Hierarchical Recurrent Neural Network for Skeleton Based Action Recognition

Yong Du, Wei Wang, Liang Wang
Center for Research on Intelligent Perception and Computing, CRIPAC
Nat'l Lab of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences
{yong.du, wangwei, wangliang}@nlpr.ia.ac.cn

Abstract

Human actions can be represented by the trajectories of skeleton joints. Traditional methods generally model the spatial structure and temporal dynamics of human skeleton with hand-crafted features and recognize human actions by well-designed classifiers. In this paper, considering that recurrent neural network (RNN) can model the long-term contextual information of temporal sequences well, we propose an end-to-end hierarchical RNN for skeleton based action recognition. Instead of taking the whole skeleton as the input, we divide the human skeleton into five parts according to human physical structure, and then separately feed them to five subnets. As the number of layers increases, the representations extracted by the subnets are hierarchically fused to be the inputs of higher layers. The final representations of the skeleton sequences are fed into a single-layer perceptron, and the temporally accumulated output of the perceptron is the final decision. We compare with five other deep RNN architectures derived from our model to verify the effectiveness of the proposed network, and also compare with several other methods on three publicly available datasets. Experimental results demonstrate that our model achieves the state-of-the-art performance with high computational efficiency.

1. Introduction

As an important branch of computer vision, action recognition has a wide range of applications, *e.g.*, intelligent video surveillance, robot vision, human-computer interaction, game control, and so on [15, 36]. Traditional studies about action recognition mainly focus on recognizing actions from videos recorded by 2D cameras. But actually, human actions are generally represented and recognized in the 3D space. Human body can be regarded as an articulated system including rigid bones and hinged joints which are further combined into four limbs and a trunk [31]. Human actions are composed of the motions of these limbs and trunk which are represented by the movements of human skeleton joints in the 3D space [37]. Currently, reliable joint coordinates can be obtained from the cost-effective depth sensor using the real-time skeleton estimation algorithms [27, 28]. Effective approaches should be investigated for skeleton based action recognition.

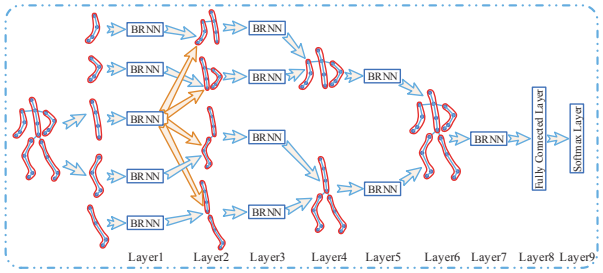


Figure 1: An illustrative sketch of the proposed hierarchical recurrent neural network. The whole skeleton is divided into five parts, which are fed into five bidirectional recurrent neural networks (BRNNs). As the number of layers increases, the representations extracted by the subnets are hierarchically fused to be the inputs of higher layers. A fully connected layer and a softmax layer are performed on the final representation to classify the actions.

Human skeleton based action recognition is generally considered as a time series problem [5, 17], in which the characteristics of body postures and their dynamics over time are extracted to represent a human action. Most of the existing skeleton based action recognition methods explicitly model the temporal dynamics of skeleton joints by using Temporal Pyramids (TPs) [19, 31, 33] and Hidden Markov Models (HMMs) [20, 34, 35]. The TPs methods are generally restricted by the width of the time windows and can only utilize limited contextual information. As for HMMs, it is very difficult to obtain the temporal aligned sequences and the corresponding emission distributions. Recently, recurrent neural networks (RNNs) with Long-Short Term Memory (LSTM) [8, 10] neurons have been used for action recognition [1, 11, 16]. All this work just uses single layer RNN as a sequence classifier without part-based

feature extraction and hierarchical fusion.

In this paper, taking full advantage of deep RNN in modelling the long-term contextual information of temporal sequences, we propose a hierarchical RNN for skeleton based action recognition. Fig. 1 shows the architecture of the proposed network, in which the temporal representations of low-level body parts are modeled by bidirectional recurrent neural networks (BRNNs) and combined into the representations of high-level parts.

Human body can be roughly decomposed into five parts, *e.g.*, two arms, two legs and one trunk, and human actions are composed of the movements of these body parts. Given this fact, we divide the human skeleton into the five corresponding parts, and feed them into five bidirectionally recurrently connected subnets (BRNNs) in the first layer. To model the movements from the neighboring skeleton parts, we concatenate the representation of the trunk subnet with those of the other four subnets, respectively, and then input these concatenated results to four BRNNs in the third layer as shown in Fig. 1. With the similar procedure, the representations of the upper body, the lower body and the whole body are obtained in the fifth and seventh layers, respectively. Up to now, we have finished the representation learning of skeleton sequences. Finally, a fully connected layer and a softmax layer are performed on the obtained representation to classify the actions. It should be noted that, to overcome the vanishing gradient problem when training RNN [8, 12], we adopt LSTM neurons in the last BRNN layer.

In the experiments, we compare with five other deep RNN architectures derived from our proposed model to verify the effectiveness of the proposed network, and compare with several methods on three publicly available datasets. Experimental results demonstrate that our method achieves the state-of-the-art performance with high computational efficiency. The main contributions of our work can be summarized as follows. Firstly, to the best of our knowledge, we are the first to provide an end-to-end solution for skeleton based action recognition by using hierarchical recurrent neural network. Secondly, by comparing with other five derived deep RNN architectures, we verify the effectiveness of the necessary parts of the proposed network, *e.g.*, bidirectional network, LSTM neurons in the last BRNN layer, hierarchical skeleton part fusion. Finally, we demonstrate that our proposed model can handle skeleton based action recognition very well without sophisticated preprocessing.

The remainder of this paper is organized as follows. In Section 2, we introduce the related work on skeleton based action recognition. In Section 3, we first review the background of RNN and LSTM, and then illustrate the details of the proposed network. Experimental results and discussion are presented in Section 4. Finally, we conclude the paper in Section 5.

2. Related Work

In this section, we briefly review the existing literature that closely relates to the proposed model, including three categories of approaches representing temporal dynamics by local features, sequential state transitions and RNN.

Approaches with local features By clustering the extracted joints into five parts, Wang *et al*. [32] use the spatial and temporal dictionaries of the parts to represent actions, which can capture the spatial structure of human body and movements. Chaudhry *et al*. [2] encode the skeleton structure with a spatial-temporal hierarchy, and exploit Linear Dynamical Systems to learn the dynamic features. Vemulapalli *et al*. [31] utilize rotations and translations to represent the 3D geometric relationships of body parts in Lie group, and then employ Dynamic Time Warping (DTW) and Fourier Temporal Pyramid (FTP) to model the temporal dynamics. Instead of modelling temporal evolution of features, Luo *et al*. [19] develop a novel dictionary learning method combined with Temporal Pyramid Matching, to keep the temporal dynamics. To represent both human motions and correlative objects, Wang *et al*. [33] first extract the local occupancy patterns from the appearance around skeleton joints, and then process them with FTP to obtain temporal structure. Zanfir *et al*. [38] propose a moving pose descriptor for capturing postures and skeleton joints. Using five joints coordinates and their temporal differences as inputs, Cho and Chen [4] perform action recognition with a hybrid multi-layer perceptron. In the above methods, the local temporal dynamics is generally represented within a certain time window or differential quantities, it cannot globally capture the temporal evolution of actions.

Approaches with sequential state transitions Lv *et al*. [20] extract local features of individual and partial combinations of joints, and train HMMs to capture the action dynamics. Based on skeletal joints features, Wu and Shao [34] adopt a deep forward neural network to estimate the emission probabilities of the hidden states in HMM, and then infer action sequences. To accurately calculate the similarity between two sequences with Dynamic Manifold Warping, Gong *et al*. [5] perform both temporal segmentation and alignment with structured time series representations. Though HMM can model the temporal evolution of actions, the input sequences have to be segmented and aligned, which in itself is a very difficult task.

Approaches with RNN The combination of RNN and perceptron can directly classify sequences without any segmentation. By obtaining sequential representations with a 3D convolutional neural network, Baccouche *et al*. [1] propose a LSTM-RNN to recognize actions. Regarding the histograms of optical flow as inputs, Grushin *et al*. [11] use LSTM-RNN for robust action recognition and achieve good results on KTH dataset. Considering that LSTM-RNNs employed in [1] and [11] are both unidirectional with only one

hidden layer, Lefebvre *et al.* [16] propose a bidirectional LSTM-RNN with one forward hidden layer and one backward hidden layer for gesture classification.

All the work above just uses RNN as a sequence classifier while we propose an end-to-end solution including both feature learning and sequence classification. Considering the fact that human actions are composed of the motions of human body parts, we use RNN in a hierarchical way.

3. Our Model

In order to put our proposed model into context, we first review recurrent neural network (RNN) and Long-Short Term Memory neuron (LSTM). Then we propose a hierarchical bidirectional RNN to solve the problem of skeleton based action recognition. Finally, five relevant deep RNNs with different architectures are also introduced.

3.1. Review of RNN and LSTM

The main difference between RNN and the feedforward networks is the presence of feedback loops which produce the recurrent connection in the unfolded network. With the recurrent structure, RNN can model the contextual information of a temporal sequence. Given an input sequence $x = (x^0, \ldots, x^{T-1})$, the hidden states of a recurrent layer $h = (h^0, \ldots, h^{T-1})$ and the output of a single hidden layer RNN $y = (y^0, \ldots, y^{T-1})$ can be derived as follows [8, 9, 10].

$$h^t = H(W_{xh}x^t + W_{hh}h^{t-1} + b_h) \quad (1)$$

$$y^t = O(W_{ho}h^t + b_o) \quad (2)$$

where W_{xh}, W_{hh}, W_{ho} denote the connection weights from the input layer x to the hidden layer h, the hidden layer h to itself and the hidden layer to the output layer y, respectively. b_h and b_o are two bias vectors, $H(\cdot)$ and $O(\cdot)$ are the activation functions in the hidden layer and the output layer.

Generally, it is very difficult to train RNNs (especially deep RNNs) with the commonly-used activation functions, *e.g.*, tanh and sigmoid functions, due to the vanishing gradient and error blowing up problems [8, 12]. To solve these problems, the Long-Short Term Memory (LSTM) architecture has been proposed [10, 13], which replaces the nonlinear units in traditional RNNs. Fig. 2 illustrates a LSTM memory block with a single cell. It contains one self-connected memory cell c and three multiplicative units, *i.e.*, the input gate i, the forget gate f and the output gate o, which can store and access the long range contextual information of a temporal sequence.

The activations of the memory cell and three gates are given as follows:

$$i^t = \sigma(W_{xi}x^t + W_{hi}h^{t-1} + W_{ci}c^{t-1} + b_i) \quad (3)$$

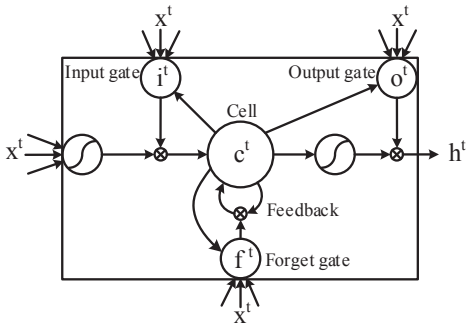


Figure 2: Long Short-Term Memory block with one cell [8].

$$f^t = \sigma(W_{xf}x^t + W_{hf}h^{t-1} + W_{cf}c^{t-1} + b_f) \quad (4)$$

$$c^t = f^t c^{t-1} + i^t tanh(W_{xc}x^t + W_{hc}h^{t-1} + b_c) \quad (5)$$

$$o^t = \sigma(W_{xo}x^t + W_{ho}h^{t-1} + W_{co}c^t + b_o) \quad (6)$$

$$h^t = o^t tanh(c^t) \quad (7)$$

where $\sigma(\cdot)$ is the sigmoid function, and all the matrices W are the connection weights between two units.

In order to utilize the past and future context for every point in the sequence, Schuster and Paliwal [26] proposed the bidirectional recurrent neural network (BRNN), which presents the sequence forwards and backwards to two separate recurrent hidden layers. These two recurrent hidden layers share the same output layer. A bidirectional recurrent neural network is illustrated in Fig. 3. It should be noted that we can easily obtain LSTM-BRNN just by replacing the nonlinear units in Fig. 3 with LSTM blocks.

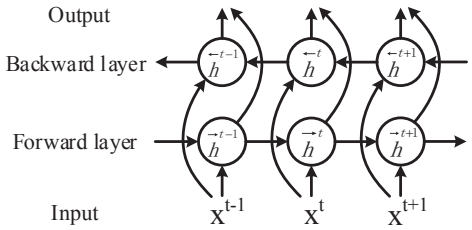


Figure 3: The architecture of bidirectional recurrent neural network [8].

3.2. Hierarchical RNN for Skeleton Based Action Recognition

According to human physical structure, the human skeleton can be decomposed into five parts, *e.g.*, two arms, two legs and one trunk. Simple human actions are performed by only one part of them, *e.g.*, punching forward and kicking forward mainly depend on swinging the arms and legs, respectively. Some actions come from moving the upper body or the lower body, *e.g.*, bending down mainly relates

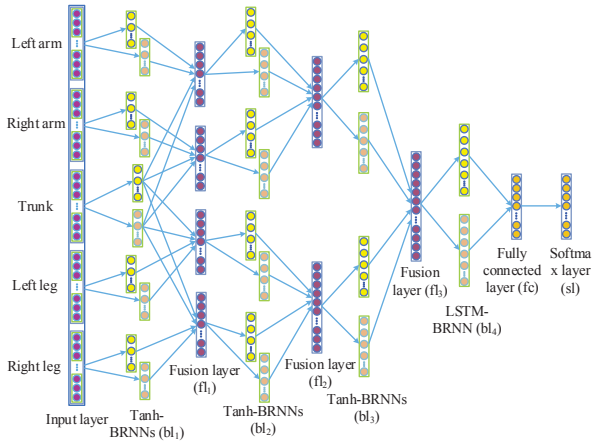


Figure 4: The architecture of our proposed model.

to the upper body. More complex actions are composed of the motions of these five parts, *e.g.*, running and swimming need to coordinate the moving of arms, legs and the trunk. To effectively recognize various human actions, modelling the movements of these individual parts and their combinations is very necessary. Benefiting from the power of RNN to access the contextual information, we propose a hierarchical bidirectional RNN for skeleton based action recognition. Different from traditional methods modelling the spatial structure and temporal dynamics with hand-crafted features and recognizing actions by well-designed classifiers, our model provides an end-to-end solution for action recognition.

The framework of the proposed model is shown in Fig. 4. We can see that our model is composed of 9 layers, *i.e.*, $bl_1 - bl_4$, $fl_1 - fl_3$, fc, and sl, each of which presents different structures and thus plays different role in the whole network. In the first layer bl_1, the five skeleton parts are fed into five corresponding bidirectionally recurrently connected subnets (BRNNs). To model the neighboring skeleton parts, *e.g.*, left arm-trunk, right arm-trunk, left leg-trunk, and right leg-trunk, we combine the representation of the trunk subnet with that of the other four subnets to obtain four new representations in the fusion layer fl_1. Similar to the layer bl_1, these resulting four representations are separately fed into four BRNNs in the layer bl_2. To model the upper and lower body, the representations of the left arm-trunk and right arm-trunk BRNNs are further combined to obtain the upper body representation while the representations of the left leg-trunk and right leg-trunk BRNNs are combined to obtain the lower body representation in the layer fl_2. Finally, the newly obtained two representations are fed into two BRNNs in the layer bl_3, and the representations of these two BRNNs in the layer bl_3 are fused again to represent the whole body in the layer fl_3. The temporal dynamics of the whole body representation is further modelled by another BRNN in the layer bl_4. From a viewpoint of feature learning, these stacked BRNNs can be considered to extract the spatial and temporal features of the skeleton sequences. After obtaining the final features of the skeleton sequence, a fully connected layer fc and a softmax layer sm are performed to classify the action.

As mentioned in Section 3.1, the LSTM architecture can effectively overcome the vanishing gradient problem while training RNNs [8, 12, 13]. However, we just adopt LSTM neurons in the last recurrent layer (bl_4). The first three BRNN layers all use the tanh activation function. This is a trade-off between improving the representation ability and avoiding overfitting. Generally, the number of weights in a LSTM block is several times more than that in a tanh neuron. It is very easy to overfit the network with limited training sequences.

3.3. Training

Training the proposed model contains a forward pass and a backward pass.

Forward pass: For the i-th BRNN layer bl_i at time t, given the q-th inputs $I^t_{i_q}$ and tanh activation function, the corresponding q-th representations of the forward layer and backward layer $\overrightarrow{h}^t_{i_q}$, $\overleftarrow{h}^t_{i_q}$ are defined as follows

$$\overrightarrow{h}^t_{i_q} = tanh(W_{I_{i_q}\overrightarrow{h}_{i_q}} I^t_{i_q} + W_{\overrightarrow{h}_{i_q}\overrightarrow{h}_{i_q}} \overrightarrow{h}^{t-1}_{i_q} + b_{\overrightarrow{h}_{i_q}}) \quad (8)$$

$$\overleftarrow{h}^t_{i_q} = tanh(W_{I_{i_q}\overleftarrow{h}_{i_q}} I^t_{i_q} + W_{\overleftarrow{h}_{i_q}\overleftarrow{h}_{i_q}} \overleftarrow{h}^{t+1}_{i_q} + b_{\overleftarrow{h}_{i_q}}) \quad (9)$$

where all the matrices W, vectors b are the corresponding connection weights and biases.

For the following fusion layer fl_i at time t, the p-th newly concatenated representation as the input of the $(i+1)$-th BRNN layer bl_{i+1} is

$$I^t_{(i+1)_p} = \overrightarrow{h}^t_{i_j} \bigoplus \overleftarrow{h}^t_{i_j} \bigoplus \overrightarrow{h}^t_{i_k} \bigoplus \overleftarrow{h}^t_{i_k} \quad (10)$$

where $\bigoplus$ denotes the concatenation operator, $\overrightarrow{h}^t_{i_j}$ and $\overleftarrow{h}^t_{i_j}$ are the hidden representations of the forward layer and backward layer of the j-th part in the i-th BRNN layer, $\overrightarrow{h}^t_{i_k}$ and $\overleftarrow{h}^t_{i_k}$ from the k-th part in the i-th layer.

For the last BRNN layer bl_4 with LSTM neurons, the forward output $\overrightarrow{h}^t_{bl_4}$ and backward output $\overleftarrow{h}^t_{bl_4}$ can be derived from Eqn. (3-7).

Combining $\overrightarrow{h}^t_{bl_4}$ and $\overleftarrow{h}^t_{bl_4}$ as the input to the fully connected layer fc, the output O^t of the layer fc is

$$O^t = W_{\overrightarrow{h}_{bl_4}} \overrightarrow{h}^t_{bl_4} + W_{\overleftarrow{h}_{bl_4}} \overleftarrow{h}^t_{bl_4} \quad (11)$$

where $W_{\overrightarrow{h}_{bl_4}}$, $W_{\overleftarrow{h}_{bl_4}}$ are the connection weights from the forward and backward layers of bl_4 to the layer fc.

Finally, the outputs of the layer fc are accumulated across the T frame sequence, and the accumulated results $\{\mathcal{A}_k\}$ are normalized by the softmax function to get each class probability $p(C_k)$:

$$\mathcal{A} = \sum_{t=0}^{T-1} O^t \tag{12}$$

$$p(C_k) = \frac{e^{\mathcal{A}_k}}{\sum_{i=0}^{C-1} e^{\mathcal{A}_i}} \tag{13}$$

Here there are C classes of human actions.

The objective function of our model is to minimize the maximum-likelihood loss function [8]:

$$\mathcal{L}(\Omega) = -\sum_{m=0}^{M-1} \ln \sum_{k=0}^{C-1} \delta(k-r) p(C_k|\Omega_m) \tag{14}$$

where $\delta(\cdot)$ is the Kronecker function, and r denotes the groundtruth label of the sequence Ω_m. There are M sequences in the training set Ω.

Backward pass: We use the back-propagation through time (BPTT) algorithm [8] to obtain the derivatives of the objective function with respect to all the weights, and minimize the objective function by stochastic gradient descent [8].

3.4. Five Comparative Architectures

In order to verify the effectiveness of the proposed network, we compare with other five different architectures derived from our proposed model. As illustrated before, we propose a hierarchically bidirectional RNN (**HBRNN-L**) for skeleton based action recognition (the suffix "-L" means that only the last recurrent layer consists of LSTM neurons, and the rest likewise). To prove the importance of the bidirectional connection, a similar network with unidirectional connection is proposed, which is called hierarchically unidirectional RNN (**HURNN-L**). To verify the role of part-based feature extraction and hierarchical fusion, we compare a deep bidirectional RNN (**DBRNN-L**), which is directly stacked with several RNNs with the whole human skeleton as the input. Furthermore, we compare a deep unidirectional RNN (**DURNN-L**) which does not adopt both the bidirectional connection and the hierarchical fusion. To further investigate whether LSTM neurons in the last recurrent layer are useful to overcome the vanishing/exploding problem in RNN, we examine another two architectures **DURNN-T** and **DBRNN-T**. Here DURNN-T and DBRNN-T are the similar networks to DURNN-L and DBRNN-L, but with the tanh activation function in all layers. It should be noted that all the six architectures have five learnable layers, *i.e.*, four recurrent hidden layers and one fully connected layer. And the number of neurons in the fully connected layer is equal to that of action categories.

4. Experiments

In this section, we evaluate our model and compare with other five different architectures and several recent work on three benchmark datasets: MSR Action3D Dataset [18], Berkeley Multimodal Human Action Dataset (Berkeley MHAD) [22], and Motion Capture Dataset HDM05 [21]. We also discuss the overfitting issues and the computational efficiency of the proposed model.

4.1. Evaluation Datasets

MSR Action3D Dataset: It is generated by a Microsoft Kinect-like depth sensor, which is widely used in action recognition. This dataset consists of 20 actions performed by 10 subjects in an unconstrained way for two or three times, 557 valid samples with 22077 frames. All sequences are captured in 15 FPS, and each frame in a sequence contains 20 skeleton joints. The low accuracy of the skeleton joint coordinates and the partial fragment missing in some sequences make this dataset very challenging.

Berkeley MHAD: It is captured by a multimodal acquisition system, in which an optical motion capture system is used to capture the 3D position of active LED markers with the frequency of 480 Hz. This dataset contains 659 sequences for 11 actions performed by 12 subjects with 5 repetitions of each action. In each frame of the sequence, there are 35 joints accurately extracted according to the 3D marker trajectory.

Motion Capture Dataset HDM05: It is captured by an optical marker-based technology with the frequency of 120 Hz, which contains 2337 sequences for 130 actions performed by 5 non-professional actors, and 31 joints in each frame. To our knowledge, this dataset is currently the largest depth sequence database which provides the skeleton joint coordinates for action recognition. As stated in [4], some samples of these 130 actions should be classified into the same category, *e.g.*, jogging starting from air and jogging starting from floor are the same action, jogging 2 steps and jogging 4 steps belong to the same "jogging" action. After sample combination, the actions are reduced to 65 categories.

4.2. Data Preprocessing and Parameter Settings

In our proposed model, all the human skeleton joints are divided into five parts, *i.e.*, two arms, two legs and one trunk, which are illustrated in Fig. 5. We can see that there are 4 joints for each part in MSR Action3D dataset. For Berkeley MHAD and HDM05 datasets, the joint numbers of arms, legs and the trunk are listed as follows: 7, 7, 7 and 7, 5, 7.

Given that human actions are independent of its absolute spatial position, we normalize the skeleton joints to an unified coordinate system. The origin of the coordinate system

Table 1: The parameter settings of our proposed model and the five compared models on three evaluation datasets. The DU.T is short for DURNN-T, and the rest likewise. The LL_i indicates the i-th learnable layer (bl_i in HBRNN-L).

Layer	MSR Action3D						Berkeley MHAD & HDM05					
	DU.T	DB.T	DU.L	DB.L	HU.L	HB.L	DU.T	DB.T	DU.L	DB.L	HU.L	HB.L
$LL_1(bl_1)$	80	40	80	40	30×5	$15 \times 2 \times 5$	90	60	80	40	40×5	$15 \times 2 \times 5$
$LL_2(bl_1)$	120	80	120	80	60×4	$30 \times 2 \times 4$	180	120	160	80	80×4	$30 \times 2 \times 4$
$LL_3(bl_3)$	240	120	180	100	90×2	$60 \times 2 \times 2$	240	120	180	100	100×2	$60 \times 2 \times 2$
$LL_4(bl_4)$	120	80	80	40	80×1	$40 \times 2 \times 1$	120	60	90	60	90×1	$60 \times 2 \times 1$

is defined as follows

$$\mathcal{O} = (\mathcal{J}_{hip\_center} + \mathcal{J}_{hip\_left} + \mathcal{J}_{hip\_right})/3 \quad (15)$$

where $\mathcal{J}_{hip\_center}$ is the 3D coordinate of the hip center, and the other two have the similar meanings.

To improve the signal to noise ratio of the raw data, we adopt a simple Savitzky-Golay smoothing filter [25] to preprocess the data. The filter is designed as follows

$$f_i = (-3x_{i-2}+12x_{i-1}+17x_i+12x_{i+1}-3x_{i+2})/35 \quad (16)$$

where x_i denotes the skeleton joint coordinate in the i-th frame, and f_i denotes the filtering result.

Considering that the trajectories of the skeleton joints vary smoothly, we sample the frames from the sequences in the fixed interval to reduce the computation cost. There are every 16 frames sampled for the Berkeley MHAD dataset and every 4 frames for the HDM05 dataset. We do not sample frames from MSR Action3D dataset due to the limited frame rates (15 FPS) and average length (less than 40 frames).

Tab. 1 shows the parameter settings of our proposed model and the five compared models on three evaluation datasets. Each value in the table indicates the number of neurons used in the corresponding layer, *e.g.*, the number 30×5 (LL_1, HU.L) means that each unidirectional subnet in the first learnable layer of HURNN-L has 30 neurons, and the number $15 \times 2 \times 5$ (LL_1, HB.L) indicates that each bidirectional subnet in the first BRNN layer (bl_1) of HBRNN-L has 15×2 neurons. These six networks on the same dataset have roughly the same number of weights.

It should be noted that the results of all the comparative methods on the three datasets are from their corresponding papers.

4.3. Experimental Results and Analysis

MSR Action3D Dataset: Although there are several validation methods summarized in [24] on this dataset, we follow the standard protocol provided in [18]. In this standard protocol, the dataset is divided into three action sets AS1, AS2 and AS3. The samples of subjects 1, 3, 5, 7, 9 are used for training while the samples of subjects 2, 4, 6, 8, 10 are used for testing. We compare the proposed model HBRNN-L with Li *et al.* [18], Chen *et al.* [3], Gowayyed *et al.* [6], Vemulapalli *et al.* [31] and other five variant architectures DURNN-T, DBRNN-T, DURNN-L, DBRNN-L, HURNN-L. The experimental results are shown in Tab. 2. We can see that our proposed HBRNN-L achieves the best average accuracy and outperforms the four methods in [3, 6, 18, 31] with hand-crafted features, and the performances of two derived models HURNN-L and DBRNN-L are promising. It should be noted that although Chen *et al.* [3] and Vemulapalli *et al.* [31] achieve the best performance in action sets AS1 and AS3, respectively, our HBRNN-L outperforms them with respect to the average accuracy. Furthermore, HBRNN-L performs consistently well on these three action sets, which indicates that HBRNN-L is more robust to various data.

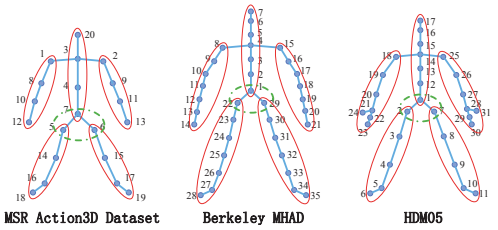


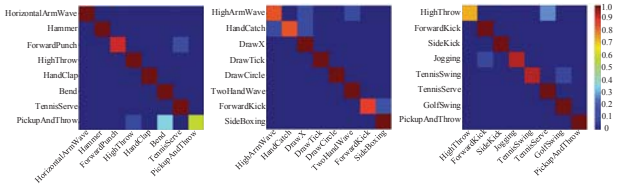
Figure 5: The human skeleton joints are divided into five parts in these three datasets.

Table 2: Experimental results on the MSR Action3D Dataset.

Method	AS1	AS2	AS3	Ave.
Li *et al.*, 2010 [18]	72.9	71.9	79.2	74.7
Chen *et al.*, 2013 [3]	**96.2**	83.2	92.0	90.47
Gowayyed *et al.*, 2013 [6]	92.39	90.18	91.43	91.26
Vemulapalli *et al.*, 2014 [31]	95.29	83.87	**98.22**	92.46
DURNN-T	75.24	75.00	81.08	77.11
DBRNN-T	81.90	80.36	88.29	83.52
DURNN-L	87.62	91.96	90.01	89.86
DBRNN-L	88.57	93.75	95.50	92.61
HURNN-L	92.38	93.75	94.59	93.57
HBRNN-L	93.33	**94.64**	95.50	**94.49**

The fact that HBRNN-L obtains higher average accuracy than HURNN-L, DBRNN-L and DURNN-L, proves the importance of bidirectional connection and hierarchical feature extraction. All the networks with LSTM neurons

in the last recurrent layer (with suffix "-L") are better than their corresponding networks with tanh activation functions (with suffix "-T"), which verifies the effectiveness of LSTM neurons in the proposed network.



(a) AS1 - 93.33% (b) AS2 - 94.64% (c) AS3 - 95.50%

Figure 6: Confusion matrices of HBRNN-L on MSR Action3D dataset.

The confusion matrices on the three action sets are shown in Fig. 6. We can see that the misclassifications mainly occur among several very similar actions. For example in Fig. 6a, the action "PickupAndThrow" is often misclassified to "Bend" while the action "ForwardPunch" is misclassified to "TennisServe". Actually, "PickupAndThrow" just has one more "throw" move than "Bend", and the "throw" move often holds few frames in the sequence. So it is very difficult to distinguish these two actions. The actions "ForwardPunch" and "TennisServe" share a large overlap in the sequences. Distinguishing them is also very challenging with only joint coordinates.

Berkeley MHAD: We follow the experimental protocol proposed in [22] on this dataset. The 384 sequences of the first 7 subjects are used for training while the 275 sequences of the last 5 subjects are used for testing. We compare our proposed model with Ofli *et al*. [23], Vantigodi *et al*. [30], Vantigodi *et al*. [29], Kapsouras *et al*. [14], Chaudhry *et al*. [2], as well as DURNN-T, DBRNN-T, DURNN-L, DBRNN-L, HURNN-L. The experimental results are shown in Tab. 3. We can see that HBRNN-L achieves the 100% accuracy with a simple preprocessing and performs better than those five derived RNN architectures, which proves the advantages of the proposed model once again. Meanwhile, the six RNN architectures obtain higher accuracy than Ofli *et al*. [23], Vantigodi *et al*. [30], Vantigodi *et al*. [29], Kapsouras *et al*. [14], and comparable results with Chaudhry *et al*. [2], which means that our proposed model provides an effective end-to-end solution for modelling temporal dynamics in action sequences.

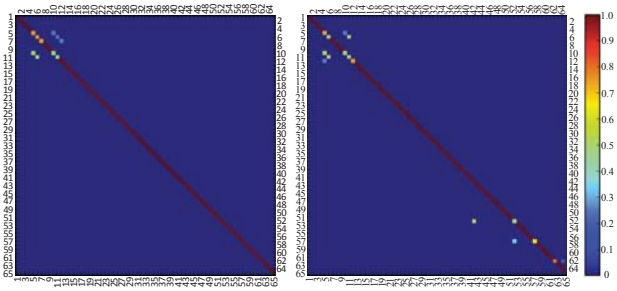
HDM05: We follow the experimental protocol proposed in [4] and perform 10-fold cross validation on this dataset. We compare our proposed model with Cho and Chen [4] and other five architectures DURNN-T, DBRNN-T, DURNN-L, DBRNN-L, HURNN-L. The experimental results are showed in Tab. 4. The proposed model HBRNN-L obtains the state-of-the-art accuracy of 96.92% with the standard deviation of 0.50. The derived models HURNN-L, DBRNN-L and DURNN-L also obtain excellent results.

Table 3: Experimental results on the Berkeley MHAD.

Method	Acc.(%)	Method	Acc.(%)
Ofli *et al*., 2014 [23]	95.37	DURNN-T	98.55
Vantigodi *et al*., 2013 [30]	96.06	DBRNN-T	99.27
Vantigodi *et al*., 2014 [29]	97.58	DURNN-L	98.55
Kapsouras *et al*., 2014 [14]	98.18	DBRNN-L	99.64
Chaudhry *et al*., 2013 [2]	99.27	HURNN-L	99.64
Chaudhry *et al*., 2013 [2]	100	**HBRNN-L**	**100**

Table 4: Experimental results on the HDM05.

Method	Ave.(%)	Std.
Cho and Chen, 2013 [4]	95.59	0.76
DURNN-T	94.63	1.16
DBRNN-T	94.79	1.11
DURNN-L	96.62	0.53
DBRNN-L	96.70	0.51
HURNN-L	96.70	0.41
HBRNN-L	**96.92**	**0.50**



(a) 97.52% (b) 96.04%

Figure 7: Two typical confusion matrices of HBRNN-L on the HDM05 dataset. The numbers on the horizontal and vertical axes correspond to the action categories [4].

Two typical confusion matrices of the 10-fold cross-validation from HBRNN-L are shown in Fig. 7. We can see that our model performs well on most of the actions. The misclassifications mainly come from the following categories: "5-depositHighR", "6-depositLowR", "7-depositMiddleR", "10-grabHighR", "11-grabLowR", and "12-grabMiddleR". Further checking the "grab" and "deposit" related skeleton sequences, we find that these two categories of actions share the similar spatial and temporal variations. Both of them can be decomposed into three sub-actions in chronological order: stretching out one hand, grabbing or depositing something, and drawing back the hand. The minor differences between grabbing and depositing something make it difficult to distinguish these two kinds of actions. It should be noted that although the original 130 actions are reduced to 65 categories, there are still

several confusing categories, *e.g.*, "39-sitDownChair" and "42-sitDownTable", which should belong to the same action. Without the context of actions, *e.g.*, recognizing chair and table from their appearances, it is very difficult to distinguish these actions just from skeleton streams.

4.4. Discussion

Overfitting issues: The experiments show that the models with suffix "-L" are easy to overfit while the others with suffix "-T" always underfit during training. It may be the vanishing gradient problem by using tanh activation function in all the layers. In order to overcome the overfitting problem in our proposed HBRNN-L and other derived networks with suffix "-L", we adopt the strategies like adding the input noise, weight noise and early stopping [7, 8]. In our practice, we find that adding the weight noise is more effective than adding the input noise, and the commonly-used dropout strategy [39] does not work here. For the underfitting problem of the models with suffix "-T", we use the retraining strategy by tuning learning rate and adding various levels of input noise and weight noise.

Computational efficiency: We take the Berkeley MHAD dataset for an example to illustrate the efficiency of HBRNN-L. With C++ implementation on a CPU 3.2GHz system, we spend 50s for each epoch consisting of 384 sequences (average 127ms per sequence) during training. After about 30 epochs, we can get an accuracy greater than 98%. During testing, it takes 52.46 ms per sequence (about 234 frames per sequence). It should be mentioned that HURNN-L, which achieves comparable performance with HBRNN-L, runs much faster and is more suitable for online applications.

5. Conclusion and Future Work

In this paper, we proposed an end-to-end hierarchical recurrent neural network for skeleton based action recognition. We first divide the human skeleton into five parts, and then feed them to five subnets. As the number of layers increases, the representations in the subnets are hierarchically fused to be the inputs of higher layers. A perceptron is performed on the learned representations of the skeleton sequences to obtain the final recognition results. Experimental results on three publicly available datasets demonstrate the effectiveness of the proposed network.

As we analyzed on the HDM05 and MSR Action3D datasets, the similar human actions are very difficult to be distinguished just from the skeleton joints. In the future, we will consider to combine more features into the proposed hierarchical recurrent neural network, *e.g.*, object appearance.

Acknowledgement

This work was supported by the National Basic Research Program of China (2012CB316300) and National Natural Science Foundation of China (61175003, 61135002, 61202328, 61420106015, U1435221).

References

[1] M. Baccouche, F. Mamalet, C. Wolf, C. Garcia, and A. Baskurt. Sequential deep learning for human action recognition. In *Human Behavior Understanding*, pages 29–39. Springer, 2011. 1, 2

[2] R. Chaudhry, F. Ofli, G. Kurillo, R. Bajcsy, and R. Vidal. Bio-inspired dynamic 3d discriminative skeletal features for human action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 471–478. IEEE, 2013. 2, 7

[3] C. Chen, K. Liu, and N. Kehtarnavaz. Real-time human action recognition based on depth motion maps. *Journal of Real-Time Image Processing*, pages 1–9, 2013. 6

[4] K. Cho and X. Chen. Classifying and visualizing motion capture sequences using deep neural networks. *CoRR*, abs/1306.3874, 2013. 2, 5, 7

[5] D. Gong, G. Medioni, and X. Zhao. Structured time series analysis for human action segmentation and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence, VOL. 36, NO. 7*, 2014. 1, 2

[6] M. A. Gowayyed, M. Torki, M. E. Hussein, and M. El-Saban. Histogram of oriented displacements (hod): describing trajectories of human joints for action recognition. In *International Joint Conference on Artificial Intelligence*, pages 1351–1357. AAAI Press, 2013. 6

[7] A. Graves. Practical variational inference for neural networks. In *Advances in Neural Information Processing Systems*, pages 2348–2356, 2011. 8

[8] A. Graves. *Supervised sequence labelling with recurrent neural networks*, volume 385. Springer, 2012. 1, 2, 3, 4, 5, 8

[9] A. Graves and N. Jaitly. Towards end-to-end speech recognition with recurrent neural networks. In *International Conference on Machine Learning*, pages 1764–1772, 2014. 3

[10] A. Graves, A. Mohamed, and G. Hinton. Speech recognition with deep recurrent neural networks. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 6645–6649. IEEE, 2013. 1, 3

[11] A. Grushin, D. D. Monner, J. A. Reggia, and A. Mishra. Robust human action recognition via long short-term memory. In *International Joint Conference on Neural Networks*, pages 1–8. IEEE, 2013. 1, 2

[12] S. Hochreiter, Y. Bengio, P. Frasconi, and J. Schmidhuber. Gradient flow in recurrent nets: the difficulty of learning long-term dependencies, 2001. 2, 3, 4

[13] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997. 3, 4

[14] I. Kapsouras and N. Nikolaidis. Action recognition on motion capture data using a dynemes and forward differ-

ences representation. *J. Vis. Comun. Image Represent.*, 25(6):1432–1445, Aug. 2014. 7

[15] J. Koutník, J. Schmidhuber, and F. Gomez. Evolving deep unsupervised convolutional networks for vision-based reinforcement learning. In *Conference on Genetic and Evolutionary Computation*, pages 541–548. ACM, 2014. 1

[16] G. Lefebvre, S. Berlemont, F. Mamalet, and C. Garcia. Blstm-rnn based 3d gesture classification. In *Artificial Neural Networks and Machine Learning*, pages 381–388. Springer, 2013. 1, 3

[17] K. Li and Y. Fu. Prediction of human activity by discovering temporal sequence patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence, VOL. 36, NO. 8*, 2014. 1

[18] W. Li, Z. Zhang, and Z. Liu. Action recognition based on a bag of 3d points. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pages 9–14. IEEE, 2010. 5, 6

[19] J. Luo, W. Wang, and H. Qi. Group sparsity and geometry constrained dictionary learning for action recognition from depth maps. In *IEEE International Conference on Computer Vision*, pages 1809–1816. IEEE, 2013. 1, 2

[20] F. Lv and R. Nevatia. Recognition and segmentation of 3-d human action using hmm and multi-class adaboost. In *European Conference on Computer Vision*, pages 359–372. Springer, 2006. 1, 2

[21] M. Müller, T. Röder, M. Clausen, B. Eberhardt, B. Krüger, and A. Weber. Documentation mocap database hdm05. Technical Report CG-2007-2, Universität Bonn, June 2007. 5

[22] F. Ofli, R. Chaudhry, G. Kurillo, R. Vidal, and R. Bajcsy. Berkeley mhad: A comprehensive multimodal human action database. In *IEEE Workshop on Applications of Computer Vision*, pages 53–60. IEEE, 2013. 5, 7

[23] F. Ofli, R. Chaudhry, G. Kurillo, R. Vidal, and R. Bajcsy. Sequence of the most informative joints (smij): A new representation for human skeletal action recognition. *Journal of Visual Communication and Image Representation*, 25(1):24–38, 2014. 7

[24] J. R. Padilla-López, A. A. Chaaraoui, and F. Flórez-Revuelta. A discussion on the validation tests employed to compare human action recognition methods using the MSR action3d dataset. *CoRR*, abs/1407.7390, 2014. 6

[25] A. Savitzky and M. J. Golay. Smoothing and differentiation of data by simplified least squares procedures. *Analytical Chemistry*, 36(8):1627–1639, 1964. 6

[26] M. Schuster and K. K. Paliwal. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681, 1997. 3

[27] J. Shotton, T. Sharp, A. Kipman, A. Fitzgibbon, M. Finocchio, A. Blake, M. Cook, and R. Moore. Real-time human pose recognition in parts from single depth images. *Communications of the ACM*, 56(1):116–124, 2013. 1

[28] J. Tompson, A. Jain, Y. LeCun, and C. Bregler. Joint training of a convolutional network and a graphical model for human pose estimation. *arXiv preprint arXiv:1406.2984*, 2014. 1

[29] S. Vantigodi and V. B. Radhakrishnan. Action recognition from motion capture data using meta-cognitive rbf network classifier. In *Intelligent Sensors, Sensor Networks and Information Processing, 2014 IEEE Ninth International Conference on*, pages 1–6. IEEE, 2014. 7

[30] S. Vantigodi and R. Venkatesh Babu. Real-time human action recognition from motion capture data. In *Computer Vision, Pattern Recognition, Image Processing and Graphics, Fourth National Conference on*, pages 1–4. IEEE, 2013. 7

[31] R. Vemulapalli, F. Arrate, and R. Chellappa. Human action recognition by representing 3d skeletons as points in a lie group. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 588–595. IEEE, 2014. 1, 2, 6

[32] C. Wang, Y. Wang, and A. L. Yuille. An approach to pose-based action recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 915–922. IEEE, 2013. 2

[33] J. Wang, Z. Liu, Y. Wu, and J. Yuan. Mining actionlet ensemble for action recognition with depth cameras. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1290–1297. IEEE, 2012. 1, 2

[34] D. Wu and L. Shao. Leveraging hierarchical parametric networks for skeletal joints based action segmentation and recognition. In *IEEE International Conference on Computer Vision*, 2014. 1, 2

[35] L. Xia, C.-C. Chen, and J. Aggarwal. View invariant human action recognition using histograms of 3d joints. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pages 20–27. IEEE, 2012. 1

[36] X. Yang and Y. Tian. Super normal vector for activity recognition using depth sequences. IEEE Conference on Computer Vision and Pattern Recognition. 1

[37] M. Ye, Q. Zhang, L. Wang, J. Zhu, R. Yang, and J. Gall. A survey on human motion analysis from depth data. In *Time-of-Flight and Depth Imaging. Sensors, Algorithms, and Applications*, pages 149–187. Springer, 2013. 1

[38] M. Zanfir, M. Leordeanu, and C. Sminchisescu. The moving pose: An efficient 3d kinematics descriptor for low-latency action recognition and detection. In *IEEE International Conference on Computer Vision*, pages 2752–2759. IEEE, 2013. 2

[39] W. Zaremba, I. Sutskever, and O. Vinyals. Recurrent neural network regularization. *CoRR*, abs/1409.2329, 2014. 8