Attention-Based Multiview Re-Observation Fusion Network for Skeletal Action Recognition

Zhaoxuan Fan, Xu Zhao, Tianwei Lin, and Haisheng Su

***Abstract*—Action recognition is an important and popular area in computer vision. Because of the helpfulness of action recognition of the skeleton and the development of related pose estimation techniques, action recognition based on skeleton data has drawn considerable attention and has been widely studied in recent years. In this paper, we propose an attention-based multiview re-observation fusion model for skeletal action recognition. The proposed model focuses on the factor of observation view of actions, which greatly influences action recognition. The model utilizes action information from multiple observation views to improve the recognition performance. In this method, we re-observe input skeleton data from several possible viewpoints, process these augmented observation data with a long short-term memory (LSTM) network separately, and, finally, fuse the outputs to generate the final recognition result. In the multiview fusion process, an attention mechanism is applied to regulate the fusion operation according to the helpfulness for the recognition of all views. In this way, the model can fuse information from multiple viewpoints to recognize actions and can learn to evaluate observation views to improve fusion performance. We also propose a multilayer feature attention method to improve the performance of the LSTM in our model. We utilize an attention mechanism to enhance the feature expression by finding and focusing on informative feature dimensions according to contextual action information. Moreover, we propose stacking multiple layers of attention operation in a multilayer LSTM network to further improve network performance. The final model is integrated into an end-to-end trainable network. Experiments conducted on two popular datasets, NTU RGB+D and SBU Kinect interaction, show that our model achieves state-of-the-art performance.**

***Index Terms*—Action recognition, multiple views, attention mechanism, skeleton, long short-term memory (LSTM).**

I. Introduction

ACTION recognition is an important and popular research topic in computer vision. The goal of action recognition is to interpret videos in a human-centered manner, greatly assisting the automatic analysis of media resources. In many realistic applications, such as intelligent surveillance, video understanding, human-machine interaction, and assistant driving, human action recognition plays an important role in improving the intelligence level of machines.

Skeleton data are commonly used for human action recognition. Skeleton data provide an accurate description of body pose that is suitable and effective for precise action recognition. The popularization of cost-effective motion capture cameras, such as Kinect, and the development of efficient skeleton reconstruction methods, such as [1], make it much easier to obtain accurate human skeleton data and greatly encourage the study of skeleton-based human action recognition. In this paper, we explore this domain.

Currently, deep-learning-based methods have achieved impressive performance and have evolved rapidly as the mainstream methods in action recognition. For skeleton-based action recognition, recurrent neural networks (RNNs), especially long short-term memory (LSTM) [2], are a popular fundamental framework because of their effectiveness for handling sequential problems. Representative related methods include dRNN [3], HBRNN [4], ST-LSTM [5] and STA-LSTM [6]. These methods construct models on the basis of RNN to fitthe action recognition problem, revise the RNN unit or design the model structure according to specific properties of action, such as skeleton structure or sequence information. Although relatively satisfactory performance has been achieved, some important factors of action recognition are often overlooked by the current methods. The most important among them is the observation view of actions.

The observation view has a significant influence on action recognition. On one hand, actions can be observed from many possible viewpoints, which increases the diversity of action expressions and greatly affects the generalization of the recognition algorithm. On the other hand, some actions are difficult to recognize from certain viewpoints but easy to recognize from others; therefore, in this situation, observing actions from suitable viewpoints greatly aids in recognition.

In this paper, we propose an attention-based multiview re-observation fusion model for skeletal action recognition that utilizes observation views to assist action recognition.

The proposed model re-observes actions from multiple viewpoints and fuses the multiview information to improve the recognition performance. Specifically, the proposed model first re-observes an input action from several possible viewpoints, processes these augmented observations with an LSTM network, and finally fuses the processed outputs from all the observations to make the final decision. Thus, the proposed model summarizes action information from multiple observation

Manuscript received March 25, 2018; revised July 2, 2018; accepted July 3, 2018. Date of publication July 26, 2018; date of current version January 24, 2019. This work was supported in part by the National Natural Science Foundation of China under Grants 61673269 and 61273285 and in part by the Cooperative Medianet Innovation Center. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Abdulmotaleb El Saddik. *(Corresponding author: Xu Zhao.)*

The authors are with the Department of Automation, Shanghai Jiao Tong University, Shanghai 200000, China (e-mail: fzx92@sjtu.edu.cn; zhaoxu@sjtu.edu.cn; wzmsltw@sjtu.edu.cn; suhaisheng@sjtu.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/TMM.2018.2859620

views in action recognition. Therefore, how to fuse multiview information efficiently is an important problem. In our model, an attention mechanism is introduced to guide the fusion process. The helpfulness of re-observation views for recognition is evaluated by the attention mechanism and is then quantified to guide fusion. The attention-guided fusion enables the model to find suitable observation views for specific actions and improves fusion performance. Essentially, the fusion of multiview information further propels the model to learn the invariant features in specific actions and therefore assists action recognition.

Furthermore, inspired by the success of the attention mechanism, a multilayer feature attention method is proposed to enhance feature expression and improve network performance. Substantial redundant or irrelevant information exists in action sequences and usually interferes with action recognition. In our model, we use attention to find and focus on informative joints or features to emphasize primary action information. In some other works, such as STA-LSTM [6], an attention mechanism is also applied to skeleton or sequence information. However, we propose to use the attention mechanism as a general feature enhancement method and stack multiple layers of attention operation in a multilayer LSTM network to further improve the performance.

The overall model, which consists of the components mentioned above, is integrated into an end-to-end trainable network. Our model is applied to two popular datasets, NTU RGB+D and SBU Kinect Interaction. A comparison with other current methods shows that our model achieves state-of-the-art performance, and further ablation experiments show the effectiveness of our proposed method.

In conclusion, there are two main contributions of our work:

1) We propose an attention-based multiview re-observation fusion model for skeletal action recognition that utilizes action information from multiple observation views to improve the recognition performance and that uses an attention mechanism to find suitable observation views for recognition.
2) We propose a multilayer feature attention method to enhance feature representation and to improve the performance of the LSTM network. We propose to use the attention mechanism as a general feature enhancement method that can be stacked in multilayer networks to further improve the performance.

The rest of this paper is organized as follows. Section II summarizes the related work on skeleton-based action recognition and attention mechanisms. Section III explains the proposed methods in detail. Section IV presents the related experimental results, and Section V concludes our work.

II. Related Work

A. Skeleton-Based Action Recognition

Human action recognition has long been an attractive topic in computer vision. Skeleton data, which represent human poses in an articulated system of joints and limbs, is a commonly used data type for action recognition. Compared to video data, skeleton data provide an explicit description of body poses and are therefore more suitable and efficient for accurate human action recognition. In recent years, with the development of efficient motion capture devices and pose estimation algorithms, it is becoming easier to obtain human skeleton information. Therefore, skeleton-based action recognition has recently attracted substantial attention.

For skeleton-based action recognition, traditional methods, such as Skeletal Quads [7], Lie Group [8], and works such as [9]–[13], follow the typical routine of many computer vision problems–design and extract features that represent the spatial information and temporal dynamics in an action sequence, then design and use a classification algorithm to classify actions. Handcrafted features require substantial experience and tests and are therefore difficult to implement.

In recent years, deep-learning-based methods have achieved greatly improved performance compared to that of traditional methods and have become the mainstream approach.

Due to its suitable nature to process sequence data, RNN is currently the most popular framework in skeleton-related action recognition. Some remarkable RNN-related methods for skeleton-based action recognition have been proposed in the past few years. Differential RNN [3] revises the original LSTM unit to emphasize salient sequence information. In dRNN, the derivative of internal LSTM states (DoS) is used to indicate the saliency of input information. The gates of the LSTM unit are revised to include the DoS term, so the DoS also influences information accumulation in the LSTM, by which the salient data input steps are emphasized. Hierarchical BRNN [4] utilizes the heuristic knowledge of the human body and designs a hierarchical LSTM network. In the network, the skeleton is split into five groups (arms, legs and trunk), which are then separately processed by RNN. Their outputs are fused from the local structure to global structure hierarchically according to the structure of the human body to accumulate action information. Traditionally, in an LSTM framework, in each time step, the skeleton data of the corresponding frame are fed into the network. Spatio-temporal LSTM [5] extends this traditional temporal domain framework to the spatio-temporal domain. In addition to the temporal action sequence, the joints in the skeleton in each time step form a spatial sequence. ST-LSTM simultaneously processes both the temporal and spatial sequence information. Furthermore, it introduces a trust gate to address possible noise and occlusion in the skeleton data. Other similar works include [14]–[17]. These methods are all based on the RNN framework and its extensions and are designed to better perform the skeleton-based action recognition task. Our work is also based on RNN, but we focus on the aspect of observation view, which is rarely considered in the above methods.

Another popular type of method is using CNN to process a transferred skeleton image. This type of method, such as [18]–[24], first designs an algorithm to encode skeleton data into an image and then uses CNN to process the encoded skeleton image to classify actions. Du *et al.* [25] propose an algorithm for skeleton-image encoding that arranges skeleton sequence data into a matrix in a certain order and then quantifies the matrix into an image. The quantified image includes the spatial skeleton representation and the temporal sequence dynamics and is

processed by CNN for classification. Hou *et al.* [26] design a more explicit method of skeleton encoding. The skeleton sequence is drawn onto canvas in order from three viewpoints, and different colors are used to highlight joint locations and sequence information. The core of this class of method is the design of the encoding algorithm, which is another type of feature design that requires considerable experience and experimentation. Our work uses the skeleton data directly, aiming for an automatic and intelligent method of action recognition.

Some other creative works have been reported. A^2 GNN [27] utilizes a graphic neural network to model the skeleton action sequence, in which the skeleton is treated as an undirected graph. A^2 GNN also adopts attention to detect salient action units. ST-NBMIM [28] proposes a spatio-temporal naive Bayes mutual information maximization algorithm to identify critical spatial and temporal information in the skeleton action sequence and forms discriminative action patterns for recognition. A multitask learning CNN is proposed in [29] to solve pose estimation and action recognition problems simultaneously. Predicted human poses are also used to help action recognition. In contrast to these methods, we focus on an RNN-based method and achieve competitive or better results.

B. Attention Mechanism

The attention mechanism is first proposed to imitate the vision mechanism of primates. In the vision process, primates tend to pay attention to a limited area, and the attention changes with time and task. Inspired by this biological discovery, an attention-based model that imitates this visual attention mechanism to analyze scene saliency is first proposed in [30].

Recently, the application of attention in some natural language processing and vision tasks has shown considerable progress. An attention mechanism is introduced into a traditional encoder-decoder translation framework [31]. In each decoding step, attention is used to direct the feature fusion of the encoder outputs by assigning relevance weights according to contextual information. An RNN model that combines a visual attention mechanism and reinforcement learning is proposed in [32]. This recurrent model of visual attention can iteratively select a sequence of interest patches from an image for specific tasks. The application of this model in image classification illustrates its effectiveness. The recurrent visual attention model in [33] is extended to an image captioning task, where it is included in a typical encoder-decoder captioning framework. The model iteratively selects areas of interest from the image for captioning in each step. These successful applications drive the exploration of attention in many other areas, such as [34]–[38].

An attention mechanism is also applied in many works on skeleton-based action recognition. Spatio-temporal attention LSTM [6] improves basic LSTM networks by adding a spatial and temporal attention mechanism. The spatial attention focuses on the importance of different body joints, while the temporal attention focuses on the importance of different action steps. Global context-aware attention LSTM [39] extends the spatio-temporal LSTM [5] with a recurrent global attention mechanism that iteratively improves attention performance based on global contextual information. These applications of attention mechanisms in skeletal action recognition achieve great results. In our work, attention is used in two new ways. First, we use attention to guide the fusion of multiple views to improve fusion performance. Second, we use attention to enhance feature expression and combine the multilayer feature attention operation with a multilayer LSTM network, which further improves the network performance.

III. Method

Our model is built on the basis of LSTM. Fig. 1 shows the framework of our proposed model. Two main methods are included in the model. First, the attention-based multiview re-observation fusion method forms the overall framework of the model. The input skeleton is first observed from several possible views, generating multiple re-observation results. These observations are processed separately by the LSTM network, and their processing outputs are fused with attention to make the final recognition decision. Second, a multilayer feature attention method is applied in the main LSTM network. The attention mechanism enhances feature expression by focusing on informative feature dimensions. Stacking multiple layers of feature attention operation in the multilayer LSTM network further improves the network performance.

A. Problem Formulation

For skeleton-based action recognition, the input is a time sequence of skeleton data $[\boldsymbol{s_1}, \boldsymbol{s_2}, ..., \boldsymbol{s_T}]$, where $\boldsymbol{s_t}$ is the locations of the joints in the skeleton at time t, arranged in a specific order.

Our goal is to learn a model F that maps the skeleton sequence $[\boldsymbol{s_1}, \boldsymbol{s_2}, ..., \boldsymbol{s_T}]$ to certain action class y.

$$y = F([\boldsymbol{s_1}, \boldsymbol{s_2}, ..., \boldsymbol{s_T}]) \tag{1}$$

The model needs to handle the spatial information and temporal dynamics of skeleton sequences to solve the mapping problem.

B. LSTM Review

The recurrent architecture of RNN is designed to process sequential data. The calculation of a basic RNN unit is shown below. The current output $\boldsymbol{h_t}$ is determined by both the current input $\boldsymbol{x_t}$ and the previous output $\boldsymbol{h_{t-1}}$, calculated with model weights $\boldsymbol{W_x}$, $\boldsymbol{W_h}$ and $\boldsymbol{b}$. This process enables the accumulation of history information.

$$\boldsymbol{h_t} = \phi(\boldsymbol{W_x} \cdot \boldsymbol{x_t} + \boldsymbol{W_h} \cdot \boldsymbol{h_{t-1}} + \boldsymbol{b}) \tag{2}$$

Theoretically, RNN can handle sequential dependencies of any length. However, as described in [2], many realistic problems, such as gradient vanishing and explosion in training, prevent the ideal realization. A long short-term memory network (LSTM) is a modified version of RNN designed to handle long-term dependencies. In contrast to the simple perceptron structure in an RNN unit, an LSTM cell consists of three gates. These gates control information flow in the cell and build linear

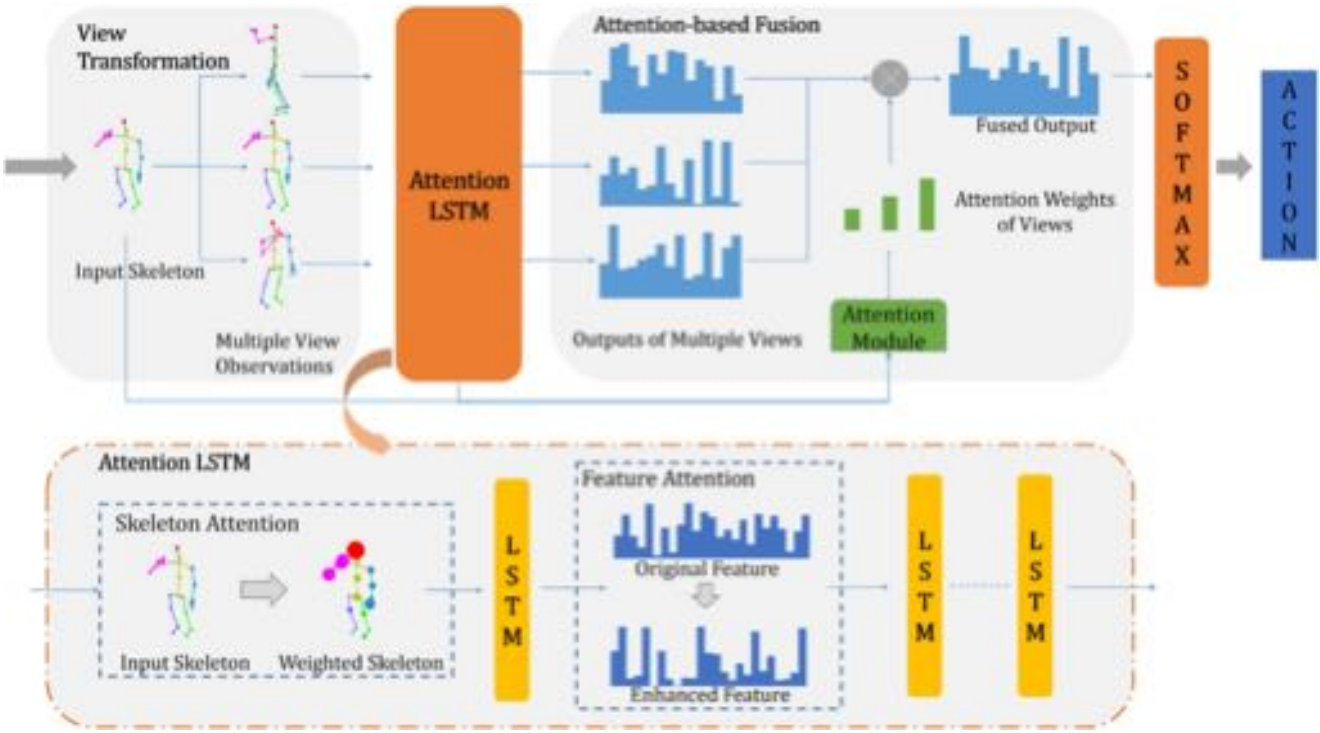

Fig. 1. Framework of our proposed model. There are two essential components in our work. 1) Attention-based multiview re-observation fusion network, which forms the overall framework of our model. The input skeleton is first transformed into several possible view observations. These skeleton observations are processed by the proposed attention LSTM separately, and their outputs are fused for the final result. In the fusion process, an attention module generates weights for different views to highlight the helpful views. 2) Attention LSTM, a multilayer LSTM network integrated with attention. In attention LSTM, the input of every LSTM layer is enhanced with feature attention.

connections that can mitigate the problem of gradient vanishing and explosion.

The computation in an LSTM cell is as below:

$$ForgetGate: \boldsymbol{f_t} = \sigma(\boldsymbol{W_f} \cdot [\boldsymbol{h_{t-1}}, \boldsymbol{x_t}] + \boldsymbol{b_f}) \quad (3)$$

$$InputGate: \boldsymbol{i_t} = \sigma(\boldsymbol{W_i} \cdot [\boldsymbol{h_{t-1}}, \boldsymbol{x_t}] + \boldsymbol{b_i}) \quad (4)$$

$$OutputGate: \boldsymbol{o_t} = \sigma(\boldsymbol{W_o} \cdot [\boldsymbol{h_{t-1}}, \boldsymbol{x_t}] + \boldsymbol{b_o}) \quad (5)$$

$$StateCandidate: \tilde{\boldsymbol{C}}_t = tanh(\boldsymbol{W_C} \cdot [\boldsymbol{h_{t-1}}, \boldsymbol{x_t}] + \boldsymbol{b_C}) \quad (6)$$

$$CellState: \boldsymbol{C_t} = \boldsymbol{f_t} * \boldsymbol{C_{t-1}} + \boldsymbol{i_t} * \tilde{\boldsymbol{C}}_t \quad (7)$$

$$Output: \boldsymbol{h_t} = \boldsymbol{o_t} * tanh(\boldsymbol{C_t}) \quad (8)$$

where $\boldsymbol{W}_*$ and $\boldsymbol{b}_*$ are learned model parameters. The cell state $\boldsymbol{C}$ refers to the information of the LSTM unit. The forget, input and output gates, which control the information flow of the LSTM, are computed based on the current input $\boldsymbol{x_i}$ and the previous cell output $\boldsymbol{h_{t-1}}$. The forget gate $\boldsymbol{f_t}$ controls how many previous cell states $\boldsymbol{C_{t-1}}$ are retained. The input gate $\boldsymbol{i_t}$ controls how many input $\boldsymbol{x_i}$ should be received. The output gate $\boldsymbol{o_t}$ controls the transformation from cell state to output.

C. Attention Mechanism

In machine learning applications, the basic idea of an attention mechanism is to identify and pay more attention to the important or relevant features according to the task requirements and contextual information. Usually, an attention mechanism can be implemented in one of two ways.

One is hard attention, which processes only the selected important features and ignores all other unimportant features. Hard attention can be considered to be a $1/0$ mask for feature elements. Networks with hard attention are non-differentiable; therefore, hard attention is usually combined with reinforcement learning for training and prediction.

The other is soft attention, which processes all the elements but assigns a different weight to each feature according to its importance or relevance. Networks with soft attention are differentiable and can be trained end-to-end with common back-propagation. Therefore, soft attention is more convenient for implementation.

In deep learning, attention is usually implemented with the framework of RNN. RNN naturally accumulates contextual information of sequential tasks, which is used in attention generation. Usually, for soft attention, the attention weights of the current RNN step $\boldsymbol{A_t}$ are generated with the RNN state of the previous step $\boldsymbol{h_{t-1}}$, which represents contextual information, and the current input $\boldsymbol{x_t}$:

$$\boldsymbol{A_t} = F_{attention}(\boldsymbol{h_{t-1}}, \boldsymbol{x_t}). \quad (9)$$

where $\boldsymbol{A_t}$ is the attention weights generated by attention model $F_{attention}$, which are then used to guide the usage of features.

D. Multiview Re-Observation Fusion With Attention

In action recognition, the observation view has a substantial influence and reflects two main aspects. First, action data

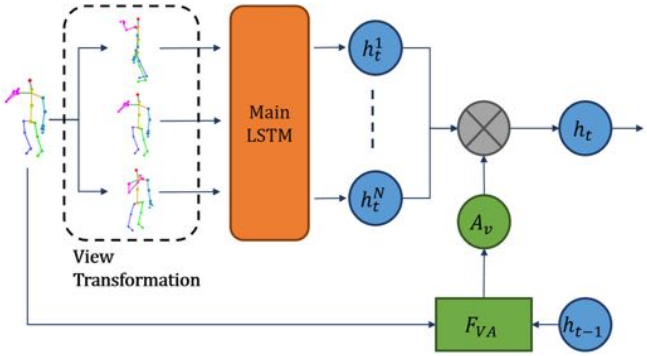

Fig. 2. Multiview fusion with attention.

can be collected from different views. A change in observation view increases data diversity, which should be considered in the model implementation and training for better performance. Second, some actions are easy to recognize when observed from certain views but hard to recognize from others. In this case, finding suitable views of the input data will help to improve the recognition performance. Based on this intuition, considering possible view observations in action recognition is no doubt beneficial. Therefore, we propose an attention-based multiview re-observation fusion method. The view changing requires the model to learn to cover a more complex and extensive feature space, which makes the processing more difficult. In our method, the model itself makes view changes to the input data to help adapt to view changing and obtain more sufficient information of action representations.

The structure of the proposed multiview re-observation fusion network is illustrated in Fig. 2. The input skeleton is first processed by several transformations, resulting in observations from different views. These skeleton observations are then processed separately by an LSTM network, after which their outputs are fused together for the final recognition result. The attention mechanism is applied during fusion to evaluate the helpfulness for the recognition of observation views and to assign fusion weights.

In practice, we obtain new view observations by rotating the input skeleton $\boldsymbol{s}$. N rotations are performed on skeleton data to generate N observations that cover possible skeleton views discretely.

In this process, skeleton data are first re-centered at the $Hip-Center$ joint to reduce skeleton shifting caused by rotation. Then, the skeleton is rotated in a 3D coordinate system. In the 3D coordinate system, the rotation of a joint in skeleton $\boldsymbol{j_i} = [j_i^x, j_i^y, j_i^z]^T$ can be represented as

$$\tilde{\boldsymbol{j_i}} = \boldsymbol{R_x R_y R_z j_i}, \tag{10}$$

where $\boldsymbol{R_x}$, $\boldsymbol{R_y}$, and $\boldsymbol{R_z}$ denote rotation around the X, Y, and Z axes, respectively. To be more specific, rotation around the Z axis can be represented as

$$\boldsymbol{R_z(\theta)} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{11}$$

where θ denotes the rotation angle around the Z axis.

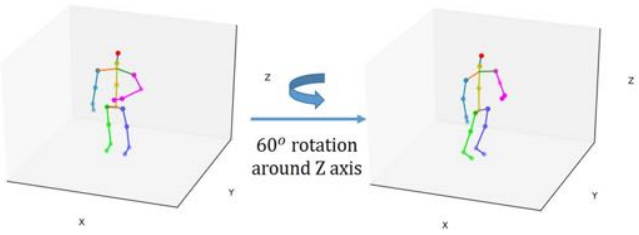

Fig. 3. Skeleton rotation around the Z axis.

In reality, most view changes are caused by observations from different horizontal angles. These view changes can be considered to be rotations of a skeleton around the Z axis in the 3D coordinate system. Therefore, to reduce the calculation complexity, we consider rotations around on the Z axis in our method. Fig. 3 illustrates the added view change in our method.

As discussed previously, of all the view observations, some are helpful for recognition but some might even be obstacles. Therefore, attention is also introduced to adjust the view fusion. Attention adjusts the weights according to contextual information. Helpful views receive higher weights while non-helpful ones receive lower weights in fusion process. In this way, the multiview fusion network learns to select appropriate views for recognition.

The attention weights $\boldsymbol{A_v}$ for multiview fusion are calculated by:

$$\boldsymbol{A_v^{'}} = \boldsymbol{U}(tanh(\boldsymbol{W_{vx}} \cdot \boldsymbol{s_t} + \boldsymbol{W_{vh}} \cdot \boldsymbol{h_{t-1}} + \boldsymbol{b_{v1}})) + \boldsymbol{b_{v2}}, \tag{12}$$

$$\boldsymbol{A_v} = Softmax(\boldsymbol{A_v^{'}}), \tag{13}$$

where $\boldsymbol{U}, \boldsymbol{W_{v*}}$, and $\boldsymbol{b_{v*}}$ are learned parameters, $\boldsymbol{s_t}$ is the skeleton input of the current step, and $\boldsymbol{h_{t-1}}$ is the model output of the last time step.

The outputs of different view transformations are fused by the weighted sum with attention weights:

$$\boldsymbol{h_t} = \sum_{i=1}^{N} \boldsymbol{A_v^i} \cdot \boldsymbol{h_t^i} \tag{14}$$

Here, $[\boldsymbol{h_t^1}, ..., \boldsymbol{h_t^N}]$ are the outputs of the attention LSTM for multiple-view skeleton observations. The fused output $\boldsymbol{h_t}$ is the weighted sum of $[\boldsymbol{h_t^i}]$, which is then processed by softmax for the final recognition.

E. Feature Attention

1) Attention of Skeleton Joints: For human actions, only a few body parts are often crucial, while most body parts are irrelevant. For example, regardless of the posture of a subject, an action is classified as eating whenever the subject is putting something into his mouth. In this case, only the interaction between the hand and mouth determines the eating action. Therefore, paying more attention to the hands and mouth will help to improve the recognition of an eating action. Based on this idea, we introduce attention into our model to learn to find crucial skeleton joints according to action information and to pay more attention to them for better recognition.

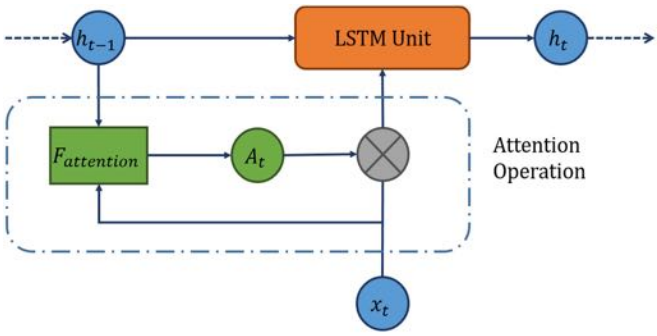

Fig. 4. Feature attention for one LSTM layer.

In our method, soft attention is used because all body parts may be involved in actions and also because of its convenience for computation. Fig. 4 demonstrates the feature attention technique adopted in our method. The recognition of skeleton data is implemented with an LSTM network. The importance of skeleton joints in the current context is predicted with the attention operation according to the contextual information. The skeleton input is then revised in reference to the predicted importance weights, and the revised input is input into the LSTM network for recognition.

The importance weights of the skeleton joints are calculated with soft attention by

$$\boldsymbol{A_j} = \boldsymbol{U_j}(tanh(\boldsymbol{W_{jh}} \cdot \boldsymbol{h_{t-1}} + \boldsymbol{W_{js}} \cdot \boldsymbol{s_t} + \boldsymbol{b_{j1}})) + \boldsymbol{b_{j2}} \tag{15}$$

Here, $\boldsymbol{U_j}, \boldsymbol{W_{j*}}$, and $\boldsymbol{b_{j*}}$ are learned parameters. The LSTM state of the previous step $\boldsymbol{h_{t-1}}$ denotes the contextual action information accumulated in the LSTM. Based on the current input $\boldsymbol{s_t}$ and previous LSTM state $\boldsymbol{h_{t-1}}$, the importance of the skeleton joints for the current context is predicted. $\boldsymbol{A_j}$ is the attention weight vector corresponding to J body joints. The attention weights are then used to revise the skeleton input, through which the joints are emphasized according to their importance. For skeleton input $\boldsymbol{s_t} = [\boldsymbol{j_{t,1}}, \boldsymbol{j_{t,2}}, ..., \boldsymbol{j_{t,J}}]$ (J joints), the revised input is

$$\tilde{\boldsymbol{s_t}} = [\boldsymbol{j_{t,1}}, \boldsymbol{j_{t,2}}, ..., \boldsymbol{j_{t,J}}] \circ \boldsymbol{A_j} \tag{16}$$

The revised inputs $\tilde{x_t}$ embody the importances of different joints. By this attention enhancement, the motion of more important joints is enlarged, while that of the less irrelevant joints is reduced. This process emphasizes the relevant sequential or interactive information of those important joints and therefore assists action recognition.

2) Multilayer Feature Attention: Inspired by the success of skeleton attention, we further consider that attention can be regarded as a universal feature enhancement method and that stacking multilayer feature attention will result in further improvement. Similar to that of joints, for general features, the importance varies for different tasks or process phrases. Paying more attention to features that are more important in specific conditions is beneficial for improving feature expression.

We implement this idea in the multilayer LSTM network as the attention LSTM. In attention LSTM, which contain 3 LSTM layers, the output of the previous LSTM layer is taken as the input feature for the next LSTM layer. Attention is applied to enhance features transfered between LSTM layers. The attention is implemented similar to that of joints. For the l-th LSTM layer ($l > 1$), the attention weights for the input feature are calculated by:

$$\boldsymbol{A_l^{'}} = tanh(\boldsymbol{W_{lh}} \cdot \boldsymbol{h_{l,t-1}} + \boldsymbol{W_{lx}} \cdot \boldsymbol{h_{l-1,t}} + \boldsymbol{b_{l1}}) \tag{17}$$

$$\boldsymbol{A_l} = \boldsymbol{U_l} \cdot \boldsymbol{A_l^{'}} + \boldsymbol{b_{l2}} \tag{18}$$

where $\boldsymbol{U_l}, \boldsymbol{W_{l*}}$, and $\boldsymbol{b_{l*}}$ are learned parameters. $\boldsymbol{h_{l-1,t}}$ is the output state of the previous LSTM layer, which is the input feature of the l-th LSTM layer. $\boldsymbol{h_{l,t-1}}$ is the previous state of the l-th LSTM layer. In the same way as for joints, the attention weights of features are computed based on input feature $\boldsymbol{h_{l-1,t}}$ and contextual information $\boldsymbol{h_{l,t-1}}$. Additionally, the input feature $\boldsymbol{h_{l-1,t}}$ of layer l at time t is revised by:

$$\tilde{\boldsymbol{h}}_{\boldsymbol{l-1,t}} = \boldsymbol{h_{l-1,t}} \circ \boldsymbol{A_l} \tag{19}$$

Fig. 5 illustrates the application of multilayer feature attention in the LSTM network.

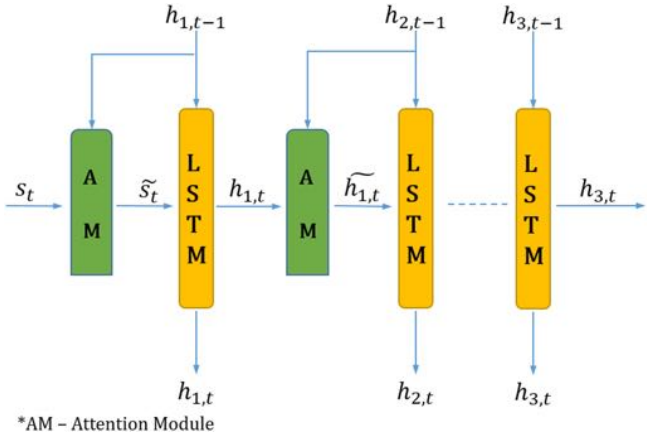

Fig. 5. Multi-layer feature attention in LSTM network. The input feature of each LSTM layer is enhanced with attention mechanism.

F. Integrated Training

The whole model, which includes multiview fusion and attention LSTM, is shown in Fig. 1. The model is end-to-end trainable with standard backpropagation, and it is trained with typical regularized log-likelihood loss,

$$L = -\sum_{i=1}^{C} y_i \log \tilde{y_i} + \lambda \|\boldsymbol{W}\|_2. \tag{20}$$

Here, y_i is the ground-truth label of the training data. $\tilde{y_i}$ is the label predicted by the model. C is the total number of classes. The first term is the log-likelihood function of the model, which drives the model to fit the probability distribution of the training data. $\|\boldsymbol{W}\|_2$ represents the $L2$ penalty of model parameters $\boldsymbol{W}$. λ is the corresponding regularization coefficient. The second term is the parameter regularization of the model, which reduces over-fitting.

IV. Experiment

A. Datasets

Our methods are applied to two popular datasets: NTU RGB+D [40] and SBU Kinect Interaction [41].

1) NTU RGB+D: The NTU RGB+D dataset is currently the largest and most challenging dataset for 3D action recognition. The dataset includes 56880 video samples of 60 action classes, which are performed by 40 subjects and captured from 3 different views. The actions in this dataset include single-person actions and two-person interactions. In this dataset, two standard evaluation methods are proposed: Cross-Subject (CS) and Cross-View (CV). In CS, 40 subjects are split equally into the training and test set. In CV, cameras 2 and 3 are used for training, and camera 1 is used for testing.

2) SBU Kinect Interaction: SBU Kinect Interaction is a dataset of two-person interactions that contains 230 samples from 8 action classes in total, which are performed by 7 subjects and captured by one fixed camera. The commonly used evaluation protocol is 5-fold cross validation proposed along with the dataset.

B. Implementation Details

Our method is implemented on the TensorFlow [42] platform. The main network contains three layers of standard LSTM. Each LSTM layer has 100 neurons for the NTU RGB+D dataset and 50 for the SBU dataset. The regularization term λ is 0.0005 0.001 for different network configurations. Dropout [43] of 0.5 is applied to avoid over-fitting. The whole network is trained end-to-end with the $Adam$ algorithm, and the initial learning rate is 0.001.

The skeleton data are sampled to the same length. For the NTU RGB+D dataset, $T = 100$ frames are sampled for each data sequence. For the SBU dataset, a length of $T = 20$ is adopted. Data sequences shorter than T are padded with zeros.

C. Comparison With State-of-the-art Methods

Table I shows the comparison with other state-of-the-art methods on the NTU RGB+D dataset. For a fair comparison, the results reported in related papers are adopted in the comparison. **LSTM + FA** refers to the 3-layer LSTM combined with the multilayer feature attention method. **LSTM + VF** refers to the 3-layer LSTM combined with the multiview fusion method. **LSTM + FA + VF** refers to our final model, the 3-layer LSTM combined with multiview fusion and multilayer feature attention. For cross-view evaluation, our method outperforms the other methods by 3% or more, while for cross-subject evaluation, our method achieves performance similar to the currently best method, GCA-LSTM. Specifically, in the cross-subject evaluation, the multilayer feature attention method results in more improvement than does the multiview fusion method because it helps to obtain the key action information from different subjects. For the cross-view evaluation, multiview fusion method results in greater improvement than does multilayer attention because the fusion of multiview information enables the model to better learn the invariant features across different views.

TABLE I
Comparison for the NTU RGB+D Dataset

Method	CS(%)	CV(%)
Skeletal Quads [7]	38.6	41.4
Lie Group [8]	50.1	52.8
HBRNN [4]	59.1	64.0
Dynamic Skeletons [44]	60.2	65.2
Deep LSTM [40]	60.7	67.3
Part-aware LSTM [40]	62.9	70.3
ST-LSTM [5]	69.2	77.7
STA-LSTM [6]	73.4	81.2
A^{2}GNN [27]	72.74	82.80
Res-TCN [20]	74.3	83.1
GCA-LSTM [39]	**74.4**	82.8
LSTM + FA	71.5	81.0
LSTM + VF	70.2	84.1
LSTM + FA + VF	73.8	**85.9**

* VF (multiview fusion).
FA (multilayer feature attention).

TABLE II
Comparison for the SBU Dataset

Method	Accuracy(%)
Joint Features + SVM [41]	80.3
HBRNN [4]	80.4
CHARM [45]	83.9
Ji *et al.* [46]	86.9
Co-occurrence LSTM [47]	90.4
STA-LSTM [6]	91.5
ST-LSTM [5]	93.9
GCA-LSTM [39]	94.1
LSTM + FA	94.2
LSTM + VF	93.2
LSTM + FA + VF	**95.0**

Table II shows the comparison with other state-of-the-art methods for the SBU Kinect Interaction dataset. The results reported in the related papers are adopted in the comparison. Our method achieves the best results. The data in the SBU dataset are collected from a fixed camera; therefore, multilayer feature attention works better than does multiview fusion, just as it does for NTU RGB+D. Moreover, the combination of these two methods achieves further improved results.

D. MultiView Re-Observation Fusion

Our proposed multiview re-observation fusion method is applied to the two datasets. As illustrated in Fig. 2, a basic 3-layer LSTM network is adopted as the main LSTM network. Different view fusion methods and configurations are tested.

Tables III and IV show the experimental results for the fusion methods. The LSTM outputs of different view observations are fused by different methods to generate the final results. Three fusion methods, namely, averaging, attention (tanh) and attention (softmax), are tested. In the experiment, for the NTU RGB+D dataset, the rotation angles are $[0^\circ, \pm 60^\circ]$, and for SBU, they are $[0^\circ, \pm 90^\circ]$.

TABLE III
VIEW FUSION METHOD EXPERIMENTS ON THE NTU RGB+D DATASET

Method	CS(%)	CV(%)
basic LSTM	66.8	77.5
LSTM + VF(ave)	67.4	82.6
LSTM + VF(tanh)	68.3	82.0
LSTM + VF(softmax)	**70.2**	**84.1**

* **ave**: view weights are $[1/N]$.
tanh: attention weights are calculated by $tanh(Wx + Wh + b)$.
softmax: attention weights are calculated by $softmax(U \cdot tanh(Wx + Wh + b))$.

TABLE IV
VIEW FUSION METHOD EXPERIMENT ON THE SBU DATASET

Method	Accuracy(%)
basic LSTM	87.5
LSTM + VF (ave)	91.4
LSTM + VF (tanh)	91.8
LSTM + VF (softmax)	**93.2**

TABLE V
VIEW SETTING EXPERIMENT ON THE NTU RGB+D DATASET

Method	CS(%)	CV(%)
basic LSTM	66.8	77.5
Rotate(0, ± 30)	69.0	82.6
Rotate(0, ± 45)	70.1	83.6
Rotate(0, ± 60)	70.2	84.1
Rotate(0, ± 90)	**70.9**	82.0
Rotate(0, ± 30, ± 60)	69.9	83.4
Rotate(0, ± 60, ± 90)	70.4	84.1
Rotate(0, ± 30, ± 60, ± 90)	70.2	**84.4**

We can see that even fusion by simply averaging the outputs of different view observations improves the accuracy, and the application of attention in view fusion results in additional substantial improvements. Moreover, the comparison shows that $softmax$ is better than $tanh$ for generating attention weights in this situation. We believe that this is because every view observation is a correct representation of the skeleton data, and $softmax$ is more suitable for fusing these observations by applying normalized weights to all the view observations.

The experimental result shows that our proposed multiview re-observation fusion method is effective in 3D action recognition. It is worth mentioning that the data in the NTU RGB+D dataset are collected from three different views while the SBU dataset has only one fixed view. In experiment, multiview fusion achieves significant improvements on both datasets, which indicates that considering multiple views of action sequences is a general method to improve 3D action recognition.

We also tested different view settings for multiview re-observation. Table V and Table VI show the experimental results. The results show that fusing any views improves the accuracy, but the appropriate views are different for different datasets. For NTU RGB+D, the data are collected from views of $[0°, \pm 45°]$. But in our experiment, view fusion of $[0°, \pm 60°]$ gives the best result. For the SBU dataset, view fusion of $[0°, \pm 90°]$ gives the best result. Furthermore, the results show that increasing the number of fused views results in a minimal increase, or even a decrease, in accuracy. Therefore, we consider fusion of only three views. According to the results, to obtain the best performance, view settings of $[0°, \pm 60°]$ for NTU RGB+D and $[0°, \pm 90°]$ for SBU are used in the later experiments.

TABLE VI
VIEW SETTING EXPERIMENT ON THE SBU DATASET

Method	Accuracy(%)
basic LSTM	87.5
Rotate(0, ± 30)	89.5
Rotate(0, ± 45)	90.5
Rotate(0, ± 60)	91.5
Rotate(0, ± 90)	93.2
Rotate(0, ± 30, ± 60)	89.9
Rotate(0, ± 60, ± 90)	93.4
Rotate(0, ± 30, ± 60, ± 90)	**93.8**

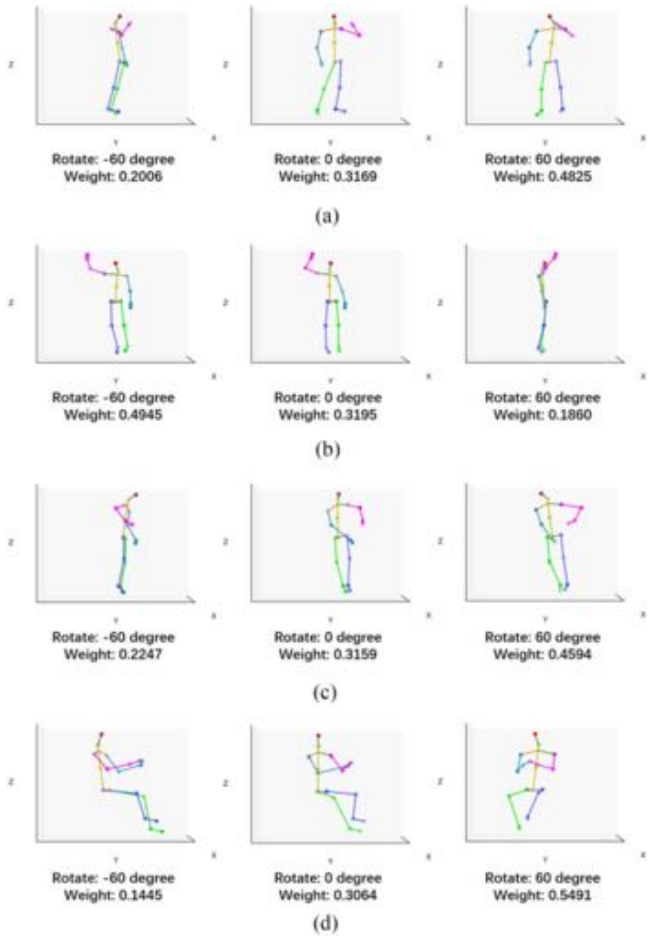

Fig. 6. Multiple view observations and their attention weights from a learned model of some examples from the NTU RGB+D dataset. (a) Drink water. (b) Take off a hat. (c) Put something inside pocket/take out something from pocket. (d) Taking a selfie.

Fig. 6 shows the view observations and their attention weights from a learned model of some examples from the NTU RGB+D dataset. From the experiment, we find that the learned model tends to observe subjects from the front view, where the skeleton joints are dispersed by the maximal level. As in Fig. 6, the front

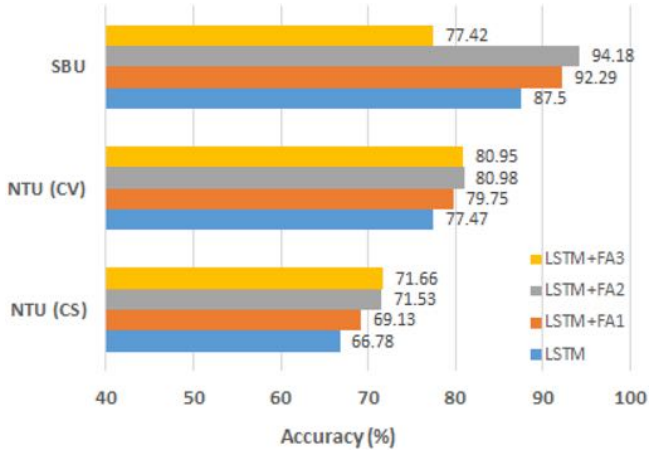

Fig. 7. Feature attention experiment on NTU RGB+D. Each cluster in the figure shows the comparison results of one dataset or one evaluation protocol. For bars in each cluster, from bottom to top, the number of attention layers increases from 0 to 3.

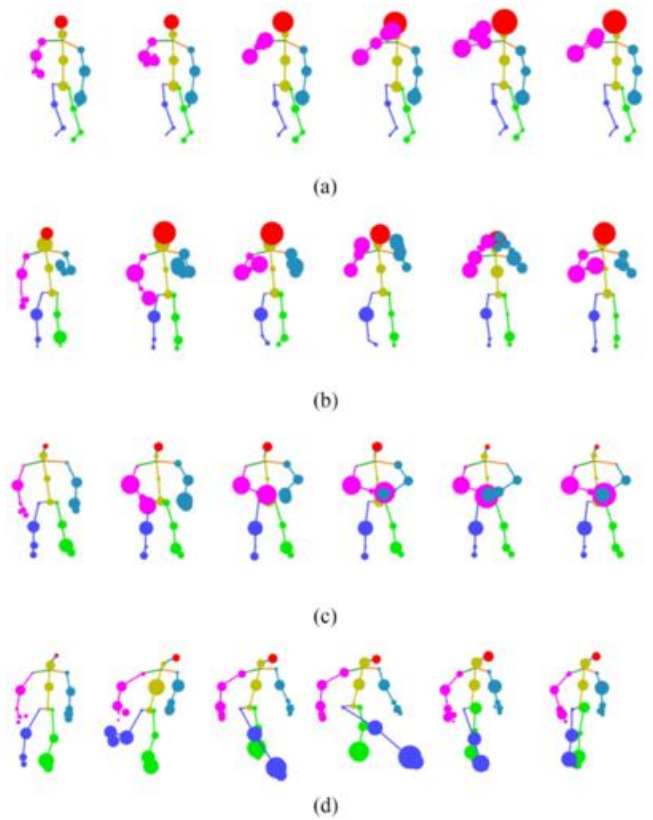

Fig. 8. Visualization of learned joint weights from attention on the NTU RGB+D dataset. The size of the joint indicates its attention weight. Each sequence visualizes an example of an action from NTU RGB+D. (a) Drink water. (b) Putting on glasses. (c) Clapping. (d) Kicking something.

views usually receive higher attention weights while the side views receive lower attention weights.

E. Multilayer Feature Attention

We evaluate the effectiveness of our proposed multilayer feature attention method on the NTU RGB+D and SBU datasets.

TABLE VII
INTEGRATION EXPERIMENT ON THE NTU RGB+D DATASET

Method	CS(%)	CV(%)
basic LSTM	66.8	77.5
LSTM + FA	71.5	81.0
LSTM + VF	70.2	84.1
LSTM + VF + FA1	72.6	85.3
LSTM + VF + FA2	**73.8**	**85.9**
LSTM + VF + FA3	73.3	85.1

* VF (skeleton multiview fusion).
FA (feature attention, n means the layer of feature attention).

In this experiment, a basic 3-layer LSTM network is used for recognition. Different layers of feature attention operation are applied in the network, and their performances are compared.

Fig. 7 shows the experimental results of the feature attention experiments on two datasets. FA-n in the figure means that the feature inputs of the first n LSTM layers in the network are processed with feature attention.

The results show that the application of feature attention significantly improves the recognition performance. Additionally, stacking two layers of feature attention results in further improvement. However, stacking three layers of feature attention does not result in evident improvement but instead can decrease the accuracy. This phenomenon is clearer on the relatively small SBU dataset. We believe that this is because multiple layers of feature attention result in additional model complexity and cause over-fitting. In conclusion, the experimental results indicate that our proposed multilayer feature attention is effective in the LSTM network. However, additional effort is required in determining the settings to avoid over-fitting when too many layers of feature attention are applied.

Fig. 8 shows the visualization of attention weights for body joints in some actions on the NTU RGB+D dataset. The learned attention weights effectively emphasize the corresponding action information. At the beginning of an action sequence, the weight of every joint is relatively small and equal. As the action proceeds, the weights vary according to the contextual action information and emphasize those informative body joints. In the action of drinking water, the interacting body parts, i.e., the head and right arm, received increasingly larger weights as the sequence proceeds. Similarly, the head and two arms in the putting on glasses action, the two hands in the clapping action, and the right leg in the kicking something action also receive higher attention weights. This result indicates that the model has learned to focus on informative body joints on the basis of contextual action information.

Another finding in our experiment is that the learned attention is greatly affected by the dataset information. For example, most of the subjects in the NTU RGB+D dataset are right-handed. Therefore, the learned model tends to assign higher weight to the right arm. In Fig. 8, we can see that even in two-handed actions, like putting on glasses and clapping, the right arm receives higher attention than the left arm. Additionally, the upper body usually has higher attention than the lower body because the dataset contains more actions involving the upper

TABLE VIII
INTEGRATION EXPERIMENT ON THE SBU DATASET

Method	Accuracy(%)
basic LSTM	87.5
LSTM + FA	94.2
LSTM + VF	93.2
LSTM + VF + FA1	94.8
LSTM + VF + FA2	**95.0**
LSTM + VF + FA3	94.1

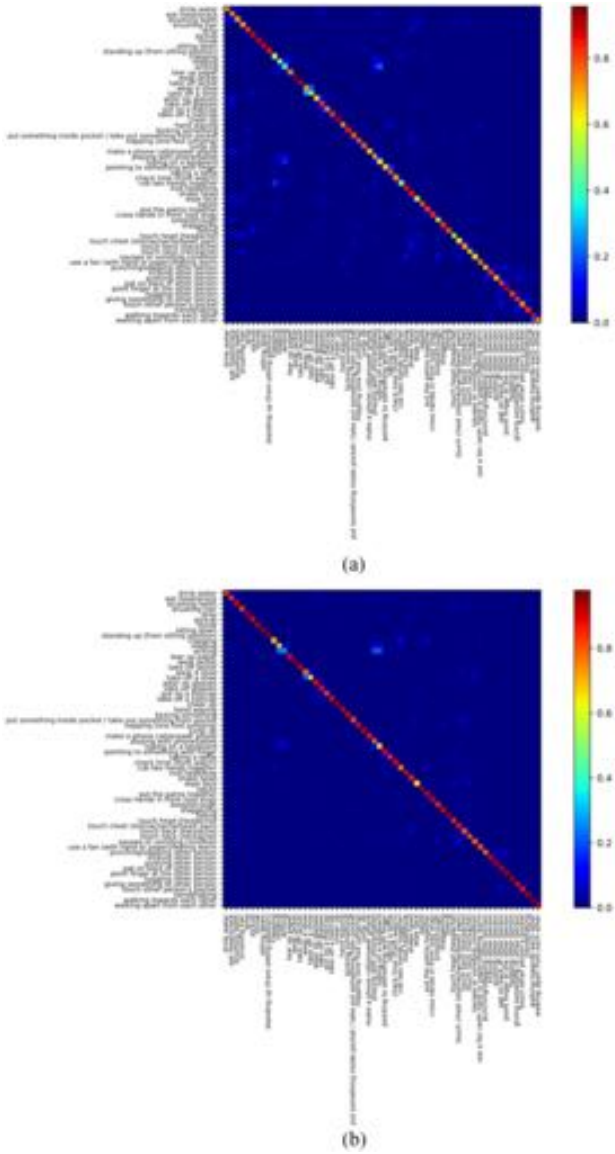

Fig. 9. Confusion matrices of the NTU RGB+D dataset. We can see that the most significant classification errors occur among classes that are physically similar. (a) Cross-subject evaluation. (b) Cross-view evaluation.

body. We believe this means the model is able to learn the dominant features in data, which is helpful for recognition.

F. Integration Experiment

Feature attention (FA) and multiview re-observation fusion (VF) by attention are combined in our final model, and the integration performance is evaluated on the NTU RGB+D and SBU datasets.

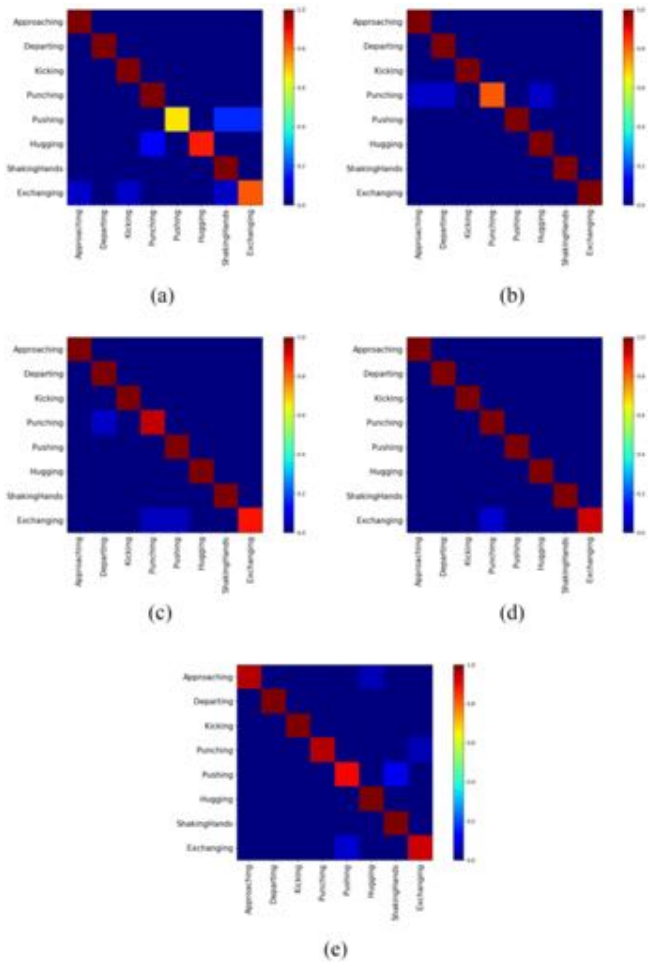

Fig. 10. Confusion matrices of the SBU Kinect Interaction dataset for 5 test sets. (a) Fold 1. (b) Fold 2. (c) Fold 3. (d) Fold 4. (e) Fold 5.

Table VII and Table VIII show the related experimental results for the two datasets. In this experiment, combining multiview fusion with one or two layers of feature attention in LSTM improves the accuracy by $1\% \sim 3\%$. However, when combining with three layers of feature attention, the accuracy decreases. We can see that the combination of feature attention and view fusion can improve recognition performance. However, as in the feature attention experiment, too many layers of feature attention can cause over-fitting, which is harmful to the performance. Therefore, according to the experimental results, our final model adopts two layers of feature attention in LSTM, which is integrated into the multiview fusion network.

G. Result Analysis

Fig. 9 and Fig. 10 show the confusion matrices for the NTU RGB+D and SBU Kinect Interaction datasets, respectively. As can be seen, the most significant classification errors occur among classes that are physically similar. These confusing classes include reading, writing, playing with phone and typing on a keyboard; clapping and rubbing two hands together; and putting on a shoe and taking off a shoe. The main reason for these

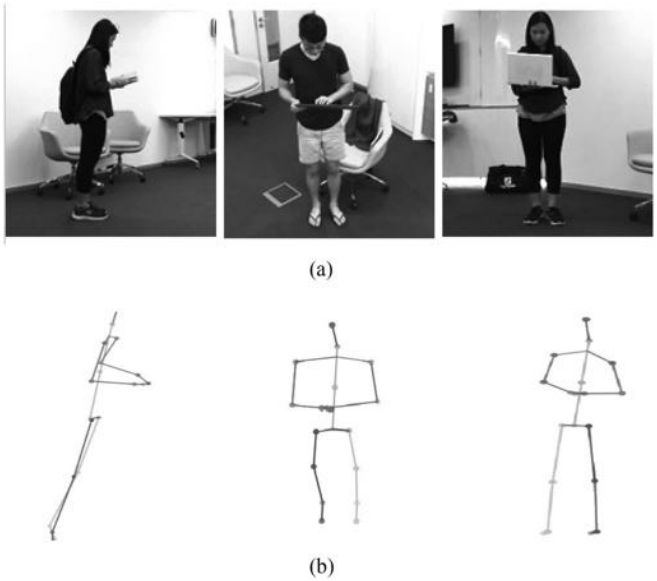

Fig. 11. Example pictures of confusing cases for the model. The physical differences between some action classes are very tiny, and it is difficult to distinguish them without the information of the interacting objects. (a) RGB Image (Reading/Playing with tablet/Typing on a keyboard). (b) Skeleton (Reading/Playing with tablet/Typing on a keyboard).

classification errors is that these actions differ by small body motions, which are difficult to capture and distinguish. Another reason is that many actions include interactions with objects, such as reading, writing, playing with phone and typing on a keyboard. In this situation, skeleton-based action recognition lacks this object information and thus performs badly. Fig. 11 shows some example pictures of these cases. This also reveals a major problem of purely skeleton-based action recognition, i.e., that it lacks object and appearance information. Therefore, it is more suitable for human-only actions.

V. Conclusion

In this paper, we propose an attention-based multiview re-observation fusion model for skeletal action recognition. The proposed model utilizes multiple-view information to improve the recognition performance by re-observing input data and fusing observed data from multiple views in recognition. The attention mechanism applied in the fusion process enables the model to evaluate the helpfulness of views for recognition and to regulate the fusion operation accordingly. In the proposed model, a multilayer feature attention method is also proposed to enhance feature expression and to improve the LSTM network performance. The proposed methods are integrated into an end-to-end-trainable network. Experiments on two popular datasets show that the proposed model achieves state-of-the-art performance, and further ablation experiments show the effectiveness of the proposed methods.

References

[1] J. Shotton *et al.*, "Real-time human pose recognition in parts from single depth images," *Commun. ACM*, vol. 56, no. 1, pp. 116–124, 2013.

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

[3] V. Veeriah, N. Zhuang, and G.-J. Qi, "Differential recurrent neural networks for action recognition," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 4041–4049.

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

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

[6] 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 Conf. Artif. Intell.*, 2017, pp. 4263–4270.

[7] G. Evangelidis, G. Singh, and R. Horaud, "Skeletal quads: Human action recognition using joint quadruples," in *Proc. 22nd Int. Conf. Pattern Recognit.*, 2014, pp. 4513–4518.

[8] R. Vemulapalli, F. Arrate, and R. Chellappa, "Human action recognition by representing 3D skeletons as points in a lie group," in *Proc. Comput. Vis. Pattern Recognit.*, 2014, pp. 588–595.

[9] J. Wang, Z. Liu, and Y. Wu, "Learning actionlet ensemble for 3D human action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 5, pp. 914–927, May 2014.

[10] M. E. Hussein, M. Torki, M. A. Gowayyed, and M. El-Saban, "Human action recognition using a temporal hierarchy of covariance descriptors on 3D joint locations," in *Proc. Int. Joint Conf. Artif. Intell.*, vol. 13, 2013, pp. 2466–2472.

[11] M. A. Gowayyed, M. Torki, M. E. Hussein, and M. El-Saban, "Histogram of oriented displacements (hod): Describing trajectories of human joints for action recognition," in *Proc. Int. Joint Conf. Artif. Intell.*, 2013, pp. 1351–1357.

[12] L. Xia, C. C. Chen, and J. K. Aggarwal, "View invariant human action recognition using histograms of 3D joints," in *Proc. Comput. Vis. Pattern Recognit. Workshops*, 2012, pp. 20–27.

[13] 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," in *Proc. Comput. Vis. Pattern Recognit. Workshops*, 2012, pp. 24–38.

[14] Y. Du, Y. Fu, and L. Wang, "Representation learning of temporal dynamics for skeleton-based action recognition," *IEEE Trans. Image Process.*, vol. 25, no. 7, pp. 3010–3022, Jul. 2016.

[15] Y. Li *et al.*, "Online human action detection using joint classification-regression recurrent neural networks," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 203–220.

[16] H. Wang and L. Wang, "Modeling temporal dynamics and spatial configurations of actions using two-stream recurrent neural networks," in *Proc. Comput. Vis. Pattern Recognit.*, 2017, pp. 3633–3642.

[17] W. Li, L. Wen, M. C. Chang, S. N. Lim, and S. Lyu, "Adaptive RNN tree for large-scale human action recognition," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 1453–1461.

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

[19] M. Liu, H. Liu, and C. Chen, "Enhanced skeleton visualization for view invariant human action recognition," *Pattern Recognit.*, vol. 68, pp. 346–362, 2017.

[20] T. S. Kim and A. Reiter, "Interpretable 3D human action analysis with temporal convolutional networks," in *Proc. Comput. Vis. Pattern Recognit. Workshops*, 2017, pp. 3633–3642.

[21] 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.

[22] C. Li, Y. Hou, P. Wang, and W. Li, "Joint distance maps based action recognition with convolutional neural networks," *IEEE Signal Process. Lett.*, vol. 24, no. 5, pp. 624–628, 2017.

[23] C. Li, Q. Zhong, D. Xie, and S. Pu, "Skeleton-based action recognition with convolutional neural networks," in *Proc. IEEE Int. Conf. Multimedia Expo. Workshops*, 2017, pp. 597–600.

[24] Q. Ke, M. Bennamoun, S. An, F. Sohel, and F. Boussaid, "A new representation of skeleton sequences for 3D action recognition," in *Proc. Comput. Vis. Pattern Recognit. Workshops*, 2017, pp. 4570–4579.

[25] Y. Du, Y. Fu, and L. Wang, "Skeleton based action recognition with convolutional neural network," in *Proc. 3rd IAPR Asian Conf. Pattern Recognit.*, 2015, pp. 579–583.

[26] Y. Hou, Z. Li, P. Wang, and W. Li, "Skeleton optical spectra based action recognition using convolutional neural networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 3, pp. 807–811, Mar. 2018.
[27] C. Li *et al.*, "Action-attending graphic neural network," *IEEE Trans. Image Process.*, vol. 27, no. 7, pp. 3657–3670, Jul. 2018.
[28] J. Weng, C. Weng, J. Yuan, and Z. Liu, "Discriminative spatio-temporal pattern discovery for 3D action recognition," *IEEE Trans. Circuits Syst. Video Technol.*, to be published, doi: 10.1109/TCSVT.2018.2818151.
[29] D. C. Luvizon, D. Picard, and H. Tabia, "2D/3D pose estimation and action recognition using multitask deep learning," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2018, vol. 2.
[30] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 11, pp. 1254–1259, Nov. 1998.
[31] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proc. Int. Conf. Learn. Represent.*, 2015, pp. 1–15.
[32] V. Mnih *et al.*, "Recurrent models of visual attention," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2204–2212.
[33] K. Xu *et al.*, "Show, attend and tell: Neural image caption generation with visual attention," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 2048–2057.
[34] J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 577–585.
[35] S. Sharma, R. Kiros, and R. Salakhutdinov, "Action recognition using visual attention," in *Proc. Int. Conf. Learn. Represent. Workshops*, 2016, pp. 1–11.
[36] C. Xiong, S. Merity, and R. Socher, "Dynamic memory networks for visual and textual question answering," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 2397–2406.
[37] A. Kumar *et al.*, "Ask me anything: Dynamic memory networks for natural language processing," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 1378–1387.
[38] Y. Yan, B. Ni, and X. Yang, "Predicting human interaction via relative attention model," in *Proc. Int. Joint Conf. Artif. Intell.*, 2017, pp. 3245–3251.
[39] J. Liu, G. Wang, P. Hu, L.-Y. Duan, and A. C. Kot, "Global context-aware attention LSTM networks for 3D action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 3671–3680.
[40] A. Shahroudy, J. Liu, T. T. Ng, and G. Wang, "NTU RGB+D: A large scale dataset for 3D human activity analysis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 1010–1019.
[41] K. Yun, J. Honorio, D. Chattopadhyay, T. L. Berg, and D. Samaras, "Two-person interaction detection using body-pose features and multiple instance learning," in *Proc. IEEE Comput. Vis. Pattern Recognit. Workshops*, 2012, pp. 28–35.
[42] M. Abadi *et al.*, "Tensorflow: A system for large-scale machine learning," in *Proc. Usenix Conf. Operating Syst. Des. Implementation*, 2016, pp. 265–283.
[43] 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, pp. 1929–1958, 2014. [Online]. Available: http://jmlr.org/papers/v15/srivastava14a.html
[44] J.-F. Hu, W.-S. Zheng, J. Lai, and J. Zhang, "Jointly learning heterogeneous features for RGB-D activity recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 5344–5352.
[45] W. Li, L. Wen, M. Choo Chuah, and S. Lyu, "Category-blind human action recognition: A practical recognition system," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 4444–4452.
[46] Y. Ji, G. Ye, and H. Cheng, "Interactive body part contrast mining for human interaction recognition," in *Proc. IEEE Int. Conf. Multimedia Expo. Workshops*, 2014, pp. 1–6.
[47] W. Zhu *et al.*, "Co-occurrence feature learning for skeleton based action recognition using regularized deep LSTM networks," in *Proc. AAAI Conf. Artif. Intell.*, vol. 2, 2016, pp. 3697–3703.

Zhaoxuan Fan received the B.E. degree in automation and the M.S. degree in control science and engineering from Shanghai Jiao Tong University, Shanghai, China, in 2015 and 2018, respectively. He is currently working as an Algorithm Engineer with Hikvision, Hangzhou, China. His research interests include visual analysis of human behavior and video understanding.

Xu Zhao received the Ph.D. degree in pattern recognition and intelligence systems from Shanghai Jiao Tong University, Shanghai, China, in 2011. He is currently an Associate Professor with the Department of Automation, Shanghai Jiao Tong University. He was a Visiting Scholar with the Beckman Institute for Advanced Science and Technology, University of Illinois at Urbana-Champaign, Champaign, IL, USA, from 2007 to 2008. He had been the Postdoctoral Research Fellow with the Northeastern University, from 2012 to 2013. His main research interests include computer vision and machine learning, especially human-centered visual computing.

Tianwei Lin received the B.E. degree from the School of Mechanical Engineering, Shanghai Jiao Tong University, Shanghai, China, in 2016, where he is currently working toward the M.S. degree with the Department of Automation. His research interests include computer vision, machine learning, and pattern recognition. He mainly focuses on action recognition and temporal action detection.

Haisheng Su received the B.E. degree in automation from Central South University, Changsha, China, in 2017, and is currently working toward the M.S. degree with the Department of Automation, Shanghai Jiao Tong University, Shanghai, China. His research interests include deep learning, video analysis, and temporal action detection.