https://doi.org/10.2352/ISSN.2470-1173.2021.10.IPAS-239

Graph Adversarial Learning for Noisy Skeleton-based Action Recognition

***Henglin Shi**[1†]**, Wei Peng**[1†]**, Xin Liu**[2,1] **and Guoying Zhao**[1∗]*
[1]***Center for Machine Vision and Signal Analysis, University of Oulu, Finland***
[2]***School of Electrical and Information Engineering, Tianjin University, China***
firstname.lastname@oulu.fi
∗ ***corresponding author***
† ***equal contribution***

Abstract

Skeleton based action recognition is playing a critical role in computer vision research, its applications have been widely deployed in many areas. Currently, benefiting from the graph convolutional networks (GCN), the performance of this task is dramatically improved due to the powerful ability of GCN for modeling the Non-Euclidean data. However, most of these works are designed for the clean skeleton data while one unavoidable drawback is such data is usually noisy in reality, since most of such data is obtained using depth camera or even estimated from RGB camera, rather than recorded by the high quality but extremely costly Motion Capture (MoCap) [1] system. Under this circumstance, we propose a novel GCN framework with adversarial training to deal with the noisy skeleton data. With the guiding of the clean data in the semantic level, a reliable graph embedding can be extracted for noisy skeleton data. Besides, a discriminator is introduced such that the feature representation could further improved since it is learned with an adversarial learning fashion. We empirically demonstrate the proposed framework based on two current largest scale skeleton-based action recognition datasets. Comparison results show the superiority of our method when compared to the state-of-the-art methods under the noisy settings.

Introduction

Human action recognition is gaining increasingly more popularity in computer vision research due to its various application in many fields, such as Human Computer Interaction (HCI) and Video Surveillance. Related researches are mainly carried out based on RGB videos and skeletons. RGB video frames are easy to capture, however, the processing is computationally costly. Conversely, skeleton data is more semantic meaningful to represent the movement of human body, but hard to record. For example previously collection of such data relies on the expensive MoCap [1] system. Fortunately, with the emergence of high-speed depth map based skeleton extraction method [2], capturing large scale of skeletons is financially feasible. However, collecting such data still relies on special equipment for capturing depth maps, and such devices are not applicable in outdoor scenarios. Because of this limitation, researchers investigated developed methods for extracting skeletons from RGB frames [3], which provide more convenience for people to utilize such high level data.

Thus, endowed by promising skeleton extract approaches, skeleton data has been widely used by more and more researchers, and a number of skeleton based action recognition methods have been developed in the recent decade. For example hand-craft methods [4, 5, 6]; Recurrent Neural Network (RNN) or Long Short-Term Memory (LSTM) [7] based methods [8, 9]; and Graph Convolutional Network (GCN) [10] based methods [11, 12, 13].

In the meantime, researchers start to face the quality issue of skeleton data, since contemporary skeleton datasets usually contain incomplete or noisy skeletons due to the occlusion or other issues during the data collection [8]. Current popular skeleton datasets are mostly captured using depth maps or RGB frames, which are not as good as data captured by the MoCap system. During the extraction, occlusion emerged in the scene will lead to inaccurate skeleton estimation or even generate incomplete skeleton. As a result, there is a need to investigate methods for handling noisy skeleton data. Liu et al. [8] proposed one of the earliest work which takes the noise of 3D skeleton data into consideration. Song et al. [12] studied the incomplete skeletons for graph convolution network based action recognition. Yu et al. [13] proposed to recover the noisy skeleton from feature level. However, study in this area is still limited which requires more attention from the community.

This work is inspired by [13] which reconstructs noisy skeleton data from feature level using a GCN and a RNN. Moreover, we reconsider the noisy data reconstruction in an adversarial point of view. Thus, we treat the PeGCN from [13] as the generator and merge it into the architecture of starGAN [14]. The generator is trained to generate features based on noisy data that is as close as s extracted from clean data, which can fool the discriminator. The discriminator is trained to distinguish the feature is extracted whether from noisy or clean skeleton data. Under this circumstance, the generator will be trained to extract 'clean' feature based on noisy data.

The main contributions of this paper are as follows. (1) We propose a novel framework which integrates the GCN and GAN architecture for skeleton feature de-noising. (2) Our end-to-end network achieves promising performance on the selected skeleton based action recognition datasets. (3) We empirically demonstrate the proposed method is effective for recovering feature from noisy skeleton data and can improve the performance of action recognition tasks. To the best of the authors' knowledge, this paper is the first work which cooperates GCN and GAN for skeleton de-nosing applications. The paper is organized as follows. Section 2 reviews related works. Section 3 explains the proposed

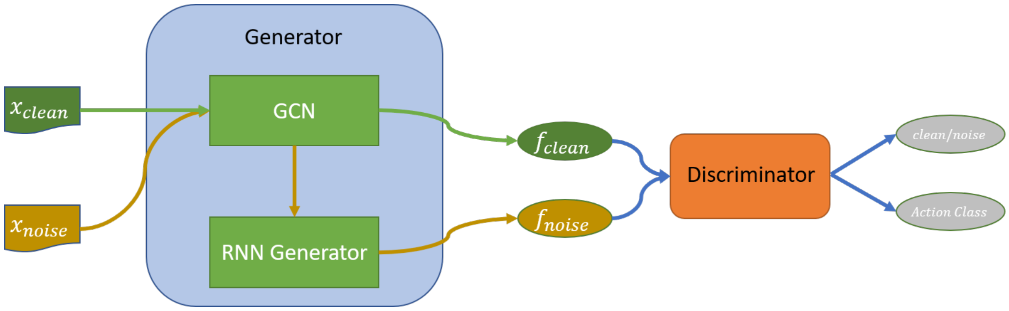

Figure 1. *The architecture of the proposed framework. The propose framework contains a generator and a discriminator. The generator contains a GCN decoder and a RNN generator. The discriminator could be any model but we use fully-connected layer in this work. In the given figure, x_{clean}, x_{noise}, f_{clean}, and f_{noise} represent clean data, noise data. clean feature, and noise feature respectively. The color of arrows indicates the flows of data.*

architecture. Section 4 demonstrates the experiments of the proposed methods on selected datasets. Discussion and conclusion are placed lastly.

Related Works

Action Recognition with Deep Learning Methods

The research of action recognition has achieved promising progress and various methods have been proposed, especially accompanying with the development of deep learning techniques. Based on the modality of the input, these methods mainly can be classified into two categories: video/image oriented and skeleton oriented. Video/image oriented methods mainly relies on Convolution Neural Networks (CNNs) [15, 16, 17, 18]. Skeleton oriented methods are mainly based on the Recurrent Neural Network (RNN) [8, 9, 19] and recently attractive Graph Convolution Network (GCN) [20, 21, 11, 22, 23, 24].

Before the emergence of GCN, analysing skeleton data using CNNs is not convenient since skeleton is unlike images or videos which have grid structures. Early explorations on action recognition using GCNs were proposed in [20], where each frame of human skeleton is reorganized as pseudo image according to the property of kinesiology so that the skeleton can be filtered by convolutional kernels. Moreover, [20] also extended the spatial GCN to temporal domain. However, one drawback of such method is that it only defines the physically connected joints as neighbour to be involved, which perhaps could ignore the patterns between distant joints. Thus, [21] proposed the AS-GCN to tackle this issue by augmenting the spatial filtering. Shi et al. [11] introduced the adaptive graph to extract feature from different layers according to the data, additionally, it also used a two-stream data as input which includes skeletal joints and bones.

Recently, Liu et al. [22] introduced a G3D module for more effective feature learning with graph convolution through spatial and temporal dimensions, and further proposed the Multi-Scale G3D for skeleton action recognition. Zhang et al. [23] proposed the SGN, a light weight model which incorporates the semantic information such as joint type and frame index for skeleton based recognition. SGN has proved that if achieves the state of the art performance with reduced parameter size. Another work which also reduced the heavy computational complexity of GCN based action recognition is shift-GCN [24]. Besides, the proposed shift-GCN also can provide flexible receptive filed in both spatial and temporal domain.

Noisy Action Recognition

The problem of noisy skeleton in action recognition has not been widely investigated, there are only few works have discussed this problem. Liu et al. [8] has investigated the the noise and occlusion in 3D skeleton data, then proposed the Trust-Gate mechanism to evaluate the credibility of the input data and further determine the portion of the input to be accepted by the model. Unlike filtering the possible noisy skeletons, Yu et al. [13] proposed to recover noisy skeleton from feature level using a recurrent module. By constraining the proposed mutual information term the model is trained to recover features from noisy skeleton data.

Generative Adversarial Network

Generative Adversarial Network (GAN) [25] has shown extensive advantages on generative tasks. In addition to the discriminator in ordinary GANs which only discriminate the realness of the generated sample, researchers have tried to add domain classification capability (e.g. classifying the expression of the generated facial image) to it. StarGAN [14] employed a multi-task discriminator which does not only identify the source of the input (real/fake), but also recognize the semantic information of the generated image.

Proposed Method

The architecture of the proposed framework is given in Figure 1, which generally contains a generator G and a multi-task discriminator D which are trained adversarially. The generator tries to generate features based on noisy skeleton which should be close to features generated based on clean data so that it can fool the discriminator. The discriminator is optimized to not only be able to identify the source of input feature (i.e. from clean data or noisy data), but also classify the feature into appropriate action class.

In the following section, We firstly formally define this task. Then, we will explain the generator and discriminator in details. Finally, we will illustrate the optimization objective of the proposed framework.

Problem formalization

In the action recognition from skeleton data task, a skeleton sequence is fed into a computational model, then the learned feature representation is utilized to make an action prediction. Here, each skeleton in the sequence can be modeled by an un-directional graph, which means a skeleton can be represented as $G = (\mathscr{V}, \mathscr{E})$, where $\mathscr{V} = \{v_1, \cdots, v_N\}$ is the node set (skeleton joint) and $\mathscr{E}$ is the edge set (skeleton bone). One can also augment graph with the feature vectors $X = \{x_{v_1}, \cdots, x_{v_N}\}$ where each feature x_{v_i} is associated with the node v_i, since there may also have attributes for each node. Then an adjacent matrix A is included to encode the node connections. Here, we leverage GCNs to embed the graph. Generally, the GCNs take as inputs $X \in \mathscr{X}_N$ and $A \in \mathscr{H}_N$, and learn neural primitives that generate individual node representations by aggregating node feature information across the graph. With the semantic representations for each node, GCNs sum up or average the node embeddings and give a final prediction for the graph. Therefore, the objective of GCNs for our action recognition task is, given a set of training graphs (sequence) with corresponding labels, to learn a function

$$h_\theta : \mathscr{X}_N \times \mathscr{H}_N \rightarrow \mathscr{Y}, \tag{1}$$

of which θ is the learnable parameter of h_θ and h_θ can be implemented by a graph convolutional network. With this function, the graph is mapped to their class label $y \in \mathscr{Y}$. By minimizing an empirical classification loss L_c on the training data, an optimal solution of the function is found. The L_c loss can be

$$L_c = \sum l(h_\theta(X, A), y) + \Omega(\theta), \tag{2}$$

where l is a loss function, e.g. cross-entropy loss, $\Omega(\cdot)$ is a regularization term. In this paper, this function h_θ is based on a spectral GCN approximated by a Chebyshev expansion method [10]. Here, we decouple the h_θ into two parts, the graph feature embedding part G_{gcn} and the classifier part, which can be a fully connected layer. Therefore, for single GCN layer, we have

$$G_{gcn}(X, A) = \sigma(\hat{A} X \theta), \tag{3}$$

where σ is an active function, $\hat{A} = D^{-1/2}(A + I)D^{-1/2}$ is the normalized adjacency matrix with $D_{ii} = 1 + \sum_j A_{ij}$. To involve higher-hops of node connections, one could stack multiple GCN layers.

Note that G_{gcn} in Eq. (3) is only for one skeleton in an action sequence. One needs also introduce a network, e.g., a temporal filter or a RNN to capture the dynamic information. Beside, the G_{gcn} can only extract feature for the clean skeleton data. In the experiment part, one will find that without introducing any other modules, it works far away from satisfaction for the noisy skeletons. Therefore, based on the spectral GCN mentioned here, we build our generative adversarial network to improve the performance of the GCN for noisy skeleton inputs.

Generator

Recent success of GCN has proved its capability on processing skeleton data. In this paper, we adopted the PeGCN from [13] as the generator. The PeGCN contains a Js-AGCN [11] and a RNN which is constructed with two layers of Gated Recurrent Units (GRU). In this paper, we use G_{gcn} to represent of GCN module of the generator, and use G_{rnn} to represent the RNN module.

We define x_{clean} as clean skeleton data, and x_{noise} as skeleton data contains noises. According to Figure 1, the clean feature can be generated by applying the GCN module on clean data, and noisy feature can be generated by applying the RNN module and GCN module on the noisy data consecutively. Each process is describe by equation (4) and (5), respectively.

$$f_{clean} = G_{gcn}(x_{clean}) \tag{4}$$

$$f_{noise} = G_{rnn}[G_{gcn}(x_{noise})] \tag{5}$$

Discriminator

Inspired by StarGAN [14], we also employ a multi-task discriminator. The discriminator consists of two part: a source discriminator D_{src} and an action discriminator D_{cls}. The source discriminator is trained to identify the source of the input feature, i.e. from clean data or noisy data. Moreover, since we plan to train a model which can extract features from noisy data and further for action recognition, so the discriminator should be able to classify action properly. In this work, we use fully-connected layers to implement the the discriminator.

Training Objective

In this work, the generator and discriminator will be optimized iteratively with respect to different losses and intervals. Firstly, we explain the loss of the discriminator. The loss of discriminator contains two part: an adversarial loss term, and a domain classification loss term. The first term measures the whether the discriminator can distinguish the the clean feature f_{clean} and noisy feature f_{noisy}. The *adversarial term* is defined as equation (6) shows. $D_{src}(\cdot)$ measures the likelihood of an arbitrary input to be recognized as clean feature. Thus, the discriminator tends to maximize it, whereas the generator tends to minimize it.

$$L_{adv} = \mathbb{E}_{f_{clean}}[\log D_{src}(f_{clean})] + \mathbb{E}_{f_{noise}}[\log(1 - D_{src}(f_{noise}))] \tag{6}$$

The second term of the discriminator loss is the domain classification term which evaluates loss of action classification. The classification loss based on clean feature f_{clean} and noisy feature f_{noisy} are illustrated as L_{cls}^{clean} and L_{cls}^{noise} in equation (7) and (8), respectively. Training the discriminator require minimize both L_{cls}^{clean} and L_{cls}^{noise}.

$$L_{cls}^{clean} = \mathbb{E}_{f_{clean}, c}[-\log D_{cls}(c|f_{clean})] \tag{7}$$

$$L_{cls}^{noise} = \mathbb{E}_{f_{noise}, c}[-\log D_{cls}(c|f_{noise})] \tag{8}$$

Thus, the full objective of training the discriminator is written as equation (9), in which maximizing L_{adv} is equivalent to minimizing $-L_{adv}$. λ_{cls} represents a hyper-parameter.

$$L_D = -L_{adv} + \lambda_{cls}(L_{cls}^{clean} + L_{cls}^{noise}) \tag{9}$$

The optimization of the generator contains three components: the adversarial term, the domain classification loss with respect to noise features, and a predictive encoding term [13] which is adopted to replace the reconstruct loss in general GANs. We have explained the previous two terms, the predictive encoding term L_{pe} is explained in equation (10), we refer [13] to readers for more details.

$$L_{pe} = -\mathop{\mathbb{E}}_{f_{clean}, f_{noise}}\left[\log \frac{p(f_{clean}, f_{noise})}{\sum_{f_{clean}} p(f_{clean})}\right] \tag{10}$$

As a result, the overall optimizing objective is to minimizing the term below:

$$L_G = L_{adv} + \lambda_{cls} L_{cls}^{noise} + \lambda_{pe} L_{pe}, \tag{11}$$

where λ_{cls} and λ_{pe} are two hyper-parameters.

Experiments

We select the NTU RGB+D [26] and Kinetics-Skeleton [20] datasets, which are the current large scale skeleton datasets, to evaluated the proposed method. In this section, we firstly briefly introduce the two selected datasets. Moreover, we explain the implementation of the experiment including the model specification, hyper-parameter setting, and the preparation of noisy data. Lastly, we present and discuss the experimental result.

Dataset

NTU RGB-D Dateset

The NTU RGB+D dataset [26] is one of the most popular dataset for action recognition in nowadays which collects 56800 action sequences from 40 subjects that are classified into 60 action classes. It is a multi-modality and multi-view dataset. It collects action samples in four modalities, i.e. rgb videos, depth maps, infrared frames, and skeletons of 25 joints. Here, we focus on the skeleton modality. There are totally 56,880 skeleton video clips which are captured from three cameras at different heights with different horizontal angles. The dataset provides two evaluation protocols: (1) cross subject; and (2) cross view. In this paper, we follow both protocols to evaluate the proposed method.

Kinetics Skeleton

The Kinetics Skeleton dataset [20] is the collection of skeletons extracted from the Kinetics video dataset [27] using OpenPose [3]. The original Kinetics dataset is a large-scale action recognition dataset which contains 300,000 action samples of 400 classes. [20] extracts the skeleton of maximally two people from each frame, for each person 18 skeletal joints are extracted. We follow the evaluation protocol provide by the original publication of Kinetics dataset [27] which 240,000 clips are assigned for training, and 20,000 clips are assigned for validation. The final result is reported based on the validation set.

Implementation

For the skeleton data, we unify the inputs such that a uniform GCN architecture is provided for these tasks. We pad the single-actor data with a second body of which the values are all 0. For samples with less than 300 frames, we repeat the samples until it reaches 300 frames.

The GCN architectures for all the experiments are performed on the PyTorch. To keep consistent with the current state-of-the-art methods [20, 11, 28] for clean skeleton data, we stack ten GCN blocks into our GCN. Each GCN block is divided into a spatial and a temporal ones. Spatial GCN block is followed by a temporal convolution with a kernel size 9 × 1, which is used to capture the dynamic information. Then a two layer GRU is utilized for the RNN in the generator. Like PeGCN [13], the last 256 feature maps work as the graph embedding for the skelton sequence. And the feature maps are averaged to a 256 dimensional vector for final class prediction.

The discriminator is implemented by two branches of fully-connected network, where one branch outputs the source of the input feature, the other branch predicts the action class. We train the generator and discriminator iteratively and adversarially. In practice, we optimize the generator once after optimizing the discriminator n times [14], in our experiment we set $n = 5$. The two hyper-parameters mentioned in section 3 are set to $\lambda_{cls} = 3$ and $\lambda_{pe} = 0.2$.

One key thing need to be explained is the preparation of noisy skeletons. We generate the noisy skeleton by following the method in [13]. Firstly, the noise level are defined which indicates the number of joints which will be applied with noises. Moreover, the joint to be modified are randomly selected based on the noise level for each frame. Lastly, random noises are applied on the selected joints. In the training phase, the noise level of involved noisy data is 5. To test the proposed method, we use the data with noise level of 0 (means original data), 1, 3, and 5.

Experimental Results

Result on NTU RGB-D dataset

Table 1 presents the evaluation result on NTU RGB-D dataset with the cross-subject protocol. We compare to one previous best method and eight state-of-the-art methods which are originally for the clean skeletons.Here, the Top-1 score of the selected dataset is reported. It can be seen from Table 1 that our method outperforms all methods on data with noise level 1, 3, and 5. Specifically, when compared to the state-of-the-art methods for clean data, our method can dramatically improve the performance for noisy skeleton. As the noisy level goes up, the superiorty can be even significant. For instance, our method can even outperform the 2S-AGCN [11] by more than 30% with noise level 5, which proves the effectiveness of our method. However, our methods does not achieve the highest result on clean data. The reason could be that our method is trained not only for classifying clean data, but also for recovering and classifying data with noise level 5, which sacrifices the performance of clean data classification.

Table 1. Performance and comparison on NTU RGB-D dataset under the cross-subject protocol. Note, numbers marked as bold format indicates the highest result among each setting.The results of these comparison methods are adopted from [13].

Top 1 Noise level Methods	0	1	3	5
ST-GCN [20]	81.57	73.38	57.76	42.73
Js-AGCN [11]	86.43	76.05	54.92	35.90
Bs-AGCN [11]	87.04	79.08	60.79	44.30
2s-AGCN [11]	**88.83**	84.31	69.40	51.27
Js-AAGCN [29]	87.49	80.31	65.87	51.79
3s RA-GCN[12]	85.87	72.02	45.12	25.59
2S RA-GCN[12]	85.83	71.97	44.41	25.35
PB-GCN [30]	86.98	77.39	56.35	73.31
PeGCN [13]	84.49	84.21	83.28	82.20
Proposed	85.04	**85.04**	**84.41**	**83.76**

Table 2 presents the result of NTU RGB-D dataset under the cross-view protocol. Similar to the result under the cross-subject protocol, our method achieves at least comparable result under the cross-view protocol. When the noisy level is higher, the advantages of our model can be more significant, especially when compared to the state-of-the-art methods for clean data. However, our method does not perform the best result when compared to PeGCN [13]. The most possible reason is that data domain shift caused by the view change can be more harmful for our network. But, still our method can achieves comparative result with the best method.

Table 2. Performance and comparison on NTU RGB-D dataset under the cross-view protocol.The performance of these comparative methods are adopted from [13]. Highest scores are marked with bold font

Top 1 Noise level Methods	0	1	3	5
ST-GCN [20]	88.76	83.14	69.18	54.07
Js-AGCN [11]	94.05	85.98	68.49	51.36
Bs-AGCN [11]	94.12	56.38	7.84	2.44
2s-AGCN [11]	**95.25**	84.12	53.05	29.39
Js-AAGCN [29]	94.61	87.87	71.81	54.37
3s RA-GCN [12]	93.51	79.77	53.59	37.21
2S RA-GCN [12]	92.97	79.58	53.34	32.46
PB-GCN [30]	93.37	80.11	54.21	33.73
PeGCN [13]	93.41	**93.21**	**92.78**	**92.24**
Proposed	92.73	92.68	92.31	91.56

Results on Kinetics Skeleton dataset

The performance on Kinetics skeleton dataset is presented in Table 3. As Table 3 shows, our framework can achieve comparable result with all other methods on clean data. Moreover, on different level of noisy data, our method outperforms the ST-GCN, Js-AGCN, and Js-AAGCN on noisy data with every noise level, and achieves comparative performance with PeGCN.

Conclusion

In this study, we propose a novel framework which integrates a feature de-noising graph convolutional network and the GAN architecture to deal with the human action recognition task for noisy skeleton task. This network is able to learn effective and reliable feature from noisy skeleton data and such that make a dramatically improvement for action classification. The empirical result reveals that our method can provide comparative or even superior performances with selected state-of-the-art methods. Especially, our method outperforms the selected comparing methods on the cross-subject protocol of the NTU RGB-D dataset. Besides, this work also provides a perspective of adopting GAN architecture to process noisy irregular data in the non-Euclidean space with promising performance, which is novel to the community.

Table 3. Performance and comparison on Kinetics skeleton dataset.Performances on noisy data of these comparison methods are adopted from [13].

Top 1 Noise level Methods	0	1	3	5
ST-GCN [20]	31.60	22.42	8.97	3.69
Js-AGCN [11]	34.39	23.06	9.13	3.81
Js-AAGCN [11]	**35.66**	27.13	11.77	4.81
PeGCN [13]	33.78	**33.34**	**32.45**	**30.90**
Proposed	32.65	31.97	31.02	29.51

Acknowledgement

This work was supported by the Academy of Finland for project MiGA (grant 316765), ICT 2023 project (grant 328115), and Infotech Oulu. As well, the authors wish to acknowledge CSC – IT Center for Science, Finland, for computational resources.

References

[1] Meinard Müller, Tido Röder, Michael Clausen, Bernhard Eberhardt, Björn Krüger, and Andreas Weber. Documentation mocap database hdm05. 2007.

[2] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from single depth images. In *CVPR 2011*, pages 1297–1304, 2011.

[3] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Openpose: realtime multi-person 2d pose estimation using part affinity fields. *arXiv preprint arXiv:1812.08008*, 2018.

[4] Xiaodong Yang and Ying Li Tian. Eigenjoints-based action recognition using naive-bayes-nearest-neighbor. In *2012 IEEE computer society conference on computer vision and pattern recognition workshops*, pages 14–19. IEEE, 2012.

[5] Jiang Wang, Zicheng Liu, Ying Wu, and Junsong Yuan. Mining actionlet ensemble for action recognition with depth cameras. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1290–1297. IEEE, 2012.

[6] Xin Liu and Guoying Zhao. 3d skeletal gesture recognition using sparse coding of time-warping invariant riemannian trajectories. *IEEE Transactions on Multimedia*, 2021.

[7] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[8] Jun Liu, Amir Shahroudy, Dong Xu, and Gang Wang. Spatio-temporal lstm with trust gates for 3d human action recognition. In *European conference on computer vision*, pages 816–833. Springer, 2016.

[9] Jun Liu, Gang Wang, Ping Hu, Ling-Yu Duan, and Alex C Kot. Global context-aware attention lstm networks for 3d action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1647–1656, 2017.

[10] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[11] Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. Two-stream adaptive graph convolutional networks for skeleton-based action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 12026–12035, 2019.

[12] Yi-Fan Song, Zhang Zhang, and Liang Wang. Richly activated graph convolutional network for action recognition with incomplete skeletons. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 1–5. IEEE, 2019.

[13] Jongmin Yu, Yongsang Yoon, and Moongu Jeon. Predictively encoded graph convolutional network for noise-robust skeleton-based action recognition. *arXiv preprint arXiv:2003.07514*, 2020.

[14] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8789–8797, 2018.

[15] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497, 2015.

[16] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in neural information processing systems*, pages 568–576, 2014.

[17] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE international conference on computer vision*, pages 6202–6211, 2019.

[18] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6299–6308, 2017.

[19] Yong Du, Wei Wang, and Liang Wang. Hierarchical recurrent neural network for skeleton based action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1110–1118, 2015.

[20] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. *arXiv preprint arXiv:1801.07455*, 2018.

[21] Maosen Li, Siheng Chen, Xu Chen, Ya Zhang, Yanfeng Wang, and Qi Tian. Actional-structural graph convolutional networks for skeleton-based action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3595–3603, 2019.

[22] Xin Liu, Henglin Shi, Xiaopeng Hong, Haoyu Chen, Dacheng Tao, and Guoying Zhao. 3d skeletal gesture recognition via hidden states exploration. *IEEE Transactions on Image Processing*, 29:4583–4597, 2020.

[23] Pengfei Zhang, Cuiling Lan, Wenjun Zeng, Junliang Xing, Jianru Xue, and Nanning Zheng. Semantics-guided neural networks for efficient skeleton-based human action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1112–1121, 2020.

[24] Ke Cheng, Yifan Zhang, Xiangyu He, Weihan Chen, Jian Cheng, and Hanqing Lu. Skeleton-based action recognition with shift graph convolutional network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 183–192, 2020.

[25] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[26] Amir Shahroudy, Jun Liu, Tian-Tsong Ng, and Gang Wang. Ntu rgb+ d: A large scale dataset for 3d human activity analysis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1010–1019, 2016.

[27] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017.

[28] Wei Peng, Xiaopeng Hong, Haoyu Chen, and Guoying Zhao. Learning graph convolutional network for skeleton-based human action recognition by neural searching. *AAAI*, 2020.

[29] Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. Skeleton-based action recognition with multi-stream adaptive graph convolutional networks. *arXiv preprint arXiv:1912.06971*, 2019.

[30] Kalpit Thakkar and PJ Narayanan. Part-based graph convolutional network for action recognition. *arXiv preprint arXiv:1809.04983*, 2018.

Author Biography

Henglin Shi is currently a Ph.D. candidate with the Center for Machine Vision and Signal Analysis, University of Oulu, Finland. He received his B.S. and M.S. degrees in Computer Science and Information Processing Science in 2012 and 2016, respectively. His research interests include machine learning and computer vision based human behavior analysis.

Wei Peng is currently a Ph.D. candidate with the Center for Machine Vision and Signal Analysis, University of Oulu, Finland. He received the M.S. degree in computer science from the Xiamen University, Xiamen, China, in 2016. His articles have published in mainstream conferences and journals, such as AAAI, ICCV, ACM Multimedia, Transactions on Image Processing. His current research interests include machine learning, affective computing, medical imaging, and human action analysis.

Xin Liu is currently an Associate Professor with the School of Electrical and Information Engineering, Tianjin University, China. He is also a senior researcher with the Center for Machine Vision and Signal Analysis, University of Oulu, Finland. He received his Ph.D. degree in Computer Science and Engineering in 2019. His research interests include human behavior analysis, 3D computer vision, image restoration, and object detection.

Guoying Zhao (IEEE Senior member, IAPR Fellow) is currently a Professor with the Center for Machine Vision and Signal Analysis, University of Oulu. She has authored or co-authored more than 240 papers in journals and conferences. She was co-publicity chair for FG2018. Now she is co-program chair for ICMI 2021, and associate editor for several journals. Her current research interests include image and video descriptors, facial-expression and micro-expression recognition, gait analysis, dynamic-texture recognition, human motion analysis, and person identification.