Skeleton-Based Human Action Recognition With Global Context-Aware Attention LSTM Networks

Jun Liu, *Student Member, IEEE*, Gang Wang, *Senior Member, IEEE*, Ling-Yu Duan, *Member, IEEE*, Kamila Abdiyeva, *Student Member, IEEE*, and Alex C. Kot, *Fellow, IEEE*

***Abstract*— Human action recognition in 3D skeleton sequences has attracted a lot of research attention. Recently, long short-term memory (LSTM) networks have shown promising performance in this task due to their strengths in modeling the dependencies and dynamics in sequential data. As not all skeletal joints are informative for action recognition, and the irrelevant joints often bring noise which can degrade the performance, we need to pay more attention to the informative ones. However, the original LSTM network does not have explicit attention ability. In this paper, we propose a new class of LSTM network, global context-aware attention LSTM, for skeleton-based action recognition, which is capable of selectively focusing on the informative joints in each frame by using a global context memory cell. To further improve the attention capability, we also introduce a recurrent attention mechanism, with which the attention performance of our network can be enhanced progressively. Besides, a two-stream framework, which leverages coarse-grained attention and fine-grained attention, is also introduced. The proposed method achieves state-of-the-art performance on five challenging datasets for skeleton-based action recognition.**

***Index Terms*— Action recognition, long short-term memory, global context memory, attention, skeleton sequence.**

I. Introduction

ACTION recognition is a very important research problem owing to its relevance to a wide range of applications, such as video surveillance, patient monitoring, robotics, human-machine interaction, etc [1]–[3]. With the development of depth sensors, such as RealSense and Kinect [4]–[6], 3D skeleton based human action recognition has received much attention, and a lot of advanced methods have been proposed during the past few years [7]–[10].

Manuscript received July 19, 2017; revised November 24, 2017; accepted December 11, 2017. Date of publication December 19, 2017; date of current version January 5, 2018. This work was supported in part by the National Research Foundation, Singapore, through the Interactive Digital Media Strategic Research Programme, in part by the National Natural Science Foundation of China under Grant U1611461 and Grant 61661146005, and in part by the National Key Research and Development Program of China under Grant 2016YFB1001501. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Jie Liang. *(Corresponding authors: Jun Liu; Ling-Yu Duan.)*

J. Liu, K. Abdiyeva, and A. C. Kot are with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798 (e-mail: jliu029@ntu.edu.sg; abdi0001@ntu.edu.sg; eackot@ntu.edu.sg).

G. Wang is with the Alibaba Group, Hangzhou 310052, China (e-mail: wanggang@ntu.edu.sg).

L.-Y. Duan is with the National Engineering Laboratory for Video Technology, Peking University, Beijing 100871, China (e-mail: lingyu@pku.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2017.2785279

Human actions can be represented by a combination of the motions of skeletal joints in 3D space [11], [12]. However, this does not indicate all joints in the skeleton sequence are informative for action recognition. For instance, the hand joints' motions are quite informative for the action *clapping*, while the movements of the foot joints are not. Different action sequences often have different informative joints, and in the same sequence, the informativeness degree of a body joint may also change over the frames. Thus, it is beneficial to selectively focus on the informative joints in each frame of the sequence, and try to ignore the features of the irrelevant ones, as the latter contribute very little for action recognition, and even bring noise which corrupts the performance [13]. This selectively focusing scheme can also be called *attention*, which has been demonstrated to be quite useful for various tasks, such as speech recognition [14], image caption generation [15], machine translation [16], and so on.

Long Short-Term Memory (LSTM) networks have strong power in handling sequential data [17]. They have been successfully applied to language modeling [18], RGB based video analysis [19]–[27], and also skeleton based action recognition [12], [28], [29]. However, the original LSTM does not have strong attention capability for action recognition. This limitation is mainly owing to LSTM's restriction in perceiving the global context information of the video sequence, which is, however, often very important for the global classification problem — skeleton based action recognition.

In order to perform reliable attention over the skeletal joints, we need to assess the informativeness degree of each joint in each frame with regarding to the global action sequence. This indicates that we need to have global contextual knowledge first. However, the available context information at each evolution step of LSTM is relatively local. In LSTM, the sequential data is fed to the network as input step by step. Accordingly, the context information (hidden representation) of each step is fed to the next one. This implies the available context at each step is the hidden representation from the previous step, which is quite local when compared to the global information.[1]

In this paper, we extend the original LSTM model and propose a Global Context-Aware Attention LSTM (GCA-LSTM) network which has strong attention capability for skeleton based action recognition. In our

[1]Though in LSTM, the hidden representations of the latter steps contain wider range of context information than that of the initial steps, their context is still relatively local, as LSTM has trouble in remembering information too far in the past [30].

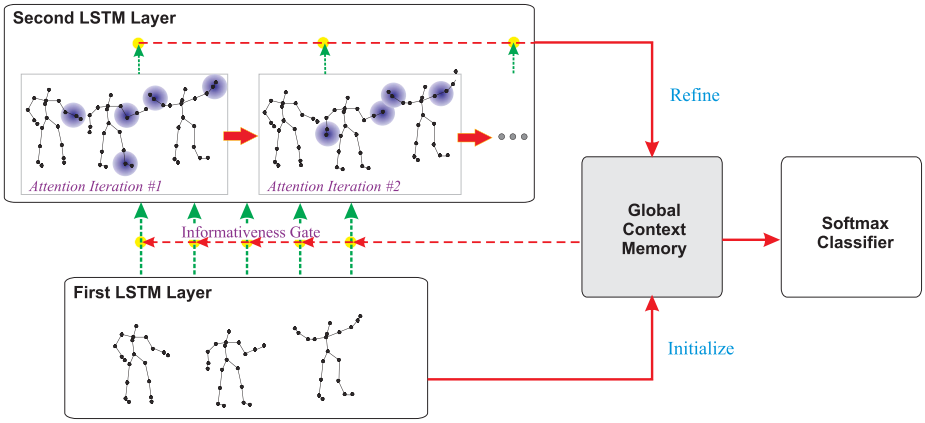


Fig. 1. Skeleton based human action recognition with the Global Context-Aware Attention LSTM network. The first LSTM layer encodes the skeleton sequence and generates an initial global context representation for the action sequence. The second layer performs attention over the inputs by using the global context memory cell to achieve an attention representation for the sequence. Then the attention representation is used back to refine the global context. Multiple attention iterations are performed to refine the global context memory progressively. Finally, the refined global context information is utilized for classification.

method, the global context information is fed to all evolution steps of the GCA-LSTM. Therefore, the network can use it to measure the informativeness scores of the new inputs at all steps, and adjust the attention weights for them accordingly, i.e., if a new input is informative regarding to the global action, then the network takes advantage of more information of it at this step, on the contrary, if it is irrelevant, then the network blocks the input at this step.

Our proposed GCA-LSTM network for skeleton based action recognition includes a global context memory cell and two LSTM layers, as illustrated in Fig. 1. The first LSTM layer is used to encode the skeleton sequence and initialize the global context memory cell. And the representation of the global context memory is then fed to the second LSTM layer to assist the network to selectively focus on the informative joints in each frame, and further generate an attention representation for the action sequence. Then the attention representation is fed back to the global context memory cell in order to refine it. Moreover, we propose a recurrent attention mechanism for our GCA-LSTM network. As a refined global context memory is produced after the attention procedure, the global context memory can be fed to the second LSTM layer again to perform attention more reliably. We carry out multiple attention iterations to optimize the global context memory progressively. Finally, the refined global context is fed to the softmax classifier to predict the action class.

In addition, we also extend the aforementioned design of our GCA-LSTM network in this paper, and further propose a two-stream GCA-LSTM, which incorporates fine-grained (joint-level) attention and coarse-grained (body part-level) attention, in order to achieve more accurate action recognition results.

The contributions of this paper are summarized as follows:

- A GCA-LSTM model is proposed, which retains the sequential modeling ability of the original LSTM, meanwhile promoting its selective attention capability by introducing a global context memory cell.
- A recurrent attention mechanism is proposed, with which the attention performance of our network can be improved progressively.
- A stepwise training scheme is proposed to more effectively train the network.
- We further extend the design of our GCA-LSTM model, and propose a more powerful two-stream GCA-LSTM network.
- The proposed end-to-end network yields state-of-the-art performance on the evaluated benchmark datasets.

This work is an extension of our preliminary conference paper [31]. Based on the previous version, we further propose a stepwise training scheme to train our network effectively and efficiently. Moreover, we extend our GCA-LSTM model and further propose a two-stream GCA-LSTM by leveraging fine-grained attention and coarse-grained attention. Besides, we extensively evaluate our method on more benchmark datasets. More empirical analysis of the proposed approach is also provided.

The rest of this paper is organized as follows. In Section II, we review the related works on skeleton based action recognition. In Section III, we introduce the proposed GCA-LSTM network. In Section IV, we introduce the two-stream attention framework. Finally, we conclude the paper in Section VI.

II. Related Work

In this section, we first briefly review the skeleton based action recognition methods which mainly focus on extracting hand-crafted features. We then introduce the RNN and LSTM based methods. Finally, we review the recent works on attention mechanism.

A. Skeleton Based Action Recognition With Hand-Crafted Features

In the past few years, different feature extractors and classifier learning methods for skeleton based action recognition have been proposed [32]–[44].

Chaudhry *et al.* [45] proposed to encode the skeleton sequences to spatial-temporal hierarchical models, and then use linear dynamical systems (LDSs) to learn the dynamic structures. Vemulapalli *et al.* [46] represented each action as a curve in a Lie group, and then utlized a

support vector machine (SVM) to classify the actions. Xia *et al.* [47] proposed to model the temporal dynamics in action sequences with the Hidden Markov models (HMMs). Wang *et al.* [48], [49] introduced an actionlet ensemble representation to model the actions meanwhile capturing the intra-class variances. Chen *et al.* [50] designed a part-based 5D feature vector to explore the relevant joints of body parts in skeleton sequences. Koniusz *et al.* [51] introduced tensor representations for capturing the high-order relationships among body joints. Wang *et al.* [52] proposed a graph-based motion representation in conjunction with a SPGK-kernel SVM for skeleton based activity recognition. Zanfir *et al.* [53] developed a moving pose framework together with a modified k-NN classifier for low-latency action recognition.

B. Skeleton Based Action Recognition With RNN and LSTM Models

Very recently, deep learning, especially recurrent neural network (RNN), based approaches have shown their strength in skeleton based action recognition. Our proposed GCA-LSTM network is based on the LSTM model which is an extension of RNN. In this part, we review the RNN and LSTM based methods as below, since they are relevant to our method.

Du *et al.* [12] introduced a hierarchical RNN model to represent the human body structure and temporal dynamics of the joints. Veeriah *et al.* [54] proposed a differential gating scheme to make the LSTM network emphasize on the change of information. Zhu *et al.* [28] proposed a mixed-norm regularization method for the LSTM network in order to drive the model towards learning co-occurrence features of the skeletal joints. They also designed an in-depth dropout mechanism to effectively train the network. Shahroudy *et al.* [55] introduced a part-aware LSTM model to push the network towards learning long-term context representations of different body parts separately. Liu *et al.* [29], [56] designed a 2D Spatio-Temporal LSTM framework to concurrently explore the hidden sources of action related context information in both temporal and spatial domains. They also introduced a trust gate mechanism [29] to deal with the inaccurate 3D coordinates of skeletal joints provided by the depth sensors.

Beside action recognition, RNN and LSTM models have also been applied to skeleton based action forecasting [57] and detection [57], [58].

Different from the aforementioned RNN/LSTM based approaches, which do not explicitly consider the informativeness of each skeletal joint with regarding to the global action sequence, our proposed GCA-LSTM network utilizes the global context information to perform attention over all the evolution steps of LSTM to selectively emphasize the informative joints in each frame, and thereby generates an attention representation for the sequence, which can be used to improve the classification performance. Furthermore, a recurrent attention mechanism is proposed to iteratively optimize the attention performance.

C. Attention Mechanism

Our approach is also related to the attention mechanism [14], [16], [59]–[63] which allows the networks to selectively focus on specific information. Luong *et al.* [62] proposed a network with attention mechanism for neural machine translation. Stollenga *et al.* [64] designed a deep attention selective network for image classification. Xu *et al.* [15] proposed to incorporate hard attention and soft attention for image caption generation. Yao *et al.* [65] introduced a temporal attention model for video caption generation.

Though a series of deep learning based models have been proposed for video analysis in [66], [67], most of them did not consider the attention mechanism. There are several works which explored attention, such as the methods in [60], [68], and [69]. However, our method is significantly different from them in the following aspects: These works use the hidden state of the previous time step of LSTM, whose context information is quite local, to measure the attention scores for the next time step. For the global classification problem - action recognition, the global information is crucial for reliably evaluating the importance (informativeness) of each input to achieve a reliable attention. Therefore, we propose a global context memory cell for LSTM, which is utilized to measure the informativeness score of the input at each step. Then the informativeness score is used as a gate (informativeness gate, similar to the input gate and forget gate) inside the LSTM unit to adjust the contribution of the input data at each step for updating the memory cell. To the best of our knowledge, we are the first to introduce a global memory cell for LSTM network to handle global classification problems. Moreover, a recurrent attention mechanism is proposed to iteratively promote the attention capability of our network, while the methods in [60], [68], and [69] performed attention only once. In addition, a two-stream attention framework incorporating fine-grained attention and coarse-grained attention is also introduced. Owing to the new contributions, our proposed network yields state-of-the-art performance on the evaluated benchmark datasets.

III. GCA-LSTM Network

In this section, we first briefly review the 2D Spatio-Temporal LSTM (ST-LSTM) as our base network. We then introduce our proposed Global Context-Aware Attention LSTM (GCA-LSTM) network in detail, which is able to selectively focus on the informative joints in each frame of the skeleton sequence by using global context information. Finally, we describe our approach to training our network effectively.

A. Spatio-Temporal LSTM

In a generic skeleton based human action recognition problem, the 3D coordinates of the major body joints in each frame are provided. The spatial dependence of different joints in the same frame and the temporal dependence of the same joint among different frames are both crucial cues for skeleton based action analysis. Very recently, Liu *et al.* [29] proposed a 2D ST-LSTM network for skeleton based action recognition, which is capable of modeling the dependency structure and context information in both spatial and temporal domains simultaneously.

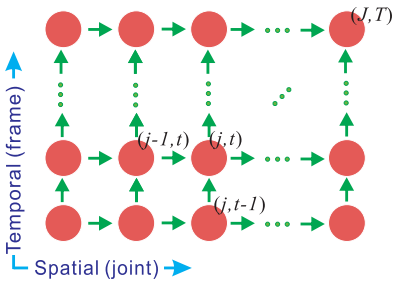


Fig. 2. Illustration of the ST-LSTM network [29]. In the spatial direction, the body joints in each frame are arranged as a chain and fed to the network as a sequence. In the temporal dimension, the body joints are fed over the frames.

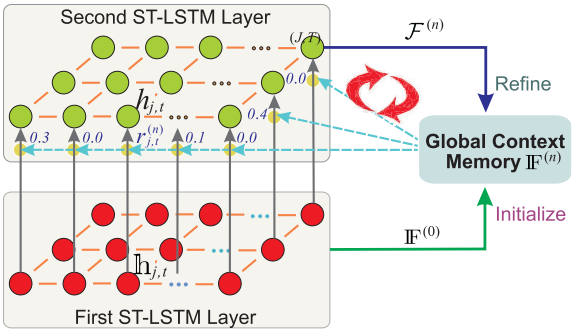


Fig. 3. Illustration of our GCA-LSTM network. Some arrows are omitted for clarity.

As depicted in Fig. 2, in ST-LSTM model, the skeletal joints in a frame are arranged and fed as a chain (the spatial direction), and the corresponding joints over different frames are also fed in a sequence (the temporal direction).

Specifically, each ST-LSTM unit is fed with a new input ($x_{j,t}$, the 3D location of joint j in frame t), the hidden representation of the same joint at the previous time step ($h_{j,t-1}$), and also the hidden representation of the previous joint in the same frame ($h_{j-1,t}$), where $j \in \{1, \ldots, J\}$ and $t \in \{1, \ldots, T\}$ denote the indices of joints and frames, respectively. The ST-LSTM unit has an input gate ($i_{j,t}$), two forget gates corresponding to the two sources of context information ($f_{j,t}^{(T)}$ for the temporal dimension, and $f_{j,t}^{(S)}$ for the spatial domain), together with an output gate ($o_{j,t}$).

The transition equations of ST-LSTM are formulated as presented in [29]:

$$\begin{pmatrix} i_{j,t} \\ f_{j,t}^{(S)} \\ f_{j,t}^{(T)} \\ o_{j,t} \\ u_{j,t} \end{pmatrix} = \begin{pmatrix} \sigma \\ \sigma \\ \sigma \\ \sigma \\ \tanh \end{pmatrix} \left(W \begin{pmatrix} x_{j,t} \\ h_{j-1,t} \\ h_{j,t-1} \end{pmatrix} \right) \tag{1}$$

$$\begin{aligned} c_{j,t} = {} & i_{j,t} \odot u_{j,t} \\ & + f_{j,t}^{(S)} \odot c_{j-1,t} \\ & + f_{j,t}^{(T)} \odot c_{j,t-1} \end{aligned} \tag{2}$$

$$h_{j,t} = o_{j,t} \odot \tanh(c_{j,t}) \tag{3}$$

where $c_{j,t}$ and $h_{j,t}$ denote the cell state and hidden representation of the unit at the spatio-temporal step (j, t), respectively, $u_{j,t}$ is the modulated input, $\odot$ denotes the element-wise product, and W is an affine transformation consisting of model parameters. Readers are referred to [29] for more details about the mechanism of ST-LSTM.

B. Global Context-Aware Attention LSTM

Several previous works [13], [50] have shown that in each action sequence, there is often a subset of informative joints which are important as they contribute much more to action analysis, while the remaining ones may be irrelevant (or even noisy) for this action. As a result, to obtain a high accuracy of action recognition, we need to identify the informative skeletal joints and concentrate more on their features, meanwhile trying to ignore the features of the irrelevant ones, i.e., selectively focusing (*attention*) on the informative joints is useful for human action recognition.

Human action can be represented by a combination of skeletal joints' movements. In order to reliably identify the informative joints in an action instance, we can evaluate the informativeness score of each joint in each frame with regarding to the global action sequence. To achieve this purpose, we need to obtain the global context information first. However, the available context at each evolution step of LSTM is the hidden representation from the previous step, which is relatively local when compared to the global action.

To mitigate the aforementioned limitation, we propose to introduce a global context memory cell for the LSTM model, which keeps the global context information of the action sequence, and can be fed to each step of LSTM to assist the attention procedure, as illustrated in Fig. 3. We call this new LSTM architecture as Global Context-Aware Attention LSTM (GCA-LSTM).

1) Overview of the GCA-LSTM Network: We illustrate the proposed GCA-LSTM network for skeleton based action recognition in Fig. 3. Our GCA-LSTM network contains three major modules. The *global context memory cell* maintains an overall representation of the whole action sequence. The *first ST-LSTM layer* encodes the skeleton sequence, and initializes the global context memory cell. The *second ST-LSTM layer* performs attention over the inputs at all spatio-temporal steps to generate an attention representation of the action sequence, which is then used to refine the global context memory.

The input at the spatio-temporal step (j, t) of the first ST-LSTM layer is the 3D coordinates of the joint j in frame t. The inputs of the second layer are the hidden representations from the first layer.

Multiple attention iterations (recurrent attention) are performed in our network to refine the global context memory iteratively. Finally, the refined global context memory can be used for classification.

To facilitate our explanation, we use $\mathbb{h}_{j,t}$ instead of $h_{j,t}$ to denote the hidden representation at the step (j, t) in the first ST-LSTM layer, while the symbols, including $h_{j,t}$, $c_{j,t}$, $i_{j,t}$, and $o_{j,t}$, which are defined in Section III-A, are utilized to represent the components in the second layer only.

2) Initializing the Global Context Memory Cell: Our GCA-LSTM network performs attention by using the global context information, therefore, we need to obtain an initial global context memory first.

A feasible scheme is utilizing the outputs of the first layer to generate a global context representation. We can average the hidden representations at all spatio-temporal steps of the first layer to compute an initial global context memory cell ($\mathbb{F}^{(0)}$) as follows:

$$\mathbb{F}^{(0)} = \frac{1}{JT}\sum_{j=1}^{J}\sum_{t=1}^{T}\mathbb{h}_{j,t} \tag{4}$$

We may also concatenate the hidden representations of the first layer and feed them to a feed-forward neural network, then use the resultant activation as $\mathbb{F}^{(0)}$. We empirically observe these two initialization schemes perform similarly.

3) Performing Attention in the Second ST-LSTM Layer: By using the global context information, we evaluate the informativeness degree of the input at each spatio-temporal step in the second ST-LSTM layer.

In the n-th attention iteration, our network learns an informativeness score ($r_{j,t}^{(n)}$) for each input ($\mathbb{h}_{j,t}$) by feeding the input itself, together with the global context memory cell ($\mathbb{F}^{(n-1)}$) generated by the previous attention iteration to a network as follows:

$$e_{j,t}^{(n)} = W_{e_1}\left(\tanh\left(W_{e_2}\begin{pmatrix}\mathbb{h}_{j,t}\\ \mathbb{F}^{(n-1)}\end{pmatrix}\right)\right) \tag{5}$$

$$r_{j,t}^{(n)} = \frac{\exp(e_{j,t}^{(n)})}{\sum_{u=1}^{J}\sum_{v=1}^{T}\exp(e_{u,v}^{(n)})} \tag{6}$$

where $r_{j,t}^{(n)} \in (0, 1)$ denotes the normalized informativeness score of the input at the step (j, t) in the n-th attention iteration, with regarding to the global context information.

The informativeness score $r_{j,t}^{(n)}$ is then used as a gate of the ST-LSTM unit, and we call it *informativeness gate*. With the assistance of the learned informativeness gate, the cell state of the unit in the second ST-LSTM layer can be updated as:

$$\begin{aligned} c_{j,t} = {} & r_{j,t}^{(n)} \odot i_{j,t} \odot u_{j,t} \\ & + (1 - r_{j,t}^{(n)}) \odot f_{j,t}^{(S)} \odot c_{j-1,t} \\ & + (1 - r_{j,t}^{(n)}) \odot f_{j,t}^{(T)} \odot c_{j,t-1} \end{aligned} \tag{7}$$

The cell state updating scheme in Eq. (7) can be explained as follows: (1) if the input ($\mathbb{h}_{j,t}$) is informative (important) with regarding to the global context representation, then we let the learning algorithm update the cell state of the second ST-LSTM layer by importing more information of it; (2) on the contrary, if the input is irrelevant, then we need to block the input gate at this step, meanwhile relying more on the history information of the cell state.

4) Refining the Global Context Memory Cell: We perform attention by adopting the cell state updating scheme in Eq. (7), and thereby obtain an attention representation of the action sequence. Concretely, the output of the last spatio-temporal step in the second layer is used as the attention representation ($\mathcal{F}^{(n)}$) for the action. Finally, the attention representation $\mathcal{F}^{(n)}$ is fed to the global context memory cell to refine it, as illustrated in Fig. 3. The refinement is formulated as follows:

$$\mathbb{F}^{(n)} = \mathrm{ReLu}\left(W_F^{(n)}\begin{pmatrix}\mathcal{F}^{(n)}\\ \mathbb{F}^{(n-1)}\end{pmatrix}\right) \tag{8}$$

where $\mathbb{F}^{(n)}$ is the refined version of $\mathbb{F}^{(n-1)}$. Note that $W_F^{(n)}$ is not shared over different iterations.

Multiple attention iterations (recurrent attention) are carried out in our GCA-LSTM network. Our motivation is that after we obtain a refined global context memory cell, we can use it to perform the attention again to more reliably identify the informative joints, and thus achieve a better attention representation, which can then be utilized to further refine the global context. After multiple iterations, the global context can be more discriminative for action classification.

5) Classifier: The last refined global context memory cell $\mathbb{F}^{(N)}$ is fed to a softmax classifier to predict the class label:

$$\hat{y} = \mathrm{softmax}\left(W_c\left(\mathbb{F}^{(N)}\right)\right) \tag{9}$$

The negative log-likelihood loss function [70] is adopted to measure the difference between the true class label y and the prediction result $\hat{y}$. The back-propagation algorithm is used to minimize the loss function. The details of the back-propagation process are described in Section III-C.

C. Training the Network

In this part, we first briefly describe the basic training method which directly optimizes the parameters of the whole network, we then propose a more advanced stepwise training scheme for our GCA-LSTM network.

1) Directly Train the Whole Network: Since the classification is performed by using the last refined global context, to train such a network, it is natural and intuitive to feed the action label as the training output at the last attention iteration, and back-propagate the errors from the last step, i.e., directly optimize the whole network as shown in Fig. 4(a).

2) Stepwise Training: Owing to the recurrent attention mechanism, there are frequent mutual interactions among different modules (the two ST-LSTM layers and the global context memory cell, see Fig. 3) in our network. Moreover, during the progress of multiple attention iterations, new parameters are also introduced. Due to these facts, it is rather difficult to simply optimize all parameters and all attention iterations of the whole network directly as mentioned above.

Therefore, we propose a stepwise training scheme for our GCA-LSTM network, which optimizes the model parameters incrementally. The details of this scheme are depicted in Fig. 4(b) and Algorithm 1.

The proposed stepwise training scheme is effective and efficient in optimizing the parameters and ensuring the convergence of the GCA-LSTM network. Specifically, at each training step n, we only need to optimize a subset of parameters and modules which are used by the attention

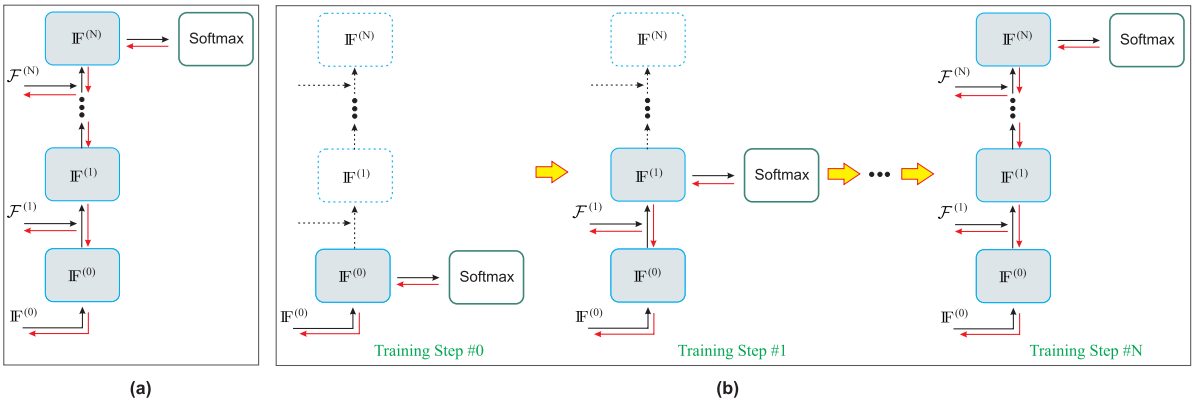


Fig. 4. Illustration of the two network training methods. (a) Directly train the whole network. (b) Stepwise optimize the network parameters. In this figure, the global context memory cell $\mathbb{F}^{(n)}$ is unfolded over the attention iterations. The training step #n corresponds to the n-th attention iteration. The black and red arrows denote the forward and backward passes, respectively. Some passes, such as those between the two ST-LSTM layers, are omitted for clarity. Better viewed in colour.

Algorithm 1 Stepwise Train the GCA-LSTM Network

1: Randomly initialize the parameters of the whole network with zero-mean Gaussian.
2: **for** n = 0 to N **do** // n is the training step
3: Feed the action label as the training output at the attention iteration n.
4: **do**
5: Training an epoch: optimizing the parameters used in the iterations 0 to n via back-propagation.
6: **while** Validation error is decreasing
7: **end for**

iterations 0 to n.[2] Training this shrunken network is more effective and efficient than directly training the whole network. At the step $n+1$, a larger scale network needs to be optimized. However, the training at step $n+1$ is also very efficient, as most of the parameters and passes have already been optimized (pre-trained well) by its previous training steps.

IV. Two-Stream GCA-LSTM Network

In the aforementioned design (Section III), the GCA-LSTM network performs action recognition by selectively focusing on the informative joints in each frame, i.e., the attention is carried out at joint level (fine-grained attention). Beside fine-grained attention, coarse-grained attention can also contribute to action analysis. This is because some actions are often performed at body part level. For these actions, all the joints from the same informative body part tend to have similar importance degrees. For example, the postures and motions of all the joints (elbow, wrist, palm, and finger) from the right hand are all important for recognizing the action *salute* in the NTU RGB+D dataset [55], i.e., we need to identify the informative body part "right hand" here. This implies coarse-grained (body part-level) attention is also useful for action recognition.

As suggested by Du *et al.* [12], the human skeleton can be divided into five body parts (torso, left hand, right hand, left leg, and right leg) based on the human physical structure. These five parts are illustrated as the right part of Fig. 5. Therefore, we can measure the informativeness degree of each body part with regarding to the action sequence, and then perform coarse-grained attention.

Specifically, we extend the design of out GCA-LSTM model, and introduce a two-stream GCA-LSTM network here, which jointly takes advantage of a fine-grained (joint-level) attention stream and a coarse-grained (body part-level) attention stream.

The architecture of the two-stream GCA-LSTM is illustrated in Fig. 5. In each attention stream, there is a global context memory cell to maintain the global attention representation of the action sequence, and also a second ST-LSTM layer to perform attention. This indicates we have two separated global context memory cells in the whole architecture, which are respectively the fine-grained attention memory cell ($\mathbb{F}_{(F)}^{(n)}$) and the coarse-grained attention memory cell ($\mathbb{F}_{(C)}^{(n)}$). The first ST-LSTM layer, which is used to encode the skeleton sequence and initialize the global context memory cells, is shared by the two attention streams.

The process flow (including initialization, attention, and refinement) in the fine-grained attention stream is the same as the GCA-LSTM model introduced in Section III. The operation in the coarse-grained attention stream is also similar. The main difference is that, in the second layer, the coarse-grained attention stream performs attention by selectively focusing on the informative body parts in each frame.

Concretely, in the attention iteration n, the network learns an informativeness score ($r_{P,t}^{(n)}$) for each body part P ($P \in \{1, 2, 3, 4, 5\}$) as:

$$e_{P,t}^{(n)} = W_{e3}\left(\tanh\left(W_{e4}\left(\begin{matrix}\bar{\mathbb{h}}_{P,t} \\ \mathbb{F}_{(C)}^{(n-1)}\end{matrix}\right)\right)\right) \tag{10}$$

$$r_{P,t}^{(n)} = \frac{\exp(e_{P,t}^{(n)})}{\sum_{u=1}^{5}\sum_{v=1}^{T}\exp(e_{u,v}^{(n)})} \tag{11}$$

where $\bar{\mathbb{h}}_{P,t}$ is the representation of the body part P at frame t, which is calculated based on the hidden representations of all

[2]Note that #0 is not an attention iteration, but the process of initializing the global context memory cell ($\mathbb{F}^{(0)}$). To facilitate the explantation of the stepwise training, we here temporally describe it as an attention iteration.

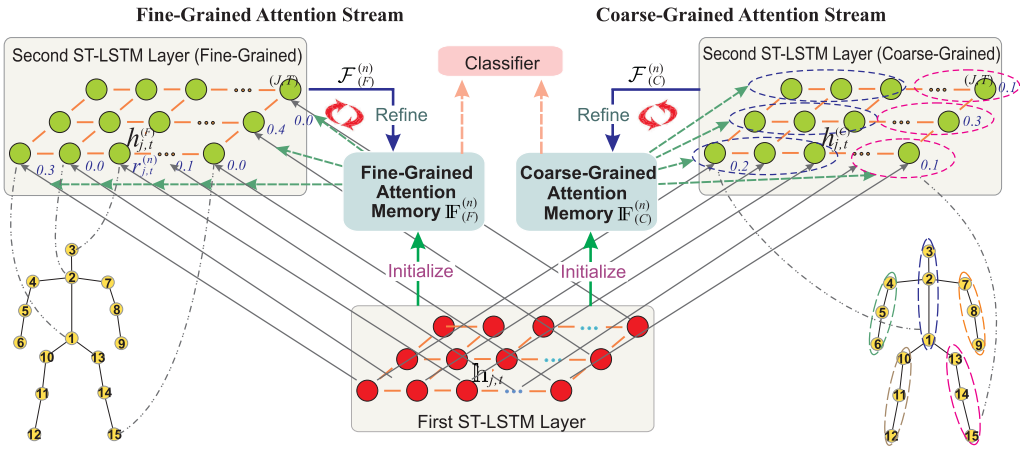


Fig. 5. Illustration of the two-stream GCA-LSTM network, which incorporates fine-grained (joint-level) attention and coarse-grained (body part-level) attention. To perform coarse-grained attention, the joints in a skeleton are divided into five body parts, and all the joints from the same body part share a same informative score. In the second ST-LSTM layer for coarse-grained attention, we only show two body parts at each frame, and other body parts are omitted for clarity.

the joints that belong to P, with average pooling as:

$$\bar{\mathbb{h}}_{P,t} = \frac{1}{J_P} \sum_{j \in P} \mathbb{h}_{j,t} \tag{12}$$

where J_P denotes the number of joints in body part P.

To perform coarse-grained attention, we allow each joint j in body part P to share the informativeness degree of P, i.e., at frame t, all the joints in P use the same informativeness score $r^{(n)}_{P,t}$, as illustrated in Fig. 5. Hence, in the coarse-grained attention stream, if $j \in P$, then the cell state of the second ST-LSTM layer is updated at the spatio-temporal step (j,t) as:

$$\begin{aligned} c_{j,t} = & r^{(n)}_{P,t} \odot i_{j,t} \odot u_{j,t} \\ & + (1 - r^{(n)}_{P,t}) \odot f^{(S)}_{j,t} \odot c_{j-1,t} \\ & + (1 - r^{(n)}_{P,t}) \odot f^{(T)}_{j,t} \odot c_{j,t-1} \end{aligned} \tag{13}$$

Multiple attention iterations are also performed in the proposed two-stream GCA-LSTM network. Finally, the refined fine-grained attention memory $\mathbb{F}^{(N)}_{(F)}$ and coarse-grained attention memory $\mathbb{F}^{(N)}_{(C)}$ are both fed to the softmax classifier, and the prediction scores of these two streams are averaged for action recognition.

The proposed step-wise training scheme can also be applied to this two-stream GCA-LSTM network, and at the training step #n, we simultaneously optimize the two attention streams, both of which correspond to the n-th attention iteration.

V. Experiments

We evaluate our proposed method on the NTU RGB + D [55], SYSU-3D [71], UT-Kinect [47], SBU-Kinect Interaction [72], and Berkeley MHAD [73] datasets. To investigate the effectiveness of our approach, we conduct extensive experiments with the following different network structures:

- "ST-LSTM + Global (1)". This network architecture is similar to the original two-layer ST-LSTM network in [29], but the hidden representations at all spatio-temporal steps of the second layer are concatenated and fed to a one-layer feed-forward network to generate a global representation of the skeleton sequence, and the classification is performed on the global representation; while in [29], the classification is performed on single hidden representation at each spatio-temporal step (local representation).
- "ST-LSTM + Global (2)". This network structure is similar to the above "ST-LSTM + Global (1)", except that the global representation is obtained by averaging the hidden representations of all spatio-temporal steps.
- "GCA-LSTM". This is the proposed Global Context-Aware Attention LSTM network. Two attention iterations are performed by this network. The classification is performed on the last refined global context memory cell. The two training methods (*direct training* and *stepwise training*) described in Section III-C are also evaluated for this network structure.

In addition, we also adopt the large scale NTU RGB+D and the challenging SYSU-3D as two major benchmark datasets to evaluate the proposed "two-stream GCA-LSTM" network.

We use Torch7 framework [74] to perform our experiments. Stochastic gradient descent (SGD) algorithm is adopted to train our end-to-end network. We set the learning rate, decay rate, and momentum to 1.5×10^{-3}, 0.95, and 0.9, respectively. The applied dropout probability [75] in our network is set to 0.5. The dimensions of the global context memory representation and the cell state of ST-LSTM are both 128.

A. Experiments on the NTU RGB + D Dataset

The NTU RGB + D dataset [55] was collected with Kinect (V2). It contains more than 56 thousand video samples. A total of 60 action classes were performed by 40 different subjects. To the best of our knowledge, this is the largest publicly available dataset for RGB + D based human action recognition. The large variations in subjects and viewpoints

TABLE I
EXPERIMENTAL RESULTS ON THE NTU RGB+D DATASET

Method	CS	CV
Skeletal Quads [76]	38.6%	41.4%
Lie Group [46]	50.1%	52.8%
Dynamic Skeletons [71]	60.2%	65.2%
HBRNN [12]	59.1%	64.0%
Deep RNN [55]	56.3%	64.1%
Deep LSTM [55]	60.7%	67.3%
Part-aware LSTM [55]	62.9%	70.3%
JTM CNN [77]	73.4%	75.2%
STA Model [68]	73.4%	81.2%
SkeletonNet [78]	75.9%	81.2%
Visualization CNN [79]	76.0%	82.6%
ST-LSTM [29]	69.2%	77.7%
ST-LSTM + Global (1)	70.5%	79.5%
ST-LSTM + Global (2)	70.7%	79.4%
GCA-LSTM (*direct training*)	74.3%	82.8%
GCA-LSTM (*stepwise training*)	**76.1%**	**84.0%**

make this dataset quite challenging. There are two standard evaluation protocols for this dataset: (1) Cross subject (CS): 20 subjects are used for training, and the remaining subjects are used for testing; (2) Cross view (CV): two camera views are used for training, and one camera view is used for testing. To extensively evaluate the proposed method, both protocols are tested in our experiment.

We compare the proposed GCA-LSTM network with state-of-the-art approaches, as shown in TABLE I. We can observe that our proposed GCA-LSTM model outperforms the other skeleton-based methods. Specifically, our GCA-LSTM network outperforms the original ST-LSTM network in [29] by 6.9% with the cross subject protocol, and 6.3% with the cross view protocol. This demonstrates that the attention mechanism in our network brings significant performance improvement.

Both "ST-LSTM + Global (1)" and "ST-LSTM + Global (2)" perform classification on the global representations, thus they achieve slightly better performance than the original ST-LSTM [29] which performs classification on local representations. We also observe "ST-LSTM + Global (1)" and "ST-LSTM + Global (2)" perform similarly.

The results in TABLE I also show that using the *stepwise training* method can improve the performance of our network in contrast to using the *direct training* method.

We also evaluate the performance of the two-stream GCA-LSTM network, and report the results in TABLE II. The results show that by incorporating fine-grained attention and coarse-grained attention, the proposed two-stream GCA-LSTM network achieves better performance than the GCA-LSTM with fine-grained attention only. We also observe the performance of two-stream GCA-LSTM can be improved with the *stepwise training* method.

B. Experiments on the SYSU-3D Dataset

The SYSU-3D dataset [71], which contains 480 skeleton sequences, was collected with Kinect. This dataset includes 12 action classes which were performed by 40 subjects. The SYSU-3D dataset is very challenging as the motion patterns are quite similar among different action classes, and there are lots of viewpoint variations in this dataset.

TABLE II
PERFORMANCE OF THE TWO-STREAM GCA-LSTM NETWORK ON THE NTU RGB+D DATASET

Method	CS	CV
GCA-LSTM (coarse-grained only)	74.1%	81.6%
GCA-LSTM (fine-grained only)	74.3%	82.8%
Two-stream GCA-LSTM	76.2%	84.7%
Two-stream GCA-LSTM with *stepwise training*	77.1%	85.1%

TABLE III
EXPERIMENTAL RESULTS ON THE SYSU-3D DATASET

Method	Accuracy
LAFF (SKL) [80]	54.2%
Dynamic Skeletons [71]	75.5%
ST-LSTM [56]	76.5%
ST-LSTM + Global (1)	76.8%
ST-LSTM + Global (2)	76.6%
GCA-LSTM (*direct training*)	77.8%
GCA-LSTM (*stepwise training*)	**78.6%**

TABLE IV
PERFORMANCE OF THE TWO-STREAM GCA-LSTM NETWORK ON THE SYSU-3D DATASET

Method	Accuracy
GCA-LSTM (coarse-grained only)	76.9%
GCA-LSTM (fine-grained only)	77.8%
Two-stream GCA-LSTM	78.8%
Two-stream GCA-LSTM with *stepwise training*	79.1%

TABLE V
EXPERIMENTAL RESULTS ON THE UT-KINECT DATASET

Method	Accuracy
Grassmann Manifold [81]	88.5%
Histogram of 3D Joints [47]	90.9%
Riemannian Manifold [82]	91.5%
Key-Pose-Motifs Mining [83]	93.5%
Action-Snippets and Activated Simplices [84]	96.5%
ST-LSTM [29]	97.0%
ST-LSTM + Global (1)	97.0%
ST-LSTM + Global (2)	97.5%
GCA-LSTM (*direct training*)	98.5%
GCA-LSTM (*stepwise training*)	**99.0%**

We follow the standard cross-validation protocol in [71] on this dataset, in which 20 subjects are adopted for training the network, and the remaining subjects are kept for testing. We report the experimental results in TABLE III. We can observe that our GCA-LSTM network surpasses the state-of-the-art skeleton-based methods in [29], [71], and [80], which demonstrates the effectiveness of our approach in handling the task of action recognition in skeleton sequences. The results also show that our proposed *stepwise training* scheme is useful for our network.

Using this challenging dataset, we also evaluate the performance of the two-stream attention model. The results in TABLE IV show that the two-stream GCA-LSTM network is effective for action recognition.

C. Experiments on the UT-Kinect Dataset

The UT-Kinect dataset [47] was recorded with a stationary Kinect. The skeleton sequences in this dataset are quite noisy.

TABLE VI

EVALUATION OF ROBUSTNESS AGAINST THE INPUT NOISE. GAUSSIAN NOISE $\mathcal{N}(0, \sigma^2)$ IS ADDED TO THE 3D COORDINATES OF THE SKELETAL JOINTS

Standard deviation (σ) of noise	$0.1cm$	$1cm$	$2cm$	$4cm$	$8cm$	$12cm$	$16cm$	$32cm$
Accuracy	100%	99.3%	98.5%	97.5%	95.6%	92.7%	80.4%	61.5%

TABLE VII

PERFORMANCE COMPARISON OF DIFFERENT ATTENTION ITERATION NUMBERS (N)

#Attention Iteration	NTU RGB+D (CS)	NTU RGB+D (CV)	UT-Kinect	SYSU-3D	Berkeley MHAD
1	72.9%	81.8%	98.0%	77.8%	**100%**
2	**76.1%**	**84.0%**	**99.0%**	**78.6%**	**100%**

TABLE VIII

PERFORMANCE COMPARISON OF DIFFERENT PARAMETER SHARING SCHEMES

#Attention Iteration	(a) w/o sharing within iteration w/ sharing cross iterations	(b) w/ sharing within iteration w/ sharing cross iterations	(c) w/o sharing within iteration w/o sharing cross iterations	(d) w/ sharing within iteration w/o sharing cross iterations
1	71.0%	72.9%	71.0%	72.9%
2	73.0%	74.3%	**73.4%**	**76.1%**
3	**73.1%**	**74.4%**	69.3%	73.2%

A total of 10 action classes were performed by 10 subjects, and each action was performed by the same subject twice.

We follow the standard leave-one-out-cross-validation protocol in [47] to evaluate our method on this dataset. Our approach yields state-of-the-art performance on this dataset, as shown in TABLE V.

D. Experiments on the SBU-Kinect Interaction Dataset

The SBU-Kinect Interaction dataset [72] includes 8 action classes for two-person interaction recognition. This dataset contains 282 sequences corresponding to 6822 frames. The SBU-Kinect Interaction dataset is challenging because of (1) the relatively low accuracies of the coordinates of skeletal joints recorded by Kinect, and (2) complicated interactions between two persons in many action sequences.

We perform 5-fold cross-validation evaluation on this dataset by following the standard protocol in [72]. The experimental results are depicted in TABLE IX. In this table, HBRNN [12], Deep LSTM [28], Co-occurrence LSTM [28], and ST-LSTM [29] are all LSTM based models for action recognition in skeleton sequences, and are very relevant to our network. We can see that the proposed GCA-LSTM network achieves the best performance among all of these methods.

E. Experiments on the Berkeley MHAD Dataset

The Berkeley MHAD dataset was recorded by using a motion capture system. It contains 659 sequences and 11 action classes, which were performed by 12 different subjects.

We adopt the standard experimental protocol on this dataset, in which 7 subjects are used for training and the remaining 5 subjects are held out for testing. The results in TABLE X show that our method achieves very high accuracy (100%) on this dataset.

As the Berkeley MHAD dataset was collected with a motion capture system rather than a Kinect, thus the coordinates of the skeletal joints are relatively accurate. To evaluate the robustness with regarding to the input noise, we also investigate the performance of our GCA-LSTM network on this dataset by adding zero mean input noise to the skeleton sequences, and show the results in TABLE VI. We can see that even if we add noise with the standard deviation (σ) set to 12 *cm* (which is significant noise in the scale of human body), the accuracy of our method is still very high (92.7%). This demonstrates that our method is quite robust against the input noise.

F. Evaluation of Attention Iteration Numbers

We also test the effect of different attention iteration numbers on our GCA-LSTM network, and show the results in TABLE VII. We can observe that increasing the iteration number can help to strength the classification performance of our network (using 2 iterations obtains higher accuracies compared to using only 1 iteration). This demonstrates that the recurrent attention mechanism proposed by us is useful for the GCA-LSTM network.

Specifically, we also evaluate the performance of 3 attention iterations by using the large scale NTU RGB + D dataset, and the results are shown in TABLE VIII. We find the performance of 3 attention iterations is slightly better than 2 iterations if we share the parameters over different attention iterations (see columns (a) and (b) in TABLE VIII). This consistently shows using multiple attention iterations can improve the performance of our network progressively. We do not try more iterations due to the GPU's memory limitation.

We also find that if we do not share the parameters over different attention iterations (see columns (c) and (d) in TABLE VIII), then too many iterations can bring performance degradation (the performance of using 3 iterations is worse than that of using 2 iterations). In our experiment, we observe the performance degradation is caused by overfitting (increasing iteration number will introduce new parameters if we do not share parameters). But the performance of

TABLE IX
EXPERIMENTAL RESULTS ON THE SBU-KINECT INTERACTION DATASET

Method	Accuracy
Yun *et al.* [72]	80.3%
CHARM [85]	83.9%
Ji *et al.* [86]	86.9%
HBRNN [12]	80.4%
Deep LSTM [28]	86.0%
Co-occurrence LSTM [28]	90.4%
SkeletonNet [78]	93.5%
ST-LSTM [29]	93.3%
GCA-LSTM (*direct training*)	94.1%
GCA-LSTM (*stepwise training*)	**94.9%**

TABLE X
EXPERIMENTAL RESULTS ON THE BERKELEY MHAD DATASET

Method	Accuracy
Ofli *et al.* [43]	95.4%
Vantigodi *et al.* [87]	96.1%
Vantigodi *et al.* [88]	97.6%
Kapsouras *et al.* [89]	98.2%
ST-LSTM [29]	100%
GCA-LSTM (*direct training*)	100%
GCA-LSTM (*stepwise training*)	100%

two iterations is still significantly better than one iteration in this case. We will also give the experimental analysis of the parameter sharing schemes detailed in Section V-G.

G. Evaluation of Parameter Sharing Schemes

As formulated in Eq. (5), the model parameters W_{e_1} and W_{e_2} are introduced for calculating the informativeness score at each spatio-temporal step in the second layer. Also multiple attention iterations are carried out in this layer. To regularize the parameter number inside our network and improve the generalization capability, we investigate two parameter sharing strategies for our network: (1) Sharing within iteration: W_{e_1} and W_{e_2} are shared by all spatio-temporal steps in the same attention iteration; (2) Sharing cross iterations: W_{e_1} and W_{e_2} are shared over different attention iterations. We investigate the effect of these two parameter sharing strategies on our GCA-LSTM network, and report the results in TABLE VIII.

In TABLE VIII, we can observe that: (1) Sharing parameters within iteration is useful for enhancing the generalization capability of our network, as the performance in columns (b) and (d) of TABLE VIII is better than (a) and (c), respectively. (2) Sharing parameters over different iterations is also helpful for handling the over-fitting issues, but it may limit the representation capacity, as the network with two attention iterations which shares parameters within iteration but does not share parameters over iterations achieves the best result (see column (d) of TABLE VIII). As a result, in our GCA-LSTM network, we only share the parameters within iteration, and two attention iterations are used.

H. Evaluation of Training Methods

The previous experiments showed that using the *stepwise training* method can improve the performance of our network in contrast to using *direct training* (see TABLE I, V, III, IX). To further investigate the performance of these two training methods, we plot the convergence curves of our GCA-LSTM network in Fig. 6.

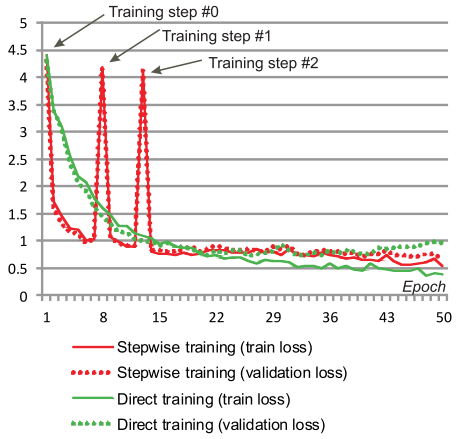


Fig. 6. Convergence curves of the GCA-LSTM network with two attention iterations by respectively using *stepwise training* (in red) and *direct training* (in green) on the NTU RGB + D dataset. Better viewed in colour.

We analyze the convergence curves (Fig. 6) of the *stepwise training* method as follows. By using the proposed *stepwise training* method, at the training step #0, we only need to train the subnetwork for initializing the global context ($\mathbb{F}^{(0)}$), i.e., only a subset of parameters and modules need to be optimized, thus the training is very efficient and the loss curve converges very fast. When the validation loss stops decreasing, we start the next training step #1. Step #1 contains new parameters and modules for the first attention iteration, which have not been optimized yet, therefore, loss increases immediately at this epoch. However, most of the parameters involved at this step have already been pre-trained well by the previous step #0, thus the network training is quite effective, and the loss drops to a very low value after only one training epoch.

By comparing the convergence curves of the two training methods, we can find (1) the network converges much faster if we use *stepwise training*, compared to *directly train* the whole network. We can also observe that (2) the network is easier to get over-fitted by using *direct training* method, as the gap between the train loss and validation loss starts to rise after the 20th epoch. These observations demonstrate that the proposed *stepwise training* scheme is quite useful for effectively and efficiently training our GCA-LSTM network.

I. Evaluation of Initialization Methods and Attention Designs

In Section III-B.2, we introduce two methods to initialize the global context memory cell ($\mathbb{F}^{(0)}$). The first is averaging the hidden representations of the first layer (see Eq. (4)), and the second is using a one-layer feed-forward network to obtain $\mathbb{F}^{(0)}$. We compare these two initialization methods in TABLE XI. The results show that these two methods perform similarly. In our experiment, we also find that by using feed-forward network, the model converges faster, thus the scheme of feed-forward network is used to initialize the global context memory cell in our GCA-LSTM network.

In the GCA-LSTM network, the informativeness score $r_{j,t}^{(n)}$ is used as a gate within LSTM neuron, as formulated in Eq. (7). We also explore to replace this scheme with soft

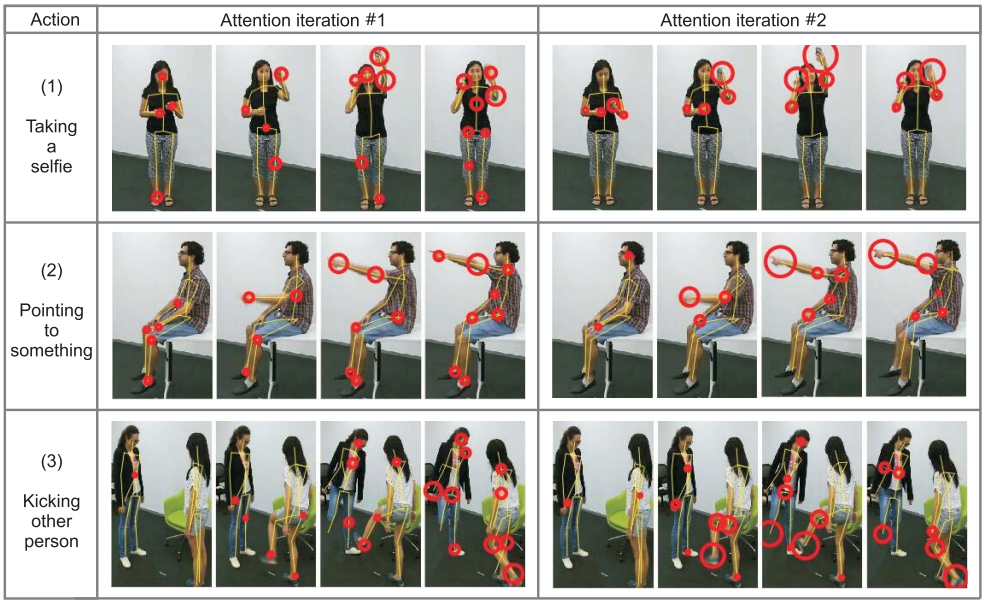


Fig. 7. Examples of qualitative results on the NTU RGB + D dataset. Three actions (*taking a selfie*, *pointing to something*, and *kicking other person*) are illustrated. The informativeness scores of two attention iterations are visualized. Four frames are shown for each iteration. The circle size indicates the magnitude of the informativeness score for the corresponding joint in a frame. For clarity, the joints with tiny informativeness scores are not shown.

TABLE XI
PERFORMANCE COMPARISON OF DIFFERENT METHODS OF INITIALIZING THE GLOBAL CONTEXT MEMORY CELL

Method	NTU RGB+D (CS)	NTU RGB+D (CV)
Averaging	73.8%	83.1%
Feed-forward network	74.3%	82.8%

attention method [15], [62], i.e., the attention representation $\mathcal{F}^{(n)}$ is calculated as $\sum_{j=1}^{J}\sum_{t=1}^{T} r_{j,t}^{(n)} \mathbb{h}_{j,t}$. Using soft attention, the accuracy drops about one percentage point on the NTU RGB + D dataset. This can be explained as equipping LSTM neuron with gate $r_{j,t}^{(n)}$ provides LSTM better insight about when to update, forget or remember. In addition, it can keep the sequential ordering information of the inputs $\mathbb{h}_{j,t}$, while soft attention loses ordering and positional information.

J. Visualizations

To better understand our network, we analyze and visualize the informativeness score evaluated by using the global context information on the large scale NTU RGB + D dataset in this section.

We analyze the variations of the informativeness scores over the two attention iterations to verify the effectiveness of the recurrent attention mechanism in our method, and show the qualitative results of three actions (*taking a selfie*, *pointing to something*, and *kicking other person*) in Fig. 7. The informativeness scores are computed with soft attention for visualization. In this figure, we can see that the attention performance increases between the two attention iterations. In the first iteration, the network tries to identify the potential informative joints over the frames. After this attention, the network achieves a good understanding of the global action. Then in the second iteration, the network can more accurately focus on the informative joints in each frame of the skeleton sequence. We can also find that the informativeness score of the same joint can vary in different frames. This indicates that *our network performs attention not only in spatial domain, but also in temporal domain*.

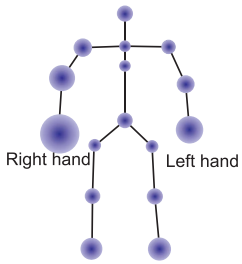


Fig. 8. Visualization of the average informativeness gates for all testing samples. The size of the circle around each joint indicates the magnitude of the corresponding informativeness score.

In order to further quantitatively evaluate the effectiveness of the attention mechanism, we analyze the classification accuracies of the three action classes in Fig. 7 among all the actions. We observe if the attention mechanism is not used, the accuracies of these three classes are 67.7%, 71.7%, and 81.5%, respectively. However, if we use one attention iteration, the accuracies rise to 67.8%, 72.4%, and 83.4%, respectively. If two attention iterations are performed, the accuracies become 67.9%, 73.6%, and 86.6%, respectively.

To roughly explore which joints are more informative for the activities in the NTU RGB + D dataset, we also average the informativeness scores of the same joint in all the testing sequences, and visualize it in Fig. 8. We can observe that averagely, more attention is assigned to the hand and foot joints. This is because in the NTU RGB + D dataset, most of the actions are related to the hand and foot postures and motions. We can also find that the average informativeness score of the right hand joint is higher than that of left hand joint. This indicates most of the subjects are right-handed.

VI. Conclusion

In this paper, we have extended the original LSTM network to construct a Global Context-Aware Attention LSTM (GCA-LSTM) network for skeleton based action recognition, which has strong ability in selectively focusing on the informative joints in each frame of the skeleton sequence with the assistance of global context information. Furthermore, we have proposed a recurrent attention mechanism for our GCA-LSTM network, in which the selectively focusing capability is improved iteratively. In addition, a two-stream attention framework is also introduced. The experimental results validate the contributions of our approach by achieving state-of-the-art performance on five challenging datasets.

Acknowledgement

This work was carried out at the Rapid-Rich Object Search (ROSE) Lab at Nanyang Technological University (NTU), Singapore.

The authors acknowledge the support of NVIDIA AI Technology Centre (NVAITC) for the donation of the Tesla K40 and K80 GPUs used for their research at the ROSE Lab. J. Liu would like to thank Qiuhong Ke from University of Western Australia for helpful discussions.

References

[1] J. Zheng, Z. Jiang, and R. Chellappa, "Cross-view action recognition via transferable dictionary learning," *IEEE Trans. Image Process.*, vol. 25, no. 6, pp. 2542–2556, Jun. 2016.

[2] F. Liu, X. Xu, S. Qiu, C. Qing, and D. Tao, "Simple to complex transfer learning for action recognition," *IEEE Trans. Image Process.*, vol. 25, no. 2, pp. 949–960, Feb. 2016.

[3] Y.-G. Jiang, Q. Dai, W. Liu, X. Xue, and C.-W. Ngo, "Human action recognition in unconstrained videos by explicit motion modeling," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 3781–3795, Nov. 2015.

[4] J. Han, L. Shao, D. Xu, and J. Shotton, "Enhanced computer vision with microsoft kinect sensor: A review," *IEEE Trans. Cybern.*, vol. 43, no. 5, pp. 1318–1334, Oct. 2013.

[5] G. Zhang, J. Liu, H. Li, Y. Q. Chen, and L. S. Davis, "Joint human detection and head pose estimation via multistream networks for RGB-D videos," *IEEE Signal Process. Lett.*, vol. 24, no. 11, pp. 1666–1670, Nov. 2017.

[6] G. Zhang *et al.*, "Robust real-time human perception with depth camera," in *Proc. ECAI*, 2016, pp. 304–310.

[7] L. L. Presti and M. La Cascia, "3D skeleton-based human action classification: A survey," *Pattern Recognit.*, vol. 53, pp. 130–147, May 2016.

[8] F. Han, B. Reily, W. Hoff, and H. Zhang, "Space-time representation of people based on 3D skeletal data: A review," *Comput. Vis. Image Understand.*, vol. 158, pp. 85–105, May 2017.

[9] J. K. Aggarwal and L. Xia, "Human activity recognition from 3D data: A review," *Pattern Recognit. Lett.*, vol. 48, pp. 70–80, Oct. 2014.

[10] J. Zhang, W. Li, P. O. Ogunbona, P. Wang, and C. Tang, "RGB-D-based action recognition datasets: A survey," *Pattern Recognit.*, vol. 60, pp. 86–105, Dec. 2016.

[11] M. Ye, Q. Zhang, L. Wang, J. Zhu, R. Yang, and J. Gall, "A survey on human motion analysis from depth data," in *Time-of-Flight and Depth Imaging. Sensors, Algorithms, and Applications*. Berlin, Germany: Springer, 2013.

[12] Y. Du, W. Wang, and L. Wang, "Hierarchical recurrent neural network for skeleton based action recognition," in *Proc. CVPR*, 2015, pp. 1110–1118.

[13] M. Jiang, J. Kong, G. Bebis, and H. Huo, "Informative joints based human action recognition using skeleton contexts," *Signal Process., Image Commun.*, vol. 33, pp. 29–40, Apr. 2015.

[14] J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," in *Proc. NIPS*, 2015, pp. 577–585.

[15] K. Xu *et al.*, "Show, attend and tell: Neural image caption generation with visual attention," in *Proc. ICML*, 2015, pp. 2048–2057.

[16] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proc. ICLR*, 2015, pp. 1–15.

[17] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[18] M. Sundermeyer, R. Schlüter, and H. Ney, "LSTM neural networks for language modeling," in *Proc. INTERSPEECH*, 2012, pp. 194–197.

[19] M. Ibrahim, S. Muralidharan, Z. Deng, A. Vahdat, and G. Mori, "A hierarchical deep temporal model for group activity recognition," in *Proc. CVPR*, 2016, pp. 1971–1980.

[20] S. Yeung, O. Russakovsky, G. Mori, and L. Fei-Fei, "End-to-end learning of action detection from frame glimpses in videos," in *Proc. CVPR*, 2016, pp. 2678–2687.

[21] J. Yue-Hei Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici, "Beyond short snippets: Deep networks for video classification," in *Proc. CVPR*, 2015, pp. 4694–4702.

[22] Z. Wu, X. Wang, Y.-G. Jiang, H. Ye, and X. Xue, "Modeling spatial-temporal clues in a hybrid deep learning framework for video classification," in *Proc. ACM MM*, 2015, pp. 461–470.

[23] J. Donahue *et al.*, "Long-term recurrent convolutional networks for visual recognition and description," in *Proc. CVPR*, 2015, pp. 2625–2634.

[24] Q. Ke, M. Bennamoun, S. An, F. Sohel, and F. Boussaid. (2017). "Leveraging structural context models and ranking score fusion for human interaction prediction." [Online]. Available: https://arxiv.org/abs/1608.05267

[25] Q. Ke, M. Bennamoun, S. An, F. Bossaid, and F. Sohel. (2016). "Leveraging structural context models and ranking score fusion for human interaction prediction." [Online]. Available: https://arxiv.org/abs/1608.05267

[26] N. Srivastava, E. Mansimov, and R. Salakhutdinov, "Unsupervised learning of video representations using LSTMs," in *Proc. ICML*, 2015, pp. 843–852.

[27] S. Ma, L. Sigal, and S. Sclaroff, "Learning activity progression in LSTMs for activity detection and early detection," in *Proc. CVPR*, 2016, pp. 1942–1950.

[28] W. Zhu *et al.*, "Co-occurrence feature learning for skeleton based action recognition using regularized deep LSTM networks," in *Proc. AAAI*, 2016, p. 8.

[29] J. Liu, A. Shahroudy, D. Xu, and G. Wang, "Spatio-temporal LSTM with trust gates for 3d human action recognition," in *Proc. ECCV*, 2016, pp. 816–833.

[30] J. Weston, S. Chopra, and A. Bordes, "Memory networks," in *Proc. ICLR*, 2015, pp. 1–15.

[31] 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. CVPR*, 2017, pp. 1647–1656.

[32] J. Luo, W. Wang, and H. Qi, "Group sparsity and geometry constrained dictionary learning for action recognition from depth maps," in *Proc. ICCV*, 2013, pp. 1809–1816.

[33] A. Shahroudy, T.-T. Ng, Q. Yang, and G. Wang, "Multimodal multipart learning for action recognition in depth videos," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 10, pp. 2123–2129, Oct. 2016.

[34] M. Meng, H. Drira, M. Daoudi, and J. Boonaert, "Human-object interaction recognition by learning the distances between the object and the skeleton joints," in *Proc. FG*, 2015, pp. 1–6.

[35] X. Yang and Y. Tian, "Effective 3D action recognition using EigenJoints," *J. Vis. Commun. Image Represent.*, vol. 25, no. 1, pp. 2–11, 2014.

[36] J. Wang and Y. Wu, "Learning maximum margin temporal warping for action recognition," in *Proc. ICCV*, 2013, pp. 2688–2695.

[37] I. Lillo, J. Carlos Niebles, and A. Soto, "A hierarchical pose-based approach to complex action understanding using dictionaries of action-lets and motion poselets," in *Proc. CVPR*, 2016, pp. 1981–1990.

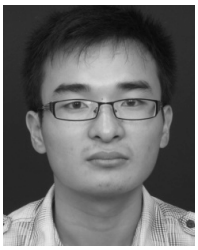
[38] H. Rahmani, A. Mahmood, D. Q. Huynh, and A. Mian, "Real time action recognition using histograms of depth gradients and random decision forests," in *Proc. WACV*, 2014, pp. 626–633.

[39] C. Chen, R. Jafari, and N. Kehtarnavaz, "Fusion of depth, skeleton, and inertial data for human action recognition," in *Proc. ICASSP*, Mar. 2016, pp. 2712–2716.

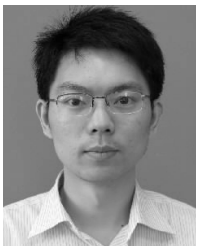
[40] L. Tao and R. Vidal, "Moving poselets: A discriminative and interpretable skeletal motion representation for action recognition," in *Proc. ICCVW*, 2015, pp. 61–69.

[41] A. Shahroudy, G. Wang, and T.-T. Ng, "Multi-modal feature fusion for action recognition in RGB-D sequences," in *Proc. ISCCSP*, 2014, pp. 1–4.

[42] P. Wang, W. Li, P. Ogunbona, Z. Gao, and H. Zhang, "Mining mid-level features for action recognition based on effective skeleton representation," in *Proc. DICTA*, 2014, pp. 1–8.
[43] F. Ofli, R. Chaudhry, G. Kurillo, R. Vidal, and R. Bajcsy, "Sequence of the most informative joints (SMIJ): A new representation for human skeletal action recognition," *J. Vis. Commun. Image Represent.*, vol. 25, no. 1, pp. 24–38, 2014.
[44] R. Anirudh, P. Turaga, J. Su, and A. Srivastava, "Elastic functional coding of human actions: From vector-fields to latent variables," in *Proc. CVPR*, 2015, pp. 3147–3155.
[45] R. Chaudhry, F. Ofli, G. Kurillo, R. Bajcsy, and R. Vidal, "Bio-inspired dynamic 3D discriminative skeletal features for human action recognition," in *Proc. CVPR*, 2013, pp. 471–478.
[46] R. Vemulapalli, F. Arrate, and R. Chellappa, "Human action recognition by representing 3d skeletons as points in a lie group," in *Proc. CVPR*, 2014, pp. 588–595.
[47] L. Xia, C.-C. Chen, and J. K. Aggarwal, "View invariant human action recognition using histograms of 3D joints," in *Proc. CVPR*, 2012, pp. 20–27.
[48] J. Wang, Z. Liu, Y. Wu, and J. Yuan, "Mining actionlet ensemble for action recognition with depth cameras," in *Proc. CVPR*, 2012, pp. 1290–1297.
[49] J. Wang, Z. Liu, Y. Wu, and J. Yuan, "Learning actionlet ensemble for 3D human action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 5, pp. 914–927, May 2014.
[50] H. Chen, G. Wang, J.-H. Xue, and L. He, "A novel hierarchical framework for human action recognition," *Pattern Recognit.*, vol. 55, pp. 148–159, Jul. 2016.
[51] P. Koniusz, A. Cherian, and F. Porikli, "Tensor representations via kernel linearization for action recognition from 3D skeletons," in *Proc. ECCV*, 2016, pp. 37–53.
[52] P. Wang, C. Yuan, W. Hu, B. Li, and Y. Zhang, "Graph based skeleton motion representation and similarity measurement for action recognition," in *Proc. ECCV*, 2016, pp. 370–385.
[53] M. Zanfir, M. Leordeanu, and C. Sminchisescu, "The moving pose: An efficient 3d kinematics descriptor for low-latency action recognition and detection," in *Proc. ICCV*, 2013, pp. 2752–2759.
[54] V. Veeriah, N. Zhuang, and G.-J. Qi, "Differential recurrent neural networks for action recognition," in *Proc. ICCV*, 2015, pp. 4041–4049.
[55] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang, "NTU RGB+D: A large scale dataset for 3D human activity analysis," in *Proc. CVPR*, 2016, pp. 1010–1019.
[56] 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 Trans. Pattern Anal. Mach. Intell.*, to be published, doi: 10.1109/TPAMI.2017.2771306.
[57] A. Jain, A. R. Zamir, S. Savarese, and A. Saxena, "Structural-RNN: Deep learning on spatio-temporal graphs," in *Proc. CVPR*, 2016, pp. 5308–5317.
[58] Y. Li, C. Lan, J. Xing, W. Zeng, C. Yuan, and J. Liu, "Online human action detection using joint classification-regression recurrent neural networks," in *Proc. ECCV*, 2016, pp. 203–220.
[59] C. Xiong, S. Merity, and R. Socher, "Dynamic memory networks for visual and textual question answering," in *Proc. ICML*, 2016, pp. 2397–2406.
[60] S. Sharma, R. Kiros, and R. Salakhutdinov, "Action recognition using visual attention," in *Proc. ICLR*, 2016, pp. 1–11.
[61] A. Kumar *et al.*, "Ask me anything: Dynamic memory networks for natural language processing," in *Proc. ICML*, 2016, pp. 1378–1387.
[62] M.-T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," in *Proc. EMNLP*, 2015, pp. 1–10.
[63] S. Sukhbaatar, A. Szlam, J. Weston, and R. Fergus, "End-to-end memory networks," in *Proc. NIPS*, 2015, pp. 2440–2448.
[64] M. F. Stollenga, J. Masci, F. Gomez, and J. Schmidhuber, "Deep networks with internal selective attention through feedback connections," in *Proc. NIPS*, 2014, pp. 3545–3553.
[65] L. Yao *et al.*, "Describing videos by exploiting temporal structure," in *Proc. ICCV*, 2015, pp. 4507–4515.
[66] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Proc. NIPS*, 2014, pp. 568–576.
[67] A. Shahroudy, T.-T. Ng, Y. Gong, and G. Wang, "Deep multimodal feature analysis for action recognition in RGB+D videos," *IEEE Trans. Pattern Anal. Mach. Intell.*, to be published, doi: 10.1109/TPAMI.2017.2691321.
[68] 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," in *Proc. AAAI*, 2017, pp. 4263–4270.
[69] Y. Wang, S. Wang, J. Tang, N. O'Hare, Y. Chang, and B. Li. (2016). "Hierarchical attention network for action recognition in videos." [Online]. Available: https://arxiv.org/abs/1607.06416
[70] A. Graves, "Supervised sequence labelling," in *Supervised Sequence Labelling with Recurrent Neural Networks*. Berlin, Germany: Springer, 2012.
[71] J.-F. Hu, W.-S. Zheng, J. Lai, and J. Zhang, "Jointly learning heterogeneous features for RGB-D activity recognition," in *Proc. CVPR*, 2015, pp. 5344–5352.
[72] 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. CVPRW*, 2012, pp. 28–35.
[73] F. Ofli, R. Chaudhry, G. Kurillo, R. Vidal, and R. Bajcsy, "Berkeley MHAD: A comprehensive multimodal human action database," in *Proc. WACV*, 2013, pp. 53–60.
[74] R. Collobert, K. Kavukcuoglu, and C. Farabet, "Torch7: A MATLAB-like environment for machine learning," in *Proc. NIPSW*, 2011, pp. 1–6.
[75] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.
[76] G. Evangelidis, G. Singh, and R. Horaud, "Skeletal quads: Human action recognition using joint quadruples," in *Proc. ICPR*, 2014, pp. 4513–4518.
[77] P. Wang, Z. Li, Y. Hou, and W. Li, "Action recognition based on joint trajectory maps using convolutional neural networks," in *Proc. ACM MM*, 2016, pp. 102–106.
[78] Q. Ke, S. An, M. Bennamoun, F. Sohel, and F. Boussaid, "SkeletonNet: Mining deep part features for 3-D action recognition," *IEEE Signal Process. Lett.*, vol. 24, no. 6, pp. 731–735, Jun. 2017.
[79] M. Liu, H. Liu, and C. Chen, "Enhanced skeleton visualization for view invariant human action recognition," *Pattern Recognit.*, vol. 68, pp. 346–362, Aug. 2017.
[80] J.-F. Hu, W.-S. Zheng, L. Ma, G. Wang, and J. Lai, "Real-time RGB-D activity prediction by soft regression," in *Proc. ECCV*, 2016, pp. 280–296.
[81] R. Slama, H. Wannous, M. Daoudi, and A. Srivastava, "Accurate 3D action recognition using learning on the grassmann manifold," *Pattern Recognit.*, vol. 48, no. 2, pp. 556–567, 2015.
[82] M. Devanne, H. Wannous, S. Berretti, P. Pala, M. Daoudi, and A. Del Bimbo, "3-D human action recognition by shape analysis of motion trajectories on riemannian manifold," *IEEE Trans. Cybern.*, vol. 45, no. 7, pp. 1340–1352, Jul. 2015.
[83] C. Wang, Y. Wang, and A. L. Yuille, "Mining 3D key-pose-motifs for action recognition," in *Proc. CVPR*, 2016, pp. 2639–2647.
[84] C. Wang, J. Flynn, Y. Wang, and A. L. Yuille, "Recognizing actions in 3d using action-snippets and activated simplices," in *Proc. AAAI*, 2016, pp. 3604–3610.
[85] W. Li, L. Wen, M. C. Chuah, and S. Lyu, "Category-blind human action recognition: A practical recognition system," in *Proc. ICCV*, 2015, pp. 4444–4452.
[86] Y. Ji, G. Ye, and H. Cheng, "Interactive body part contrast mining for human interaction recognition," in *Proc. ICMEW*, 2014, pp. 1–6.
[87] S. Vantigodi and R. V. Babu, "Real-time human action recognition from motion capture data," in *Proc. NCVPRIPG*, 2013, pp. 1–4.
[88] S. Vantigodi and V. B. Radhakrishnan, "Action recognition from motion capture data using meta-cognitive RBF network classifier," in *Proc. ISSNIP*, 2014, pp. 1–6.
[89] I. Kapsouras and N. Nikolaidis, "Action recognition on motion capture data using a dynemes and forward differences representation," *J. Vis. Commun. Image Represent.*, vol. 25, no. 6, pp. 1432–1445, 2014.



Jun Liu received the B.Eng. degree from Central South University, China, in 2011, and the M.Sc. degree from Fudan University, China, in 2014. He is currently pursuing the Ph.D. degree with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. His research interests include video analysis, human action recognition, and deep learning.



Gang Wang was an Associate Professor with the School of Electrical and Electronic Engineering, Nanyang Technological University. He is currently a Researcher and a Technique Leader with Alibaba.

He received the B.Eng. degree in electrical engineering from the Harbin Institute of Technology and the Ph.D. degree in electrical and computer engineering from the University of Illinois at Urbana–Champaign. He was a recipient of the MIT Technology Review Innovator Under 35 Award (Asia).

He is the Area Chair of ICCV'17 and CVPR'18 and an Associate Editor of the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE.

Ling-Yu Duan received the Ph.D. degree in information technology from The University of Newcastle in 2007. Since 2008, he has been with Peking University, China. Since 2012, he has been the Deputy Director of the Rapid-Rich Object Search Laboratory, a joint laboratory between Nanyang Technological University and Peking University. He is currently a Full Professor with the School of Electrical Engineering and Computer Science, Peking University. He is leading the group of visual search with the Institute of Digital Media, Peking University.

His research interests include the areas of visual search, augmented reality, and multimedia content analysis.

Kamila Abdiyeva received the B.Eng. degree in computer science from Nazarbayev University, Kazakhstan, in 2015. She is currently pursuing the Ph.D. degree with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. Her research interests include computer vision, machine learning, and deep learning.

Alex C. Kot (F'05) has been with Nanyang Technological University, since 1991. He is currently the Director of Rapid-Rich Object Search Laboratory and the Director of the NTU-PKU Joint Research Institute.

He has co-authored for several best paper awards including ICPR, WIFS, and IWDW. He served as an Associate Editor for the IEEE TRANSACTIONS ON SIGNAL PROCESSING, the IEEE TRANSACTIONS ON MULTIMEDIA, the IEEE *Signal Processing Magazine*, the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, and the IEEE TRANSACTIONS ON IMAGE PROCESSING. He is the IEEE Distinguished Lecturer of the Signal Processing Society and the Circuits and Systems Society and a fellow of the IES. He is a fellow of the Academy of Engineering, Singapore.