IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received January 20, 2021, accepted February 12, 2021, date of publication February 16, 2021, date of current version April 13, 2021.

*Digital Object Identifier 10.1109/ACCESS.2021.3059650*

# Skeleton-Based Square Grid for Human Action Recognition With 3D Convolutional Neural Network

**WENWEN DING** [1], **CHONGYANG DING** [2], **GUANG LI** [2], **AND KAI LIU** [2]

[1] School of Mathematical Sciences, Huaibei Normal University, Anhui 235000, China
[2] School of Computer Science and Technology, Xidian University, Xi'an 710071, China

Corresponding author: Kai Liu (kailiu@mail.xidian.edu.cn)

This work was supported in part by the National Science Foundation of China for International Young Scientists under Grant 61850410523, in part by the National Natural Science Foundation of China under Grant 61806155, in part by the Nature Science Foundation of Anhui Province under Grant 1908085MF186, and in part by the Natural Science Foundation of the Anhui Higher Education Institutions of China under Grant KJ2020ZD008, Grant KJ2018A0384, Grant KJ2019A0589, and Grant KJ2019A0592.

**ABSTRACT** Convolutional neural networks (CNNs) can effectively handle grid-structured data but not dynamic skeletons, which are usually expressed as graph structures. In this study, we first propose a skeleton-based square grid (SSG) for transforming dynamic skeletons into three-dimensional (3D) grid-structured data so that CNNs can be applied to such data. Each SSG contains a joint-based square grid (JSG) and a rigid-based square grid (RSG) based on intrinsic and extrinsic dependencies of various body parts, respectively. Next, to enhance the ability of deep features to capture the correlations among 3D grid-structured data, a two-stream 3D CNN is constructed to learn spatiotemporal features using the JSG and RSG sequences. Finally, we introduce a soft attention model that selectively focuses on the informative body parts in the skeleton sequences. We validate our model in terms of action recognition using three datasets: NTU RGB+D, Kinetics Motion, and SBU Kinect Interaction datasets. Our experimental results demonstrate the effectiveness of the proposed approach as well as its superior performance when compared with those of state-of-the-art methods.

**INDEX TERMS** 3D convolutional neural networks, skeleton action recognition, neural network, attention mechanism.

## I. INTRODUCTION

Human action recognition has received significant attention in computer vision, owing to its wide application in video surveillance, medical rehabilitation, animal behavior analysis, virtual reality, and human-computer interactions. However, owing to the influence of external environmental factors such as changes in appearance, surrounding distractions, and variations in viewpoints, it is still a challenging task to precisely represent human action features. With the advent of cost-efficient depth sensors such as Kinect, the approaches to obtaining human skeleton information have changed significantly.

Deep learning techniques have achieved remarkable progress in the field of computer vision. The success of these techniques mostly relies on convolutional neural networks (CNNs) [1], [2] and recurrent neural networks (RNNs) [3], [4]. RNNs with long short-term memory (LSTM) can learn temporal dependencies; however, it is difficult to train a stacked LSTM in practice [5]. CNNs use a standard convolution, which can be applied only to grid-structured data. In [6]–[8], a skeleton sequence was treated as a *still image*, and thus, spatiotemporal information could be learned using CNNs. However, these approaches do not consider the motion information encoded in multiple contiguous frames; therefore, the dependencies between joints and rigid bodies cannot be expressed fully.

A skeleton can be considered a graph structure, with bone joints and rigid bodies representing the vertices and edges of the graph, respectively. The recently proposed graph convolutional networks (GCNs) [9], [10], which generalize CNNs to arbitrary graphs, have been effective in learning spatiotemporal information. Yan *et al.* [11] leveraged the spatial connections between bone joints and connected the same

The associate editor coordinating the review of this manuscript and approving it for publication was Byung Cheol Song.

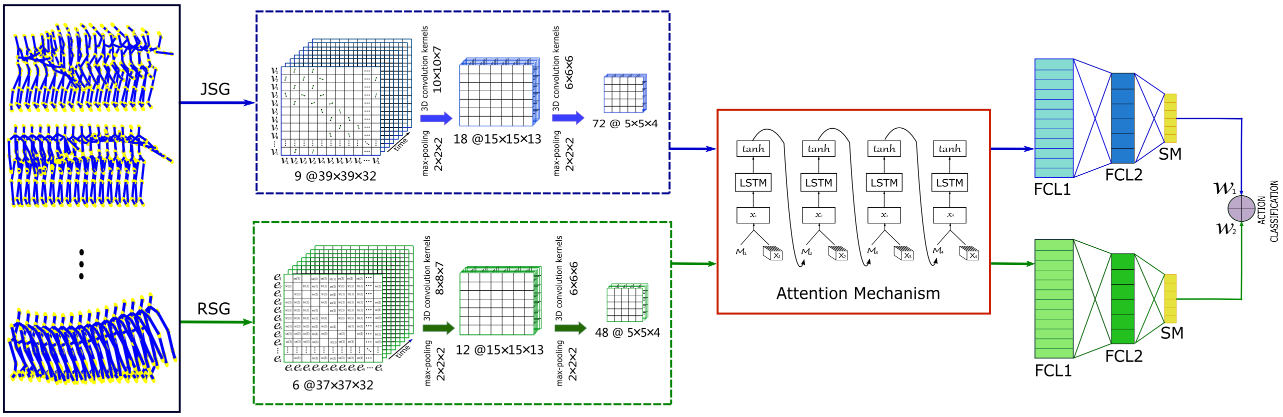


**FIGURE 1. General framework of the proposed approach. Two 3D CNN streams are used to process the intrinsic and extrinsic dependencies in a skeleton.**

joints over time to form a spatiotemporal graph (STG). Spatiotemporal graph convolutional networks (ST-GCNs) apply GCNs to model STGs. However, this approach is limited to a heuristic design of the sampling function for the graph convolutional operation and represents only physical dependencies, rendering this approach unsuitable for human action recognition. For example, the hand and the head are physically disconnected, but their dependency is important for recognizing actions such as *drinking water* or *answering the telephone*. To extract the global relationship, Ding *et al.* [12] proposed STG-IN to perform long-range temporal modeling over an STG.

To resolve these issues, we propose an effective method for transforming each skeleton into a skeleton-based square grid (SSG) containing a joint-based square grid (JSG) and a rigid-based square grid (RSG). Both the horizontal and vertical axes of the SSG are expressed as a chain order generated by traversing the skeleton tree based on a depth-first order. The JSG preserves the *intrinsic* dependency (physical connection) of the skeleton structure, as depicted in Fig. 3a. The RSG enables the CNN to extract the *extrinsic* dependency (physical disconnection) among various body parts in a skeleton, as depicted in Fig. 3b. After the skeleton sequence is transformed into a grid sequence, as depicted in Fig. 1, a two-stream 3D CNN is introduced to extract dependency relationships and discriminative spatiotemporal features from the JSG and RSG sequences, respectively. Furthermore, certain body parts in the skeleton and crucial frames in the sequences are more informative for recognizing actions, such as the hands in the action of *waving hands*. Such body parts and frames must have high importance when modeling dynamic skeletons. Therefore, a soft attention model with LSTM is incorporated into the two-stream 3D CNN to allocate attention masks to various body parts. Finally, the two-stream 3D CNN and the attention network are cascaded as an entire network, which is trained in an end-to-end manner using the input SSG data.

The main contributions of this study are as follows: 1) An SSG is proposed so that the skeleton structure can be adaptively learned, and the spatial relationships among various body parts can be determined. 2) A soft attention mechanism is introduced to learn the importance of human body parts in a skeleton and the crucial frames in the action sequences. Next, attention weights are used to refine the output of the two-stream 3D CNN. 3) The proposed model exceeded the performances of state-of-the-art methods by a significant margin, based on the results on three datasets for skeleton-based action recognition. The disadvantage of the proposed method is that it does not have universal applicability. The proposed grid can preserve only the adjacency relationship for the human skeleton graph. For any other graph, if there is no prior domain knowledge, the extracted relationship is unknown after it is transformed into this type of grid.

The remainder of this paper is organized as follows. Section II presents related work. Section III describes the SSG developed to adaptively learn the features of a two-stream 3D CNN. A soft attention mechanism is then introduced for feature refinement using attention weights. Section IV presents the experimental results and discussion. Section V concludes the paper.

## II. RELATED WORK

Deep neural networks can realize automatic feature extraction to replace hand-crafted features. CNNs have been proven effective in extracting local-to-global features. In [6], [7], it was proposed that skeleton sequences be represented as 2D grayscale images, called *skeleton images*. Subsequently, a CNN was used to learn a spatiotemporal representation. Each row of the *skeleton images* was typically arranged by simply concatenating all joints in a predefined chain order, and each column represented the temporal evolution of a joint. It was demonstrated that the relations between adjacent joints in a skeleton were not expressed in *skeleton images*. Liu *et al.* [13] proposed skeleton images with "Skepxels" for a better representation of joint correlations. Yang *et al.* [8] designed skeleton images using depth-first traversal on skeleton trees. In these approaches, the skeleton sequences were treated as still images, and the motion information encoded in multiple contiguous frames was not considered.

To adapt the skeleton data to a suitable view, a view-adaptive LSTM [14] was introduced for better action recognition. Pham *et al.* [15] proposed building a compact image to represent skeleton poses and their motions. Pham *et al.* [16] designed a deep neural network and trained it to learn a direct 2D-to-3D mapping and predict human poses in 3D space.

3D CNNs with 3D convolution kernels and 3D pooling can not only acquire the spatial features of each skeleton but also express the change in adjacent joints over time. Ji *et al.* [17] proposed a simple and efficient method for automatic spatiotemporal feature learning using a 3D CNN. Cao *et al.* [18] verified that 3D CNNs are more suitable for spatiotemporal feature learning, and a 3 × 3 × 3 convolution kernel is the best choice in convolutional layers. In general, 3D CNNs can automatically capture correlations in the spatiotemporal information. Our SSG transforms dynamic skeletons into 3D grid-structured data. It differs from previous approaches in that it can extract intrinsic and extrinsic dependencies with temporal dynamics. It also differs from [19] in that it harnesses the 3D CNNs whereas [19] uses the growing grid neural networks. To consider time dependency, another approach has investigated the combination of an RNN and LSTM. Du *et al.* [20] proposed an end-to-end hierarchical RNN to encode the relative motion between the joints in the skeleton. Shahroudy *et al.* [21] proposed a part-aware LSTM with part-based memory sub-cells and a new gating mechanism. To learn the co-occurrence features of the joints, Zhu *et al.* [22] used an end-to-end fully connected deep LSTM network. However, LSTM networks cannot memorize all the information of an entire action sequence [23]; therefore, they cannot efficiently learn the structure of the human skeleton.

Attention mechanisms simulate human perception and focus more on certain parts of the information. All the joints are not equally important in an action. It was demonstrated in [24]–[27] that some actions are related to a certain set of joint points, whereas some others are related to other joints. For example, a *telephone call* is closely related to the joints of the head, shoulder, elbow, and wrist. It has little relationship with the joints of the leg. In contrast, *walking* can be identified primarily through the observation of the joints of the leg. Therefore, the importance of each posture is not the same in action recognition.

The various attention models that have been proposed can be classified into *soft attention* and *hard attention* models. Soft attention is deterministic, and the importance (score) of each body part in action sequences is measured and added to form the final representation. Hard attention is stochastic, and a single element is selected exclusively. Liu *et al.* [26] proposed global context-aware attention LSTM to extract the global contextual information by measuring the scores of the new inputs at all steps and adjusting the attention weights accordingly. Zang *et al.* [27] implemented an attention mechanism in a temporally weighted multi-stream CNN, focusing on critical segments rather than processing all sampled frames.

## III. MODEL ARCHITECTURE

### A. GRAPH REPRESENTATION OF SKELETON

The human skeleton is represented using a group of 3D spatial joint coordinates. A non-grid graph $\mathcal{S} = (\mathcal{V}, \mathcal{E})$ can model the spatial relations of the joints in the skeleton. $\mathcal{V} = \{v_i\}_{i=1}^N$ indicates the set of nodes representing the joints in the skeleton, where $N$ is the number of joints. $\mathcal{E} = \{e_i\}_{i=1}^M$ indicates the set of edges, that is, oriented rigid bodies, where $M = N - 1$. Fig. 2a depicts an example skeleton with 20 joints and 19 rigid bodies. We define the signal of each joint $v_i$ using its 3D spatial coordinates $Coor(v_i) = (x_{v_i}, y_{v_i}, z_{v_i})$. Let $e_{is}$ and $e_{ie}$ denote the starting and ending points, respectively, of the rigid body $e_i$. We can construct a feature $Rig(e_{is}, e_{ie}) \in \mathcal{R}^9$ as

$$Rig(e_{is}, e_{ie}) = (Coor(e_{is}), Coor(e_{ie}), Ori(e_{is}, e_{ie})). \quad (1)$$

where the orientation vector of a rigid body is defined as $Ori(e_{is}, e_{ie}) = (x_{e_{is}} - x_{e_{ie}}, y_{e_{is}} - y_{e_{ie}}, z_{e_{is}} - z_{e_{ie}})$.

Thus, each rigid body $e_i$ is represented by $Rig(e_{is}, e_{ie})$, which can reflect the intrinsic dependency (i.e., physical connection). The intrinsic dependency of the skeleton structure can be preserved using a JSG.

To learn the relationships among various body parts, we introduce the relative geometry of a pair of rigid bodies $e_m$ and $e_n$, which can be denoted as $p_m = R_{m,n}p_n + \overrightarrow{d}_{m,nd}$ by obtaining the coordinates of point $p$ from rigid bodies $e_m$ to $e_n$. Using rotation matrix $R_{m,n}$ and translation vector $\overrightarrow{d}_{m,n}$, a rigid-body transformation (rotation and translation) to align one body part can be written as

$$\begin{bmatrix} p_m \\ 1 \end{bmatrix} = \begin{bmatrix} R_{n,m} & \overrightarrow{d}_{n,m} \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} p_n \\ 1 \end{bmatrix}. \quad (2)$$

In particular, the relative geometry of $e_m$ and $e_n$ can be described as

$$P(e_m, e_n) = \begin{bmatrix} R_{m,n} & \overrightarrow{d}_{m,n} \\ 0^T & 1 \end{bmatrix} \quad (3)$$

$$P(e_n, e_m) = \begin{bmatrix} R_{n,m} & \overrightarrow{d}_{n,m} \\ 0^T & 1 \end{bmatrix}. \quad (4)$$

Note that both $P_{m,n}$ and $P_{n,m}$ do not change only when both $e_m$ and $e_n$ undergo the same rotation or translation, that is, only when there is no relative motion between them. Therefore, we use both $P_{m,n}$ and $P_{n,m}$ to represent the relative geometry of $e_m$ and $e_n$.

This rigid-body geometrical transformation is in the form of the matrix of the Lie group SE(3), which can be mapped to its Lie algebra by a 6D vector representation using Equ. 5 the matrix logarithm [28]. Formally,

$$\begin{aligned} p_v(e_m, e_n) &= vec(log(P(e_m, e_n))) \\ &= vec(\begin{bmatrix} 0 & -\omega_3 & \omega_2 & d_1 \\ \omega_3 & 0 & -\omega_1 & d_2 \\ -\omega_2 & \omega_1 & 0 & d_3 \\ 0 & 0 & 0 & 0 \end{bmatrix}) \\ &= [\omega_1, \omega_2, \omega_3, d_1, d_2, d_3] \in \mathcal{R}^6 \end{aligned} \quad (5)$$

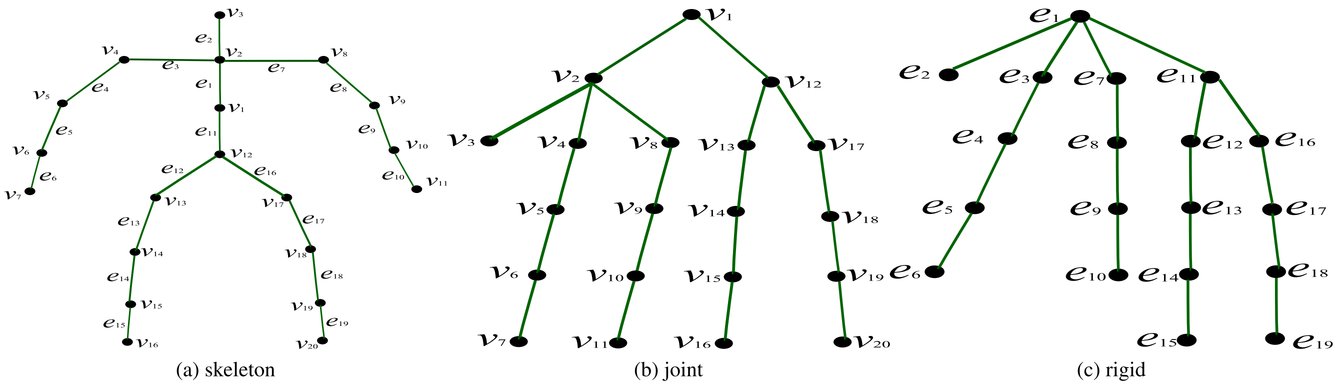


**FIGURE 2.** (a) Example skeleton consisting of 20 joints and 19 rigid bodies. (b) Skeleton tree depicting joints in the human body $v_i$ and their physical connections. (c) Skeleton tree depicting rigid bodies $e_i$ and their physical connections.

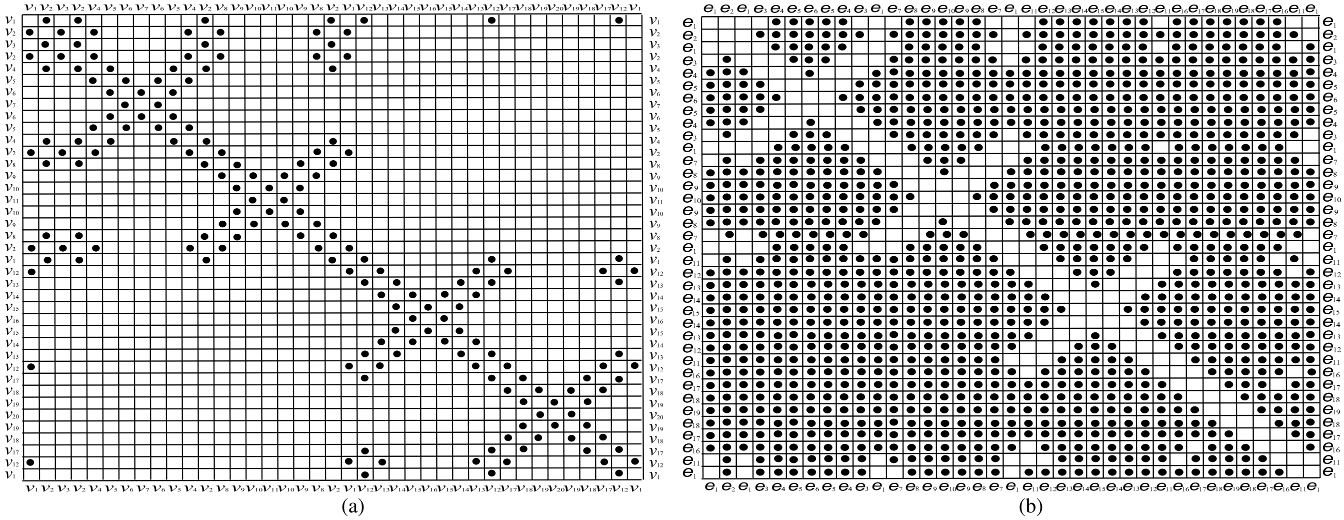


**FIGURE 3.** (a) JSG with intrinsic dependency. The size of JSG is 39 × 39 × 9, and the skeleton has 20 joints. (b) RSG with extrinsic dependency. The size of RSG is 37 × 37 × 6, and the skeleton has 19 rigid bodies.

Thus, the rigid-body geometrical transformation is represented by a 6D vector $p_v(e_m, e_n)$, which can reflect the extrinsic dependencies (i.e., physical disconnection) among various body parts. An RSG is introduced to encode the extrinsic dependency.

## B. GRID REPRESENTATION OF SKELETON STRUCTURE

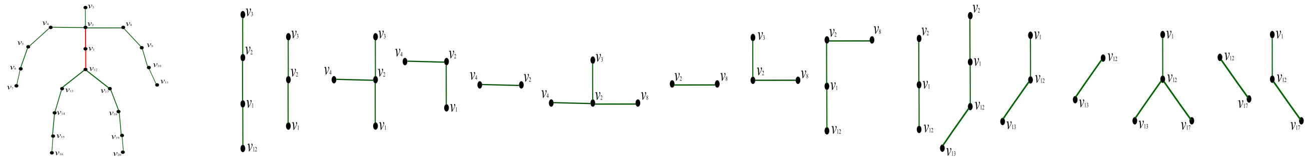
Given a dynamic skeleton sequence, actions can be represented by a set of bone joints or rigid bodies, as well as their relationships in space and time. In this study, we call the relationship between joints *intrinsic* dependency (i.e., physical connection) and that between rigid bodies *extrinsic* dependency (i.e., physical disconnection). An important characteristic is that each pair of connected joints moves together. The *intrinsic* dependency is maintained during the movement owing to the force imposed by the rigid bodies. The *extrinsic* dependency is dependent on the movement of other joints through "invisible" rigid bodies. For example, the hand and the head are physically disconnected, but their dependency is important for recognizing the action of *drinking water* or *answering the telephone*. In this sense, extrinsic dependency contributes as much as does intrinsic dependency to action recognition.

### 1) GRID REPRESENTATION OF INTRINSIC DEPENDENCY

To model the kinematic dependency among physically connected joints, the skeleton is transformed into a tree structure, as depicted in Fig. 2b. In the simple joint chain model, the joint visiting order is $v_1 \rightarrow v_2 \rightarrow \ldots \rightarrow v_N$, where $N$ is the number of joints in the skeleton. The skeleton tree can be unfolded into a chain with a depth-first order, with the joint visiting order being $order^{L_J}$ : $v_1 \rightarrow v_2 \rightarrow v_3 \rightarrow v_2 \rightarrow v_4 \rightarrow v_5 \rightarrow v_6 \rightarrow v_7 \rightarrow v_6 \rightarrow v_5 \rightarrow v_4 \rightarrow v_2 \rightarrow \ldots \rightarrow v_{19} \rightarrow v_{18} \rightarrow v_{17} \rightarrow v_{12} \rightarrow v_1$, where $L_J = 2*N-1$ is the number of joints in the first order. With the proposed $order^{L_J}$, we map the skeleton to a $JSG \in \mathcal{R}^{L_J \times L_J \times 9}$, as depicted in Fig. 3a. Each cell and the horizontal and vertical axes in $JSG$ can be identified with $order^{L_J}$. To model the intrinsic dependency accurately, adjacency matrix $A^J \in \mathcal{R}^{L_J \times L_J}$ is constructed. Here, we set $a^J_{ij} = 0$ to discard not only self-connections but

| | $v_1$ | $v_2$ | $v_3$ | $v_2$ | $v_4$ | $v_5$ | $v_6$ | $v_7$ | $v_6$ | $v_5$ | $v_4$ | $v_2$ | $v_8$ | $v_9$ | $v_{10}$ | $v_{11}$ | $v_{10}$ | $v_9$ | $v_8$ | $v_2$ | $v_1$ | $v_{12}$ | $v_{13}$ | $v_{14}$ | $v_{15}$ | $v_{16}$ | $v_{15}$ | $v_{14}$ | $v_{13}$ | $v_{12}$ | $v_{17}$ | $v_{18}$ | $v_{19}$ | $v_{20}$ | $v_{19}$ | $v_{18}$ | $v_{17}$ | $v_{12}$ | $v_1$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $v_2$ | ● | | ● | | ● | | | | | | ● | | ● | | | | | | ● | | ● | | | | | | | | | | | | | | | | | | |
| $v_1$ | | ● | | ● | | | | | | | | ● | | | | | | | | ● | | ● | | | | | | | | ● | | | | | | | | ● | |
| $v_{12}$ | ● | | | | | | | | | | | | | | | | | | | | ● | | ● | | | | | | ● | | ● | | | | | | ● | | ● |

(a)



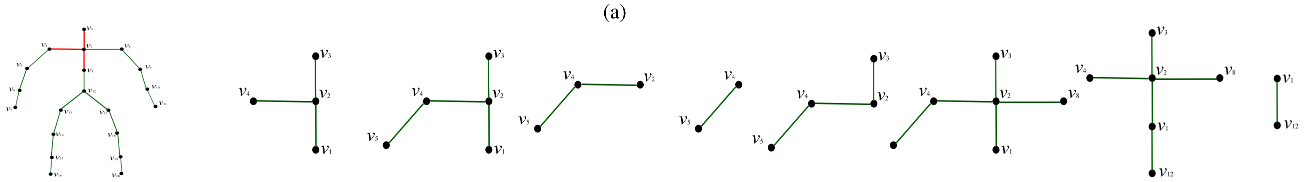
(b)

(c)

FIGURE 4. (a) Part of *JSG* depicting the intrinsic dependency among three joints $\{v_1, v_2, v_{12}\}$ and other joints in the human skeleton. (b) The three joints $\{v_1, v_2, v_{12}\}$ correspond to the torso as indicated by the red marker. (c) Given a convolution operator with a kernel size of 3 × 3, the neighboring rigid bodies with respect to the torso can be automatically programmed. Various parts of the body centered on the trunk can be convoluted.

| | $v_1$ | $v_2$ | $v_3$ | $v_2$ | $v_4$ | $v_5$ | $v_6$ | $v_7$ | $v_6$ | $v_5$ | $v_4$ | $v_2$ | $v_8$ | $v_9$ | $v_{10}$ | $v_{11}$ | $v_{10}$ | $v_9$ | $v_8$ | $v_2$ | $v_1$ | $v_{12}$ | $v_{13}$ | $v_{14}$ | $v_{15}$ | $v_{16}$ | $v_{15}$ | $v_{14}$ | $v_{13}$ | $v_{12}$ | $v_{17}$ | $v_{18}$ | $v_{19}$ | $v_{20}$ | $v_{19}$ | $v_{18}$ | $v_{17}$ | $v_{12}$ | $v_1$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $v_1$ | | ● | | ● | | | | | | | | ● | | | | | | | | ● | | ● | | | | | | | | ● | | | | | | | | ● | |
| $v_2$ | ● | | ● | | ● | | | | | | ● | | ● | | | | | | ● | | ● | | | | | | | | | | | | | | | | | | ● |
| $v_3$ | | ● | | ● | | | | | | | | ● | | | | | | | | ● | | | | | | | | | | | | | | | | | | | |
| $v_2$ | ● | | ● | | ● | | | | | | ● | | ● | | | | | | ● | | ● | | | | | | | | | | | | | | | | | | |
| $v_4$ | | ● | | ● | | ● | | | | ● | | ● | | | | | | | | ● | | | | | | | | | | | | | | | | | | | |

(a)



(b)

(c)

FIGURE 5. (a) Part of *JSG* depicting the intrinsic dependency among four joints $\{v_1, v_2, v_3, v_4\}$ and other joints in the human skeleton. (b) and (c) depict the 5 × 5 convolution filter applied to joints $\{v_1, v_2, v_3, v_4\}$.

also physically disconnected joints. As depicted in Fig. 3a, when $a^J_{ij} = 1$, there is a black dot in the corresponding cell $JSG(i, j)$, indicating that the feature vector of a rigid body $Rig(order^{L_J}(i), order^{L_J}(j))$ is computed using Eq. 1. When $a^J_{ij} = 0$, there is nothing in the corresponding cell $JSG(i, j)$, indicating that there is a vector whose values are all zero. Formally,

$$JSG(i, j) = \begin{cases} Rig(order^{L_J}(i), order^{L_J}(j)), & if\ a^J_{ij} = 1 \\ 0, & else. \end{cases} \quad (6)$$

The implementation of graph-based convolution is not as straightforward as that of 2D or 3D convolution. The sampling function of ST-GCN [11] is defined using only the 1-distance neighbor set of a joint. A partitioning strategy is required to divide the adjacent sets of a joint into a fixed number of subsets. This is empirically designed and difficult to generalize to various skeleton structures. In *JSG*, each cell will have the same number of neighbors and the same relationships to a neighbor in a given direction; thus, a sampling function is not necessary. The values read by the receptive fields are transformed into a linear layer and fed into a convolutional architecture. The receptive fields are not limited to the 1-distance neighbor set, as depicted in Figs. 4 and 5.

### 2) GRID REPRESENTATION OF EXTRINSIC DEPENDENCY

As mentioned previously, the skeleton graph employed in ST-GCN [11] merely represents the physical structure and cannot be used to learn extrinsic dependencies. To resolve this, we introduce *RSG* to learn the relations among various body parts. First, based on the principle that each rigid body in the skeleton is regarded as a vertex in a tree, the skeleton is transformed into a tree structure, as depicted in Fig. 2c. As mentioned previously, the skeleton tree can be unfolded into a chain with a depth-first order $order^{L_R}$: $e_1 \rightarrow e_2 \rightarrow e_1 \rightarrow e_3 \rightarrow e_4 \rightarrow e_5 \rightarrow e_6 \rightarrow e_5 \rightarrow e_4 \rightarrow e_3 \rightarrow e_1 \rightarrow e_7 \rightarrow \ldots \rightarrow e_{17} \rightarrow e_{16} \rightarrow e_{11} \rightarrow e_1$, where $L_R = 2 * (N - 1) - 1$ is the number of rigid bodies in the depth-first order. With the proposed order, we map the skeleton to an $RSG \in \mathcal{R}^{L_R \times L_R \times 6}$, as depicted in Fig. 3b. The extrinsic dependency among various body parts can be encoded using *RSG*. Similarly, to obtain further information about extrinsic dependency, we construct adjacency matrix $A^R \in \mathcal{R}^{L_R \times L_R}$. Here, we set $a^R_{ij} = 0$ to discard not only self-connections but also physically connected rigid bodies with intrinsic dependency. As depicted in Fig. 3b, when $a^R_{ij} = 1$, there is a black dot in the corresponding cell $RSG(i, j)$, indicating that feature vector $p_v(order^{L_R}(i), order^{L_R}(j))$ in cell $(i, j)$ representing the relative geometry of $e_i$ and $e_j$ will be given by Eq. 5. When $a^R_{ij} = 0$, there is nothing in the corresponding cell $RSG(i, j)$, indicating that there is a vector whose values are all zero. Formally,

$$RSG(i, j) = \begin{cases} p_v(order^{L_R}(i), order^{L_R}(j)), & if\ a^R_{ij} = 1 \\ 0, & else. \end{cases} \tag{7}$$

In $RSG$, the receptive field at any level can represent the relative geometry of various parts of the human body. As depicted in Fig. 6, $\{e_4, e_5, e_6\}$ can be expressed as a connected set of rigid segments (right arm), and $\{e_8, e_9, e_{10}\}$ can be considered the left arm. The red receptive field describes the relative geometry from the right to the left arm. Similarly, the blue receptive field describes the relative geometry from the left to the right arm. Therefore, for an arbitrary body part $b_i$, we can completely and hierarchically learn the relative geometry of $b_i$ and other body parts from this $RSG$.

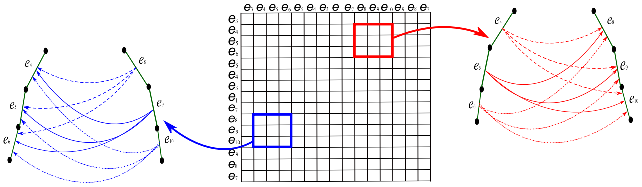

e4
e8
e5
e9
e6
e10
e3 e4 e5 e6 e5 e4 e3 e1 e7 e8 e9 e10 e9 e8 e7
e3 e4 e5 e6 e5 e4 e3 e1 e7 e8 e9 e10 e9 e8 e7
e4
e8
e5
e9
e6
e10

**FIGURE 6. Relative geometry of various body parts can be represented using the relative geometry of all pairs of rigid bodies.**

Each skeleton sequence has a different duration. To normalize the temporal length of gestures, we first resampled each gesture sequence to $T$ frames using the nearest neighbor interpolation by dropping or repeating frames. Therefore, for a dynamic skeleton of length $T$, we can formulate an action sequence as a stream of grids $SSG = (SSG_1, SSG_2, \ldots, SSG_t, \ldots, SSG_T)$, where $SSG_t = \langle JSG_t, RSG_t \rangle$ denotes the skeleton at the $t$-th time slice.

## C. 3D CONVOLUTIONAL NEURAL NETWORKS

CNNs are generally composed of convolutional, pooling, and fully connected layers. In conventional CNNs, 2D convolution extracts 2D feature blocks from local neighborhoods, which correspond to a 2D convolutional kernel. In 3D convolutional layers, a 3D convolution kernel is used to extract a 3D feature block into a cube composed of a group of neurons. Different 3D convolution kernels are used to process different feature input blocks, each of which corresponds to a convolution kernel. On an input 3D cube with the same 3D convolution kernel, the 3D feature map is obtained by overlapping convolution. Compared with 2D CNNs, 3D CNNs can simultaneously learn features from both spatiotemporal dimensions by capturing correlations among 3D signals.

We present the proposed strategy of adopting a two-stream 3D CNN architecture to capture spatiotemporal information by performing 3D convolutions, thereby capturing the motion information encoded in multiple adjacent human postures [17]. As depicted in Fig. 1, the architecture of the JSG stream is the same as that of the RSG stream. For an individual stream, each 3D CNN consists of two layers of 3D convolution followed by max-pooling. We extract the output of the last convolutional layer by feeding $(JSG_1, JSG_2, \ldots, JSG_T)$ and $(RSG_1, RSG_2, \ldots, RSG_T)$ to the corresponding 3D CNN. There are two sets of grids $X^{JSG} = \{X^{JSG}_1, \ldots, X^{JSG}_s, \ldots, X^{JSG}_S\}$ and $X^{RSG} = \{X^{RSG}_1, \ldots, X^{RSG}_s, \ldots, X^{RSG}_S\}$. For example, $X^{RSG}$ is a 4D cube of the form $K \times K \times D \times S$, as depicted in Fig. 7, where $K = 5$, $S = 4$, and $D = 48$. Therefore, on each grid of $X^{JSG}_s$ and $X^{RSG}_s$, we extract $K^2$ $D_1$-dimensional and $K^2$ $D_2$-dimensional vectors, respectively. We refer to these vectors as feature slices in a video snippet.

$$\begin{aligned} X^{JSG}_s &= [X^{JSG}_{s,1}, X^{JSG}_{s,2}, \ldots, X^{JSG}_{s,K^2}], \qquad X^{RSG}_{s,i} \in \mathbb{R}^{D_1}, \\ X^{RSG}_s &= [X^{RSG}_{s,1}, X^{RSG}_{s,2}, \ldots, X^{RSG}_{s,K^2}], \qquad X^{RSG}_{s,i} \in \mathbb{R}^{D_2} \end{aligned} \tag{8}$$

where $X^{JSG}_s$ and $X^{RSG}_s$ are the feature cubes of the $s$-th grid. These represent a subsequence of an action, and each element $X^{RSG}_{s,i}$ is the feature of a body part in this video snippet.

The $K^2$ vectors in each grid correspond to $K^2$ body parts in a subsequence, which essentially encode not only the spatial structure but also the temporal information. For action recognition over a period of time, not every body part is relevant. The proposed model focuses on those $K^2$ body parts where the action occurs. For better readability, $X_s$ represents the feature vectors $X^{JSG}_s$ and $X^{RSG}_s$ as the input to the attention model.

## D. ATTENTION MECHANISM

LSTM [22], [29] can preserve sequence information over time and capture long-term dependencies. We follow the LSTM implementation in [27], that is,

$$\begin{pmatrix} i_t \\ f_t \\ o_t \\ g_t \end{pmatrix} = \begin{pmatrix} \sigma \\ \sigma \\ \sigma \\ tanh \end{pmatrix} T_{d+D,4d} \begin{pmatrix} h_{t-1} \\ x_t \end{pmatrix},$$

$$c_t = f_t \odot c_{t-1} + i_t \odot g_t h_t = o_t \odot tanh(c_t) \tag{9}$$

where $i_t$, $f_t$, and $o_t$ are the input, forget, and output gates, respectively. $c_t$ is the cell (memory) state, and $h_t$ is the hidden state. $T : \mathbb{R}^{d+D} \rightarrow \mathbb{R}^{4d}$ is an affine transformation with trainable parameters, where $d$ is the dimensionality of $i_t, f_t, o_t, g_t, c_t$, and $h_t$. $\sigma(\cdot)$ is the sigmoid function, and $\odot$ denotes the Hadmard product.

Therefore, it is natural to design an LSTM subnetwork that assigns attention masks $\mathcal{M}_s$ to various body parts of the skeleton based on the content of the video snippet, as depicted in Fig. 8a. Because video frames are sequential, different video snippets have strong dependencies. We can use the encoded $X_{s-1}$ to predict the attention masks $\mathcal{M}_s$ at $X_s$, and then use the attention masks to refine the input to the LSTM, as depicted in Fig. 8b. In particular, we use a location softmax function over $K \times K$ locations with $tanh$ activation to predict the importance of the $K^2$ locations in the frame, which can be

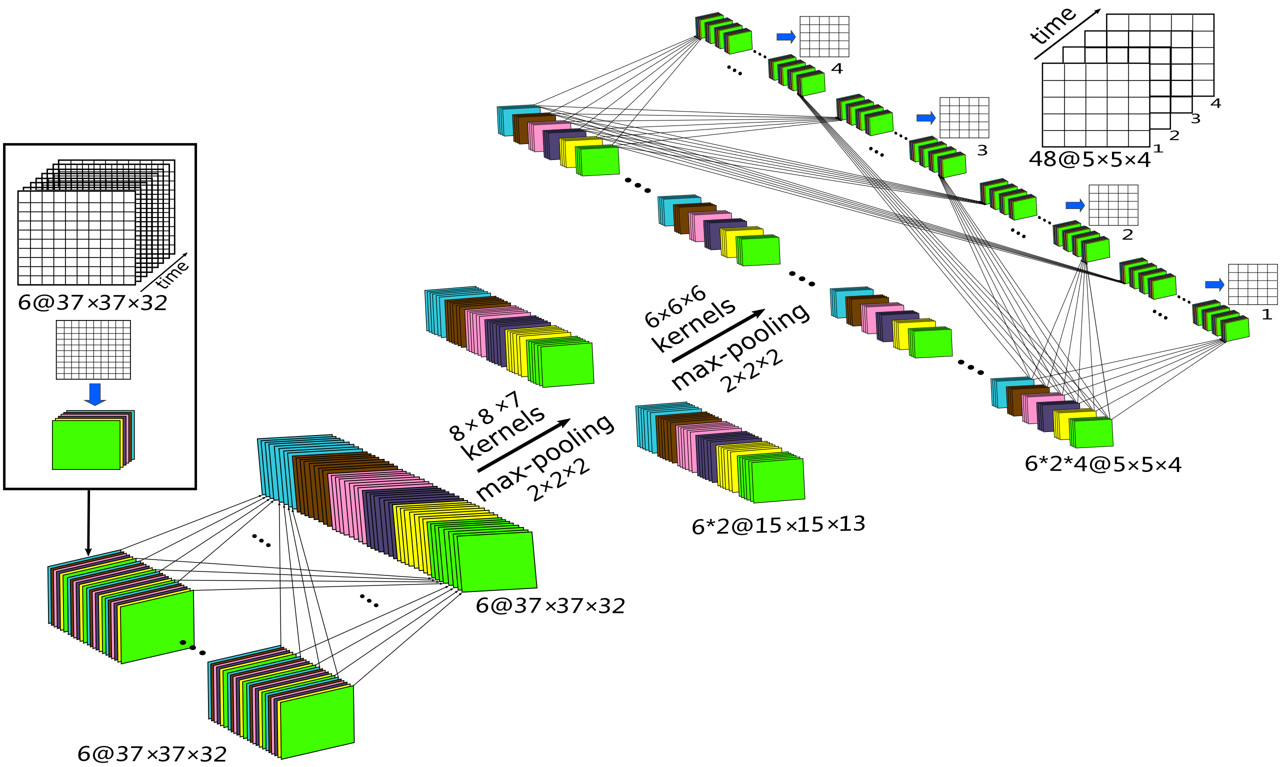


**FIGURE 7.** 3D CNN architecture to capture spatiotemporal information of RSG by performing 3D convolutions and max-pooling, considering 20 joints of the human skeleton as an example.

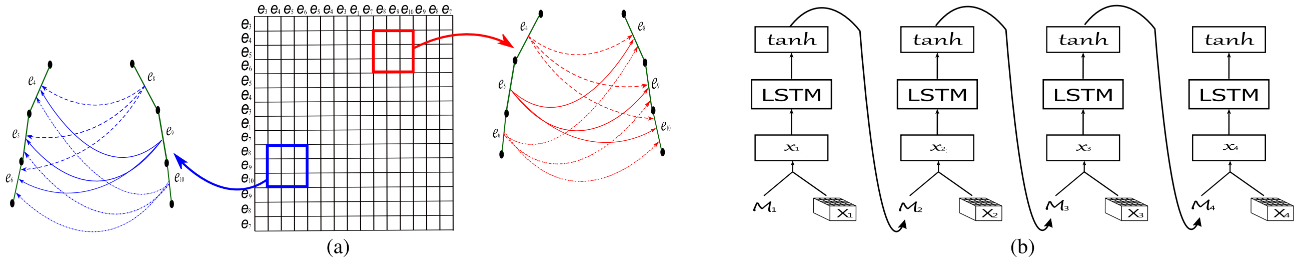


**FIGURE 8.** Structure of soft attention network.

expressed as

$$\mathcal{M}_{s,i} = \frac{exp(w_i^T h_{s-1})}{\sum_{j=1}^{K^2} exp(w_j^T h_{s-1})}, \quad i \in 1 \ldots K^2 \tag{10}$$

where $\mathcal{M}_{s,i}$ is the importance weight of the $i$-th body part of the $s$-th grid, and $W = \{w_1, w_2, \ldots, w_{K^2}\} \in \mathbb{R}^{2D\times K^2}$ are the weights of the softmax function [30]. Note that we compute two attention weights $\mathcal{M}_{s,i}^{JSG}$ and $\mathcal{M}_{s,i}^{RSG}$ separately. *JSG* and *RSG* capture the intrinsic dependency among joints and extrinsic dependency among rigid bodies, respectively. Therefore, two sets of attention masks for $X_s^{JSG}$ and $X_s^{RSG}$ must be calculated separately. With the aforementioned attention masks, the inputs to the two LSTMs are the weighted averages of the various locations as

$$x_s^{JSG} = \sum_{i=1}^{K^2} \mathcal{M}_{s,i}^{JSG} X_{s,i}^{JSG} \text{ and } x_s^{RSG} = \sum_{i=1}^{K^2} \mathcal{M}_{s,i}^{RSG} X_{s,i}^{RSG} \tag{11}$$

### E. ACTION RECOGNITION

The output $x_s^{JSG}$ and $x_s^{RSG}$ of the attention layer encode complementary information from the *intrinsic* and *extrinsic* dependencies among joints or rigid bodies. $x_s^{JSG}$ and $x_s^{RSG}$ are fed to two fully connected layers with 512 and 256 neurons, respectively. Therefore, the action classifier consists of two networks: a joint grid network and a rigid-body grid network, with network parameters $W_J$ and $W_R$, respectively. Given the action sequence *SSG*, we multiply the class membership probabilities $P(C|x, W_J)*P(C|x, W_R)$ from the two networks for the action classifier, and the class label is predicted as $c^* = argmaxP(C|x, W_J) * P(C|x, W_R)$.

The negative log-likelihood loss function [31] for dataset $D$ is used to measure the difference between the true label $c$ and the predicted result $c^*$ as follows:

$$L(W_J, W_R, D) = -\frac{1}{|D|}\sum_{i=0}^{|D|} log(P(C^{(i)}|x^{(i)}, W_J, W_R). \tag{12}$$

The entire model is differentiable with the embedded attention mechanism; therefore, back-propagation can be used to minimize the loss function, and the entire framework can be trained end-to-end.

## IV. EXPERIMENTS AND RESULTS

The skeleton sequences under various perspectives are significantly different, even for the same action. Therefore, the skeleton sequences are transformed by placing the coordinates of the hip center at the origin (0, 0). Motivated by the recent work [14], skeleton sequences are transformed into a suitable view for better action recognition. With the new origin of the hip center, the skeleton rotates parallel to the horizontal axis X. Moreover, the lengths of all body parts are normalized with respect to a reference skeleton.

The proposed method was evaluated on three action recognition datasets: NTU RGB+D [21], SBU Kinect Interaction [32], and Kinetics [33]. To evaluate the effectiveness of the proposed model, we performed extensive experiments using four different configurations as follows:

- JSG is the baseline, and the physical structure without extrinsic dependency is considered. The input data contain nine channels.
- RSG considers only extrinsic dependency using the rigid-body geometrical transformation. The input data contain six channels.
- SSG (JSG+RSG) is a network considering both intrinsic and extrinsic dependencies. To combine the two types of input information, we model them using two 3D CNNs. The outputs of the softmax layer in the CNNs are added to obtain the final score, as depicted in Fig. 1.
- SSG+attention is a network where the attention mechanism is added.

### A. IMPLEMENTATION

The model takes an SSG (JSG and RSG) sequence as the input, which is a 4D tensor. The architecture of the network consists mainly of a 3D CNN, an attention mechanism, and fully connected layers. Finally, the output is fed to a softmax classifier [34].

As depicted in Fig. 7, the human body comprises 20 major joints. Therefore, each action sequence can be represented as a $39 \times 39 \times 9 \times 32$ tensor for JSG (intrinsic dependency) and $37 \times 37 \times 6 \times 32$ tensor for RSG (extrinsic dependency). Considering RSG as an example, the $37 \times 37 \times 6 \times 32$ tensor was fed to a 3D CNN with two convolutional layers, $H_1$ and $H_2$. First, we applied 3D convolutions with a kernel size of $8 \times 8 \times 7$ ($8 \times 8$ in the spatial dimension and 7 in the temporal dimension) on each channel separately. In the subsequent max-pooling, we applied $2 \times 2 \times 2$ subsampling. The next convolution and max-pooling were obtained by applying a kernel size of $6 \times 6 \times 6$ and max pooling $2 \times 2 \times 2$. After the two layers of convolution and max-pooling, the 32 frames were converted into a 4D cube of the form $5 \times 5 \times 48 \times 4$. Finally, we extracted $5^2$ 48-dimensional vectors to evaluate the importance weight of the body part.

In this study, the 3D CNN architecture consisted of two layers of 3D convolution followed by max-pooling. The reason for using these two layers for the 3D CNN architecture is as follows: for each training action, we generated a 4D cube of a form beyond a feature vector encoding the long-term action information. In this design, it is desirable to selectively focus on the information of the body parts by introducing a soft attention model.

The implementation details are based on those of the original CNN, as described in [38] and [39]. We trained the network from scratch. All the model parameters were randomly initialized as in [38] and learned using the stochastic gradient descent algorithm [39]. The applied dropout probability was 0.5 to avoid over-fitting. All experiments were performed in the PyTorch deep learning framework [40] on two Nvidia GTX 1080i GPUs with a batch size of 16. The initial learning rate was set to 0.1 and reduced by multiplying it by 0.1 every 20 epochs. The training process ended at the 80th epoch. In the attention mechanism layer, the classic LSTM unit was employed to model the dependency among various video snippets, where the dimension of the hidden units was set to 256.

In the network, the most important parameters ($k$ and $t$) are the sizes of the receptive fields. $k$ and $t$ are the sizes in the spatial and temporal domains, respectively. As $k$ and $t$ increase, the local filtering region becomes larger. Note that when $k = 1$ and $t > 1$, only temporal filtering is performed, and the convolution on a grid is only in the cell itself (with no neighbors). When $t = 1$ and $k > 1$, only spatial filtering is performed. By considering the impact of various values of $k$ and $t$ in different datasets, we conclude that spatial and temporal filtering together can improve the action recognition performance.

### B. EXPERIMENTS ON NTU RGB+D DATASET

**NTU RGB+D dataset.** The NTU RGB+D dataset [21] is a large-scale RGBD dataset for skeleton-based action recognition. It contains 56880 action sequences and four million frames. The dataset is based on 40 human subjects and has 60 action classes, including 50 single-person actions and 10 actions performed by two people. The dataset was obtained using Kinect. Each human body is represented by 25 major joints. Each action was captured by three cameras at the same height simultaneously but from different horizontal angles: −45°, 0°, and 45°. Every action was performed twice, with the performer facing the left or right sensor. Moreover, the height of the sensors and their distances to the action performer were adjusted to obtain further viewpoint variations. We applied a normalization preprocessing step, as in [21], for position and view invariance [21]. To avoid affecting the continuity of a sequence, no temporal down-sampling was performed.

We evaluated the proposed model using two standard evaluation protocols proposed in [21]: cross-subject (X-Sub) and cross-view (X-View). In the cross-subject evaluation, 40320 action sequences from 20 subjects were used for

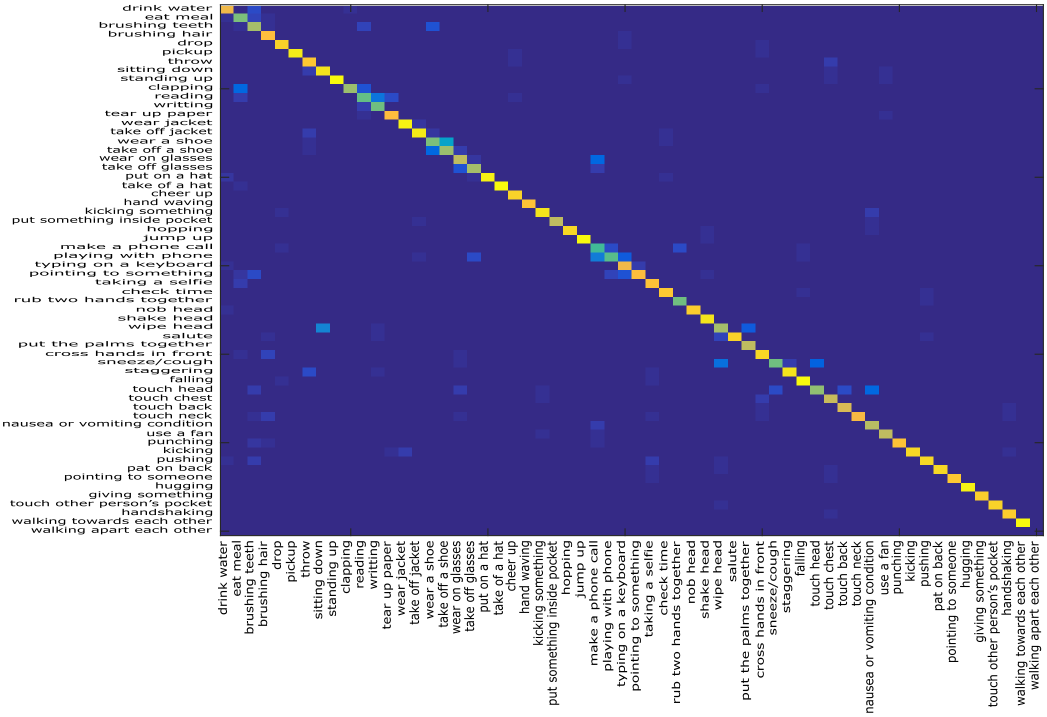


**FIGURE 9.** Confusion matrix on the NTU RGB+D dataset with the cross-subject evaluation protocol.

training, and the remaining 16560 action sequences were used for testing. In the cross-view evaluation, 37920 action sequences captured from cameras 2 and 3 were used for training, and the other 18960 action sequences from camera 1 were used for testing.

Each human body skeleton is represented by the 3D spatial coordinates of 25 major body joints. Therefore, each action sequence can be represented as a $49 \times 49 \times 9 \times 32$ tensor for intrinsic dependency and a $47 \times 47 \times 6 \times 32$ tensor for extrinsic dependency. These two tensors were fed to a 3D CNN with two convolutional layers. The temporal kernel size $t$ was selected between 1 and 10, and the spatial kernel size $k$ was selected from $\{1, 2, 4, 6\}$. Cross-comparison results are depicted in Fig. 10. The number of channels was four or eight. We adopted two max-pooling layers of size $2 \times 2 \times 2$ after each convolution layer.

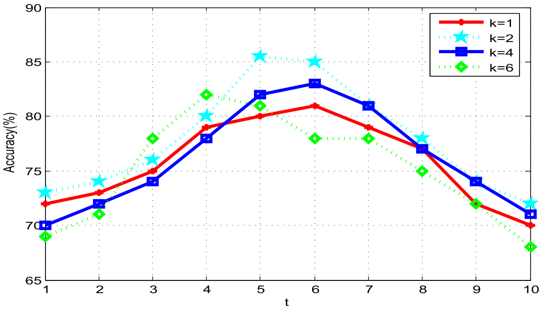


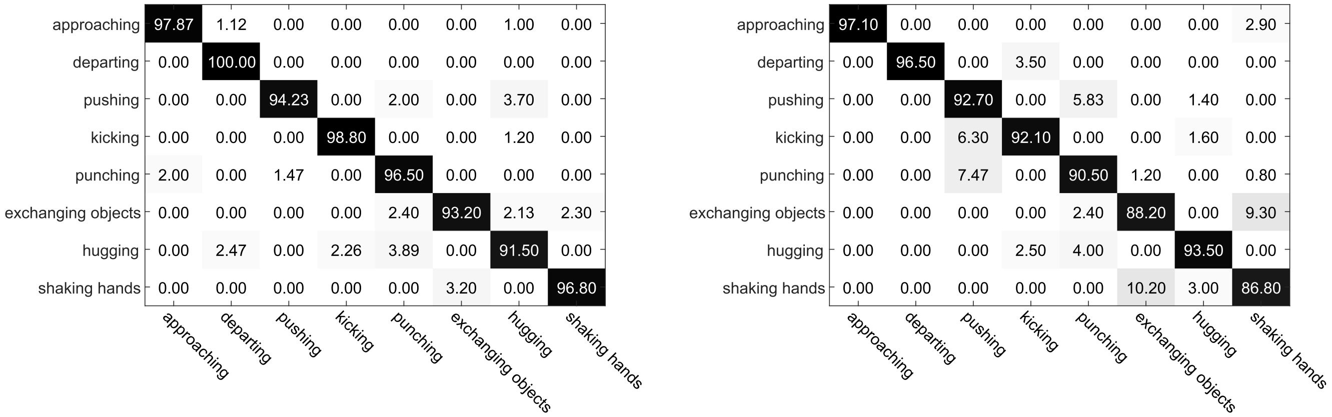
**FIGURE 10.** Comparison of the sizes of various convolutional kernels on the NTU RGB+D dataset with the cross-subject protocol.

Table 1 lists the performances of various methods on NTU RGB+D. It is evident that the performance of deep learning approaches is generally better than those of the hand-crafted feature methods. This demonstrates that JSG and RSG streams are complementary, as their fusion significantly improves on each of them (4.2% over JSG and 2.6% over RSG).

**TABLE 1.** Recognition rates (%) with state-of-the-art approaches on the NTU RGB+D dataset with cross-subject and cross-view settings.

| Methods | Cross-Subject | Cross-View |
|---|---|---|
| C-CNN+MTLN [6] | 79.6 | 84.8 |
| Temporal Conv [7] | 74.3 | 83.1 |
| ST-LSTM+TS [35] | 69.2 | 77.7 |
| TSSI+GLAN+SSAN [8] | 82.4 | 89.1 |
| GE-GCN [36] | 84 | 89.4 |
| ST-GCN [11] | 81.5 | 88.3 |
| motif-GCNs+non-local VTDB [37] | 84.2 | 90.2 |
| STG-IN [12] | 85.8 | 88.7 |
| VA-fusion(aug.) [14] | 89.4 | 95.0 |
| JSG | 81.7 | 87.2 |
| RSG | 82.8 | 88.9 |
| SSG | 85.7 | 91.3 |
| SSG+attention | 86.5 | 91.9 |
| SSG+attention+view | **90.2** | **95.7** |

Based on the data listed in Table 1, it is evident that the proposed network performs poorly when compared with the performances of the VA-fusion models [14]. This is because action recognition with a view adaptation subnetwork [14] extracts the features of action sequences after determining the virtual observation viewpoints. Adding this subnetwork



(a)

(b)

**FIGURE 11.** Confusion matrices for five-fold cross validation using (a) SSG+attention and (b) baseline method (JSG) on the SBU Kinect Interaction dataset.

to the SSG+attention model leads to a 3.8% increase for the cross-view evaluation protocol, even better than that of VA-fusion (0.7%).

The confusion matrix on the NTU RGB+D dataset with the cross-subject setting is depicted in Fig. 9. It is evident that the proposed model makes a good distinction between an action involving one person and an interaction between two persons. Single-person actions are still confused because of the similarity of two actions in motion patterns, such as *reading* and *writing*.

### C. EXPERIMENTS ON SBU KINECT INTERACTION DATASET

**SBU Kinect Interaction dataset.** The SBU Kinect Interaction dataset [32] is an interaction dataset with each action performed by two experimental subjects in the same laboratory environment, such as *approaching and departing* and *shaking hands and changing objects*. A total of 282 skeleton sequences are performed by seven experimental subjects, corresponding to 6822 frames, with an average of 25 frames per skeleton sequence. It contains eight action classes: *approaching*, *departing*, *pushing*, *kicking*, *punching*, *exchanging objects*, *hugging*, and *shaking hands*. The evaluation was performed using a five-fold cross-validation protocol (i.e., four used for training and one for testing).

Each human body skeleton is represented by the 3D spatial coordinates of 15 major body joints. Each action sequence can be represented as a 29 × 29 × 9 × 24 tensor for JSG (intrinsic dependency) and 27 × 27 × 6 × 24 tensor for RSG (extrinsic dependency). When there are two people in the action recognition task, two skeleton trees are unfolded into two chains in the depth-first order. Each action sequence of the two-person interaction can be represented as a 58 × 58 × 9 × 24 tensor for JSG (intrinsic dependency) and a 54 × 54 × 6 × 24 tensor for RSG (extrinsic dependency). The corresponding convolution kernel size $t$ is 9 × 9 × 3 and 8 × 8 × 3, respectively. We adopted two max-pooling layers of size 2 × 2 × 2 after each convolution layer.

Fig. 11 depicts the comparison of the confusion matrices for SSG+attention and the baseline method on the SBU Kinect Interaction dataset. It is clear that when two persons have relatively simple body interactions, such as approaching or departing, both methods are effective. However, when the interaction is more complex, such as hugging or punching, the proposed method is more effective than the joint feature method. It is evident that both models are effective when there is relatively simple body interaction between two persons, such as *leaving* and *approaching*. However, when there are more complex interactions, such as *kicking*, *punching*, or *hugging*, SSG+attention is more effective than the baseline method, which considers only the intrinsic dependency.

A comparison of the proposed network with state-of-the-art methods is presented in Table 2. The proposed SSG+attention framework exhibits consistently high recognition performance on the SBU Kinect Interaction dataset. It achieves a 2.9% improvement over the baseline method.

**TABLE 2.** Recognition rates (%) on the SBU Kinect Interaction dataset.

| Methods | Accuracy |
|---|---|
| Hierarchical RNN [41] | 80.35 |
| CHARM [42] | 83.9 |
| Deep LSTM [22] | 86.03 |
| Joint Feature [43] | 86.9 |
| ST-LSTM+Trust Gate [35] | 93.3 |
| Clips+CNN+MTLN [6] | 93.6 |
| TSSI+SSAN+GLAN [8] | 95.7 |
| JSG | 92.2 |
| RSG | 94.1 |
| SSG | 95.3 |
| SSG+attention+view | **96.1** |

### D. EXPERIMENTS ON Kinetics–Motion DATASET

**Kinetics–Motion dataset** The Kinetics dataset [33] is one of the largest human action datasets. It contains 300,000 video clips, each of approximately 10 s duration. To cover as many real occasions as possible, Kinetics comprises action sequences from YouTube, and 400 human action classes are covered. The dataset provides only raw video clips without skeleton data. To perform joint-based action recognition, we used the pre-calculated estimated poses provided in [44].

The Kinetics–Motion dataset is proposed for a better evaluation of skeleton-based methods on estimated joints, which is a 30-class subset of Kinetics with action labels strongly related to body motion. Each human body skeleton is represented by the 3D spatial coordinates of 18 major body joints. The 30 selected classes are as follows: *belly dancing*, *punching bag*, *capoeira*, *squat*, *windsurfing*, *skipping rope*, *swimming backstroke*, *hammer throw*, *throwing discus*, *tobogganing*, *hopscotch*, *hitting baseball*, *roller skating*, *arm wrestling*, *snatch weight lifting*, *tai chi*, *riding mechanical bull*, *salsa dancing*, *hurling* (*sport*), *lunge*, *skateboarding*, *country line dancing*, *juggling balls*, *surfing crowd*, *dead lifting*, *clean and jerk*, *crawling baby*, *push up*, *front raises*, and *pullups*.

We evaluated the proposed method on the Kinetics–Motion dataset containing 266440 samples. The samples were divided into the training set (246534 clips) and validation set (19906 clips). Following the evaluation method in [33], we trained the models on the training set and reported both the top-1 and top-5 accuracies on the validation set. The data listed in Table 3 indicate that the proposed network achieved superior performance to that of existing skeleton-based methods.

**TABLE 3. Comparison with state-of-the-art methods on the Kinetics dataset.**

| Method | Top-1(%) | Top-5(%) |
|---|---|---|
| Feature Enx. [45] | 14.9 | 25.8 |
| Deep LSTM [22] | 16.4 | 35.3 |
| TCN [7] | 20.3 | 40.0 |
| ST-GCN [11] | 30.7 | 52.8 |
| 2S-ASGCN [46] | 34.5 | 56.9 |
| SSG+attention+view | 39.8 | 62.6 |

## V. CONCLUSION AND FUTURE WORK

We proposed a two-stream 3D CNN for skeleton-based action recognition. To extract the dependencies among various body parts, we transformed the skeleton data into JSG and RSG, and then performed 3D convolution and 3D max-pooling on the JSG and RSG streams, respectively. To detect the salient action units crucial for identifying motions, we introduced an attention mechanism to learn masks on JSG and RSG. The output of the action attention layer was finally fed to a fully connected layer and a softmax layer to predict the class label. Extensive experiments and analyses indicated that the modules of the 3D CNN and attention masks can further improve the recognition performance. In the future, we will focus on various methods for encoding skeleton data to adapt the CNN more effectively.

## REFERENCES

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, May 2017.

[2] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1725–1732.

[3] A. Graves, A.-R. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, May 2013, pp. 6645–6649.

[4] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2013, pp. 1310–1318.

[5] R. Pascanu, C. Gulcehre, K. Cho, and Y. Bengio, "How to construct deep recurrent neural networks," 2013, *arXiv:1312.6026*. [Online]. Available: http://arxiv.org/abs/1312.6026

[6] Q. Ke, M. Bennamoun, S. An, F. Sohel, and F. Boussaid, "A new representation of skeleton sequences for 3D action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3288–3297.

[7] T. S. Kim and A. Reiter, "Interpretable 3D human action analysis with temporal convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2017, pp. 1623–1631.

[8] Z. Yang, Y. Li, J. Yang, and J. Luo, "Action recognition with spatio–temporal visual attention on skeleton image sequences," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 8, pp. 2405–2415, Aug. 2019.

[9] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," 2016, *arXiv:1609.02907*. [Online]. Available: http://arxiv.org/abs/1609.02907

[10] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 29, 2016, pp. 3844–3852.

[11] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 11–20.

[12] W. Ding, X. Li, G. Li, and Y. Wei, "Global relational reasoning with spatial temporal graph interaction networks for skeleton-based action recognition," *Signal Process., Image Commun.*, vol. 83, Apr. 2020, Art. no. 115776.

[13] J. Liu, N. Akhtar, and A. Mian, "Skepxels: Spatio-temporal image representation of human skeleton joints for action recognition," in *Proc. CVPR Workshops*, 2019, pp. 1–10.

[14] P. Zhang, C. Lan, J. Xing, W. Zeng, J. Xue, and N. Zheng, "View adaptive neural networks for high performance skeleton-based human action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 8, pp. 1963–1978, Aug. 2019.

[15] H. H. Pham, H. Salmane, L. Khoudour, A. Crouzil, P. Zegers, and S. A. Velastin, "Spatio–temporal image representation of 3D skeletal movements for view-invariant action recognition with deep convolutional neural networks," *Sensors*, vol. 19, no. 8, p. 1932, Apr. 2019.

[16] H. H. Pham, H. Salmane, L. Khoudour, A. Crouzil, S. A. Velastin, and P. Zegers, "A unified deep framework for joint 3D pose estimation and action recognition from a single RGB camera," *Sensors*, vol. 20, no. 7, p. 1825, Mar. 2020.

[17] S. Ji, W. Xu, M. Yang, and K. Yu, "3D convolutional neural networks for human action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 221–231, Jan. 2013.

[18] C. Cao, Y. Zhang, C. Zhang, and H. Lu, "Body joint guided 3-D deep convolutional descriptors for action recognition," *IEEE Trans. Cybern.*, vol. 48, no. 3, pp. 1095–1108, Mar. 2018.

[19] Z. Gharaee, "Hierarchical growing grid networks for skeleton based action recognition," *Cognit. Syst. Res.*, vol. 63, pp. 11–29, Oct. 2020.

[20] Y. Du, W. Wang, and L. Wang, "Hierarchical recurrent neural network for skeleton based action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1110–1118.

[21] 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 Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1010–1019.

[22] W. Zhu, C. Lan, J. Xing, W. Zeng, Y. Li, L. Shen, and X. Xie, "Co-occurrence feature learning for skeleton based action recognition using regularized deep LSTM networks," in *Proc. 13th AAAI Conf. Artif. Intell.*, 2016, pp. 1–7.

[23] Q. Ke, M. Bennamoun, S. An, F. Sohel, and F. Boussaid, "Learning clip representations for skeleton-based 3D action recognition," *IEEE Trans. Image Process.*, vol. 27, no. 6, pp. 2842–2855, Jun. 2018.

[24] J. Thompson and R. Parasuraman, "Attention, biological motion, and action recognition," *NeuroImage*, vol. 59, no. 1, pp. 4–13, Jan. 2012.

[25] S. Song, C. Lan, J. Xing, W. Zeng, and J. Liu, "An end-to-end spatio-temporal attention model for human action recognition from skeleton data," 2016, *arXiv:1611.06067*. [Online]. Available: http://arxiv.org/abs/1611.06067

[26] J. Liu, G. Wang, P. Hu, L.-Y. Duan, and A. C. Kot, "Global context-aware attention LSTM networks for 3D action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1647–1656.

[27] J. Zang, L. Wang, Z. Liu, Q. Zhang, G. Hua, and N. Zheng, "Attention-based temporal weighted convolutional neural network for action recognition," in *Proc. 14th IFIP WG 12.5 Int. Conf. Artif. Intell. Appl. Innov. (AIAI)*. Rhodes, Greece: Springer, May 2018, pp. 97–108.

[28] R. M. Murray, Z. Li, S. S. Sastry, and S. S. Sastry, *A Mathematical Introduction to Robotic Manipulation*. Boca Raton, FL, USA: CRC Press, 1994.

[29] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-C. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 28, 2015, pp. 802–810.

[30] J. Goodman, "Classes for fast maximum entropy training," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 1, May 2001, pp. 561–564.

[31] A. Graves, "Supervised sequence labelling," in *Supervised Sequence Labelling With Recurrent Neural Networks* (Studies in Computational Intelligence), vol. 385. Berlin, Germany: Springer, 2012, doi: 10.1007/978-3-642-24797-2_2.

[32] K. Yun, J. Honorio, D. Chattopadhyay, T. L. Berg, and D. Samaras, "Two-person interaction detection using body-pose features and multiple instance learning," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2012, pp. 28–35.

[33] W. Kay, S. Vijayanarasimhan, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, F. Viola, T. Green, T. Back, P. Natsev, M. Suleyman, and A. Zisserman, "The kinetics human action video dataset," 2017, *arXiv:1705.06950*. [Online]. Available: http://arxiv.org/abs/1705.06950

[34] K. Duan, S. S. Keerthi, W. Chu, S. K. Shevade, and A. N. Poo, "Multi-category classification by soft-max combination of binary classifiers," in *Multiple Classifier Systems* (Lecture Notes in Computer Science), vol. 2709, T. Windeatt and F. Roli, Eds. Berlin, Germany: Springer, 2003, doi: 10.1007/3-540-44938-8_13.

[35] J. Liu, A. Shahroudy, D. Xu, and G. Wang, "Spatio-temporal LSTM with trust gates for 3D human action recognition," in *Computer Vision—ECCV* (Lecture Notes in Computer Science), vol. 9907, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. Cham, Switzerland: Springer, 2016, doi: 10.1007/978-3-319-46487-9_50.

[36] X. Zhang, C. Xu, X. Tian, and D. Tao, "Graph edge convolutional neural networks for skeleton-based action recognition," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 8, pp. 3047–3060, Aug. 2020.

[37] Y.-H. Wen, L. Gao, H. Fu, F.-L. Zhang, and S. Xia, "Graph CNNS with motif and variable temporal block for skeleton-based action recognition," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 8989–8996.

[38] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[39] Y. A. LeCun, L. Bottou, G. B. Orr, and K. R. Müller, "Efficient BackProp," in *Neural Networks: Tricks of the Trade* (Lecture Notes in Computer Science), vol. 7700, G. Montavon, G. B. Orr, and K. R. Müller, Eds. Berlin, Germany: Springer, 2012, doi: 10.1007/978-3-642-35289-8_3.

[40] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in PyTorch," in *Proc. 31st Conf. Neural Inf. Process. Syst. (NIPS) NeurIPS Workshop*, Long Beach, CA, USA, 2017.

[41] J.-F. Hu, W.-S. Zheng, J. Lai, and J. Zhang, "Jointly learning heterogeneous features for RGB-D activity recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 5344–5352.

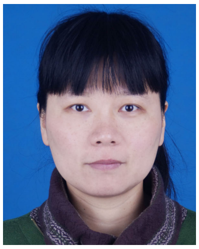
[42] W. Li, L. Wen, M. C. Chuah, and S. Lyu, "Category-blind human action recognition: A practical recognition system," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 4444–4452.

[43] Y. Ji, G. Ye, and H. Cheng, "Interactive body part contrast mining for human interaction recognition," in *Proc. IEEE Int. Conf. Multimedia Expo Workshops (ICMEW)*, Jul. 2014, pp. 1–6.

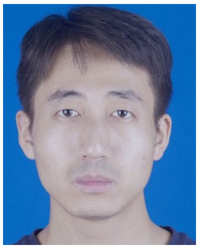
[44] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2D pose estimation using part affinity fields," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 7291–7299.

[45] B. Fernando, E. Gavves, M. J. Oramas, A. Ghodrati, and T. Tuytelaars, "Modeling video evolution for action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 5378–5387.

[46] L. Shi, Y. Zhang, J. Cheng, and H. Lu, "Adaptive spectral graph convolutional networks for skeleton-based action recognition," 2018, *arXiv:1805.07694*. [Online]. Available: http://arxiv.org/abs/1805.07694



**WENWEN DING** received the B.S. degree in computer science from the Hefei University of Technology, in 2002, and the M.S. and Ph.D. degrees in computer technology from Xidian University, in 2007 and 2017, respectively. She is currently an Associate Professor with the School of Mathematical Sciences, Huaibei Normal University, Anhui, China. Her research interests include human activity recognition, video semantics understanding, and so on.



**CHONGYANG DING** received the B.E. degree from Xidian University, Xi'an, China, in 2016, where he is currently pursuing the Ph.D. degree with the Department of Computer Science and Technology. His research interests include action recognition and video understanding.

**GUANG LI** received the M.S. degree in computer science from the Inner Mongolia University of Science and Technology, in 2009. He is currently pursuing the Ph.D. degree with Xidian University. He joined the School of Foreign Languages, Inner Mongolia University of Science and Technology, in 2009, where he is currently a Teacher. His research interests include action recognition, especially skeleton-based action recognition.

**KAI LIU** received the B.S. and M.S. degrees in computer science, and the Ph.D. degree in signal processing from Xidian University, Xi'an, China, in 1999, 2002, and 2005, respectively. He is currently a Professor of computer science and technology with Xidian University. He has attended China lunar explore project from Chang'er-1, 2, and 3 which he has been a chief designer for the core part of image compression hardware in the Chang'er satellite. He was also the chief designer for the first remote image compression chip which was used in several China satellite missions. He is the Chief Designer for the spectral images and Laser Induced Breakdown Spectroscopy (LIBS) images compression in the China Mars mission. He currently works with Huawei on the hardware architecture of single shot detection using CNN network in the field of ADAS. He has published more than 50 articles on image/data compression and image processing, including tracking, detection, and behavior analysis. His main research interests include FPGA/ASIC design for image and text coding. He has received several awards from different organizations, including the Shaanxi Science and Technology Award by Province Government, in 2009, the Best Teacher Award from EMC corporation, in 2009, 2010, the Excellent Cooperation Award from Huawei Corporation for his work on the GSM/UTMS data compression, in 2013, the Excellent Cooperation Teacher Award from IBM Corporation for his work on IBM-XIDIAN education project and SSD data processing project, in 2014.