Space-Time Skeletal Analysis with Jointly Dual-Stream ConvNet for Action Recognition

Thien Huynh-The*, Cam-Hao Hua†, Nguyen Anh Tu‡, and Dong-Seong Kim*
*ICT Convergence Research Center, Kumoh National Institute of Technology, Republic of Korea
† Department of Computer Science and Engineering, Kyung Hee University, Republic of Korea
‡ Department of Computer Science, Nazarbayev University, Republic of Kazakhstan
Email: {thienht,dskim}@kumoh.ac.kr, hao.hua@oslab.khu.ac.kr, tu.nguyen@nu.edu.kz

***Abstract*—In this decade, although numerous conventional methods have been introduced for three-dimensional (3D) skeleton-based human action recognition, they have posed a primary limitation of learning a vulnerable recognition model from low-level handcrafted features. This paper proposes an effective deep convolutional neural network (CNN) with a dual-stream architecture to simultaneously learn the geometric-based static pose and dynamic motion features for high-performance action recognition. Each stream consists of several advanced blocks of regular and grouped convolutional layers, wherein various kernel sizes are configured to enrich representational features. Remarkably, the blocks in each stream are associated via skip-connection scheme to overcome the vanishing gradient problem, meanwhile, the blocks of two stream are jointly connected via a customized layer to partly share high-relevant knowledge gained during the model training process. In the experiments, the action recognition method is intensively evaluated on the NTU RGB+D dataset and its upgraded version with up to 120 action classes, where the proposed CNN achieves a competitive performance in terms of accuracy and complexity compared to several other deep models.**

***Index Terms*—Action recognition, convolutional network, dual-stream architecture, geometric feature, 3D skeleton data.**

I. Introduction

In the last decades, analyzing and understanding human action from videos have been taken into consideration in various applications, such as behavior monitoring [1], intelligent surveillance [2], and smart healthcare [3]. Numerous previous action recognition methods have studied locally visual descriptive features from RGB image, which are critically borne several challenging problems [4], including dynamic illumination condition, object occlusion, and viewpoint variation [5]. By taking advantages of depth imaging technology developed in depth camera (for example, Microsoft Kinect and Intel RealSense) to overcome the above-mentioned problems, several recent approaches have leveraged the skeletal maps to improve the recognition performance in terms of accuracy and processing speed [6]. It is worth noting that some advanced pose estimation algorithms are initially embedded in these camera devices, where the accurate skeletal information (such as human body joint coordinates) of an object detected in range is specified to an additional output channel besides conventional RGB [7]. However, learning traditional classification models [8] with weakly-representational handcrafted features (such as 3D descriptive and geometric characteristics) have revealed the obscurity of discriminating actions which share similar static skeletal patterns [9].

This research was financially supported by National Research Foundation of Korea (NRF) through Creativity Challenge Research-based Project (2019R1I1A1A01063781), and in part by the Priority Research Centers Program through the NRF funded by the Ministry of Education, Science and Technology (2018R1A6A1A03024003).

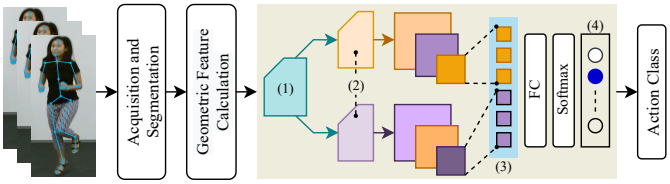

Fig. 1. The overall framework for 3D action recognition with a dual-stream CNN for learning geometric distance features of static pose and action dynamic. Annotation: (1)−extracted geometric features, (2)−static pose and dynamic motion feature maps, (3)−feature concatenation, and (4)−scores of predicted classes.

Recently, with the favorable specification of mining high-level representational features from high-dimensional unstructured data, deep learning technique has been intensively exploited for image classification and pixel-wise segmentation [10]–[12]. Recurrent neural network (RNN) [13], [14] and long short-term memory (LSTM) [15], [16] network were studied for 3D human action recognition. Despite the fact that these networks are capable of mining the spatial correlations between different skeletal joints in time for a better action discrimination, they cannot take into consideration of learning an entire action thoroughly [17]. To seize the high-level correlations between 3D body joints in both spatial and temporal domains concurrently, some methods take advantage of convolutional neural network (CNN) for multi-scale feature extraction. Several approaches either plot skeletal trajectory as graphical image or transform 3D joint coordinate to pixel value for a spatiotemporal action representation. In [18], the joint trajectories and dynamics were encoded into 2D color images, called joint trajectory maps (JTM), for mining the discriminative features using multi-stream CNN. Ke et al. [19] and Caetano et al. [20] converted coordinate values to illumination values for encoding each 3D

joint as a color pixel. Accordingly, the skeletal observation is represented by a color image, where the image size (width $\times$ height) is identical to the number of human joints in a full skeleton set and the number of video frames. In another work, relative geometric features [21] were taken into account for action image generation to overcome the viewpoint variation problem. The joint-distance-based static pose and dynamic motion features [22] were individually performed by two processing streams in CNN for feature learning, wherein each stream is inspired by Inception-v3 architecture. Some approaches combine CNNs with late fusion (or decision-level) scheme to boost the recognition accuracy [23]. For effectively dealing with the overfitting issue when training CNN on small-sized datasets, some novel data augmentation techniques [24], [25] have been proposed for enrich the representational information in encoded action images. Some CNN backbones initially introduced for image classification, such as VGG-19, GoogleNet, ResNet-101, and Inception-v3, are recommended for learning skeleton-based action recognition model with a transfer learning technique [26], [27]. As a primary limitation of many existing 3D skeleton-based action recognition approaches, static pose and dynamic motion are usually analyzed independently without jointly informative association [28], which subsequently alleviate the capability of discriminating either pose-shared or motion-shared actions.

In this paper, a novel deep convolutional network is introduced for 3D human action recognition via analyzing skeletal information acquired by depth camera. The network is structured by two processing streams of multiple stacks of convolutional layers, which allows learning the relevant information of action representation from geometric feature sets of static pose and dynamic motion separately. Each stream is organized by several modules mainly consisting of convolutional blocks, where each block is specified by not only regular convolutional layers but also grouped convolutional layers with various kernel size for feature enrichment. Significantly, two streams are jointly associated via a customized layer for the objective of cross-knowledge sharing between two streams. In the experiments, the performance of network in terms of recognition accuracy is evaluated on NTU RGB+D, a large-scale dataset of action recognition, and its extension with 120 classes. The proposed dual-stream architecture is superior to the single-stream scenarios and further outperforms other recent deep models for skeleton-based action recognition.

II. Proposed Method

Regarding the overall framework shown in Fig. 1, the relatively geometric features representing the static pose and dynamic motion of an action are calculated from an skeleton sequence having arbitrary length. Afterwards, they are learned by two convolutional streams configured for being able to jointly share the highly relevant information from multi-scale feature maps.

A. Relatively Geometric Feature Calculation

Instead of directly formulating the skeletal data (e.g., joint coordinates) of an action into a high-dimensional array [19], [20] that is critically sensitive to viewpoint variation, we utilize the joint-to-joint distance, one of the most appropriate geometric attributes for subject representation in the 3D space. In particular, we calculate the intra-frame joint distance for static pose description and the inter-frame joint distance for dynamic motion explanation [22]. Given a skeletal sequence $\mathcal{A}$ having T frames, each frame may contain one or multiple human skeletons. Note that we assume each skeleton consists of m 3D body joints of $p = (x, y, z)$ in space $\mathbb{R}^3$. Each distance feature $\boldsymbol{\vartheta}^{ij}$ is formed by triple values $\left(\vartheta^{ij}_{oxy}, \vartheta^{ij}_{oyz}, \vartheta^{ij}_{ozx}\right)$ of Euclidean distances between two arbitrary joints i and j when projecting onto different original planes

$$\begin{aligned} \vartheta^{ij}_{oxy} &= \left\| p^i_{z=0} - p^j_{z=0} \right\|, \\ \vartheta^{ij}_{oyz} &= \left\| p^i_{x=0} - p^j_{x=0} \right\|, \\ \vartheta^{ij}_{ozx} &= \left\| p^i_{y=0} - p^j_{y=0} \right\|. \end{aligned} \tag{1}$$

For the static pose description, we measure the distance features regarding the subjects within a frame, denoted $\boldsymbol{\vartheta}^{ij}_t$. Meanwhile, the dynamic motion is represented by the features $\boldsymbol{\vartheta}^{ij}_{\Delta t}$ of the subjects observed in two consecutive frames to explicitly capture the movement of body joints in time. It is worth noting that the joints i and j belong either a same skeleton of a single action such as *clapping* or two different skeletons of an interaction like *pushing*. The distance features are accumulated for static pose $\mathbb{F}^{\mathrm{S}}$ and dynamic motion $\mathbb{F}^{\mathrm{D}}$ groups by monitoring T frames in the video $\mathcal{A}$, which are expressed as follows:

$$\begin{aligned} \mathbb{F}^{\mathrm{S}} &= \left[\boldsymbol{\vartheta}^{ij}_t \big|^{i=1,\ldots,m;j=1,\ldots,m}_{t=1,\ldots,T} \right], \\ \mathbb{F}^{\mathrm{D}} &= \left[\boldsymbol{\vartheta}^{ij}_t \big|^{i=1,\ldots,m;j=1,\ldots,m}_{\Delta t=2,\ldots,T} \right]. \end{aligned} \tag{2}$$

With respect to an input skeletal sequence, a handcrafted geometric feature set $\mathbb{F}^{\mathbb{G}} = \left\{ \mathbb{F}^{\mathrm{S}}, \mathbb{F}^{\mathrm{D}} \right\}$ (assembled in a 3D data array) is generated for deep learning. At this point, a regular min-max value normalization is carried out to re-scale the distance features in the range of $[0, 1]$ for the prevention of dataset bias and erratically updated weight.

B. J2$^{\mathrm{S}}$Net: Jointly Dual-Stream ConvNet

Although some previous methods have recommended a multi-stream CNN for learning an action recognition model [18], [22], [28], the multiple convolutional streams in networks are processed independently for different kinds of input, where the outputs of those streams are usually fused at the global average pooling layer [22] and the softmax layer [18], [28]. Without jointly sharing the relevant information between streams during the convolutional-based feature extraction process, the deep CNNs cannot learn the inter-stream correlations to maximize action discrimination. In this work, we introduce an efficient deep CNN, namely Jointly Dual-Stream ConvNet (J2$^{\mathrm{S}}$Net), for skeleton-based action

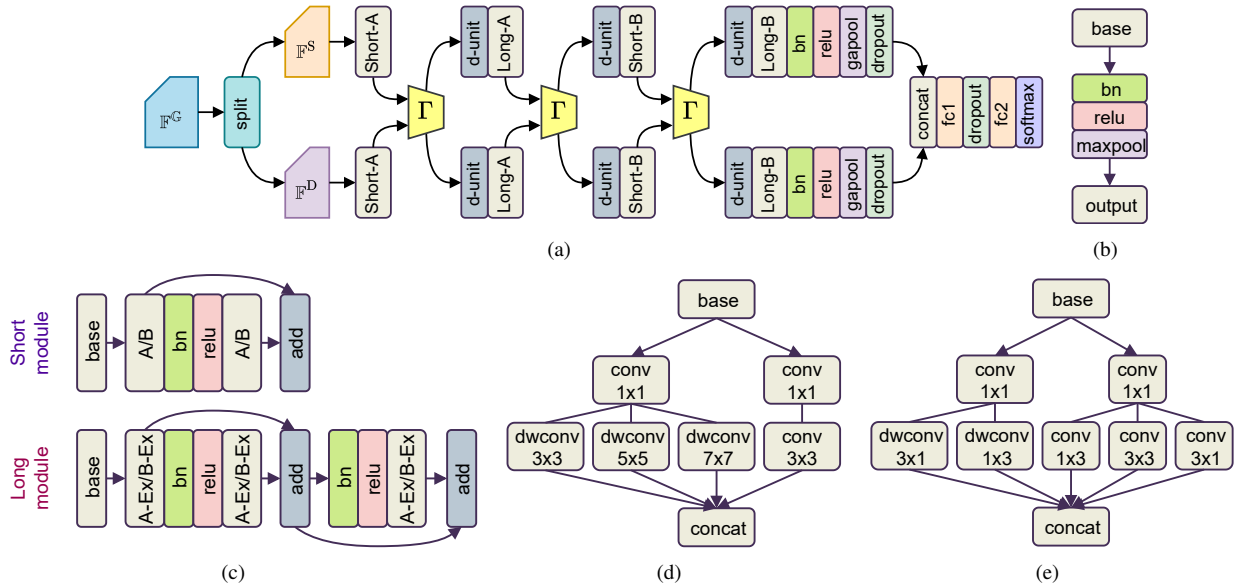

Fig. 2. Description of J2^{S}Net developed for action recognition: (a) the overall CNN architecture with two jointly connected streams processing static pose and dynamic motion features, where Γ refers to as a novel sharing layer, (b) the down-sampling unit (denoted d-unit), (c) the structure of short and long modules: the short module consists of two standard convolutional blocks (A or B) associated via skip-connection, meanwhile, the long module is an expansion form with three blocks (denoted A-Ex/B-Ex), where the block-A-Ex/block-B-Ex is the advancement of block-A/block-B with more convolution kernels configured in layers, (d) and (e) the structures of convolutional block-A and block-B, respectively.

recognition, in which the network architecture is specified by two jointly connected streams of convolutional layers. As the overall network architecture shown in Fig. 2(a), J2^{S}Net is able to analyze the spatially static human pose and the temporally dynamic body motion at multiple scales comprehensively thanks to the two processing streams, where their relevant features are tightly incorporated to promisingly enhance the recognition accuracy.

For the details of network architecture, the input layer is facilitated with the handcrafted geometric feature set $\mathbb{F}^{\mathbb{G}}$ which contains the skeletal information corresponding to a predefined action category. A customized layer, denoted split in Fig. 2(a), is defined to partition the feature set $\mathbb{F}^{\mathbb{G}}$ into the input maps of $\mathbb{F}^{\mathrm{S}}$ and $\mathbb{F}^{\mathrm{D}}$. These maps are separately processed by two learnable streams sharing same configurations, for instance, the kernel size and the number of kernels in convolutional layers. Each stream is primarily specified by several short and long modules alternately connected for learning geometric features at multi-scale representations. With regard to the detailed structure shown in Fig. 2(c), the short module consists of two standard convolutional block (either block-A or block-B), batch normalization (bn) layer, rectified linear unit (relu) layer, and addition layer. The long module, meanwhile, the expansion of short form, has three advanced blocks (either block-A-Ex or block-B-Ex). Notably, in these modules, the blocks are associated via skip-connection scheme which was introduced for assembling residual unit in ResNet [29]. In details, the block-A (see the arrangement given Fig. 2(d)) consists of two unit layers with the kernel (a.k.a. filter) of size 1×1, one regular convolutional layer with the kernel of size 3×3, and three grouped convolutional layers (wherein the channel-wise/depth-wise separable convolution operation are performed) with the kernels of size 3×3, 5×5, and 7×7. The block-B (see the pattern illustrated in Fig. 2(e)) has two unit layers, three regular convolutional layers with the kernels of size 1×3, 3×1, and 3×3, and two grouped convolutional layers with the kernels of size 1×3 and 3×1. The output of these convolutional blocks accumulates feature maps of same spatial size via a depth-wise concatenation layer (concat). While the block-A aims to capture the global features with symmetric larger-sized kernels, the block-B learns the local features with asymmetric smaller-sized kernels. Compared with the standard block-A/block-B, the advanced block-A-Ex/block-B-Ex is configured with more kernels in convolutional layers. The detailed description of block configuration is summarized in Table I. The modules short-A/B and long-A/B, denoted in Fig. 2(a), are configured by the block-A/B and the block-A-Ex/B-Ex, respectively.

TABLE I
CONFIGURATION OF THE PROPOSED NETWORK ARCHITECTURE.

Item	Description
Block-A/**A-Ex**	$2 \times [32/\mathbf{32}$ conv $(1 \times 1)]$ $1 \times [32/\mathbf{64}$ conv $(3 \times 3)]$ $1 \times [32$ groups $\times\ 1/\mathbf{2}$ gconv $(3 \times 3)]$ $1 \times [32$ groups $\times\ 1/\mathbf{2}$ gconv $(5 \times 5)]$ $1 \times [32$ groups $\times\ 1/\mathbf{2}$ gconv $(7 \times 7)]$
Block-B/**B-Ex**	$2 \times [80/\mathbf{80}$ conv $(1 \times 1)]$ $1 \times [32/\mathbf{64}$ conv $(3 \times 3)]$ $1 \times [64/\mathbf{128}$ conv $(1 \times 3)]$ $1 \times [64/\mathbf{128}$ conv $(3 \times 1)]$ $1 \times [80$ groups $\times\ 1/\mathbf{2}$ gconv $(1 \times 3)]$ $1 \times [80$ groups $\times\ 1/\mathbf{2}$ gconv $(3 \times 1)]$

As mentioned before, in each module, skip-connection

mechanism is performed via an addition layer to prevent the vanishing gradient problem and maintain the information identity which can be attenuated over multiple convolution and nonlinear operations. In Fig. 2(c), the modules are finalized by skip-connection strategy with the output as follows

- Short-module:

$$out_{\mathtt{short-A/B}} = out^{\mathtt{1st}}_{\mathtt{A/B}} + out^{\mathtt{2nd}}_{\mathtt{A/B}}, \tag{3}$$

where $out^{\mathtt{1st}}$ and $out^{\mathtt{2nd}}$ stand for the outputs of the first and the second block-A/B in a short module, respectively, in which $out^{\mathtt{2nd}}_{\mathtt{A/B}} = \mathtt{conv}_{\mathtt{A/B}}\left(out^{\mathtt{1st}}_{\mathtt{A/B}}\right)$ with $\mathtt{conv}_{\mathtt{A/B}}$ referring to as the overall convolution operation of the block-A/B.

- Long-module:

$$out_{\mathtt{long-A/B}} = out^{\mathtt{1nd}}_{\mathtt{A-Ex/B-Ex}} + out^{\mathtt{2nd}}_{\mathtt{A-Ex/B-Ex}} + out^{\mathtt{3rd}}_{\mathtt{A-Ex/B-Ex}}, \tag{4}$$

where

$$\begin{aligned} out^{\mathtt{2nd}}_{\mathtt{A-Ex/B-Ex}} &= \mathtt{conv}_{\mathtt{A-Ex/B-Ex}}\left(out^{\mathtt{1st}}_{\mathtt{A-Ex/B-Ex}}\right), \\ out^{\mathtt{3nd}}_{\mathtt{A-Ex/B-Ex}} &= \mathtt{conv}_{\mathtt{A-Ex/B-Ex}}\left(out^{\mathtt{2st}}_{\mathtt{A-Ex/B-Ex}}\right), \end{aligned} \tag{5}$$

with $\mathtt{conv}_{\mathtt{A-Ex/B-Ex}}$ referring to as the overall convolution operation of the block-A-Ex/B-Ex.

In the training process, the knowledge gained in two processing streams is partly shared together via an intermediately connected layer Γ in Fig. 2(a). The layer is designed with two inputs, denoted in_1 and in_2, for reading the feature maps individually acquired from two streams. As a result, the layer returns two outputs correspondingly, at which sharing knowledge is achieved via some arithmetic operations performed inside as follows

$$\begin{aligned} out_1 &= \Gamma(in_1) = in_1 + \lambda \times \mathtt{maxpool}(in_2), \\ out_2 &= \Gamma(in_2) = in_2 + \lambda \times \mathtt{maxpool}(in_1), \end{aligned} \tag{6}$$

where $0 \leqslant \lambda \leqslant 1$ refers to as an association impact and `maxpool` stands for a max pooling layer which returns the maximum feature value of each 3×3 region with stride 1 (that means, without down-sampling). For this research, λ is set to 0.5 for the simultaneous deployment of cross-knowledge sharing between two streams and information maintaining of each stream. The output of Γ layer is forwarded directly to the down-sample unit, where the spatial size of feature maps is down-scaled by two times with a max pooling layer specified by pool of size 3×3 and stride of 2.

Each stream is finalized with a global average pooling layer, at which the size of output array is identical to the channel dimension of the feature maps. For example, given the long module with block-B-Ex outputs a feature map having 640 channels, the expected output vector shall have size of 1×640. Afterwards, the global average features of two streams are merged via a concatenation layer before feeding to fully connected (or dense) layers and softmax layer for classification. Furthermore, some dropout layers with the probability of 0.5 are arranged to effectively handle the overfitting issue in model training. J2$^{\mathrm{S}}$Net is trained with randomly initialized weights in 100 epochs using stochastic gradient descent optimizer, initial learning rate of 0.01 (which is reduced to 0.001 after 50 epochs), and mini-batch size of 32.

TABLE II
COMPARISON OF RECOGNITION ACCURACY (%)

NTU RGB+D		
Methods	**C-Subject**	**C-View**
Deep RNN [7]	56.3	64.1
Part-Aware LSTM [7]	62.9	70.3
ST-LSTM [16]	65.2	76.1
ST-LSTM + Trust Gate [16]	69.2	77.7
SkeleMotion [20]	69.6	80.1
Two-Stream Attention LSTM [15]	76.1	84.0
JTM + ConvNet [18]	76.3	81.1
PoF2I + Inception-v3 [25]	82.5	89.5
J2$^{\mathrm{S}}$Net (Single-stream $\mathbb{F}^{\mathrm{S}}$)	77.7	85.5
J2$^{\mathrm{S}}$Net (Single-stream $\mathbb{F}^{\mathrm{D}}$)	77.9	86.1
J2$^{\mathrm{S}}$Net (Dual-stream $\mathbb{F}^{\mathbb{G}}$)	79.8	87.1
NTU RGB+D 120		
Methods	**C-Subject**	**C-Setup**
Part-Aware LSTM [7]	25.5	26.3
Soft RNN [13]	36.3	44.9
Dynamic Skeleton [5]	50.8	54.7
ST-LSTM + Trust Gate [16]	58.2	60.9
Two-Stream Attention LSTM [15]	61.2	63.3
SkeleMotion [20]	67.7	66.9
DGPoT-2$^{\mathrm{S}}$CNN (GoogleNet) [22]	71.8	73.9
J2$^{\mathrm{S}}$Net (Single-stream $\mathbb{F}^{\mathrm{S}}$)	67.2	71.2
J2$^{\mathrm{S}}$Net (Single-stream $\mathbb{F}^{\mathrm{D}}$)	68.1	71.8
J2$^{\mathrm{S}}$Net (Dual-stream $\mathbb{F}^{\mathbb{G}}$)	70.5	73.8

III. PERFORMANCE EVALUATION

A. Datasets

NTU RGB+D: This dataset [7] has 56,880 skeleton sequences (using Kinectv2 for data collection) of 60 action categories, including single-person actions, human-object interactions, and human-human interactions. Because of intra-class and view-point variations, NTU RGB+D is widely used for evaluating the performance of 3D skeleton-based action recognition methods. Accordingly, two evaluation protocols (cross-subject and cross-view) are given for experimental configuration.

NTU RGB+D 120 [30]: The database is extended from NTU RGB+D [7] with 57,600 videos of 60 new actions. In particular, NTU RGB+D 120 contains 114,480 skeletal sequences which present up to 120 daily actions, mutual action, and medical conditions. The cross-subject and cross-setup evaluation protocols are taken in the experiments. The details of data splitting for performance benchmark are mentioned in the dataset papers [7], [30].

B. Experimental Results

The accuracy results of the proposed 3D action recognition method evaluated on NTU RGB+D and NTU RGB+D 120 are reported in Table II. At first, we perform an ablation study, where the performance of J2$^{\mathrm{S}}$Net is investigated with different geometric feature sets, including the static pose $\mathbb{F}^{\mathrm{S}}$, the dynamic motion $\mathbb{F}^{\mathrm{D}}$, and the merged set $\mathbb{F}^{\mathbb{G}}$ (noted that only

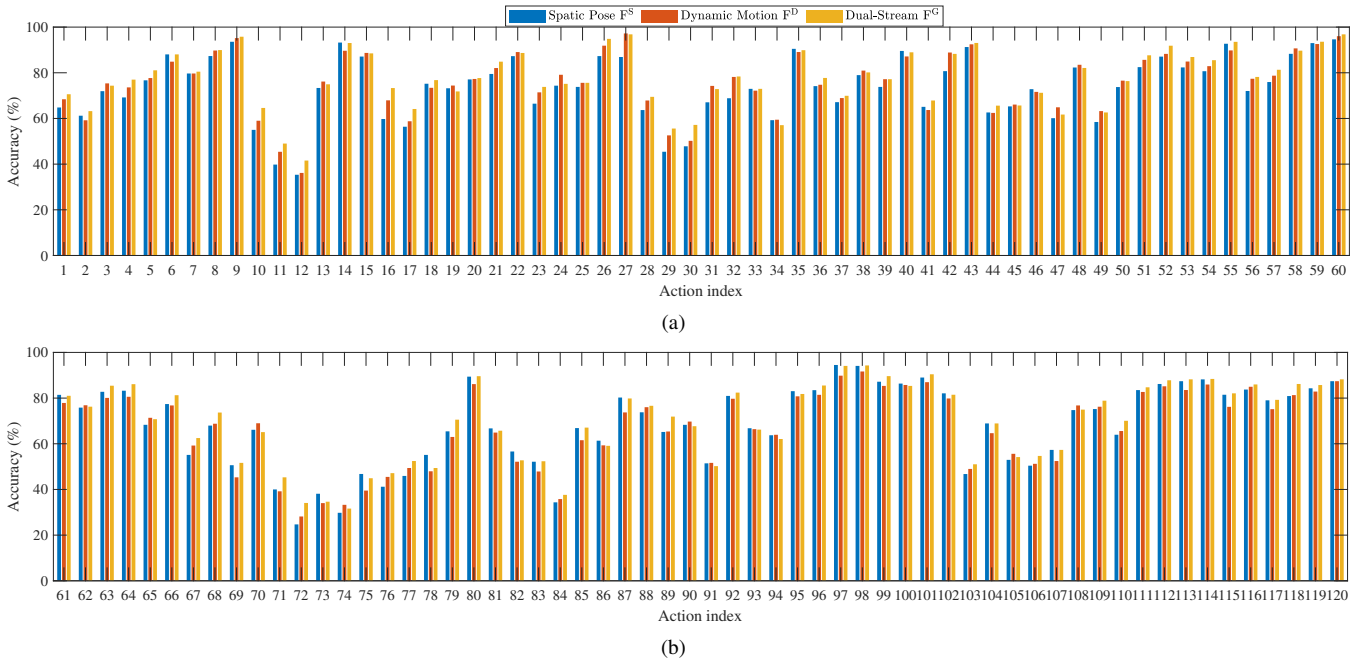

Fig. 3. Recognition accuracy of 120 actions in the NTU RGB+D 120 dataset with respect to the cross-setup evaluation protocol.

single stream of the network is carried out for model learning). It is observed that J2$^{\mathrm{S}}$Net with dual-stream processing achieves some improvements of recognition rate if compared with single-stream processing. Concretely, J2$^{\mathrm{S}}$Net with $\mathbb{F}^{\mathbb{G}}$ recognizes more accurately than that with $\mathbb{F}^{\mathrm{S}}$ by 2.1% and 1.6% with respect to the cross-subject and cross-view of NTU RGB+D, respectively. Meanwhile, the network obtains the improvement of 3.3% for the cross-subject and 2.6% for the cross-setup of NTU RGB+D 120. Besides, the dynamic motion for capturing the body transition in time is more beneficial than the static pose to impressively pattern an action. Statistically, the single-stream network with $\mathbb{F}^{\mathrm{D}}$ performs recognition more precisely than that with $\mathbb{F}^{\mathrm{S}}$ by the overall higher accuracy of $0.2 - 0.9\%$. Compared with NTU RGB+D, the updated one with 60 new actions is more challenging with large intra-class variation, high variety of subject characteristics, and more environment configurations for data collection. Consequently, the recognition accuracy on NTU RGB+D 120 is worse than that on NTU RGB+D as a smaller and less challenging version (for example, approximately 9.3% regarding the cross-subject protocol). For more details of performance analysis, we additionally provide the accuracy results of 120 actions with respect to the cross-setup configuration in Fig. 3, where the efficiency of dual-stream $\mathbb{F}^{\mathbb{G}}$, compared with that of the single-stream configuration, can be perceived explicitly. Some actions which are mostly represented by some very similar sets of skeletal pipeline (for instance, *reading* vs *writing* and *make victory sign* vs *make OK sign*) suffer much confusion besides other uncommon daily actions (e.g., *counting money* and *staple book*).

Secondly, we provide a method comparison in terms of recognition rate, where the proposed method with J2$^{\mathrm{S}}$Net combats against other state-of-the-art deep models, including LSTM and CNNs for learning skeleton information of action. Based the comparison results summarized in Table II, the proposed CNN significantly outperforms several RNN and LSTM models with respect to both datasets. For example, J2$^{\mathrm{S}}$Net is better than Spatio-Temporal LSTM with Trust Gate (ST-LSMT + Trust Gate) [16] by approximately $9.4 - 12.9\%$. In [15], an advanced LSTM with two-stream architecture and attention mechanism is recommended to increase the accuracy of action recognition system. Some approaches, which exploits CNNs for modeling 3D skeleton information, performs action recognition more advantageously than RNNs and LSTM. SkeleMotion [20] is better than ST-LSTM [16] by around 4.0% for cross-subject and 4.4% for cross-view of NTU RGB+D evaluation. Meanwhile, J2$^{\mathrm{S}}$Net with dual-stream processing achieves the second best results (less than of PoF2I + Inception-v3 [25] by $2.4 - 2.7\%$ regarding NTU RGB+D and DGPoT-2$^{\mathrm{S}}$CNN (GoogleNet) [22] by $0.1 - 1.3\%$ regarding NTU RGB+D 120). However, it is noted that compared with J2$^{\mathrm{S}}$Net trained from scratch with randomly initialized weights, two approaches PoF2I and DGPoT-2$^{\mathrm{S}}$CNN fine-tune the pre-trained networks (e.g., Inception-v3 and GoogleNet), where the learned rich features are manipulated for transfer learning of encoded action images.

C. Complexity Analysis

In this experiment, we report a complexity analysis via the measurement of network size and processing speed between J2$^{\mathrm{S}}$Net and two other CNN-based models, particularly, PoF2I + Inception-v3 [25] and DGPoT-2$^{\mathrm{S}}$CNN (GoogleNet) [22], where the comparison results of the inference (or prediction) time and the number of parameters are given in Table III.

TABLE III
COMPARISON OF COMPUTATIONAL COMPLEXITY

Methods	Parameters (M)	Time (ms)
PoF2I + Inception-v3 [25]	23.9	6.0
DGPoT-2^{S}CNN (GoogleNet) [22]	7.0	4.5
J2^{S}Net (Dual-stream $\mathbb{F}^{\mathbb{G}}$)	2.2	3.6

With the configuration described in Table I, J2^{S}Net has approximately 2.2M trainable parameters (including weights and bias). Based on the system equipped by a single NVIDIA 1080Ti GPU, J2^{S}Net takes around 3.6 ms as an average inference time for processing a segmented action video. Despite being worse than PoF2I and DGPoT-2^{S}CNN (where they take advantages of learned rich feature representations by fine-tuning Inception-v3 and GoogleNet) in terms of accuracy, J2^{S}Net is more lightweight to execute faster.

IV. CONCLUSION

In this paper, we have proposed J2^{S}Net, a novel CNN with dual-stream architecture for concurrently learning the geometric static pose and dynamic motion from 3D skeleton data, for human action recognition. Each processing stream in network is structured by several cascaded blocks of regular and grouped convolutional layers which are specified by various kernel sizes to enrich the representational features. A jointly associated connection of two streams is established via a customized layer for sharing high-relevant knowledge. Compared with several recent deep models, the proposed network has achieved a competitive performance in terms of recognition rate and inference time on two large and challenging datasets.

REFERENCES

[1] O. Banos, C. Villalonga, J. Bang, T. Hur, D. Kang, S. Park, T. Huynh-The, V. Le-Ba, M. Amin, M. Razzaq, W. K. C. Hong, and S. Lee, "Human behavior analysis by means of multimodal context mining," *Sensors*, vol. 16, p. 1264, 2016.

[2] T. Huynh-The, C.-H. Hua, N. A. Tu, and D.-S. Kim, "Learning 3D spatiotemporal gait feature by convolutional network for person identification," *Neurocomputing*, vol. 397, pp. 192 – 202, 2020.

[3] O. Banos, J. Bang, T. Hur, M. H. Siddiqi, H. Thien, L. Vui, W. Ali Khan, T. Ali, C. Villalonga, and S. Lee, "Mining human behavior for health promotion," in *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2015, pp. 5062–5065.

[4] N. A. Tu, T. Huynh-The, K. U. Khan, and Y. Lee, "ML-HDP: A hierarchical bayesian nonparametric model for recognizing human actions in video," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 3, pp. 800–814, 2019.

[5] J. Hu, W. Zheng, J. Lai, and J. Zhang, "Jointly learning heterogeneous features for RGB-D activity recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 11, pp. 2186–2200, Nov 2017.

[6] T. Huynh-The, C.-H. Hua, N. A. Tu], T. Hur, J. Bang, D. Kim, M. B. Amin, B. H. Kang, H. Seung, S.-Y. Shin, E.-S. Kim, and S. Lee, "Hierarchical topic modeling with pose-transition feature for action recognition using 3D skeleton data," *Information Sciences*, vol. 444, pp. 20–35, 2018.

[7] A. Shahroudy, J. Liu, T. Ng, and G. Wang, "NTU RGB+D: A large scale dataset for 3D human activity analysis," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 1010–1019.

[8] T. Huynh-The, B. Le, S. Lee, and Y. Yoon, "Interactive activity recognition using pose-based spatio–temporal relation features and four-level Pachinko allocation model," *Information Sciences*, vol. 369, pp. 317 – 333, 2016.

[9] T. Huynh-The, B. Le, and S. Lee, "Describing body-pose feature - poselet - activity relationship using Pachinko allocation model," in *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2016, pp. 40–45.

[10] C. Hua, T. Huynh-The, and S. Lee, "Convolutional networks with bracket-style decoder for semantic scene segmentation," in *Proc. 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Oct. 2018, pp. 2980–2985.

[11] C.-H. Hua, T. Huynh-The, S.-H. Bae, and S. Lee, "Cross-attentional bracket-shaped convolutional network for semantic image segmentation," *Information Sciences*, vol. 539, pp. 277 – 294, 2020.

[12] C. Hua, T. Huynh-The, and S. Lee, "Retinal vessel segmentation using round-wise features aggregation on bracket-shaped convolutional neural networks," in *Proc. 2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Jul. 2019, pp. 36–39.

[13] J. Hu, W. Zheng, L. Ma, G. Wang, J. Lai, and J. Zhang, "Early action prediction by soft regression," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 11, pp. 2568–2583, 2019.

[14] W. Zheng, L. Li, Z. Zhang, Y. Huang, and L. Wang, "Relational network for skeleton-based action recognition," in *2019 IEEE International Conference on Multimedia and Expo (ICME)*, 2019, pp. 826–831.

[15] J. Liu, G. Wang, L. Duan, K. Abdiyeva, and A. C. Kot, "Skeleton-based human action recognition with global context-aware attention LSTM networks," *IEEE Transactions on Image Processing*, vol. 27, no. 4, pp. 1586–1599, 2018.

[16] J. Liu, A. Shahroudy, D. Xu, A. C. Kot, and G. Wang, "Skeleton-based action recognition using spatio-temporal LSTM network with trust gates," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 12, pp. 3007–3021, 2018.

[17] L. Wang, D. Q. Huynh, and P. Koniusz, "A comparative review of recent kinect-based action recognition algorithms," *IEEE Transactions on Image Processing*, vol. 29, pp. 15–28, 2020.

[18] P. Wang, W. Li, C. Li, and Y. Hou, "Action recognition based on joint trajectory maps with convolutional neural networks," *Knowledge-Based Systems*, vol. 158, pp. 43 – 53, 2018.

[19] Q. Ke, M. Bennamoun, H. Rahmani, S. An, F. Sohel, and F. Boussaid, "Learning latent global network for skeleton-based action prediction," *IEEE Transactions on Image Processing*, vol. 29, pp. 959–970, 2020.

[20] C. Caetano, J. Sena, F. Brémond, J. A. Dos Santos, and W. R. Schwartz, "SkeleMotion: A new representation of skeleton joint sequences based on motion information for 3D action recognition," in *2019 16th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, 2019, pp. 1–8.

[21] T. Huynh-The, C.-H. Hua, T.-T. Ngo, and D.-S. Kim, "Image representation of pose-transition feature for 3D skeleton-based action recognition," *Information Sciences*, vol. 513, pp. 112 – 126, 2020.

[22] T. Huynh-The, C. Hua, N. A. Tu, and D. Kim, "Learning geometric features with dual–stream cnn for 3D action recognition," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 2353–2357.

[23] M. Liu and J. Yuan, "Recognizing human actions as the evolution of pose estimation maps," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1159–1168.

[24] T. Huynh-The and D. Kim, "Data augmentation for CNN-based 3D action recognition on small-scale datasets," in *2019 IEEE 17th International Conference on Industrial Informatics (INDIN)*, vol. 1, 2019, pp. 239–244.

[25] T. Huynh-The, C. Hua, and D. Kim, "Encoding pose features to images with data augmentation for 3-D action recognition," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 5, pp. 3100–3111, 2020.

[26] T. Huynh-The, C. Hua, N. A. Tu, J. Kim, S. Kim, and D. Kim, "3D action recognition exploiting hierarchical deep feature fusion model," in *2020 14th International Conference on Ubiquitous Information Management and Communication (IMCOM)*, Jan. 2020, pp. 1–3.

[27] T. Huynh-The, C. Hua, and D. Kim, "Learning action images using deep convolutional neural networks for 3D action recognition," in *Proc. 2019 IEEE Sensors Applications Symposium (SAS)*, Mar. 2019, pp. 1–6.

[28] K. Liu, L. Gao, N. M. Khan, L. Qi, and L. Guan, "A multi-stream graph convolutional networks-hidden conditional random field model for

skeleton-based action recognition," *IEEE Transactions on Multimedia*, pp. 1–1, 2020.

[29] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[30] J. Liu, A. Shahroudy, M. L. Perez, G. Wang, L. Duan, and A. Kot Chichung, "NTU RGB+D 120: A large-scale benchmark for 3D human activity understanding," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2019.