to the past frames in the conventional rank pooling and exploits both the higher order and non-linear dynamics of depth data simultaneously. Specifically, DDIs mainly exploit the dynamic of postures, and DDNI and DDMNI, built upon norm vectors, effectively exploit the 3D structural information captured by depth maps. Such representations make it possible to use a standard ConvNet architecture to learn discriminative "dynamic" features and to tune ConvNet models trained from image data on small annotated depth data without training the parameters of the ConvNet from scratch. For instance, the large-scale isolated gesture recognition challenge [19] has on average only 144 video clips per class compared to 1200 images per class in ImageNet. Experimental results have shown that the three representations can improve the accuracy of action recognition substantially compared to the state-of-the-art methods.

Part of the work [20], [21] was reported in the ChaLearn

Looking at People (LAP) challenge [1][2] in conjunction with ICPR 2016. The key novelty of the method is to encode the geometric, motion and structural information of human actions into three dynamic depth images though ranking pooling. Compared to the two workshop papers [20], [21], the extension includes (1) the bidirectional rank pooling in the method of [20] is replaced with *hierarchical* and *bidirectional* rank pooling to capture both high order and non-linear dynamics more effectively; (2) The extended method is applied to continuous action recognition; (3) the method is also evaluated on the large and challenging NTU RGB-D dataset in addition to the ChaLearn LAP datasets and state-of-the-art results are achieved on the three large datasets using depth modality only; and (4) more analysis are presented in this paper.

The rest of this paper is organized as follows. Section II briefly reviews the related works. Details of the proposed method are described in Section III. Experimental results and analysis are presented in Section IV. Section V concludes the paper.

II. RELATED WORK

In this section, recent works related to the paper, including action segmentation, action recognition based on depth and deep learning-based action recognition, are briefly reviewed. Comprehensive reviews can be found in [22]–[26].

A. Action Segmentation

Most existing works on action recognition reported to date focus on the classification of individual actions by assuming that instances of individual actions have been isolated or segmented from a video or a stream of depth maps before the classification. In the case of continuous recognition, the input stream usually contains unknown numbers, unknown orders and unknown boundaries of actions and both segmentation and recognition have to be solved at the same time. There are three common approaches to such continuous recognition. The first approach is to tackle the segmentation and classification of actions separately and sequentially. The key advantages of this approach are that different features can be used for segmentation and classification and existing classification methods can be leveraged. The disadvantages are that both segmentation and classification could be the bottleneck of the systems and they can hardly be optimized together. The second approach is to apply classification to a sliding temporal window and aggregate the window based classification to achieve final segmentation and classification. One of the key challenges in this approach is the difficulty of setting the size of sliding window because durations of different gestures or actions can vary significantly. The third approach is to perform the segmentation and classification simultaneously. With respect to action segmentation, the popular and widely used method employs dynamic time warping (DTW) to determine the delimiting frames of individual actions [27]–[29]. Difference

images are first obtained by subtracting two consecutive grey-scale images and each difference image is partitioned into a grid of 3×3 cells. Each cell is then represented by the average value of pixels within this cell. The matrix of the cells in a difference image is flattened as a vector called motion feature and calculated for each frame in the video, excluding the final frame. This results in a $9 \times (K-1)$ matrix of motion features for a video with K frames. The motion feature matrix is extracted from both test video and training video which consists of multiple actions. The two matrices are treated as two temporal sequences with each motion feature as a feature vector at an instant of time. The distance between two feature vectors is defined as the negative Euclidean distance and a matrix containing DTW distances (measuring similarity between two temporal sequences) between the two sequences is then calculated and analyzed by Viterbi algorithm [30] to segment the actions. Another category of action segmentation methods from a multi-action video is based on appearance. Upon the general assumption that the *start* and *end* frames of adjacent actions are similar, correlation coefficients [31] and K-nearest neighbour algorithm with histogram of oriented gradient (HOG) [32] are used to identify the *start* and *end* frames of actions. Jiang et al. [33] proposed a method based on quantity of movement (QOM) by assuming the same *start* pose among different actions. Candidate delimiting frames are chosen based on the global QOM. After a refining stage which employs a sliding window to keep the frame with minimum QOM in each windowing session, the *start* and *end* frames are assumed to be the remaining frames. This paper adopts the QOM based method and its details will be presented in Section III-A.

B. Depth Based Action Recognition

Many methods have been reported to date on depth map-based action recognition. Li et al. [1] sampled points from a depth map to obtain a bag of 3D points to encode spatial information and employ an expandable graphical model to encode temporal information [34]. Yang et al. [35] stacked differences between projected depth maps as a depth motion map (DMM) and then used HOG to extract relevant features from the DMM. This method transforms the problem of action recognition from spatio-temporal space to spatial space. In [36], a feature called Histogram of Oriented 4D Normals (HON4D) was proposed; surface normal is extended to 4D space and quantized by regular polychorons. Following this method, Yang and Tian [37] cluster hypersurface normals and form the polynormal which can be used to jointly capture the local motion and geometry information. Super Normal Vector (SNV) is generated by aggregating the low-level polynormals. In [38], a fast binary range-sample feature was proposed based on a test statistic by carefully designing the sampling scheme to exclude most pixels that fall into the background and to incorporate spatio-temporal cues. All of previous works are based on hand-crafted features which is shallow and dataset-dependent. With the development of deep learning, Wang et al. [39], [40] applied ConvNets to depth action recognition based on the variants of DMM [35], which is sensitive to

---

[1] http://gesture.chalearn.org/icpr2016_contest

[2] The team won the second place in the Isolated Gesture Recognition, third place in the Continuous Gesture Recognition, and two best paper awards
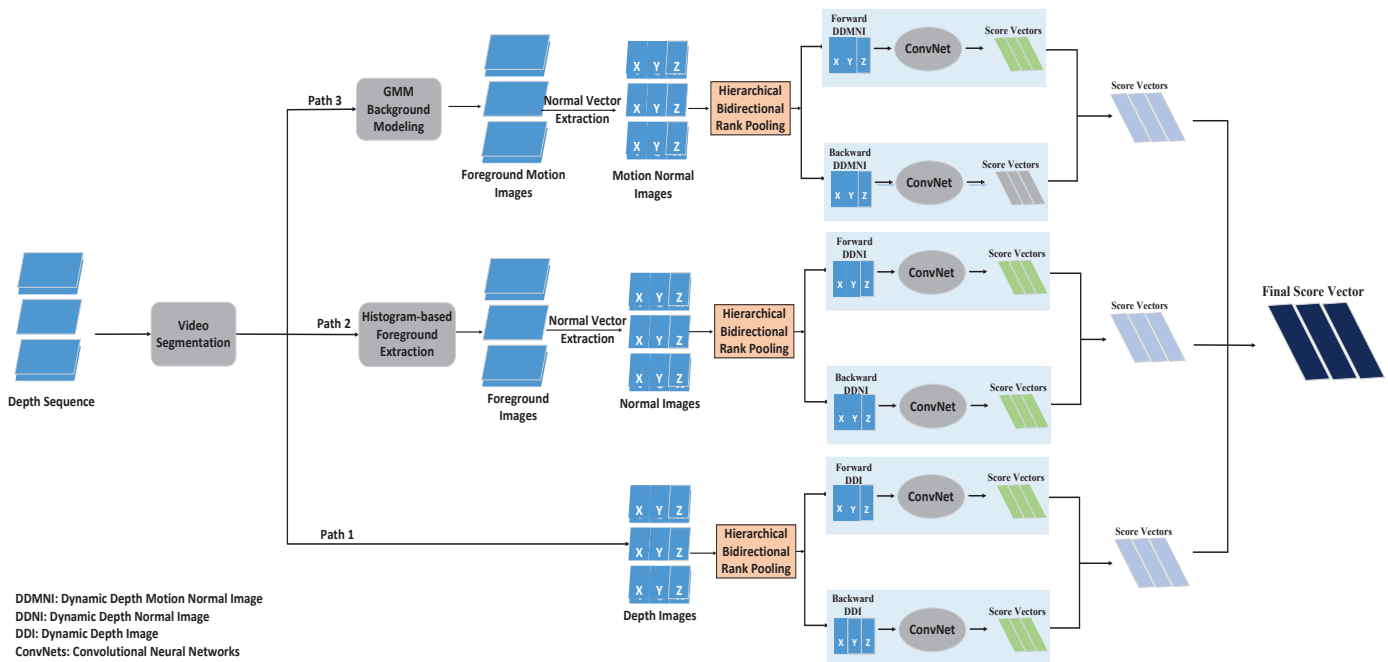
Fig. 1. The framework of the proposed method.

noise and cannot work well in fine-grained action recognition. Wu. et al. [41] adopted a 3D ConvNet to extract features from depth data, which requires a large amount of training data to achieve the best performance. Wang et al. [42] and Hou et al. [43] proposed the concept of structured images for depth based action recognition, but the proposed methods need the help of skeleton data to locate the position of human body, parts and joints.

C. Deep Leaning Based Motion Recognition

Existing deep learning approaches can generally be divided into four categories based on how the video is represented and fed to a deep neural network. The first category views a video either as a set of still images [8] or as a short and smooth transition between similar frames [9], and each color channel of the images is fed to one channel of a ConvNet. Although obviously suboptimal, considering the video as a bag of static frames performs reasonably well. The second category is to represent a video as a volume and extends ConvNet to a third, temporal dimension [11], [12] replacing 2D filters with their 3D equivalents. So far, this approach has produced little benefit, probably due to the lack of annotated training data. The third category is to treat a video as a sequence of images and feed the sequence to a RNN [5]–[7], [14]. A RNN is typically considered as memory cells, which are sensitive to both short as well as long term patterns. It parses the video frames sequentially and encode the frame-level information in the memory. However, using RNNs did not give an improvement over temporal pooling of convolutional features [8] or over hand-crafted features. The last category is to represent a video in one or multiple compact images and adopt available trained ConvNet architectures for fine-

tuning [17], [39], [40], [44]–[46]. This category has achieved state-of-the-art results of action recognition on many RGB and depth/skeleton datasets. The proposed method in this paper falls into the last category.

III. PROPOSED METHOD

The proposed method consists of four stages: action segmentation, construction of the three sets of dynamic images, ConvNets training and score fusion for classification. The framework is illustrated in Fig. 1. Given a sequence of depth maps consisting of multiple actions, the *start* and *end* frames of each action are identified based on quantity of movement (QOM) [33]. Then, three sets of dynamic images are constructed for each action segment and used as the input to six ConvNets for multiple score fusion-based classification. Details are presented in the rest of this section.

A. Action Segmentation

Previous works on action recognition mainly focus on the classification of segmented actions. In the case of continuous recognition, both segmentation and recognition have to be solved. This paper tackles the segmentation and classification of actions separately and sequentially.

Given a sequence of depth maps that contains multiple actions, each frame has the relevant movement with respect to its adjacent frame and the first frame. The *start* and *end* frames of each action is detected based on quantity of movement (QOM) [33] by assuming that all actions starts from a similar pose. For a multi-action depth sequence I, the QOM for frame t is defined as

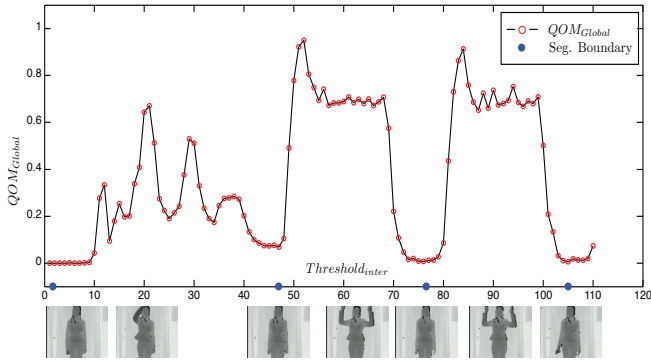$$QOM_{Global}(I, t) = \sum_{m,n} \psi(I_t(m, n), I_1(m, n)), \quad (1)$$

Fig. 2. An example of illustrating the inter-action segmentation results. Figure from [33].

where (m, n) is the pixel location and the indicator function $\psi(x, y)$ is defined as

$$\psi(x, y) = \begin{cases} 1 & if\, |x - y| \geqslant Threshold_{QOM}; \\ 0 & \text{otherwise} \end{cases}$$

$Threshold_{QOM} = 60$ is a threshold predefined emperically. A set of frame indices of candidate delimiting frames is initialized by choosing frames with lower global QOMs than a $threshold_{inter}$. The $threshold_{inter}$ is calculated by adding the mean to twice the standard deviation of the global QOMs extracted from first and last 12.5% of the average action sequence length L calculated from the training actions. A sliding window of size $\frac{L}{2}$ is then used to refine the candidate set. In each windowing session only the frame with a minimum global QOM is considered to be the boundary frame of two consecutive actions. After the refinement, the remaining frames are expected to be the delimiting frames of actions, as shown in Fig. 2.

B. Construction of Dynamic Images

The three sets of dynamic images, Dynamic Depth Images (DDIs), Dynamic Depth Normal Images (DDNIs) and Dynamic Depth Motion Normal Images (DDMNIs) are constructed from a segmented sequence of depth maps through hierarchical bidirectional rank pooling. They aim to exploit shape, motion and structural information captured by a depth sequence at different spatial and temporal scales. To this end, the conventional ranking pooling [17] is extended to the hierarchical bidirectional rank pooling.

The conventional rank pooling [17] aggregates spatio-temporal information from one video sequence into one dynamic image. It defines a function that maps a video clip into one feature vector [17]. A *rank pooling function* is formally defined as follows.

Rank Pooling. Let a depth map sequence with k frames be represented as $< d_1, d_2, ..., d_t, ..., d_k >$, where d_t is the average of depth features over the frames up to t-timestamp. At each time t, a score $r_t = \omega^T \cdot d_t$ is assigned. The score satisfies $r_i > r_j \Longleftrightarrow i > j$. In general, more recent frames are associated with larger scores. The process of rank pooling is to find ω^* that satisfies the following objective function:

$$\arg \min_{\omega} \frac{1}{2} \parallel \omega \parallel^2 + \lambda \sum_{i > j} \xi_{ij}, \qquad (2)$$
$$s.t. \quad \omega^T \cdot (d_i - d_j) \geq 1 - \xi_{ij}, \xi_{ij} \geq 0$$

where ξ_{ij} is a slack variable. Since the score r_i assigned to frame i is often defined as the order of the frame in the sequence, ω^* aggregates information from all of the frames in the sequence and can be used as a descriptor of the sequence. In this paper, the rank pooling is directly applied on the pixels of depth maps and the ω^* is of the same size as depth maps and forms a dynamic depth image (DDI).

However, the conventional ranking pooling method has two drawbacks. Firstly, it treats a video sequence in a single temporal scale which is usually too shallow [18]. Secondly, since in rank pooling the averaged feature up to time t is used to classify frame t, the pooled feature is biased towards beginning frames of a depth sequence, hence, frames at the beginning has more influence to ω^*. This is not justifiable in action recognition as there is no prior knowledge on which frames are more important than other frames.

To overcome the first drawback, it is proposed that the ranking pooling is applied recursively to sliding windows over several *rank pooling layer*. This recursive process can effectively explore the high-order and non-linear dynamics of a depth sequence. The *rank pooling layer* is defined as follows:

Definition 2 (Rank Pooling Layer). Let $I^{(l)} = \left\langle i_1^{(l)}, ..., i_n^{(l)} \right\rangle$ denote the input sequence/subsequence that contains n frames; M_l is the window size; and S_l is a stride in the l_{th} layer. The subsequences of $I^{(l)}$ can be defined as $I_t^{(l)} = \left\langle i_t^{(l)}, ..., i_{t+M_l-1}^{(l)} \right\rangle$, where $t \in \{1, S_l + 1, 2S_l + 1, ...\}$. By applying the *rank pooling function* on the subsequences respectively, the outputs of l_{th} layer constitute the $(l + 1)_{th}$ layer, which can be represented as $I^{(l+1)} = \left\langle ..., i_t^{(l+1)}, ... \right\rangle$.

$I^{(l)}$ to $I^{(l+1)}$ forms one layer of temporal hierarchy. Multiple *rank pooling layers* can be stacked together to make the pooling higher-order. In this case, each successive layer obtains the dynamics of the previous layer. Figure 3 shows a hierarchical rank pooling with two layers. For the first layer, the sequence is the input depth sequence, thus $l = 1$, $n = 5$; for the second layer, $l = 2$, $n = 3$. By adjusting the window size and stride of each layer, the hierarchical rank pooling can explore high-order and non-linear dynamics effectively.

To address the second drawback, it is proposed to to apply the rank pooling bidirectionally.

Bidirectional Rank Pooling is to apply the rank pooling forward and backward to a sequence of depth maps. In the forward rank pooling, the r_i is defined in the same order as the time-stamps of the frames. In the backward rank pooling, r_i is defined in the reverse order of the time-stamps of the frames. When bidirectional rank pooling is applied to a sequence of depth maps, two DDIs, forward DDI and backward DDI, are generated.

By employing the hierarchical and bidirectional pooling together, the hierarchical bidirectional rank pooling exploits the dynamics of a depth sequence at different temporal scales and bidirectionally at the same time. It has been empirically
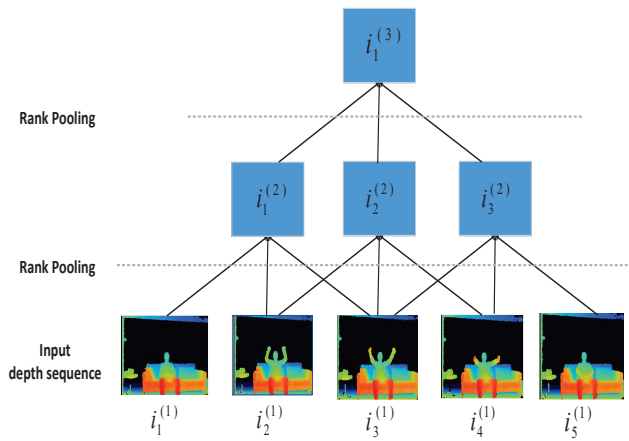
Fig. 3. Illustration of a two layered rank pooling with window size three (M_l = 3) and stride one (S_l = 1).

Fig. 4. Samples of generated forward and backward DIs [17], DDIs, DDNIs and DDMNIs for gesture Mudra1/Ardhapataka.

observed that, for most actions with relatively short durations, two layers of bidirectional rank pooling is sufficient.

1) Construction of DDI: Given a segmented sequence of depth maps, the hierarchical bidirectional rank pooling method described above is employed directly on the depth pixels to generate two dynamic depth images (DDIs), forward DDI and backward DDI. Even though rank pooling method exploits the evolution of videos and aims to encode both the spatial and motion information into one image, it is likely to lose much motion information due to the insensitivity of depth pixels to motion. As shown in Fig. 4, DDIs effectively capture the posture information, similar to key poses. Moreover, compared with the dynamic images (DIs [17]), the DDIs are more effective, without having interfering texture on the body.

2) Construction of DDNI: Depth images well represent the geometry of surfaces in the scene, and norm vectors is sensitive to motion of depth pixels. In order to simultaneously exploit the spatial and motion information in depth sequences, it is proposed to extract normals from depth maps and construct the so-called DDNIs (dynamic depth normal images). For each depth map, a surface normal (n_x, n_y, n_z) is calculated at each pixel. Three channels (N_x, N_y, N_z), referred to as a Depth Normal Image, are generated from the normals, where (N_x, N_y, N_z) are respectively normal images of the three components (n_x, n_y, n_z). The sequence of each DNI goes through hierarchical bidirectional rank pooling to generate two DDNIs, one being the forward DDNI and the other is the backward DDNI.

To minimize the interference of the background, it is assumed that the background in the histogram of depth maps occupies the last peak representing far distances. Specifically, pixels whose depth values are greater than a threshold defined by the last peak of the depth histogram minus a fixed tolerance are considered as background and removed from the calculation of DDNIs by re-setting their depth values to zero. Through this simple process, most of the background can be removed and has much contribution to the DDNIs. Samples of DDNIs can be seen in Fig. 4.
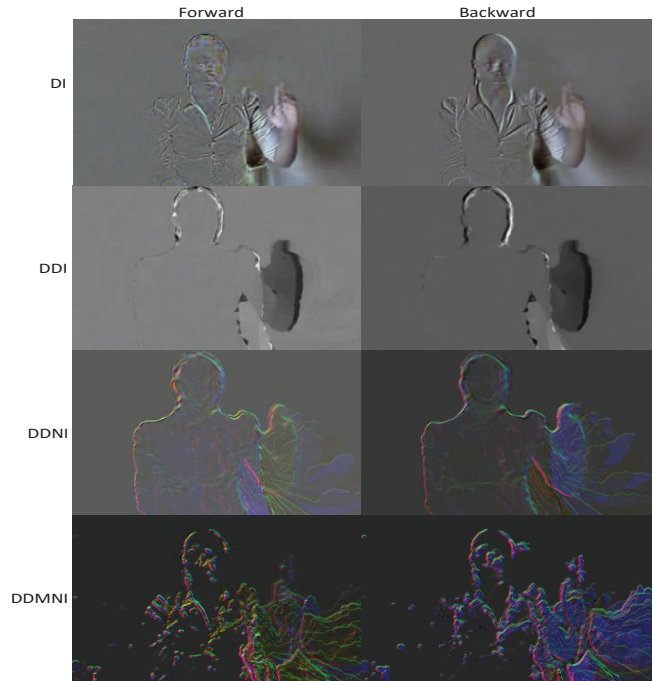
3) Construction of DDMNI: The purpose of constructing a DDMNI is to further exploit the motion in depth maps. Gaussian mixture model (GMM) is applied to depth sequences in order to detect moving foreground. The norm vectors are extracted from the moving foreground and Depth Normal Image is constructed from the norm vectors for each depth map. Hierarchical bidirectional rank pooling is applied to the Depth Norm Image sequence, and two DDMNIs, forward DDMNI and backward DDMNI, are generated, which capture the motion information specifically well (see the illustration in Fig. 4).

C. Network Training

After the construction of DDIs, DDNIs and DDMNIs, there are six dynamic images, as illustrated in Fig. 4, for each depth map sequence. Six ConvNets were trained on the six channels individually. VGG-16 [47] is adopted in this paper. The implementation is derived from the publicly available Caffe toolbox [48] based on three NVIDIA Tesla K40 GPU cards and one Pascal TITAN X.

The training procedure is similar to those in [47]. The network weights are learned using the mini-batch stochastic gradient descent with the momentum set to 0.9 and weight decay set to 0.0005. All hidden weight layers use the rectification (RELU) activation function. At each iteration, a mini-batch of 32 samples is constructed by sampling 256 shuffled training samples, and all the images are resized to 224 × 224. The learning rate is set to 10^{-3} for fine-tuning with pre-trained models on ILSVRC-2012, and then it is decreased according to a fixed schedule, which is kept the same for all training sets. The training undergoes 100 epochs and the

learning rate decreases every 30 epochs for each ConvNet. The dropout regularization ratio is set to 0.9 to reduce complex co-adaptations of neurons in nets.

D. Score Fusion for Classification

Given a test depth video sequence (sample), three pairs of dynamic images (DDIs, DDNIs, DDMNIs) are generated and fed into six different trained ConvNets. For each image pair, multiple-score fusion was used. The score vector output from the two pair of ConvNets are multiplied in an element-wise manner and the resultant score vectors are normalized using L_1 norm. The three normalized score vectors are then multiplied in an element-wise fashion and the max score in the resultant vector is assigned as the probability of the test sequence being the recognized class. The index of this max score corresponds to the recognized class label and expressed as follows:

$$label = Fin(max(v_1 \circ v_2 \circ v_3 \circ v_4 \circ v_5 \circ v_6)) \quad (3)$$

where v is a score vector, $\circ$ refers to element-wise multiplication and $Fin(\cdot)$ is a function to find the index of the element having the maximum score.

IV. Experiments

In this section, the Large-scale Isolated and Continuous Gesture Recognition datasets at the ChaLearn LAP challenge 2016 (ChaLearn LAP IsoGD Dataset and ChaLearn LAP ConGD Dataset) [49], the NTU RGB+D dataset [15], and the corresponding evaluation protocols and results & analysis are described. On ChaLearn LAP ConGD Dataset, action segmentation was first conducted to segment the continuous actions to isolated actions.

A. Settings

This section presents empirical studies to set the parameters for segmentation, hierarchical rank pooling, choice of networks and classifiers.

1) Segmentation: A sliding window is used to refine the candidate set and in each windowing session only location of the frame with a minimum global QOM is retained as the boundary of two gestures. The window size is determined by the average action sequence length L which is learned from the training samples. Theoretically, the sliding window size would affect the segmentation results. Several experiments were conducted using the DDI forward channel on the validation subset of the Chalearn LAP ConGD dataset to evaluate how the sliding window size would affect the performance. Results have shown that too large or too small sliding window size may affect adversely to the final recognition. But any value between an half or a quarter of the average gesture length worked well. Therefore, we set the window size to $\frac{L}{2}$ in all of our experiments.

2) Parameters for hierarchical rank pooling: For hierarchical rank pooling, generally speaking, the value of l is decided by the complexity of actions/gestures in the dataset, and the n is decided by the window size M_l and stride S_l. It has been empirically observed that, for most actions

with relatively short durations, two layers of bidirectional rank pooling is sufficient. To investigate the affects of these two parameters, we conducted several experiments on the ChaLearn LAP IsoGD dataset using the DDI forward channel. The results have demonstrated that the final performance is not sensitive to the parameters (l, M_1, S_l). Consequently, two layered hierarchical bidirectional rank pooling method is adopted with window size $M_l = 3$ and stride step $S_l = 1$ for all the experiments reported in the paper.

3) Choice of network architecture and classifier: Experimental studies were also conducted on other other CNNs include AlexNet, GoogleNet, VGG-16, VGG-19 and the new CNN architecture, MobileNetV2 [50]. Results have shown that VGG-16 (36.92%) and VGG-19 (36.96%) performed better than AlexNet (32.22%), GoogleNet (36.11%) and MobileNet V2 (34.56%) on the validation set. However, VGG-16 performed comparable to VGG-19. For the sake of computational cost, VGG-16 was chosen in our experiments.

Several experiments were also conducted by using kNN with $k = 1, 5, 15, 25$, linear SVM on the features from the convolutional layers of VGG-16 and the full VGG-16 on the DDI forward channel of the ChaLearn LAP IsoGD dataset. VGG-16 performed favourably to others.

B. ChaLearn LAP IsoGD Dataset

1) Description: The ChaLearn LAP IsoGD Dataset is derived from the ChaLearn Gesture Dataset (CGD) [51]. It includes 47933 RGB-D depth sequences, each RGB-D video representing one gesture instance. There are 249 gestures performed by 21 different individuals. The detailed information of this dataset are shown in Table I. In this paper, only depth maps are used to evaluate the performance of the proposed method.

2) Evaluation Protocol: The dataset is divided into training, validation and test sets. All three sets consist of samples of different subjects in order to ensure that the gestures of one subject in validation and test sets do not appear in the training set.

For the isolated gesture recognition challenge, recognition rate r is used as the evaluation criteria. The recognition rate is calculated as:

$$r = \frac{1}{n}\delta(p_l(i), t_l(i)) \quad (4)$$

where n is the number of samples; p_l is the predicted label; t_l is the ground truth; $\delta(j_1, j_2) = 1$, if $j_1 = j_2$, otherwise $\delta(j_1, j_2) = 0$.

3) Experimental Results: Table II shows the results of each channel. From the results we can see that DDIs achieved much better results than DDNIs and DDMNIs, and the reasons are as follows: first, the depth values are not the real depth, but they are normalized to [0,255], which distort the true 3D structure information and affects the norm vectors extraction; second, for storage benefit, the videos are compressed at a loss level, which leads to lots of compression blocking artifacts, which makes the extraction of moving foreground and norm vectors very noisy. Even though, the three kinds of dynamic images still provide complimentary information to each other.

TABLE I
INFORMATION OF THE CHALEARN LAP ISOGD DATASET.

Sets	# of labels	# of gestures	# of RGB videos	# of depth videos	# of subjects	label provided
Training	249	35878	35878	35878	17	Yes
Validation	249	5784	5784	5784	2	No
Testing	249	6271	6271	6271	2	No
All	249	47933	47933	47933	21	-

In addition, it can be seen that the bidirectional rank pooling exploits more useful information compared to one-way rank pooling [17], and by adopting multiply score fusion method, the accuracy is largely improved. Moreover, hierarchical rank pooling encodes the dynamic of depth sequences better compared with the conventional rank pooling method.

TABLE II
COMPARATIVE ACCURACY OF THE THREE SET OF DYNAMIC IMAGES ON THE VALIDATION SET OF THE CHALEARN LAP ISOGD DATASET. RP DENOTES CONVENTIONAL RANK POOLING; HRP REPRESENTS HIERARCHICAL RANK POOLING.

Method	Accuracy for RP	Accuracy for HRP
DDI (forward)	36.13%	36.92%
DDI (backward)	30.45%	31.24%
DDI (fusion)	37.52%	37.68%
DDNI (forward)	24.86%	25.02%
DDNI (backward)	24.58%	24.64%
DDNI (fusion)	29.26%	29.48%
DDMNI (forward)	24.81%	24.69%
DDMNI (backward)	23.14%	23.57%
DDMNI (fusion)	27.75%	27.89%
Fusion All	42.56%	43.72%

The results obtained by the proposed method on the validation and test sets are listed and compared with previous methods in Table III. These methods include MFSK combined 3D SMoSIFT [52] with (HOG, HOF and MBH) [53] descriptors. MFSK+DeepID further included Deep hidden IDentity (Deep ID) feature [54]. Thus, these two methods utilized not only hand-crafted features but also deep learning features. Moreover, they extracted features from RGB and depth separately, concatenated them together, and adopted Bag-of-Words (BoW) model as the final video representation. The other methods, WHDMM+SDI [17], [40], extracted features and conducted classification with ConvNets from depth and RGB individually and adopted multiply score fusion for final recognition. SFAM [55] adopted scene flow to extract features and encoded the flow vectors into action maps, which fused RGB and depth data from the onset of the process. C3D [56] applied 3D convolutional networks to both depth and RGB channels and fused them in a late fusion method. Pyramidal 3D CNN [57] adopted 3D convolutional networks to pyramid input to recognize gesture from both clip videos and entire video. It is noteworthy that the results of the proposed method have been obtained using a single modality viz., depth data, while all compared methods are based on RGB and depth modalities. From this table, we can see that the proposed method outperformed all of these recent works significantly, and illustrated its effectiveness.

Fig 5 visually compares the DDI, DDNI and DDMI and the SFAM-D, SFAM-S and SFAM-RP proposed in [55]. It can be seen that the proposed DDIs do not suffer the blocky artifacts

TABLE III
COMPARATIVE ACCURACY OF PROPOSED METHOD AND BASELINE METHODS ON THE CHALEARN LAP ISOGD DATASET.

Method	Set	Recognition rate r
MFSK [58]	Validation	18.65%
MFSK+DeepID [58]	Validation	18.23%
SDI [17]	Validation	20.83%
WHDMM [40]	Validation	25.10%
Scene Flow [55]	Validation	36.27%
Proposed Method	Validation	43.72%
MFSK [58]	Testing	24.19%
MFSK+DeepID [58]	Testing	23.67%
Pyramidal 3D CNN [57]	Testing	50.93%
C3D [56]	Testing	56.90%
Proposed Method	Testing	59.21%

TABLE IV
ACCURACIES OF THE PROPOSED METHOD AND PREVIOUS METHODS ON THE CHALEARN LAP CONGD DATASET.

Method	Set	Mean Jaccard Index $\overline{J_S}$
MFSK [58]	Validation	0.0918
MFSK+DeepID [58]	Validation	0.0902
Proposed Method	Validation	0.3905
MFSK [58]	Testing	0.1464
MFSK+DeepID [58]	Testing	0.1435
IDMM + ConvNet [21]	Testing	0.2655
C3D [59]	Testing	0.2692
Two-stream RNNs [60]	Testing	0.2869
Proposed Method	Testing	0.4109

as the SFAMs do and the motion information in DDIs becomes much clean.

C. ChaLearn LAP ConGD Dataset

1) Description: The ChaLearn LAP ConGD Dataset is also derived from the ChaLearn Gesture Dataset (CGD) [51]. It has 47933 RGB-D gesture instances in 22535 RGB-D gesture videos. Each RGB-D video may contain one or more gestures. There are 249 gestures performed by 21 different individuals. The detailed information of this dataset is shown in Table V. In this paper, only depth data was used in the proposed method.

2) Evaluation Protocol: The dataset is divided into training, validation and test sets by the challenge organizers. All three sets include data from different subjects and the gestures of one subject in validation and test sets do not appear in the training set. Jaccard index (the higher the better) is adopted to measure the performance. The Jaccard index measures the average relative overlap between true and predicted sequences of frames for a given gesture. For a sequence s, let $G_{s,i}$ and $P_{s,i}$ be binary indicator vectors for which 1-values correspond to frames in which the ith gesture label is being performed. The Jaccard Index for the ith class is defined for the sequence s as:

TABLE V
INFORMATION OF THE CHALEARN LAP CONGD DATASET.

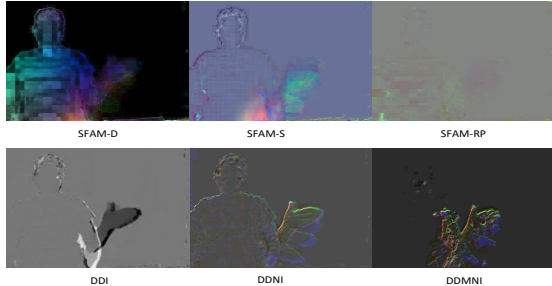| Sets	# of labels	# of gestures	# of RGB videos	# of depth videos	# of subjects	label provided	temporal segment provided
Training	249	30442	14134	14134	17	Yes	Yes
Validation	249	8889	4179	4179	2	No	No
Testing	249	8602	4042	4042	2	No	No
All	249	47933	22535	22535	21	-	-

Fig. 5. Visual comparion of the DDIs in the proposed method with SFAM [55].

$$J_{s,i} = \frac{G_{s,i} \bigcap P_{s,i}}{G_{s,i} \bigcup P_{s,i}}, \qquad (5)$$

where $G_{s,i}$ is the ground truth of the ith gesture label in sequence s, and $P_{s,i}$ is the prediction for the ith label in sequence s. When $G_{s,i}$ and $P_{s,i}$ are empty, $J_{(s,i)}$ is defined to be 0. Then for the sequence s with l_s true labels, the Jaccard Index J_s is calculated as:

$$J_s = \frac{1}{l_s} \sum_{i=1}^{L} J_{s,i}. \qquad (6)$$

For all testing sequences $S = s_1, ..., s_n$ with n gestures, the mean Jaccard Index $\overline{J_S}$ is used as the evaluation criteria and calculated as:

$$\overline{J_S} = \frac{1}{n} \sum_{j=1}^{n} J_{s_j}. \qquad (7)$$

3) Experimental Results: To measure the performance of the segmention, the Levenshtein distance as used in [61] between the segmented sequence and the ground truth segments is caculated as the metric. The QOM method have achived on average score of 71.44 on the ChaLearn LAP ConGD Dataset, which is higher than what the best result, i.e. 66.56 in [61], on a similar dataset probably due to the clear transitional pose between two gestures in the ChaLearn LAP ConGD Dataset.

The recognition results of the proposed method on the validation and test sets and their comparisons with the results of previous methods are shown in Table IV. MFSK and MFSK+DeepID [58] methods first segmented the continuous videos to segments and then extracted the features over the segments over two modalities to train and classify the actions. IDMM + ConvNet [21] also adopted the action segmentation method and then extracted one improved depth motion map using color coding method over the segments, and ConvNet was adopted to train and classify segmented actions. C3D [59]

applied 3D convolutional networks to RGB video and jointly learn the features and classifier. Two-stream RNNs [60] first adopted R-CNN to extract the hand and then conducted temporal segmentation. Two-stream RNNs were adopted to fuse multi-modality features for final recognition based on segments. The results showed that the proposed method outperformed all previous methods largely, even though only single modality, i.e. depth data, was used.

D. NTU RGB+D Dataset

1) Description: To our best knowledge, NTU RGB+D Dataset is currently the largest action recognition dataset in terms of training samples for each action. The 3D data is captured by Kinect v2 cameras. The dataset has more than 56 thousand sequences and 4 million frames, containing 60 actions performed by 40 subjects aged between 10 and 35. It consists of front view, two side views and left, right 45 degree views. This dataset is challenging due to large intra-class and viewpoint variations.

2) Evaluation Protocol: For fair comparison and evaluation, the same protocol as that in [15] was used. It has both cross-subject and cross-view evaluation. In the cross-subject evaluation, samples of subjects 1, 2, 4, 5, 8, 9, 13, 14, 15, 16, 17, 18, 19, 25, 27, 28, 31, 34, 35 and 38 were used as training and samples of the remaining subjects were reserved for testing. In the cross-view evaluation, samples taken by cameras 2 and 3 were used as training, while the testing set includes samples from camera 1.

3) Experimental Results: Similarly to LAP IsoGD Dataset, we conducted several experiments to compare the three set of dynamic images using conventional rank pooling method and the proposed hierarchical bidirectional rank pooling method. The comparisons are shown in Table VI. From the Table it can be seen that compared with DDIs, DDNIs achieved much better results than DDI in cross-subject setting, due to the sensitivity of norm vectors to motion over real depth values. This justified the effectiveness of proposed depth norm images for rank pooling. However, due to the sensitivity of norm vectors to motion and view angles, in cross-view setting, much worse results were achieved for DDNIs and DDMNIs. From the final fusion results we can see that the three set of dynamic images exploit the shape and motion at different levels, and provide complimentary information to each other.

Table VII lists the performance of the proposed method and those previous works. The proposed method was compared with some skeleton-based methods and depth-based methods previously reported on this dataset. We can see that the proposed method outperformed all the previous works significantly.

TABLE VI
COMPARATIVE ACCURACY OF THE THREE SET OF DYNAMIC IMAGES ON THE NTU RGB+D DATASET. RP DENOTES CONVENTIONAL RANK POOLING; HRP REPRESENTS HIERARCHICAL RANK POOLING.

Method	Cross subject Accuracy for RP	Cross subject Accuracy for HRP	Cross view Accuracy for RP	Cross view Accuracy for HRP
DDI (forward)	75.80%	76.10%	76.50%	76.75%
DDI (backward)	70.99%	75.45%	75.62%	75.48%
DDI (fusion)	81.66%	82.01%	81.53%	81.60%
DDNI (forward)	79.79%	79.98%	54.57%	55.01%
DDNI (backward)	81.46%	81.28%	56.61%	57.43%
DDNI (fusion)	84.18%	84.24%	61.07%	62.35%
DDMNI (forward)	68.89%	69.33%	50.01%	50.67%
DDMNI (backward)	70.04%	71.11%	49.53%	49.27%
DDMNI (fusion)	73.56%	74.27%	54.98%	55.09%
Fusion All	86.72%	87.08%	83.75%	84.22%

TABLE VII
ACCURACIES OF THE PROPOSED METHOD AND PREVIOUS METHODS ON NTU RGB+D DATASET.

Method	Modality	Cross Subject	Cross View
Lie Group [62]	Skeleton	50.08%	52.76%
HBRNN [5]	Skeleton	59.07%	63.97%
2 Layer RNN [15]	Skeleton	56.29%	64.09%
2 Layer LSTM [15]	Skeleton	60.69%	67.29%
Part-aware LSTM [15]	Skeleton	62.93%	70.27%
ST-LSTM [7]	Skeleton	65.20%	76.10%
ST-LSTM+ Trust Gate [7]	Skeleton	69.20%	77.70%
JTM [44]	Skeleton	73.40%	75.20%
HON4D [36]	Depth	30.56%	7.26%
SNV [37]	Depth	31.82%	13.61%
SLTEP [63]	Depth	58 .22%	–
Proposed Method	Depth	87.08%	84.22%

The confusion matrix of the cross subject setting is shown in Fig 6. We can see that the method distinguishes well the single subject based actions from the two subject based actions. Human-object interactions like "Make a phone call" may be confused with other actions with similar motion patterns such as "Eat meal/snack" and "brush teeth", because the objects that involved in the actions are hard to distinguish in the depth maps, which is the weakness of using depth modality alone. Other single subject based actions such as "Rub two hands together" and "Clapping" are also easily confused with each other due to the similarity of the two actions in motion patterns.

E. Discussions

1) Cross-dataset analysis: It is not feasible to train the model on a dataset such as NTU RGB+D and directly test on another dataset such as ChaLearn LAP IsoGD because of the different sets of actions in the two datasets. However, we conducted the following experiments. The model trained over the DDI forward channel on NTU RGB+D dataset was adopted as pre-trained model and fine-tuned on the ChaLearn LAP IsoGD dataset and vice verse. No noticeable improvement on the results have been demonstrated as shown in Table VIII. It is probably because both datasets are large enough to train a good model over the pre-trained model obtained from ImageNet.
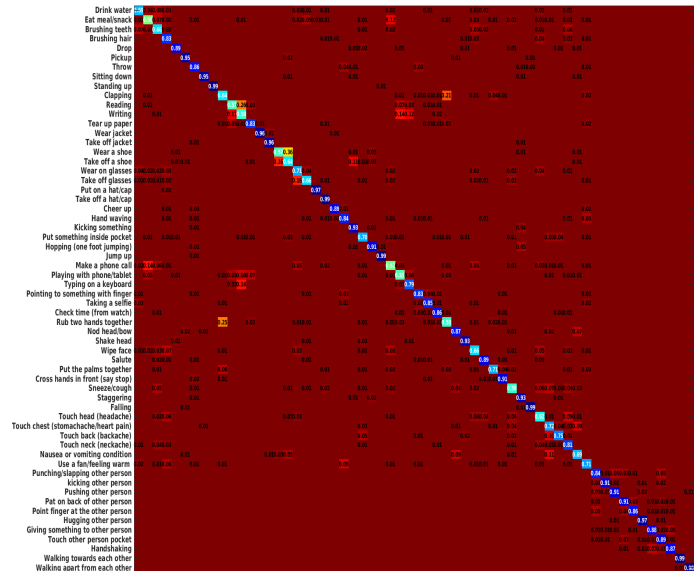

Fig. 6. The confusion matrix for the cross subject setting on NTU RGB+D dataset.

TABLE VIII
CROSS-DATASET TRAINING.

Setting	Accuracy
Fine-tune on ChaLearn LAP IsoGD from ImageNet model	36.92%
Fine-tune on NTU RGB+D from ImageNet model	76.10%
Pre-train on NTU RGB+D dataset(cross-subject setting) Fine-tune on ChaLearn LAP IsoGD dataset	36.72%
Pre-train on ChaLearn LAP IsoGD dataset Fine-tune on NTU RGB+D dataset (cross-subject setting)	75.95%

2) Different fusion methods: In this paper, we adopt product score fusion to fuse the classifications obtained from the six channels. The other two commonly used late score fusion methods are average and maximum score fusion. The comparisons among the three late score fusion methods are shown in Table IX. We can see that the product score fusion method achieved the best results on all the three large datasets. This verifies to a large extent that the six channels are likely to be statistically independent and product score fusion is the right choice. We also extracted the convnet features (the last fc 4096 feature) from the six models over ChaLearn LAP Iso dataset

and concatenate the six feature vectors into one long feature vector (4096*6) and using linear SVM to do clissification. The result on the validation set is 40.88% compared with 43.73% achieved by the proposed method.

TABLE IX
COMPARISON OF THREE DIFFERENT LATE SCORE FUSION METHODS ON THE SIX DATASETS.

Dataset	Score Fusion Method		
	Max	Average	Product
ChaLearn LAP IsoGD	40.38%	42.21%	43.72%
ChaLearn LAP ConGD	0.3591	0.3768	0.3905
NTU RGB+D (cross subject)	84.62%	85.77%	87.08%
NTU RGB+D (cross view)	82.47%	83.56%	84.22%

3) Channel fusion: We conducted several experiments over ChaLearn LAP Iso dataset by using two and three path fusion to analysis the channel fusion cases, and the results are shown in Table X. From the Table we can see that

TABLE X
COMPARATIVE RESULTS FOR TWO PATHS FUSION OVER CHALEARN LAP ISO VALIDATION DATASET.

Fusion Channels	Accuracy
DDI+DDNI	41.45%
DDI+DDMNI	39.12%
DDNI+DDMNI	33.64%
DDI+DDNI+DDMNI	43.72%

the fusion of DDI with either DDNI or DDMNI achieved better results than the fusion of DDNI and DDMNI, and the fusion of the three representation has led to the best results.

4) Computational cost: we compared the computational cost of the proposed method in the test phase without any code optimization with the computational cost of the RGB+D-based methods MFSK+DeepID [58], SFAM [55], depth-based method WHDMM [40] on ChaLearn LAP IsoGD dataset. The results are reported as follows Though the proposed method

TABLE XI
CCOMPUTATIONAL COST COMPARISON OF THE PROPOSED METHOD (WITHOUT ANY CODE OPTIMIZATION) WITH PREVIOUS METHODS ON THE CHALEAN LAP ISOGD DATASET. CPU TIME REFERS THE COMPUTATION OF REPRESENTATION (DDI/DDNI/DDMNI) AND THE GPU TIME REPRESENTS THE TIME TAKEN BY THE CONVNET FEATURE EXTRACTION. FOR EACH PATH IN THE PROPOSED METHOD THE COMPUTATINAL TIME IS ROUGHLY SAME DUE TO THE FAST PREPROCESSING MODULE.

Method	Computational Cost
MFSK+DeepID [58]	41.00s(CPU)++00.00ms(GPU)
SFAM [55]	6.30s(CPU)+31.00ms(GPU)
WHDMM [40]	0.60s(CPU time)+16.00ms(GPU time)
Proposed Method	62.00s(CPU)+31.00ms(GPU time)

requires more computation compared with the previous methods, the significant improvement on performance would justify the additional computation is worthwhile.

V. CONCLUSIONS

To fully exploit the spatial, temporal and structural information in a depth sequence for action recognition at different time-scale, this paper presents three simple, compact yet effective representations. These representations are constructed through the extended ranking pooling, namely, the hierarchical bidirectional rank pooling. The representations enable fine-tuning on depth data existing ConvNets models that are learned from RGB video. Experimental results on three large datasets ChaLearn LAP IsoGD, ChaLearn LAP ConGD and NTU RGB+D have verified the efficacy of the proposed representations.

REFERENCES

[1] W. Li, Z. Zhang, and Z. Liu, "Action recognition based on a bag of 3D points," in *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2010, pp. 9–14.
[2] J. K. Aggarwal and L. Xia, "Human activity recognition from 3d data: A review," *Pattern Recognition Letters*, vol. 48, pp. 70–80, 2014.
[3] L. L. Presti and M. La Cascia, "3d skeleton-based human action classification: A survey," *Pattern Recognition*, vol. 53, pp. 130–147, 2016.
[4] J. Zhang, W. Li, P. O. Ogunbona, P. Wang, and C. Tang, "RGB-D-based action recognition datasets: A survey," *Pattern Recognition*, vol. 60, pp. 86–105, 2016.
[5] Y. Du, W. Wang, and L. Wang, "Hierarchical recurrent neural network for skeleton based action recognition," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1110–1118.
[6] V. Veeriah, N. Zhuang, and G.-J. Qi, "Differential recurrent neural networks for action recognition," in *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 4041–4049.
[7] J. Liu, A. Shahroudy, D. Xu, and G. Wang, "Spatio-temporal LSTM with trust gates for 3D human action recognition," in *Proc. European Conference on Computer Vision*, 2016, pp. 816–833.
[8] J. Yue-Hei Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici, "Beyond short snippets: Deep networks for video classification," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 4694–4702.
[9] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Proc. Annual Conference on Neural Information Processing Systems (NIPS)*, 2014, pp. 568–576.
[10] D. Jayaraman and K. Grauman, "Slow and steady feature analysis: higher order temporal coherence in video," in *CVPR*, 2016.
[11] S. Ji, W. Xu, M. Yang, and K. Yu, "3D convolutional neural networks for human action recognition," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 35, no. 1, pp. 221–231, 2013.
[12] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3D convolutional networks," in *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 4489–4497.
[13] N. Srivastava, E. Mansimov, and R. Salakhudinov, "Unsupervised learning of video representations using lstms," in *ICML*, 2015, pp. 843–852.
[14] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell, "Long-term recurrent convolutional networks for visual recognition and description," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 2625–2634.
[15] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang, "NTU RGB+ D: A large scale dataset for 3D human activity analysis," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
[16] B. Fernando, E. Gavves, J. Oramas, A. Ghodrati, and T. Tuytelaars, "Modeling video evolution for action recognition," in *CVPR*, 2015.
[17] H. Bilen, B. Fernando, E. Gavves, A. Vedaldi, and S. Gould, "Dynamic image networks for action recognition," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[18] B. Fernando, P. Anderson, M. Hutter, and S. Gould, "Discriminative hierarchical rank pooling for activity recognition," in *CVPR*, 2016.

[19] J. Wan, S. Z. Li, Y. Zhao, S. Zhou, I. Guyon, and S. Escalera, "Chalearn looking at people rgb-d isolated and continuous datasets for gesture recognition," in *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2016, pp. 1–9.

[20] P. Wang, W. Li, S. Liu, Z. Gao, C. Tang, and P. Ogunbona, "Large-scale isolated gesture recognition using convolutional neural networks," in *Proceedings of ICPRW*, 2016.

[21] P. Wang, W. Li, S. Liu, Y. Zhang, Z. Gao, and P. Ogunbona, "Large-scale continuous gesture recognition using convolutional neural networks," in *Proceedings of ICPRW*, 2016.

[22] T. DOrazio, R. Marani, V. Renò, and G. Cicirelli, "Recent trends in gesture recognition: how depth data has improved classical approaches," *Image and Vision Computing*, 2016.

[23] P. K. Pisharady and M. Saerbeck, "Recent methods and databases in vision-based hand gesture recognition: A review," *Computer Vision and Image Understanding*, vol. 141, pp. 152–165, 2015.

[24] S. Escalera, V. Athitsos, and I. Guyon, "Challenges in multimodal gesture recognition," *Journal of Machine Learning Research*, vol. 17, no. 72, pp. 1–54, 2016.

[25] H. Cheng, L. Yang, and Z. Liu, "Survey on 3d hand gesture recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 9, pp. 1659–1673, Sept 2016.

[26] S. Herath, M. Harandi, and F. Porikli, "Going deeper into action recognition: A survey," *arXiv preprint arXiv:1605.04988*, 2016.

[27] J. Wan, Q. Ruan, W. Li, and S. Deng, "One-shot learning gesture recognition from rgb-d data using bag of features," *Journal of Machine Learning Research*, vol. 14, pp. 2549–2582, 2013.

[28] J. Wan, V. Athitsos, P. Jangyodsuk, H. J. Escalante, Q. Ruan, and I. Guyon, "Csmmi: Class-specific maximization of mutual information for action and gesture recognition," *IEEE Transactions on Image Processing*, vol. 23, no. 7, pp. 3152–3165, July 2014.

[29] H. J. Escalante, I. Guyon, V. Athitsos, P. Jangyodsuk, and J. Wan, "Principal motion components for one-shot gesture recognition," *Pattern Analysis and Applications*, pp. 1–16, 2015.

[30] G. D. Forney, "The viterbi algorithm," *Proceedings of the IEEE*, vol. 61, no. 3, pp. 268–278, March 1973.

[31] Y. M. Lui, "Human gesture recognition on product manifolds," *Journal of Machine Learning Research*, vol. 13, no. 1, pp. 3297–3321, Nov 2012.

[32] D. Wu, F. Zhu, and L. Shao, "One shot learning gesture recognition from rgbd images," in *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, June 2012, pp. 7–12.

[33] F. Jiang, S. Zhang, S. Wu, Y. Gao, and D. Zhao, "Multi-layered gesture recognition with kinect," *Journal of Machine Learning Research*, vol. 16, no. 2, pp. 227–254, 2015.

[34] W. Li, Z. Zhang, and Z. Liu, "Expandable data-driven graphical modeling of human actions based on salient postures," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 18, no. 11, pp. 1499–1510, 2008.

[35] X. Yang, C. Zhang, and Y. Tian, "Recognizing actions using depth motion maps-based histograms of oriented gradients," in *Proc. ACM international conference on Multimedia (ACM MM)*, 2012, pp. 1057–1060.

[36] O. Oreifej and Z. Liu, "HON4D: Histogram of oriented 4D normals for activity recognition from depth sequences," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013, pp. 716–723.

[37] X. Yang and Y. Tian, "Super normal vector for activity recognition using depth sequences," in *Proc. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 804–811.

[38] C. Lu, J. Jia, and C.-K. Tang, "Range-sample depth feature for action recognition," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 772–779.

[39] P. Wang, W. Li, Z. Gao, C. Tang, J. Zhang, and P. O. Ogunbona, "Convnets-based action recognition from depth maps through virtual cameras and pseudocoloring," in *Proc. ACM international conference on Multimedia (ACM MM)*, 2015, pp. 1119–1122.

[40] P. Wang, W. Li, Z. Gao, J. Zhang, C. Tang, and P. Ogunbona, "Action recognition from depth maps using deep convolutional neural networks," *Human-Machine Systems, IEEE Transactions on*, vol. 46, no. 4, pp. 498–509, 2016.

[41] D. Wu, L. Pigou, P. J. Kindermans, N. D. H. Le, L. Shao, J. Dambre, and J. M. Odobez, "Deep dynamic neural networks for multimodal gesture segmentation and recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 8, pp. 1583–1597, Aug 2016.

[42] P. Wang, S. Wang, Z. Gao, Y. Hou, and W. Li, "Structured images for rgb-d action recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1005–1014.

[43] Y. Hou, S. Wang, P. Wang, Z. Gao, and W. Li, "Spatially and temporally structured global to local aggregation of dynamic depth information for action recognition," *IEEE Access*, vol. 6, pp. 2206–2219, 2018.

[44] P. Wang, Z. Li, Y. Hou, and W. Li, "Action recognition based on joint trajectory maps using convolutional neural networks," in *Proc. ACM international conference on Multimedia (ACM MM)*, 2016, pp. 102–106.

[45] Y. Hou, Z. Li, P. Wang, and W. Li, "Skeleton optical spectra based action recognition using convolutional neural networks," in *Circuits and Systems for Video Technology, IEEE Transactions on*, 2016, pp. 1–5.

[46] P. Wang, W. Li, J. Wan, P. Ogunbona, and X. Liu, "Cooperative training of deep aggregation networks for rgb-d action recognition," in *AAAI*, 2018.

[47] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[48] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. B. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding." in *Proc. ACM international conference on Multimedia (ACM MM)*, 2014, pp. 675–678.

[49] H. J. Escalante, V. Ponce-Lpez, J. Wan, M. A. Riegler, B. Chen, A. Claps, S. Escalera, I. Guyon, X. Bar, P. Halvorsen, H. Mller, and M. Larson, "Chalearn joint contest on multimedia challenges beyond visual analysis: An overview," in *Proceedings of ICPRW*, 2016.

[50] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation," *arXiv preprint arXiv:1801.04381*, 2018.

[51] I. Guyon, V. Athitsos, P. Jangyodsuk, and H. J. Escalante, "The chalearn gesture dataset (CGD 2011)," *Machine Vision and Applications*, vol. 25, no. 8, pp. 1929–1951, 2014.

[52] J. Wan, Q. Ruan, W. Li, G. An, and R. Zhao, "3d smosift: three-dimensional sparse motion scale invariant feature transform for activity recognition from rgb-d videos," *Journal of Electronic Imaging*, vol. 23, no. 2, 2014.

[53] H. Wang and C. Schmid, "Action recognition with improved trajectories," in *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2013, pp. 3551–3558.

[54] Y. Sun, X. Wang, and X. Tang, "Deep learning face representation from predicting 10,000 classes," in *CVPR*, 2014.

[55] P. Wang, W. Li, Z. Gao, Y. Zhang, C. Tang, and P. Ogunbona, "Scene flow to action map: A new representation for rgb-d based action recognition with convolutional neural networks," in *CVPR*, 2017.

[56] Y. Li, Q. Miao, K. Tian, Y. Fan, X. Xu, R. Li, and J. Song., "Large-scale gesture recognition with a fusion of rgb-d data based on the c3d model," in *Proceedings of ICPRW*, 2016.

[57] G. Zhu, L. Zhang, L. Mei, J. Shao, J. Song, and P. Shen, "Large-scale isolated gesture recognition using pyramidal 3d convolutional networks," in *Proceedings of ICPRW*, 2016.

[58] J. Wan, G. Guo, and S. Z. Li, "Explore efficient local features from rgb-d data for one-shot learning gesture recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 8, pp. 1626–1639, Aug 2016.

[59] N. C. Camgoz, S. Hadfield, O. Koller, and R. Bowden, "Using convolutional 3d neural networks for user-independent continuous gesture recognition," in *Proceedings of ICPRW*, 2016.

[60] X. Chai, Z. Liu, F. Yin, Z. Liu, and X. Chen, "Two streams recurrent neural networks for large-scale continuous gesture recognition," in *Proceedings of ICPRW*, 2016.

[61] C. Lea, A. Reiter, R. Vidal, and G. D. Hager, "Segmental spatiotemporal cnns for fine-grained action segmentation," in *14the European Conference on Computer Vision (ECCV)*, 2016.

[62] R. Vemulapalli, F. Arrate, and R. Chellappa, "Human action recognition by representing 3D skeletons as points in a lie group," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 588–595.

[63] X. Ji, J. Cheng, D. Tao, X. Wu, and W. Feng, "The spatial laplacian and temporal energy pyramid representation for human action recognition using depth sequences," *Knowledge-Based Systems*, 2017.

Pichao Wang (M'17) received his PhD in computer science from The University of Wollongong, Australia. He was a recipient of the 2016 ChaLearn Isolated Gesture Recognition & Continuous Gesture Recognition Challenge Awards, at ICPR 2016. He was also a recipient of two Best Paper Awards at the ChaLearn Look at People Challenge, 2016. He has published 30+ peerreviewed papers, including those in highly regarded journals and conferences such as IEEE TMM, IEEE THMS, CVPR, AAAI, ACM MM, etc. He served on the Technical Program Committees of ICCV 2017, CVPR 2018, ICME 2018, and IJCAI 2018. His current research interests include computer vision and machine learning. He is now a researcher in Motovis Inc. He is a member of IEEE.

Wanqing Li (SM'12) received his PhD in electronic engineering from The University of Western Australia. He was an Associate Professor (91-92) at Zhejiang University, a Senior Researcher and later a Principal Researcher (98-03) at Motorola Research Lab, and a visiting researcher (08,10 and 13) at Microsoft Research US. He is currently an Associate Professor and Co-Director of Advanced Multimedia Research Lab (AMRL) of University of Wollongong, Australia. His research areas are 3D computer vision, 3D multimedia signal processing and medical image analysis. Dr. Li is a Senior Member of IEEE.
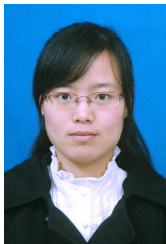
Zhimin Gao received the BS degree from North China Electric Power University, China in 2010, the MS degree at Tianjin University, China in 2013 and the PhD degree with the School of Computing and Information Technology, University of Wollongong, Australia in 2018. Her research interests include Computer Vision and Machine Learning.

Chang Tang (S'13) received his Ph.D. degree from Tianjin University, Tianjin, China in 2016. He joined the AMRL Lab of the University of Wollongong between Sep. 2014 and Sep. 2015. He is now an associate professor at the School of Computer Science, China University of Geosciences, Wuhan, China. His current research interests include machine learning and data mining.

Philip O. Ogunbona (SM'97) earned his PhD, DIC (Electrical Engineering) from Imperial College, London. His Bachelors degree was in Electronic and Electrical Engineering from the University of Ife, Nigeria (now named Obafemi Awolowo University). He is a Professor and Co-Director of the Advanced Multimedia Research Lab, University of Wollongong, Australia. His research interests include computer vision, pattern recognition and machine learning. He is a Senior Member of IEEE and Fellow of Australian Computer Society.