Feedback Graph Convolutional Network for Skeleton-based Action Recognition

Hao Yang*, Dan Yan*, Li Zhang, Yunda Sun, Dong Li, and Stephen J. Maybank, *Fellow IEEE*

Abstract—Skeleton-based action recognition has attracted considerable attention since the skeleton data is more robust to the dynamic circumstances and complicated backgrounds than other modalities. Recently, many researchers have used the Graph Convolutional Network (GCN) to model spatial-temporal features of skeleton sequences by an end-to-end optimization. However, conventional GCNs are feedforward networks for which it is impossible for the shallower layers to access semantic information in the high-level layers. In this paper, we propose a novel network, named Feedback Graph Convolutional Network (FGCN). This is the first work that introduces a feedback mechanism into GCNs for action recognition. Compared with conventional GCNs, FGCN has the following advantages: (1) A multi-stage temporal sampling strategy is designed to extract spatial-temporal features for action recognition in a coarse to fine process; (2) A Feedback Graph Convolutional Block (FGCB) is proposed to introduce dense feedback connections into the GCNs. It transmits the high-level semantic features to the shallower layers and conveys temporal information stage by stage to model video level spatial-temporal features for action recognition; (3) The FGCN model provides predictions on-the-fly. In the early stages, its predictions are relatively coarse. These coarse predictions are treated as priors to guide the feature learning in later stages, to obtain more accurate predictions. Extensive experiments on three datasets, NTU-RGB+D, NTU-RGB+D120 and Northwestern-UCLA, demonstrate that the proposed FGCN is effective for action recognition. It achieves the state-of-the-art performance on all three datasets.

Index Terms—Feedback Mechanism, Graph Convolutional Network, Skeleton, Action Recognition

1 Introduction

In recent years, the quantity of videos uploaded from various terminals has exploded. This has driven the demand for automatic human action analysis based on the content of videos. In particular, human action recognition using skeletons has attracted worldwide attention because the skeleton data is more robust to dynamic circumstances and complicated backgrounds, compared with other modalities such as RGB [1] and optical flow [2]. Early deep learning methods using skeletons for action recognition usually represent the skeleton data as a sequence of vectors [3], [4], [5], [6] or a pseudo-image [7], [8], [9]. Then the data is modeled by a Recurrent Neural Network (RNN) or Convolutional Neural Network (CNN) respectively. However, these methods do not explicitly exploit the spatial dependencies among correlated joints, even though the spatial dependencies are informative for understanding human actions. More recently, some methods [10], [11], [12], [13] construct spatial-temporal graphs based on the natural connections in the human body and on temporal edges which connect the same joints between adjacent frames. These methods then use a GCN to extract spatial-temporal features. However, conventional GCNs [10], [11], [12], [13] are all single-pass feedforward networks that are fed with the entire skeleton sequence. The single-pass feedforward networks cannot access the high-level semantic information in the shallower layers. It is difficult for these models to extract effective spatial-temporal features, because the useful information is usually submerged by the motion-irrelevant or undiscriminating clips when they are fed with entire skeletons. For example, in the action “kicking something”, most clips show “standing upright”, and in the action “wear a shoe”, most clips show a subject sitting on a chair. The input of the entire skeleton sequence also increases the computational complexity of the model.

Motivated by these observation, we propose a novel neural network, named Feedback Graph Convolutional Network (FGCN), to extract effective spatial-temporal features from skeleton data in a coarse to fine process. The FGCN model is the first to introduce a feedback mechanism into GCNs for action recognition. Unlike conventional GCNs, the FGCN model uses a multi-stage temporal sampling strategy to sparsely sample input skeleton clips. This avoids inputting the entire skeleton sequence. The input skeleton sequence is divided into multiple stages in the temporal domain. Skeleton clips are sampled from each temporal stage. Each sampled clip is fed into a graph convolutional network to extract local spatial-temporal features for each stage. A Feedback Graph Convolutional Block (FGCB) is proposed to model video level spatial-temporal features by fusing the local features in a progressive process. The FGCB is a locally dense graph convolutional network with lateral connections from each stage to the next stage. The feedback block FGCB introduces feedback connections into conventional GCNs. From a semantic point of view, it works in a top down manner, which makes it possible for the shallower

- ** Equal Contribution*
- *Hao Yang, Dan Yan and Yunda Sun are with the R&D Center of Artificial Intelligent, NUCTECH Company Limited, Beijing, China. E-mail: {yanghao1, yandan, sunyunda}@nuctech.com*
- *Li Zhang and Dong Li are with the Department of Engineering Physics, Tsinghua University, Beijing, China.*
  E-mail: {zli, lid19}@mail.tsinghua.edu.cn
- *Stephen J. Maybank is with the Department of Computer Science and Information Systems, Birkbeck College. London, United Kingdom.*
  E-mail: steve.maybank@bbk.ac.uk

Manuscript submitted March, 2021.

convolutional layers to access the semantic information in the high-level layers. The feedback mechanism in FGCB works with a sequence of causes and effects. The output of each stage, except the last one, is transmitted to the next stage to modulate its input.

Another advantage of the FGCN is that it provides early predictions in a fraction of the action duration. This is valuable in many applications such as robotics or autonomous driving, in which a short latency time is very crucial. The predictions are provided by the multi-stage coarse to fine optimization process. In the early stages the FGCN is only fed with a part of the skeleton sequence. The information about the action is limited, so the predictions of the action are relatively coarse. These predictions are treated as priors to guide the feature learning in later stages. In the later stages, the model receives more information about the action. Thus its predictions tend to be more accurate. Several temporal fusion strategies are proposed to fuse local predictions from multiple temporal stages to obtain a video level prediction.

The main contributions of this paper are summarized as follows:

- We propose a novel Feedback Graph Convolutional Network (FGCN) for skeleton-based action recognition. It extracts spatial-temporal features of actions by a coarse to fine process and provides predictions on-the-fly. To our knowledge, this is the first work that introduces a feedback mechanism into GCNs for action recognition.
- A multi-stage temporal sampling strategy is proposed to sparsely sample input skeleton clips in the temporal domain. It supports the FGCN extracting spatial-temporal features for action recognition in a progressive process.
- We propose a densely connected Feedback Graph Convolutional Block (FGCB) with lateral connections between adjacent temporal stages. Functionally, it transmits high-level semantic features as priors, to guide feature learning in the shallower layers.
- The proposed FGCN model is extensively evaluated on three datasets, NTU-RGB+D, NTU-RGB+D120 and Northwestern-UCLA. It achieves state-of-the-art performances on all three datasets.

2 Related Works

2.1 Skeleton-based Action Recognition

As the depth sensor technologies (*e.g.* Kinect [14]) and pose estimation algorithms [15], [16] matured, it became possible to capture skeleton data in real time by locating the key joints. The skeleton data is robust to illumination change, scene variation, and complex backgrounds. This robustness facilitates data-driven methods for skeleton-based action recognition. Conventional action recognition methods usually extract hand-crafted features from skeleton sequences. Some traditional methods [17], [18], [19], [20] rely on view-invariant features of actions. Examples of these features are body part-based skeletal quads [17], [18], group sparsity based class-specific dictionary coding [19], and canonical view transformed features [20]. Other traditional methods integrate the information from the different modalities that are available in 3D action datasets. Many works [21], [22], [23], [24] combine depth information with the skeleton to improve performance. The depth information is represented by HOG features [21], [22] and Fourier Temporal Pyramids [24], or it is modeled by random decision forests [23]. The recent successes of deep learning have led to a surge of deep network based skeleton modeling methods. The most widely used models are RNN and CNN. RNN-based models [3], [4], [5], [6] usually concatenate the coordinates (2D or 3D) of all joints in each frame as a vector and then model the features of actions by an RNN fed with a sequence of vectors. LSTM-IRN [25] proposes the Interaction Relational Network to ensure that the interaction patterns can be properly learned. CNN-based models [7], [8], [9] stack a sequence of vectors to obtain a pseudo-image, then reduce the skeleton-based action recognition to an image classification task. The two-stream based model [26] combines RNN and CNN, operating on vectors of skeletons and RGB images respectively, to improve performance beyond what can be obtained with a single network. However, these methods do not explicitly model the spatial dependence between correlated joints, which is crucial for understanding human actions.

2.2 GCN based Action Recognition

The Graph Convolutional Networks (GCNs) [27], [28], [29], [30], [31] generalize the convolutional operation to deal with graphs. There are two main ways of constructing GCNs: spatial perspective and spectral perspective. Spatial perspective methods [27], [28] directly apply convolution filters to graph vertexes and their neighbors. In contrast, spectral perspective methods [29], [30], [31] consider the graph convolution as a form of spectral analysis by utilizing the eigenvalues and eigenvectors of the graph Laplacian matrices. This work follows the spatial perspective based methods [10], [11], [12], [13]. The ST-GCN model [10] overcomes the limitations of hand-crafted parts and traversal rules used in previous methods. It operates on a spatial-temporal graph to model the structured information about the joints along both the spatial and temporal dimensions. Based on ST-GCN, the 2s-AGCN model [11] proposes a two-stream adaptive graph convolutional network, in order to exploit the second-order information of the skeleton for action recognition. The DGNN model [12] represents the skeleton data as a directed acyclic graph based on the kinematic dependency between the joints and bones. The AS-GCN model [13] proposes an actional-structural graph convolutional network by generating the skeleton graph with actional links and structural links. However, conventional GCNs are all feedforward networks in which shallower layers cannot access the semantic information in high-level layers.

2.3 Feedback Network

A feedback mechanism exists in the human visual cortex [32], [33]. It has been a focus of research in psychology [34] and control theory [35], [36]. In recent years, the feedback

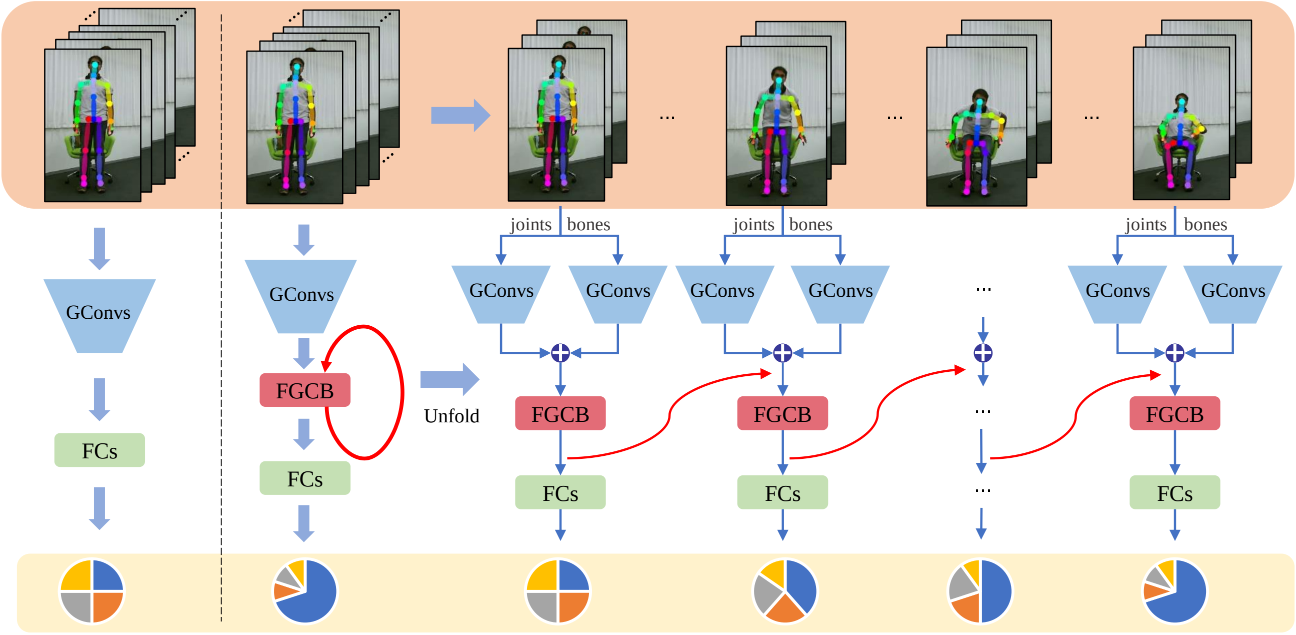


Fig. 1. Comparison of the conventional GCNs (left) and the proposed FGCN (right). FGCN models spatial-temporal features in a coarse to fine process with feedback mechanism. Red arrows represent the feedback connections of the Feedback Graph Convolutional Block (FGCB). The pie charts represent the predicted probabilities of actions.

mechanism has been introduced into deep neural networks in computer vision [37], [38], [39], [40], [41], [42], because it allows a network to use the information from the output to correct previous states. In action recognition, the shuttleNet [40] is a biologically-inspired deep network which is loop connected in order to mimic the brain's feedforward and feedback connections. In object recognition, the dasNet model [37] exploits the feedback structure to alter its convolutional filter sensitivities during classification and to focus its internal attention on some of its convolutional filters. The Feedback Network [38] firstly introduces the feedback mechanism into the convolutional recurrent neural network, which transmits the output with high-level information to the input layer. In super resolution, the DBPN model [41] proposes a deep back-projection network to achieve error feedback. The SRFBN model [39] designs a feedback block to handle the feedback connections and refine low-level representations with high-level information. In human pose estimation, Joao et al. [42] propose an iterative error feedback (IEF) by iteratively estimating and applying a self-correction to the current pose estimation.

3 The Method

3.1 Graph Convolutional Network

GCNs [27], [28] generalize the convolution operation to learn effective representations from graph structured data. In action recognition, a skeleton of the human body is defined as an undirected graph in which each joint in the skeleton corresponds to a vertex of the graph and each bone in the skeleton corresponds to an edge of the graph. Following [10], we construct the spatial temporal graph on skeleton sequences in two steps. First, the joints within one frame are connected according to the connectivity of the human body. Second, each joint in a given frame is connected to the same joint in the adjacent frames. In this paper, the spatial temporal graph associated with a video is denoted by $\mathbf{G} = \{\mathbf{V}, \mathbf{E}\}$, where $\mathbf{V}$ is the set of vertices in the graph and $\mathbf{E}$ is the set of edges in the graph. The vertex set is denoted as $\mathbf{V} = \{v_{ti}|t = 1, \dots, T; i = 1, \dots, N\}$. The edge set $\mathbf{E}$ consists of two subsets. The first subset specifies the intra-skeleton connections in each frame, denoted as $E_S(t) = \{v_{ti}v_{tj}|(i,j) \in \mathbf{Q}, t = 1, \dots, T\}$, where $\mathbf{Q}$ is the set of naturally connected joint pairs in the human body. The second subset contains the inter-frame edges which connect the same joints in adjacent frames, $E_T(i) = \{v_{ti}v_{(t+1)i}|t = 1, ..., T-1, i = 1, \dots, N\}$.

The graph convolution operation is defined on each vertex and its neighbor set. For a vertex v_{ti} in the graph, its neighbor set is denoted as $\mathbf{N}(v_{ti}) = \{v_{tj}|d(v_{ti}, v_{tj}) \le D\}$, where $d(v_{ti}, v_{tj})$ is the number of edges in the shortest path from v_{tj} to v_{ti}. We set $D = 1$ for the 1-distance neighbor set. The graph convolution operating on the vertex v_{ti} and its neighbor set $\mathbf{N}(v_{ti})$ is formulated as:

$$\mathbf{F}_{out}(v_{ti}) = \sum_{v_{tj} \in \mathbf{N}(v_{ti})} \frac{1}{Z[l(v_{tj})]} \mathbf{F}_{in}(v_{tj}) \mathbf{W}[l(v_{tj})], \quad (1)$$

where $\mathbf{F}_{in}$ and $\mathbf{F}_{out}$ denote the input and output features of this convolutional layer. $l(v_{tj})$ is the label function which allocates a label from 0 to $K-1$ for each vertex in $\mathbf{N}(v_{ti})$. Following the spatial configuration partition strategy proposed in ST-GCN [10], we set $K = 3$ to partition the neighbor set $\mathbf{N}(v_{ti})$ into 3 subsets. $\mathbf{W}(\cdot)$ is the weighting function which provides a weight vector according to the label $l(v_{tj})$. $Z[l(v_{tj})]$ denotes the number of vertices in the subset of $\mathbf{N}(v_{ti})$ with the label $l(v_{tj})$.

In the implementation, the connections between vertices in a graph are recorded in an $N \times N$ adjacency matrix $\mathbf{A}$.

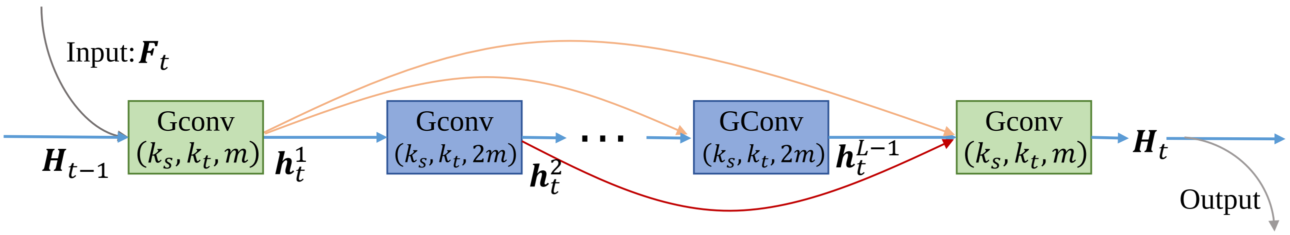


Fig. 2. The detailed architecture of the proposed Feedback Graph Convolutional Block (FGCB).It transmits the high-level semantic features to the shallower layers. Temporal information is accumulated stage by stage to model global spatial-temporal features for action recognition.

The adjacency matrix corresponding the k_{th} subset of the neighbor set $\mathbf{N}(v_{ti})$ is denoted as $\mathbf{A}_k$. With this adjacency matrix, the operation of graph convolution in Eqn. 1 can be formulated as:

$$\mathbf{F}_{out} = \sum_{k=0}^{K-1} \mathbf{W}_k (\Lambda_k^{-\frac{1}{2}} \mathbf{A}_k \Lambda_k^{-\frac{1}{2}} \mathbf{F}_{in}) \odot (\mathbf{M}_k), \tag{2}$$

where $\odot$ denotes the dot product and $\Lambda_k^{ii} = \sum_j \mathbf{A}_k^{ij}$ is a diagonal matrix. $\mathbf{W}_k$ is the weight vector of the convolution operation, which corresponds to the weighting function $\mathbf{W}(\cdot)$ in Eqn. 1. In practice, $\mathbf{A}_k$ is allocated with a learnable weight matrix $\mathbf{M}_k$ which is an $N \times N$ attention map that indicates the importance of each vertex. It is initialized as an all-one matrix.

3.2 Feedback Graph Convolutional Network

Traditional GCNs based methods [10], [11], [12], [13] for action recognition are all fed with the entire skeleton sequence in a feedforward network. However, the useful information is usually submerged by the motion-irrelevant and undiscriminating clips when the networks are fed with the entire skeleton sequence. Single-pass feedforward networks cannot access semantic information in the shallower layers. To tackle these problems, we propose a Feedback Graph Convolutional Network (FGCN) which extracts spatial-temporal features by a multi-stage progressive process. The architecture of the FGCN is shown in Fig. 1. Specifically, in the FGCN model, a multi-stage temporal sampling strategy is designed to sparsely sample a sequence of input clips from the skeleton data. These clips are first fed into graph convolutional layers to extract the local spatial-temporal features. Then, a Feedback Graph Convolutional Block (FGCB) is proposed to fuse the local spatial-temporal features from multiple temporal stages by transmitting the high-level information in each stage to the next stage to modulate its input. Finally, several temporal fusion strategies are proposed to fuse the local predictions from all temporal stages to give a video level prediction.

Formally, the multi-stage temporal sampling strategy samples input skeleton clips from the skeleton sequence S in two steps. First, it divides each skeleton sequence into T temporal stages with equal durations, denoted as $S = \{s_1, s_2, \ldots, s_t, \ldots, s_T\}$. Second, in each temporal stage, a skeleton clip is sampled randomly as the input of the deep model, denoted as $\{\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_t, \ldots, \mathbf{c}_T\}$, where $\mathbf{c}_t$ is the skeleton clip sampled from the corresponding stage s_t. Each sampled clip $\mathbf{c}_t$ is fed into the backbone network to extract the local spatial-temporal features in the corresponding temporal stage, formulated as:

$$\mathbf{F}_t = f_{GConvs}(\mathbf{c}_t), t = 1, 2, \ldots, T \tag{3}$$

where $\mathbf{F}_t$ is the local spatial-temporal features extracted by graph convolutional layers of backbone network. The ST-GCN model [10] is used as the backbone of the FGCN. It is denoted as $f_{GConvs}(\cdot)$ in Eqn. 3 and *GConvs* in Fig. 1.

All the local features extracted from the T temporal stages flow into the Feedback Graph Convolutional Block (FGCB) to enable the learning of global spatial-temporal features for action recognition. Each local feature is fed into the corresponding temporal step of the FGCB feedback block. As shown in Fig. 2, FGCB receives two inputs at stage t: one input is the output features from the previous stage $t-1$, denoted as $\mathbf{H}_{t-1}$; the other is the local features from the current stage, denoted as $\mathbf{F}_t$. Particularly, the input feature at the first stage, $\mathbf{F}_1$, is regarded as the initial feature $\mathbf{H}_0$. Based on these two inputs, the feedback process of FGCB is formulated as:

$$\mathbf{H}_t = f_{FGCB}(\mathbf{H}_{t-1}, \mathbf{F}_t), 1 \leq t \leq T, \tag{4}$$

where $\mathbf{H}_t$ is the output of FGCB at stage t, and the function $f_{FGCB}(\cdot)$ represents the feedback block. More details about FGCB can be found in Section 3.3.

Following FGCB, a fully connected layer and a softmax loss layer are used at each stage to predict actions. The prediction process from the output $\mathbf{H}_t$ of FGCB is formulated as:

$$\mathbf{P}_t = f_{pred}(\mathbf{H}_t), 1 \leq t \leq T, \tag{5}$$

where $\mathbf{P}_t \in R^C$ denotes the local prediction at stage t and C is the number of actions. The function $f_{pred}(\cdot)$ represents the operations of the fully connected layer and the softmax layer. After operating on T temporal stages, we obtain T local predictions $\{\mathbf{P}_1, \mathbf{P}_2, \ldots, \mathbf{P}_T\}$. Several temporal fusion strategies are proposed to fuse these local predictions obtained from different stages for a video level prediction $\mathbf{P}_S$. It is computed as:

$$\mathbf{P}_S = f_{tf}(\mathbf{P}_1, \mathbf{P}_2, \ldots, \mathbf{P}_T), \tag{6}$$

where $\mathbf{P}_S \in R^C$ and the function $f_{tf}(\cdot)$ defines a temporal fusion strategy. In this paper, we propose three temporal fusion strategies, *i.e.* last-win-all fusion, average fusion and weighting fusion. The FGCN model is trained end-to-end with the cross-entropy loss as follows:

$$L(y, \mathbf{P}_S) = -\sum_{i=1}^{C} \mathbf{Y}^i \log(\mathbf{P}_S^i), \tag{7}$$

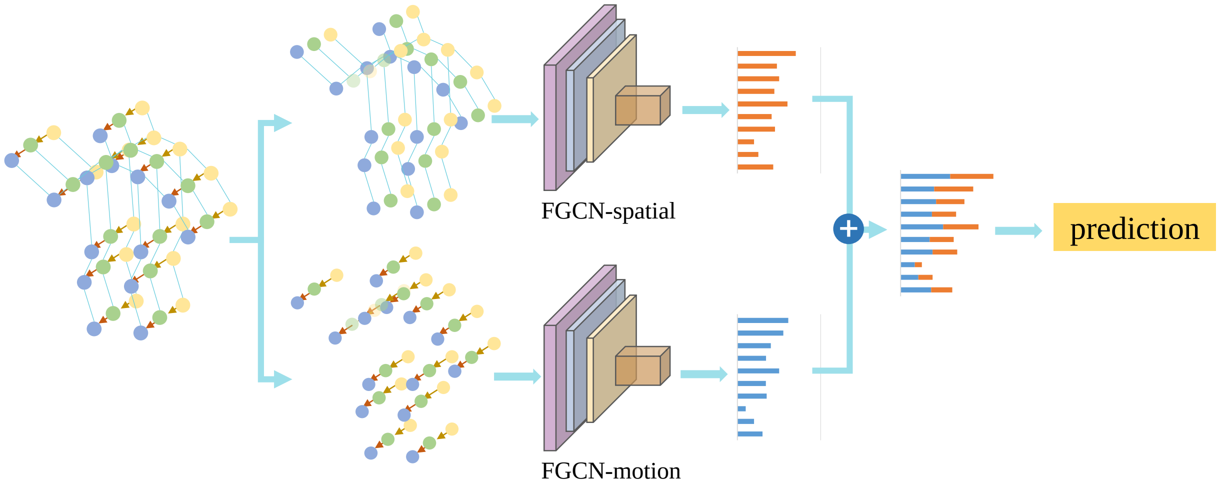


Fig. 3. The predictions of FGCN-spatial and FGCN-motion are fused for final action prediction. The model FGCN-spatial is fed with spatial graphs, and the other model, FGCN-motion, is fed with motion graphs.

where y is the action label of the skeleton S, if $y = i$, $\mathbf{Y}^i$ is set as 1, otherwise it is set as 0.

3.3 Feedback Graph Convolutional Block

The Feedback Graph Convolutional Block (FGCB) is the core component of the FGCN model. On the one hand, the feedback block FGCB transmits the high-level semantic information to the shallower layers to refine their encoded features. On the other hand, the output of each stage flows into the next stage to modulate its input. To enable FGCB to effectively transmit information from high-level layers to shallower layers and from the previous stage to the next stage, we propose a densely connected graph convolutional network, *i.e.*, FGCB. It adds shortcut connections from each layer to all subsequent layers. At a temporal stage t, FGCB receives the high-level information from the output $\mathbf{H}_{t-1}$ of the previous stage in order to modulate the middle-level feature $\mathbf{F}_t$ at the current stage. In our model, FGCB consists of L spatial-temporal graph convolutional layers. The spatial-temporal graph convolutional layer is denoted as $GConv(k_s, k_t, m)$ in Fig. 2, where k_s and k_t are the sizes of convolution kernel in the spatial and temporal domains respectively, and m denotes output channels of the graph convolutional layer.

As shown in Fig. 2, the first convolutional layer in FGCB receives two inputs $\mathbf{H}_{t-1}$ and $\mathbf{F}_t$. $\mathbf{H}_{t-1}$ is the high-level semantic features from the output of the last temporal stage $t-1$ of FGCB. The middle-level feature $\mathbf{F}_t$ is extracted from the input clip at stage t by the backbone network, as formulated in Eqn. 3. Then $\mathbf{H}_{t-1}$ and $\mathbf{F}_t$ are concatenated in the channel dimension, denoted as $[\mathbf{F}_t, \mathbf{H}_{t-1}]$. The output of the first layer in FGCB is formulated as:

$$\mathbf{h}_t^1 = f_{FGCB}^1([\mathbf{F}_t, \mathbf{H}_{t-1}]), \tag{8}$$

where $t = 1, 2, \dots, T$ and the function $f_{FGCB}^1(\cdot)$ denotes the first graph convolution layer of FGCB, and $\mathbf{h}_t^1$ denotes the output features of the first layer. Following the first layer, the l_{th} layer receives the output features from all preceding layers, $\mathbf{h}_t^1, \mathbf{h}_t^2, \dots, \mathbf{h}_t^{l-1}$, as input:

$$\mathbf{h}_t^l = f_{FGCB}^l([\mathbf{h}_t^1, \mathbf{h}_t^2, \dots, \mathbf{h}_t^{l-1}]), \tag{9}$$

where $l = 1, 2, \dots, L$ and $[\mathbf{h}_t^1, \mathbf{h}_t^2, \dots, \mathbf{h}_t^{l-1}]$ refers to the concatenated features of the preceding layers. Similar to the first layer, the final layer in FGCB compresses and fuses the features from the outputs of all preceding layers to produce the output of FGCB:

$$\mathbf{h}_t^L = f_{FGCB}^L([\mathbf{h}_t^1, \mathbf{h}_t^2, \dots, \mathbf{h}_t^{L-1}]), \tag{10}$$

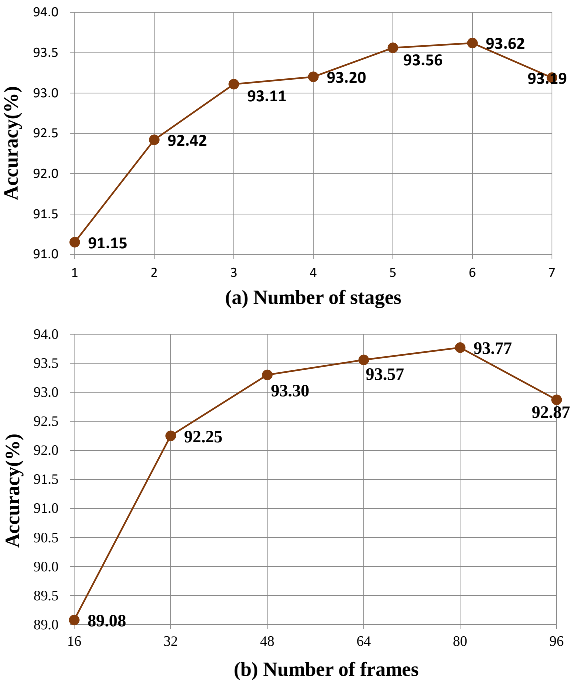
The features, $\mathbf{h}_t^L$, are treated as the output of the feedback block.

$$\mathbf{H}_t = \mathbf{h}_t^L. \tag{11}$$

3.4 Two-stream Framework of FGCN

The joints and bones of a skeleton only contain spatial information of actions. However, many actions are difficult to recognize from the spatial information alone, for example "wear a shoe" versus "take off a shoe", "wear glasses" versus "take off glasses", *etc*. Inspired by [12], we model the spatial-temporal features by exploiting both the spatial information and the temporal movement information of skeleton sequences. Based on the defined spatial temporal graph $\mathbf{G}$ in Section 3.1, the joints and bones of the spatial graph $\mathbf{G}_t = \{\mathbf{V}_t, \mathbf{E}_t\}$ are specified as $\mathbf{V}_t = \{v_{ti} | i = 1, \dots, N\}$ and $\mathbf{E}_t = \{v_{ti}v_{tj} | (i,j) \in \mathbf{Q}\}$. The joint or bone of the motion graph is defined as the difference of the corresponding joint vectors or bone vectors in two adjacent frames.

Given the joints and bones from two adjacent frames, denoted as v_{ti}, $v_{(t+1)i}$ and $v_{ti}v_{tj}$, $v_{(t+1)i}v_{(t+1)j}$ respectively, the joint of the motion graph is defined as $m(v_{ti}) = v_{(t+1)i} - v_{ti}$. Similarly, the bone of the motion graph is defined as $m(v_{ti}v_{tj}) = v_{(t+1)i}v_{(t+1)j} - v_{ti}v_{tj}$. The motion graph is formulated as $\mathbf{G}_t^m = \{\mathbf{V}_t^m, \mathbf{E}_t^m\}$, $\mathbf{V}_t^m = \{m(v_{ti}) | i = 1, \dots, N\}$ and $\mathbf{E}_t^m = \{m(v_{ti}v_{tj}) | (i,j) \in \mathbf{Q}\}$. In this paper, the spatial graph $\mathbf{G}_t$ and the motion graph $\mathbf{G}_t^m$ are fed into two separate FGCN models to predict action labels. The model fed with spatial graphs $\mathbf{G}_t$ is denoted as FGCN-spatial, the other fed with motion graphs $\mathbf{G}_t^m$ is denoted as FGCN-motion. The final prediction is obtained by weighting the output scores of the softmax layers from the two models, as shown in Fig. 3.



(a) Number of stages

(b) Number of frames

Fig. 4. Evaluating the influence of two key hyper-parameters on the cross-view benchmark of the NTU-RGB+D dataset, (a) the influence of the number of stages, (b) the influence of the number of frames in each stage.

4 Experiments

In this section, we evaluate the proposed FGCN method by conducting extensive experiments on three 3D skeleton action datasets, NTU-RGB+D, NTU-RGB+D120, and Northwestern-UCLA. First, the ablation experiments are conducted on the widely used dataset NTU-RGB+D. Then, our FGCN model is compared with state-of-the-art methods on all three datasets.

4.1 Datasets

NTU-RGB+D [43] is a widely used dataset for skeleton-based action recognition. The dataset contains more than 56,000 skeleton sequences categorized into 60 action classes. It provides 25 major body joints with 3D coordinates for every human in each frame. Two benchmark evaluations are recommended: cross-subject and cross-view. For cross-subject, both training and test sets consist of 20 subjects, and have 40,320 and 16,560 sequences respectively. The cross-view setup divides the data according to camera views. The training set has 37,920 sequences captured from the front and two side views, while the test set has 18,960 sequences captured from left and right 45 degree views.

NTU-RGB+D120 [44] is currently the largest in-door captured 3D skeleton dataset. It is an extension of the NTU-RGB+D with 120 action classes and more than 114,000 video samples. The newly added action classes make the action recognition task more challenging. For example, different actions may have similar body motions but different subjects. There may be fine-grained hand or finger motions, *etc.* The dataset has 106 subjects and 32 setup IDs. Cross-subject and cross-setup benchmarks are defined. For cross-subject, 53 subjects constitute the training set, and the remaining 53 subjects constitute the test set. Similarly,the 32 setup IDs are also divided equally into two parts for training and testing in cross-setup.

TABLE 1
Comparing different temporal fusion strategies on the cross-view benchmark of the NTU-RGB+D dataset.

Temporal Fusion Strategies	Weights					Cross-view
	w_1	w_2	w_3	w_4	w_5	
Last-win-all fusion	0	0	0	0	1	89.88
Weight fusion-1	0.05	0.05	0.1	0.2	0.6	93.09
Weight fusion-2	0.1	0.15	0.2	0.25	0.3	93.05
Weight fusion-3	0.1	0.2	0.4	0.2	0.1	92.61
Average fusion	0.2	0.2	0.2	0.2	0.2	**93.57**

Northwestern-UCLA [45] is a multi-view 3D event dataset captured simultaneously by three Kinect cameras from different viewpoints. This dataset includes 1494 video sequences covering 10 action categories performed by 10 subjects from 1 to 6 times. It provides 3D spatial coordinates of 20 major body joints. As reported in [45], all the samples from the first two cameras are picked for training. The samples from the remaining camera are for testing.

4.2 Implementation Details

All experiments are implemented with the PyTorch deep learning framework. The Stochastic Gradient Descent (SGD) optimizer is used during training with a batch size of 32 and a momentum of 0.9. The initial learning rate is set as 0.1. The learning rate is divided by 10 at the 40th and 60th epoch. The training process stops at the 80th epoch. In our experiments, the input video is divided into 5 stages temporally and 64 consecutive frames are sampled randomly from each stage to form an input clip. To make a fair comparison with the baseline model ST-GCN [10], ST-GCN is used as the backbone of the FGCN model. The hyper-parameters of FGCB are set empirically. We set the number of layers in FGCB as 4 (*i.e.*, $L = 4$) to balance the computational complexity and performance. The spatial kernel size, temporal kernel size and output channels of the convolutional layers in FGCB are set as $k_s = 3$, $k_t = 3$ and $m = 256$ respectively.

4.3 Ablation Study

In this section, we design four ablation experiments to evaluate the influence of the multi-stage temporal sampling strategy, the feedback block FGCB, temporal fusion strategies and different inputs on the performance of our FGCN model. These experiments are all conducted on the challenging skeleton dataset NTU-RGB+D.

In the first experiment, we evaluate the influence of two key hyper-parameters, *i.e.*, the number of stages and the number of frames sampled in each stage, on the performance of our FGCN model. In this experiment, the FGCN model is fed with joints only, using the average temporal fusion strategy. It is evaluated on the cross-view benchmark of the NTU-RGB+D dataset. In Fig. 4(a), the performances of FGCN with different numbers of stages are reported. Up to a certain point, the performance of FGCN increases as the number of stages increases, because more high-level information is fed back and used by the model. When

TABLE 2
Evaluating the effectiveness of the proposed multi-stage temporal sampling strategy and feedback block FGCB on the NTU-RGB+D dataset.

Models	Cross-subject	Cross-view
ST-GCN [10]	81.5	88.3
Multi-stage ST-GCN	84.25	91.54
FGCN (FGCB+multi-stage sampling)	87.04	93.57

the number of stages surpasses a threshold (*i.e.*, 7 stages), there is a drop in the performance of FGCN. Because most videos contain 300∼400 frames. If the FGCN model is fed with 7 stages and fed with 64 frames in each stage, there are large overlaps between adjacent temporal stages. So there is no randomness for the input clip sampling process. The FGCN model achieves similar performances when the skeleton sequence is divided into 5 or 6 stages. In the subsequent experiments, we set the number of temporal stages as 5, to balance performance against computational cost. In Fig. 4(b), we evaluate the performance of FGCN fed with different numbers of frames in each stage. The FGCN model achieves the similar performances when it is fed with 80 frames or 64 frames in each temporal stage. The computational cost of the model fed with 80 frames in each stage is much higher than the cost of the model fed with 64 frames. To balance performance against computational cost, we set the number of frames as 64 in the subsequent experiments.

In the second experiment, we compare three different temporal fusion strategies, *i.e.* last-win-all fusion, average fusion, and weight fusion, on the cross-view benchmark of the NTU-RGB+D dataset. All models in this experiment are fed with joints only. The results of these models are listed in Tab. 1. Among these three fusion strategies, the average fusion strategy achieves the best performance. Based on the results, we use the average fusion strategy to fuse the local predictions for the video level prediction in the subsequent experiments.

In the third experiment, we evaluate the effectiveness of the proposed multi-stage temporal sampling strategy and the feedback block FGCB separately. Firstly, we propose the Multi-stage ST-GCN model by introducing the proposed multi-stage sampling strategy into the baseline model ST-GCN [10]. The Multi-stage ST-GCN model achieves 91.54% on the cross-view benchmark of the NTU-RGB+D dataset, as shown in Tab. 2. It outperforms the original ST-GCN model [10] by 3.24%. This improvement demonstrates the effectiveness of the proposed multi-stage sampling strategy. The FGCN model involves the densely connected feedback block FGCB and the multi-stage temporal sampling strategy. As shown in Tab. 2, FGCN fed with joints only achieves 93.57% on the cross-view benchmark of the NTU-RGB+D dataset. It outperforms both the original ST-GCN model [10] and the Multi-stage ST-GCN model. The FGCN model outperforms the baseline model ST-GCN by 5.54% and 5.27% on the cross-subject and cross-view benchmarks of the NTU-RGB+D dataset respectively. These improvements demonstrate the effectiveness of the proposed feedback block FGCB. The confusion matrices of FGCN and ST-GCN models for the former 30 actions are shown in Fig. 5, and the complete confusion matrices are shown in the supplementary materials. These improvements of FGCN indicate that introducing the feedback mechanism into GCNs is very effective for action recognition.

TABLE 3
Evaluating the effectiveness of the FGCN model fed with different inputs on the NTU-RGB+D dataset.

Models	Cross-subject	Cross-view
FGCN-joint	87.04	93.57
FGCN-bone	86.96	93.22
FGCN-joint+FGCN-bone	89.24	95.28
FGCN-spatial	88.32	94.82
FGCN-motion	85.96	93.57
FGCN-spatial+FGCN-motion	**90.22**	**96.25**

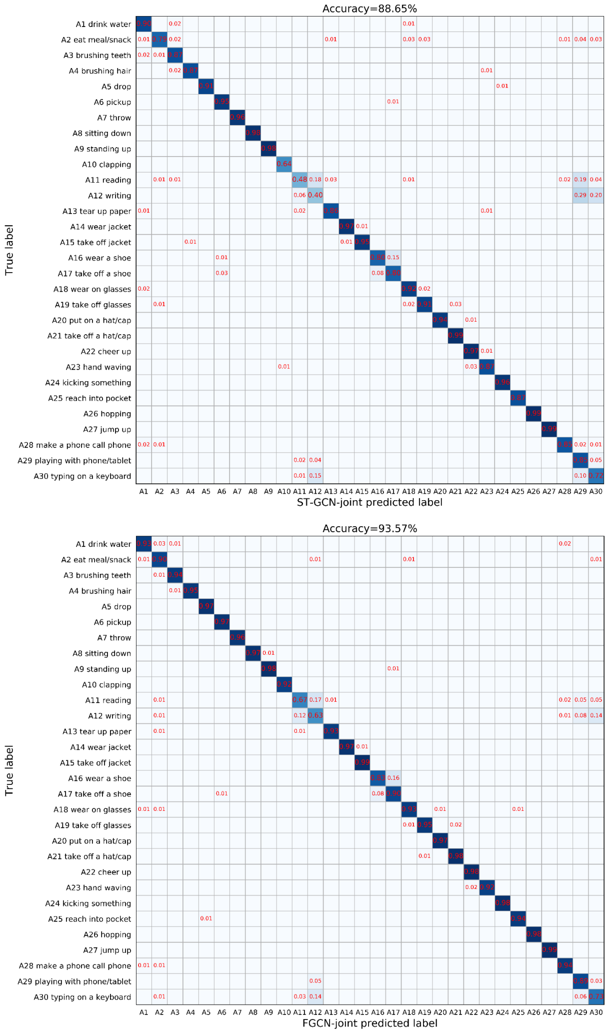


Fig. 5. The confusion matrices of the models, ST-GCN-joint and FGCN-joint, on the NTU-RGB+D dataset.

In the fourth experiment, we evaluate the effectiveness of the fusion of FGCN models that are fed with different inputs, *i.e.*, joints and bones, spatial graphs and motion graphs, on the NTU-RGB+D dataset. Firstly, we fuse the softmax scores of two FGCN models, where one model is fed with joints of the spatial graph, denoted as FGCN-joint, the other is FGCN-bone which is fed with bones of the spa-

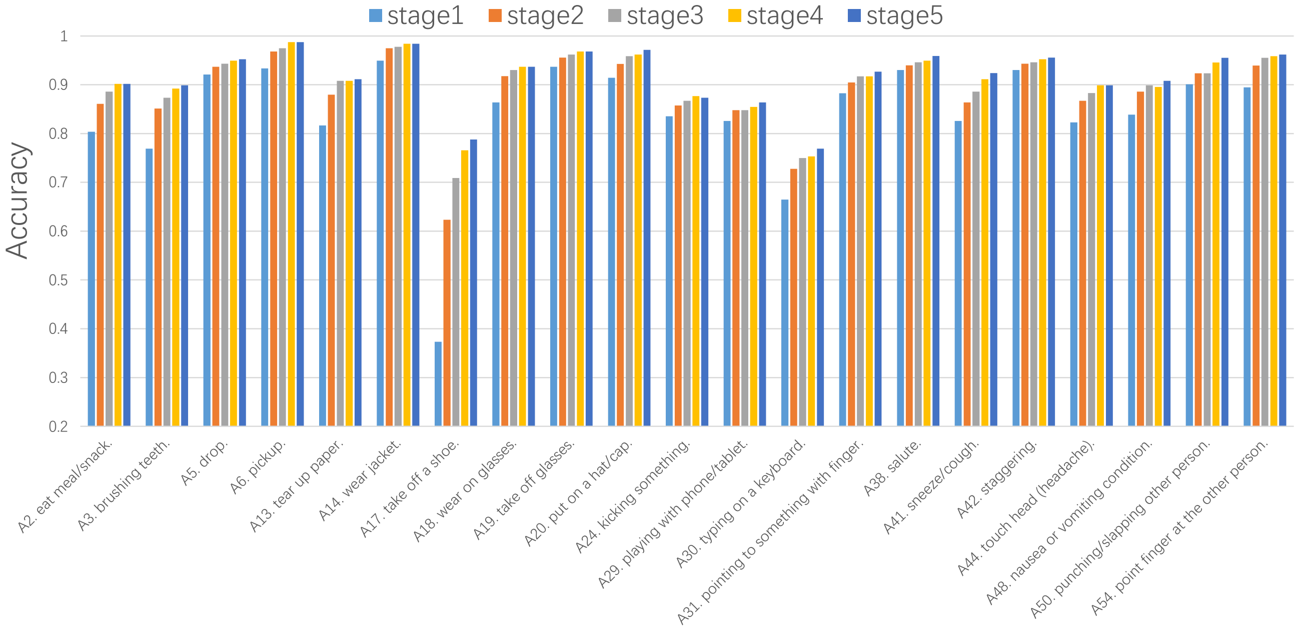


Fig. 6. Early prediction of the proposed FGCN model. The prediction accuracies at each stage of actions are presented. The FGCN model is fed with joints only and it is evaluated on the cross-view benchmark of the NTU-RGB+D dataset.

TABLE 4
The accuracies of FGCN with different observation ratios of actions.

Stage	1st stage	2nd stage	3rd stage	4th stage	5th stage
Observation ratios	20%	40%	60%	80%	100%
Accuracy (%)	91.27	92.38	93.49	93.65	93.97

TABLE 5
Comparisons with the state-of-the-art methods on the NTU-RGB+D dataset.

Models	Cross-subject	Cross-view
ResNet152-3S (ICMEW 2017) [46]	85.0	92.3
ST-GCN (AAAI 2018) [10]	81.5	88.3
DPRL+GCNN (CVPR 2018) [47]	83.5	89.8
SR-TSL (ECCV 2018) [48]	84.8	92.4
PB-GCN (BMVC 2018) [49]	87.5	93.2
Bayesian GC-LSTM (ICCV 2019) [50]	81.8	89.0
AS-GCN (CVPR 2019) [13]	86.8	94.2
AGC-LSTM (CVPR 2019) [51]	89.2	95.0
2s-AGCN (CVPR 2019) [11]	88.5	95.1
DGNN (CVPR 2019) [12]	89.9	96.1
GR-GCN (ACM MM 2019) [52]	87.5	94.3
BAGCN (arXiv 2019) [53]	90.3	96.3
MS-G3D (CVPR 2020) [54]	91.5	96.2
STIGCN (ACM MM 2020) [55]	90.1	96.1
CGCN (arXiv 2020) [56]	90.3	96.4
NAS-GCN (AAAI 2020) [57]	89.4	95.7
Sym-GNN (T-PAMI 2021) [58]	90.1	96.4
AGE-Ens:S(J+B)&V(J+B) (arXiv 2021) [59]	91.0	96.1
FGCN (ours)	**90.2**	**96.3**

tial graph. As shown in the upper part of Tab. 3, the fusion model FGCN-joint+FGCN-bone achieves a clear improvement over both the FGCN-joint and the FGCN-bone. Then, we report the experimental results of FGCN fed with spatial graphs and motion graphs in Tab. 3. The FGCN-spatial model fed with spatial graphs (joints and bones) achieves 88.32% on cross-subject and 94.82% on cross-view. It is comparable with the performance of the FGCN-joint+FGCN-bone model that fuses the softmax scores of two models. The FGCN-motion fed with motion graphs achieves 85.96% on cross-subject and 93.57% on cross-view. Finally, we fuse the softmax scores of FGCN models fed with spatial graphs and motion graphs. The FGCN-spatial+FGCN-motion achieves 90.22% on cross-subject and 96.25% on cross-view, and it achieves a clear improvement over both the FGCN-spatial and the FGCN-motion.

4.4 Early Predictions

In this section, we present the prediction accuracies at each stage of actions to show the advantages of the early prediction by the FGCN model. The FGCN model is fed with joints only and it is evaluated on the cross-view benchmark of the NTU-RGB+D dataset. The early prediction benefits from the multi-stage coarse to fine process. The predictions at multiple temporal stages become more accurate stage by stage. As shown in Fig. 6, the predictions of FGCN are relatively coarse in the early stages, because the FGCN model is only fed with a part of the action sequence, in which the information about the action is limited. In the later stages the predictions of FGCN model become more accurate, because the model receives more information about the action and it is guided by the prior information in former stages.

Based on the results in this experiment, we report the accuracies with different observation ratios (*i.e.*, 20%, 40%, 60%, 80% and 100%) of actions in Tab. 4. The accuracy in each stage is the average of the accuracies over all actions. As shown in Tab. 4, the FGCN model achieves 91.27% in the first stage. It outperforms the most comparable baseline model ST-GCN [10] by 3%. When FGCN is fed with more observations of actions in the subsequent stages, it gets higher accuracies.

TABLE 6
Comparisons with the state-of-the-art methods on the NTU-RGB+D120 dataset.

Models	Cross-subject	Cross-setup
Internal Feature Fusion (T-PAMI 2017) [60]	58.2	60.9
Multi-Task Learning (CVPR 2017) [7]	58.4	57.9
Skeleton Visualization (PR 2017) [8]	60.3	63.2
TS Attention LSTM (TIP 2017) [61]	61.2	63.3
Multi-Task RotClips (TIP 2018) [62]	62.2	61.8
ST-GCN (AAAI 2018) (reported in [63])	72.4	71.3
AS-GCN (CVPR 2019) (reported in [63])	77.7	78.9
FSNet (T-PAMI 2019) (reported in [44])	59.9	62.4
TSRJI (SIBGRAPI 2019) [64]	67.9	62.8
Shift-GCN (2-stream) (CVPR 2020) [65]	85.3	86.6
Shift-GCN (4-stream) (CVPR 2020) [65]	85.9	87.6
ST-TR (CVIU 2021) [66]	85.1	87.1
GVFE + AS-GCN (ICPR 2021) [63]	79.2	81.2
FGCN (ours)	**85.4**	**87.4**

4.5 Comparison with State-of-the-art

In this section, we compare the performance of the FGCN model with the state-of-the-art methods developed in recent years on the NTU-RGB+D dataset, the NTU-RGB+D120 dataset, and the Northwestern-UCLA dataset.

For the NTU-RGB+D dataset, we display the accuracy of skeleton-based action recognition methods, such as CNN-based methods [46], RNN-based methods [48], [50], [51] and GCN based methods [10], [11], [12], [13]. As shown in Tab. 5, the proposed FGCN model achieves 8.7% and 8.0% improvements on the cross-subject and cross-view benchmarks respectively over the most comparable method ST-GCN [10]. The STIGCN model [55] is constructed with the novel simple and highly modularized graph convolutional blocks that aggregate multi-granularity information from both the spatial and temporal paths. It achieves 90.1% and 96.1% on the cross-subject and cross-view benchmarks of the NTU-RGB+D dataset respectively. The FGCN model outperforms it on both the cross-subject and cross-view benchmarks. In [57], the authors exploit the Neural Architecture Search (NAS) to automatically design a GCN for skeleton-based human action recognition. The resulting model, NAS-GCN, achieves 89.4% and 95.7% on the cross-subject and cross-view benchmarks of the NTU-RGB+D dataset respectively. FGCN outperforms it by about 1% on both the cross-subject and cross-view benchmarks. Moreover, the FGCN model outperforms other state-of-the-art methods, such as 2s-AGCN [11], AS-GCN [13], and GR-GCN [52]. Our FGCN model achieves the state-of-the-art performance on both cross-subject and cross-view benchmarks of the NTU-RGB+D dataset.

For the NTU-RGB+D120 dataset, the results on cross-subject and cross-setup benchmarks of the recent state-of-the-art methods are listed in Tab. 6. The proposed FGCN model achieves 85.4% on cross-subject and 87.4% on cross-setup and it outperforms the most comparable ST-GCN model [10] by 13.0% and 16.1% on the cross-subject and cross-setup benchmarks respectively. The FSNet [67] proposes a novel window scale selection method to predict ongoing actions. It achieves 59.9% and 62.4% on the cross-subject and cross-setup benchmarks of the NTU-RGB+D120 dataset respectively. Our FGCN model outperforms it on both the cross-subject and cross-setup benchmarks. Shift-GCN [65] is constructed with the novel shift graph operations and lightweight point-wise convolutions. The fusion of 4 networks, *i.e.,*Shift-GCN (4-stream), achieves 85.9% and 87.6% respectively on the cross-subject and cross-setup benchmarks of the NTU-RGB+D120 dataset. Our FGCN model achieves a comparative performance with the Shift-GCN (4-stream) model. The FGCN model fuses the predictions of two networks (*i.e.*, FGCN-spatial and FGCN-motion). It outperforms the most comparable Shift-GCN (2-stream) model by nearly 1% on the cross-setup benchmark of the NTU-RGB+D120 dataset. The FGCN model outperforms other state-of-the-art methods with large margins. For example, the FGCN model outperforms Two-Stream Attention LSTM [61] by over 24% on both the cross-subject and cross-setup benchmarks. Our FGCN model outperforms the most recent methods, such as GVFE + AS-GCN [63] and ST-TR [66] on both the cross-subject and cross-setup benchmarks of the NTU-RGB+D120 dataset.

TABLE 7
Comparisons with the state-of-the-art methods on the Northwestern-UCLA dataset.

Models	Accuracy(%)
Actionlet ensemble (T-PAMI 2013) [24]	76.0
Lie group (CVPR 2014) [18]	74.2
HBRNN-L(CVPR 2015) [3]	78.5
Skeleton Visualization (PR 2017) [8]	86.1
Ensemble TS-LSTM (ICCV 2017) [71]	89.2
AGC-LSTM (CVPR 2019) [51]	93.3
JS+JM+BS+BM (ICME 2019) [68]	91.3
HiGCN (ICIG 2019) [69]	88.9
MSNN (CSVT 2020) [70]	89.4
FGCN (ours)	**95.3**

For the typical 3D action recognition dataset Northwestern-UCLA, we compare the proposed FGCN model with the state-of-the-art methods in recent years. The results of these models are reported in Tab. 7. The FGCN model outperforms the part-based hierarchical recurrent neural network HBRNN-L [3] by 16.8%. The recent method AGC-LSTM proposes an attention enhanced graph convolutional LSTM network to capture discriminative features from the co-occurrence relationship between spatial configuration and temporal dynamics. The FGCN model outperforms it by 2%. Moreover, the FGCN model outperforms the recent methods, such as JS+JM+BS+BM [68], HiGCN [69] and MSNN [70]. The proposed FGCN model achieves state-of-the-art performance on the Northwestern-UCLA dataset.

5 Conclusion

In this paper, we propose a novel FGCN model to extract effective spatial-temporal features of actions in a coarse to fine process. Firstly, we propose a multi-stage temporal sampling strategy to sample sparse skeleton clips in multiple temporal stages and exploit graph convolutional layers to extract local spatial-temporal features for each stage. Then, we introduce the feedback mechanism into conventional GCNs by proposing the feedback block FGCB which is a densely connected graph convolutional network. The FGCB transmits the semantic information from high-level layers to shallower layers and from the former stages to the later stages. Moreover, the FGCN provides early predictions which help agents in applications to make timely decisions on-the-fly. The proposed FGCN model is extensively evaluated on the NTU-RGB+D, NTU-RGB+D120 and Northwestern-UCLA datasets. It has achieved state-of-the-art performance on all three datasets. In future work, we will extend the feedback mechanism into CNNs to deal with RGB videos for action recognition, *e.g.*, the feedback block FGCB is stacked on a CNN to model the motions in RGB videos by a coarse to fine process.

REFERENCES

[1] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1725–1732.

[2] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Advances in Neural Information Processing Systems*, 2014, pp. 568–576.

[3] Y. Du, W. Wang, and L. Wang, "Hierarchical recurrent neural network for skeleton based action recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1110–1118.

[4] 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 *AAAI Conference on Artificial Intelligence*, 2017, pp. 4263–4270.

[5] P. Zhang, C. Lan, J. Xing, W. Zeng, J. Xue, and N. Zheng, "View adaptive recurrent neural networks for high performance human action recognition from skeleton data," in *IEEE International Conference on Computer Vision*, 2017, pp. 2117–2126.

[6] S. Li, W. Li, C. Cook, C. Zhu, and Y. Gao, "Independently recurrent neural network (IndRNN): Building a longer and deeper RNN," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5457–5466.

[7] Q. Ke, M. Bennamoun, S. An, F. Sohel, and F. Boussaid, "A new representation of skeleton sequences for 3D action recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 3288–3297.

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

[9] H. Liu, J. Tu, and M. Liu, "Two-stream 3D convolutional neural network for skeleton-based action recognition," *arXiv preprint arXiv:1705.08106*, 2017.

[10] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *AAAI Conference on Artificial Intelligence*, 2018, pp. 7444–7452.

[11] L. Shi, Y. Zhang, J. Cheng, and H. Lu, "Two-stream adaptive graph convolutional networks for skeleton-based action recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 12 026–12 035.

[12] ——, "Skeleton-based action recognition with directed graph neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 7912–7921.

[13] M. Li, S. Chen, X. Chen, Y. Zhang, Y. Wang, and Q. Tian, "Actional-structural graph convolutional networks for skeleton-based action recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3595–3603.

[14] Z. Zhang, "Microsoft kinect sensor and its effect," *IEEE Transaction on Multimedia*, vol. 19, no. 2, pp. 4–10, 2012.

[15] A. Toshev and C. Szegedy, "Deeppose: Human pose estimation via deep neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1653–1660.

[16] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2D pose estimation using part affinity fields," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 7291–7299.

[17] G. Evangelidis, G. Singh, and R. Horaud, "Skeletal quads: Human action recognition using joint quadruples," in *IEEE International Conference on Pattern Recognition*, 2014, pp. 4513–4518.

[18] R. Vemulapalli, F. Arrate, and R. Chellappa, "Human action recognition by representing 3D skeletons as points in a lie group," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 588–595.

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

[20] H. Rahmani and A. Mian, "Learning a non-linear knowledge transfer model for cross-view action recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 2458–2466.

[21] J.-F. Hu, W.-S. Zheng, J. Lai, and J. Zhang, "Jointly learning heterogeneous features for RGB-D activity recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5344–5352.

[22] E. Ohn-Bar and M. Trivedi, "Joint angles similarities and HOG2 for action recognition," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2013, pp. 465–470.

[23] H. Rahmani, A. Mahmood, D. Q. Huynh, and A. Mian, "Real time action recognition using histograms of depth gradients and random decision forests," in *IEEE Winter Conference on Applications of Computer Vision*, 2014, pp. 626–633.

[24] J. Wang, Z. Liu, Y. Wu, and J. Yuan, "Learning actionlet ensemble for 3D human action recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 5, pp. 914–927, 2013.

[25] M. Perez, J. Liu, and A. C. Kot, "Interaction relational network for mutual action recognition," *IEEE Transactions on Multimedia*, vol. PP, no. 99, 2021.

[26] R. Zhao, H. Ali, and P. Van der Smagt, "Two-stream RNN/CNN for action recognition in 3D videos," in *IEEE International Conference on Intelligent Robots and Systems*, 2017, pp. 4260–4267.

[27] J. Bruna, W. Zaremba, A. Szlam, and Y. Lecun, "Spectral networks and locally connected networks on graphs," in *International Conference on Learning Representations*, 2014.

[28] M. Niepert, M. Ahmed, and K. Kutzkov, "Learning convolutional neural networks for graphs," in *IEEE International Conference on Machine Learning*, 2016, pp. 2014–2023.

[29] D. K. Duvenaud, D. Maclaurin, J. Iparraguirre, R. Bombarell, T. Hirzel, A. Aspuru-Guzik, and R. P. Adams, "Convolutional networks on graphs for learning molecular fingerprints," in *Advances in Neural Information Processing Systems*, 2015, pp. 2224–2232.

[30] M. Henaff, J. Bruna, and Y. Lecun, "Deep convolutional networks on graph-structured data," *arXiv preprint arXiv:1506.05163*, 2015.

[31] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *International Conference on Learning Representations*, 2017.

[32] J. Hupé, A. James, B. Payne, S. Lomber, P. Girard, and J. Bullier, "Cortical feedback improves discrimination between figure and background by v1, v2 and v3 neurons," *Nature*, vol. 394, no. 6695, pp. 784–787, 1998.

[33] C. D. Gilbert and M. Sigman, "Brain states: top-down influences in sensory processing," *Neuron*, vol. 54, no. 5, pp. 677–696, 2007.

[34] S. J. Ashford and L. L. Cummings, "Feedback as an individual resource: Personal strategies of creating information," *Organizational Behavior and Human Performance*, vol. 32, no. 3, pp. 370–398, 1983.

[35] E. B. Lee and L. Markus, "Foundations of optimal control theory," Minnesota Univ Minneapolis Center For Control Sciences, Tech. Rep., 1967.

[36] A. G. Parlos, K. T. Chong, and A. F. Atiya, "Application of the recurrent multilayer perceptron in modeling complex process dynamics," *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 255–266, 1994.

[37] M. F. Stollenga, J. Masci, F. Gomez, and J. Schmidhuber, "Deep networks with internal selective attention through feedback connections," in *Advances in Neural Information Processing Systems*, 2014, pp. 3545–3553.

[38] A. R. Zamir, T.-L. Wu, L. Sun, W. B. Shen, B. E. Shi, J. Malik, and S. Savarese, "Feedback networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1308–1317.

[39] Z. Li, J. Yang, Z. Liu, X. Yang, G. Jeon, and W. Wu, "Feedback network for image super-resolution," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3867–3876.

[40] Y. Shi, Y. Tian, Y. Wang, Z. Wei, and T. Huang, "Learning long-term dependencies for action recognition with a biologically-inspired deep network," in *IEEE International Conference on Computer Vision*, 2017, pp. 716–725.

[41] M. Haris, G. Shakhnarovich, and N. Ukita, "Deep back-projection networks for super-resolution," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1664–1673.

[42] J. Carreira, P. Agrawal, K. Fragkiadaki, and J. Malik, “Human pose estimation with iterative error feedback,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4733–4742.

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

[44] J. Liu, A. Shahroudy, M. L. Perez, G. Wang, L.-Y. Duan, and A. K. Chichung, “NTU RGB+D 120: A large-scale benchmark for 3D human activity understanding,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[45] J. Wang, X. Nie, Y. Xia, Y. Wu, and S. C. Zhu, “Cross-view action modeling, learning and recognition,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2649–2656.

[46] B. Li, Y. Dai, X. Cheng, H. Chen, Y. Lin, and M. He, “Skeleton based action recognition using translation-scale invariant image mapping and multi-scale deep CNN,” in *IEEE International Conference on Multimedia and Expo Workshops*, 2017, pp. 601–604.

[47] Y. Tang, Y. Tian, J. Lu, P. Li, and J. Zhou, “Deep progressive reinforcement learning for skeleton-based action recognition,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5323–5332.

[48] C. Si, Y. Jing, W. Wang, L. Wang, and T. Tan, “Skeleton-based action recognition with spatial reasoning and temporal stack learning,” in *Springer European Conference on Computer Vision*, 2018, pp. 103–118.

[49] K. Thakkar and P. Narayanan, “Part-based graph convolutional network for action recognition,” in *British Machine Vision Conference*, 2018, pp. 1–13.

[50] R. Zhao, K. Wang, H. Su, and Q. Ji, “Bayesian graph convolution LSTM for skeleton based action recognition,” in *IEEE International Conference on Computer Vision*, 2019, pp. 6882–6892.

[51] C. Si, W. Chen, W. Wang, L. Wang, and T. Tan, “An attention enhanced graph convolutional LSTM network for skeleton-based action recognition,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1227–1236.

[52] X. Gao, W. Hu, J. Tang, J. Liu, and Z. Guo, “Optimized skeleton-based action recognition via sparsified graph regression,” in *ACM International Conference on Multimedia*, 2019, pp. 601–610.

[53] J. Gao, T. He, X. Zhou, and S. Ge, “Focusing and diffusion: Bidirectional attentive graph convolutional networks for skeleton-based action recognition,” *arXiv preprint arXiv:1912.11521*, 2019.

[54] Z. Liu, H. Zhang, Z. Chen, Z. Wang, and W. Ouyang, “Disentangling and unifying graph convolutions for skeleton-based action recognition,” in *IEEE conference on computer vision and pattern recognition*, 2020, pp. 143–152.

[55] Z. Huang, X. Shen, X. Tian, H. Li, J. Huang, and X.-S. Hua, “Spatio-temporal inception graph convolutional networks for skeleton-based action recognition,” in *ACM International Conference on Multimedia*, 2020, pp. 2122–2130.

[56] D. Yang, M. M. Li, H. Fu, J. Fan, and H. Leung, “Centrality graph convolutional networks for skeleton-based action recognition,” *arXiv preprint arXiv:2003.03007*, 2020.

[57] W. Peng, X. Hong, H. Chen, and G. Zhao, “Learning graph convolutional network for skeleton-based human action recognition by neural searching,” in *AAAI Conference on Artificial Intelligence*, vol. 34, no. 03, 2020, pp. 2669–2676.

[58] M. Li, S. Chen, X. Chen, Y. Zhang, Y. Wang, and Q. Tian, “Symbiotic graph neural networks for 3d skeleton-based human action recognition and motion prediction,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[59] Z. Qin, Y. Liu, P. Ji, D. Kim, L. Wang, B. McKay, S. Anwar, and T. Gedeon, “Leveraging third-order features in skeleton-based action recognition,” *arXiv preprint arXiv:2105.01563*, 2021.

[60] J. Liu, A. Shahroudy, D. Xu, A. C. Kot, and G. Wang, “Skeleton-based action recognition using spatio-temporal LSTM network with trust gates,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 12, pp. 3007–3021, 2017.

[61] J. Liu, G. Wang, L.-Y. Duan, K. Abdiyeva, and A. C. Kot, “Skeleton-based human action recognition with global context-aware attention LSTM networks,” *IEEE Transactions on Image Processing*, vol. 27, no. 4, pp. 1586–1599, 2017.

[62] Q. Ke, M. Bennamoun, S. An, F. Sohel, and F. Boussaid, “Learning clip representations for skeleton-based 3D action recognition,” *IEEE Transactions on Image Processing*, vol. 27, no. 6, pp. 2842–2855, 2018.

[63] K. Papadopoulos, E. Ghorbel, D. Aouada, and B. Ottersten, “Vertex feature encoding and hierarchical temporal modeling in a spatio-temporal graph convolutional network for action recognition,” in *IEEE International Conference on Pattern Recognition*, 2021, pp. 10–15.

[64] C. Caetano, F. Brémond, and W. R. Schwartz, “Skeleton image representation for 3D action recognition based on tree structure and reference joints,” in *SIBGRAPI Conference on Graphics, Patterns and Images*. IEEE, 2019, pp. 16–23.

[65] K. Cheng, Y. Zhang, X. He, W. Chen, J. Cheng, and H. Lu, “Skeleton-based action recognition with shift graph convolutional network,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 183–192.

[66] C. Plizzari, M. Cannici, and M. Matteucci, “Skeleton-based action recognition via spatial and temporal transformer networks,” *Computer Vision and Image Understanding*, vol. 208, p. 103219, 2021.

[67] J. Liu, A. Shahroudy, G. Wang, L.-Y. Duan, and A. K. Chichung, “Skeleton-based online action prediction using scale selection network,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[68] Y. Li, R. Xia, X. Liu, and Q. Huang, “Learning shape-motion representations from geometric algebra spatio-temporal model for skeleton-based action recognition,” in *IEEE International Conference on Multimedia and Expo*, 2019, pp. 1066–1071.

[69] L. Huang, Y. Huang, W. Ouyang, and L. Wang, “Hierarchical graph convolutional network for skeleton-based action recognition,” in *Springer International Conference on Image and Graphics*, 2019, pp. 93–102.

[70] Z. Shao, Y. Li, and H. Zhang, “Learning representations from skeletal self-similarities for cross-view action recognition,” *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 160–174, 2020.

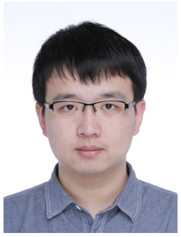
[71] I. Lee, D. Kim, S. Kang, and S. Lee, “Ensemble deep learning for skeleton-based action recognition using temporal sliding LSTM networks,” in *IEEE International Conference on Computer Vision*, 2017, pp. 1012–1020.

Hao Yang received the Ph.D degree from the Institute of Automation, Chinese Academy of Sciences (CASIA), Beijing, China, in 2019. Currently, he is working as an algorithm research fellow in the R&D Center of Artificial Intelligent, Nuctech Company Limited, Beijing, China. His research interests include deep learning, motion analyses and face recognition.

Dan Yan received the graduate degree from the Nanjing University Of Science & Technology (NJUST), Nanjing, China, in 2018. Currently, she is working as an algorithm engineer in the R&D Center of Artificial Intelligent, Nuctech Company Limited, Nanjing, China. Her research interests include action recognition, video structure and neural architecture search.

Li Zhang received her academic degrees from Tsinghua University and China Institute of Atomic Energy. Currently she is a full professor in Department of Engineering Physics, Tsinghua University. She has published more than 120 papers on international journals and conferences and has been granted more than 100 patents worldwide. Her research interests include radiation imaging theory, 3D reconstruction and visual analysis.

Yunda Sun received the Ph.D degree from Beijing Jiaotong University, China in 2006. He has been granted more than 40 patents worldwide. Currently, he is working as a team leader in the R&D Center of Artificial Intelligence, Nuctech Company Limited. His research interests include visual analysis and object recognition.



Dong Li received the bachelor's degree in automation and the master's degree in control science and engineering from Tsinghua University, China, in 2008 and 2011. He worked as a senior software engineer in the Xilinx R&D Center Beijing from 2012 to 2015. Since 2016, he has been a research scientist in the R&D Center of Artificial Intelligence, Nuctech Company Limitied. He is currently also a doctoral student in the D.E. program of Tsinghua University. His research interests include computer vision and efficient representation of deep learning models.

Stephen J. Maybank received the BA degree in mathematics from Kings College Cambridge in 1976, and the Ph.D. degree in computer science from Birkbeck College, University of London in 1988. He is currently a professor emeritus in the Department of Computer Science and Information Systems, Birkbeck College. His research interests include the geometry of multiple images, camera calibration, visual surveillance, etc. He is a fellow of the IEEE.