GHand: A Graph Convolution Network for 3D Hand Pose Estimation

Pengsheng Wang[1], Guangtao Xue[1], Ping Li[2], Jinman Kim[3], Bin Sheng[1], and Lijuan Mao[4]

[1] Shanghai Jiao Tong University, Shanghai, People's Republic of China
shengbin@sjtu.edu.cn
[2] The Hong Kong Polytechnic University, Hong Kong, People's Republic of China
[3] The University of Sydney, Sydney, Australia
[4] Shanghai University of Sport, Shanghai, People's Republic of China
maolijuan@sus.edu.cn

Abstract. Vision-based 3D hand pose estimation plays an important role in the field of human-computer interaction. In recent years, with the development of convolutional neural networks (CNN), the field of 3D hand pose estimation has made a great progress, but there is still a long way to go before the problem is solved. Although recent studies based on CNN networks have greatly improved the recognition accuracy, they usually only pay attention on the regression ability of the network itself, and ignore the structural information of the hands, thus leads to a low accuracy in contrast. In this paper we proposed a new hand pose estimation network, which can fully learn the structural information of hands through an adaptive graph convolutional neural network. The experiment on the public dataset shows the accuracy of our graph convolution network exceeds the SOTA methods in 3D hand pose estimation.

Keywords: 3D hand pose estimation · Adaptive graph convolution · Depth image

1 Introduction

In natural society, hand pose plays an important role when we communicate with each other. It is widely used in the field of AR/VR and human-computer interaction due to its rich expressive ability and comfortable and convenient expression [5,18,25]. With the popularity of depth cameras, depth-image-based hand pose estimation gains more attention and is one of the hottest topic in hand pose estimation [9,22]. Recently, with the development of convolutional neural networks, great progress has been made in this field, but it is still a changeling problem because of the large variations in hand orientations, high flexibility, and severe self-occlusion.

In previous researches, convolution neural network is usually used to regress the 3D coordinates of hand pose joints directly, thus the dependencies between the joints was ignored, which will result in low accuracy and deformed gestures.

CNN has been proved successful in tackling grid-like structure data and RNN in sequence data, but many tasks, e.g. social networks, molecular structures, can only be represented in a form of graph-structure data. To overcome their limitations, recently graph convolution neural networks were introduced to process the graph-structure data due to its effective representations. In this paper we proposed a GCN to regress the 3D co-ordinates of hand joints from a depth image in an end-to-end way. Our main contributions can be summarized as follow:

- We proposed a Graph Convolution Network (GCN) for 3D hand pose estimation, GHand, which can regress the 3D coordinates from a depth image in an end-to-end way.
- For the first time we recommend an adaptive adjacent matrix to learn the structural information of the hands, thus the dependencies between the different joints can be fully exploited.
- Through self-comparison experiment in public dataset, we show that the GCN can significantly improve the accuracy of the network. We also compared our approach with other state-of-the-art models and our approach has a better performance.

2 Related Work

3D hand pose estimation is a hot topic in computer vision, because of its wide use in many scenes. We refer to [3] for an overview of the previous works, and they can be divided into two types of approach. One is based on RGB images and another is based on depth image. With the popularity of depth cameras, depth-image based methods gain more attention. In this paper, we focus on the 3D coordinates regression from a single depth image.

2.1 3D Hand Pose Estimation

Structure information of the hand has proven helpful when predicting the 3D position of the hand joints [22]. In order to utilize the structural properties of hands, many methods have been proposed [10,11]. The main ideas of them can be divided into two types. One way is to treat the structure information as a prior [14,16]. Calculate a prior through the PCA method and directly add it to the convolutional neural network model. [23] designed some handcraft constraints and put them into the loss function. Although these methods can improve the recognition accuracy to a certain extent, handcraft prior of the structure information will also damage the learning ability of the model and thus will reduce the representation ability of the model. Another way is to design a branch network which is similar with the hand structure. [24] designed a three-branch network, where the three branches correspond to the thumb, index finger, and the three other fingers, according to the differences in the functional importance of different fingers. These studies have demonstrated that handling different parts of the hand via a multi-branch CNN can improve the accuracy of 3D hand pose

estimation. However, not all dependencies between joints are taken into account. To capture the better structure information of hands, we adopt a GCN to model the hand structure in a learnable way.

2.2 Graph Convolution Method

Graph convolution networks allow learning high-level representations of the relationships between the nodes of graph-based data. [8] used graph convolutional networks for skeleton-based action recognition. [4] designed an adaptive graph convolution network to regress the 3D position of the gesture-object key-points from an RGB image. In this paper we adopt an adaptive convolutional network to model the structural information of hands, and regress the 3D coordinates of the hand joints in an end-to-end way.

The model starts with 4 Residual Blocks as the backbone network to extract the image feature vector and predict the initial 3D coordinates. The coordinates concatenated with the image features vector used as the features of the input graph of a 2-layered graph convolution to exploit the structure information of hands to estimate the better 3D pose.

3 Methodology

3.1 Overall Network Architecture

Figure 1 illustrates the overall architecture of the proposed GCN-based 3D hand pose estimation methods. The proposed network mainly consists of two parts: a backbone convolution neural network to extract the features from an input depth image; a joint regression Graph Convolution Network, which consists two graph convolution layers, to regress the 3D coordinates of the hand pose joints. The input depth image is firstly fed into backbone network for features extraction and initial 3D coordinates regression of the hand pose joints. Then, the GCN take the obtained features and the initial 3D coordinates from the backbone network as an input graph and predict the final 3D coordinates of the hand pose joints.

3.2 Backbone Network

The backbone network of the proposed 3D hand pose estimation method refers to [21], as described in Table 1. Different with [21], our backbone network has only four residual blocks to extract the feature maps from the input depth image. Each of the residual blocks consists of two 3×3 convolutional layer and a 1×1 convolutional layer instead of the identity skip connection when the output dimension of the residual block is increased. Max-pooling layers for down-sampling are appended after each residual block except for the last one block. Following the last block, a global max pooling layer is used to convert the feature maps to a 256D feature vector. Then a fully connection layer regresses the initial 3D

coordinates of the hand pose joints. Inspired by the architecture of [12], we concatenate these features with the initial 3D predictions of each joint, yielding a graph with 259 features (256 image features plus initial estimates of x, y and z) for each node in sub-net GCN.

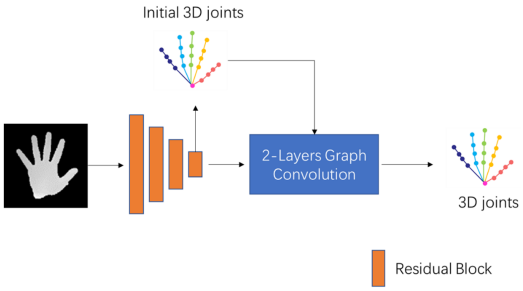

Fig. 1. The overall architecture of the proposed method for 3D hand pose estimation.

Table 1. Detailed architecture of the backbone network for feature vector extraction.

Layers	Kernel size	Channels	Output size
Residual block	3×3	64	96×96
Max pooling	2×2	64	48×48
Residual block	3×3	64	48×48
Max pooling	2×2	64	24×24
Residual block	3×3	128	24×24
Max pooling	2×2	128	12×12
Residual block	3×3	256	12×12
Global average pooling	–	–	256
Fully connection	–	–	42

3.3 GCN Network

Our GCN network consist of two layers graph convolution, which is inspired by [4]: the output features of a graph convolution layer for an input graph with N nodes, k input features, and l output features for each node is computed as,

$$Y = \sigma(\widetilde{A}XW) \tag{1}$$

where σ is the activation function, $W \in R^{k \times l}$ is the trainable weights matrix, $X \in R^{N \times k}$ is the matrix of input features, and $A \in R^{N \times N}$ is the row-normalized adjacency matrix of the graph,

$$\widetilde{A} = \hat{D}^{-\frac{1}{2}} \hat{A} \hat{D}^{-\frac{1}{2}} \tag{2}$$

where $\hat{A} = A + I$ and $\hat{D}$ is the diagonal node degree matrix. $\hat{A}$ simply defines the extent to which each node uses other nodes' features. So $\hat{A}X$ is the new feature matrix in which each node's features are the averaged features of the node itself and its adjacent nodes.

In order to learn the structure info between each joint, inspired by [4], we also use a learnable adjacency matrix (A) in our graph convolution layer. This approach allows us to fully exploit the dependencies between different joints on different fingers or on same finger.

3.4 Loss Function

Our loss function for training the model has two parts. The first part is the loss for the initial 3D coordinates predicted by Backbone network (L_{init3D}). The other is the loss that calculated from the final 3D coordinates (L_{3D}),

$$L = \alpha L_{init3D} + L_{3D} \tag{3}$$

4 Experiment Results

4.1 Dataset and Evaluate Metrics

We implemented our experiment on the popular public dataset: NYU dataset [20], which was captured with three Microsoft Kinects and contains 72k training and 8k testing images from three different views. The training set was collected from one subject, while the testing set was collected from two subjects. To evaluate the performance of the different 3D hand pose estimation methods, we used two metrics. The first metric is the average 3D distance error between the ground truth and predicted 3D position for each joint. The second one is the percent-age of succeeded frames whose errors for all joints are within a threshold which is the same as [19].

4.2 Self-comparisons

To analyze the function of the GCN, we trained a network without the GCN. The 3D coordinates are directly regressed from the feature vector which extracted by the backbone network from the input of a single depth image. As shown in Fig. 2, the proposal GCN can significantly improve the performance of the network.

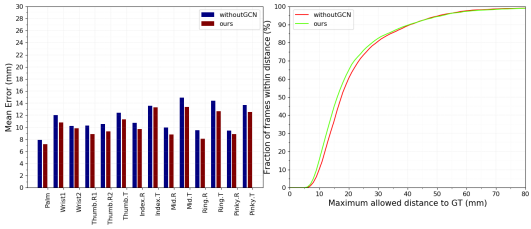

Fig. 2. Self-comparison results. Left: 3D distance errors (mm) per hand joint. Right: percentage of success frames over different error thresholds.

4.3 Comparison with State-of-the-Art Methods

We compared our proposed network on NYU datasets with the most recently proposed methods, including DeepPrior [14], its improved version Deep Prior++ [16], Feed-back [15], REN-9 × 6 × 6 [7], Pose-REN [2], Generalized [17] and DeepModel [23], as well as methods using 3D inputs, includes 3D CNN [6], SHPR-Net [1]. As show in Table 2 and Fig. 3, our results outperform the results of all the state-of-the-art methods no matter whose input is 2D depth map or 3D points cloud.

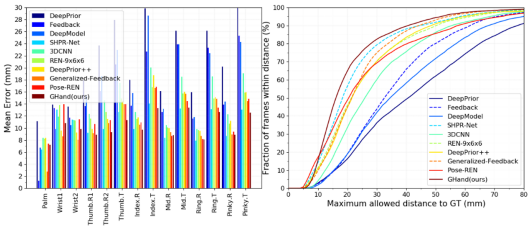

Fig. 3. Compare with other state-of-the-art methods.

Table 2. Comparison with state-of-the-art methods on NYU dataset.

Method	Mean error(mm)	Inputs
DeepPrior [14]	20.75	2D
Deep Prior++ [16]	12.23	2D
Feedback [15]	15.97	2D
$REN-9\times6\times6$ [7]	12.69	2D
Pose-REN [2]	11.81	2D
Generalized [17]	10.89	2D
DeepModel [16]	17.03	2D
GHand (ours)	**10.33**	**2D**
3D CNN [6]	14.11	3D
SHPR-Net [1]	10.77	3D

5 Conclusion

In this paper, we introduced a Graph Convolution Network for the 3D hand pose estimation from a single depth image. We have experimentally shown that the pro-posed approach outperforms the state-of-the art on the publicly available dataset: NYU hand dataset.

Through experiments on the public dataset, it is shown that Graph Convolution Network works effectively to exploit the structure information of hands and can improve the accuracy of the prediction. For the future work, GCN based classification can be integrated into this network for the classification of the gestures, thus the whole framework can be used for the applications, such as driving control, UAV control.

Acknowledgments. This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFF0300903, in part by the National Natural Science Foundation of China under Grant 61872241 and Grant 61572316, and in part by the Science and Technology Commission of Shanghai Municipality under Grant 15490503200, Grant 18410750700, Grant 17411952600, and Grant 16DZ0501100.

References

1. Chen, X., Wang, G., Zhang, C., Kim, T., Ji, X.: SHPR-Net: deep semantic hand pose regression from point clouds. IEEE Access **6**, 43425–43439 (2018)
2. Chen, X., Wang, G., Guo, H., Zhang, C.: Pose guided structured region ensemble network for cascaded hand pose estimation. Neurocomputing **395**, 138–149 (2020)
3. Doosti, B.: Hand pose estimation: a survey. Computer Vision and Pattern Recognition (2019)
4. Doosti, B., Naha, S., Mirbagheri, M., Crandall, D.: HOPE-Net: a graph-based model for hand-object pose estimation. arxiv.org/abs/2004.00060 (2020)
5. Erol, A., Bebis, G., Nicolescu, M., Boyle, R.D., Twombly, X.: Vision-based hand pose estimation: a review. Comput. Vis. Image Underst. **108**(1), 52–73 (2007)
6. Ge, L., Liang, H., Yuan, J.: 3D Convolutional Neural Networks for Efficient and Robust Hand Pose Estimation from Single Depth Images. Supplementary Material. 1–5 (n.d.)
7. Guo, H., Wang, G., Chen, X., Zhang, C., Qiao, F., Yang, H.: Region ensemble network: improving convolutional network for hand pose estimation. arxiv.org/abs/1702.02447 (2017)
8. Li, M., Chen, S., Chen, X., Zhang, Y., Wang, Y., Tian, Q.: Actional-structural graph convolutional networks for skeleton-based action recognition. In: Computer Vision and Pattern Recognition, pp. 3595–3603 (2019)
9. Lu, P., Sheng, B., Luo, S., Jia, X., Wu, W.: Image-based non-photorealistic rendering for realtime virtual sculpting. Multimedia Tools Appl. **74**(21), 9697–9714 (2014). https://doi.org/10.1007/s11042-014-2146-4
10. Kamel, A., Sheng, B., Yang, P., Li, P., Shen, R., Feng, D.D.: Deep convolutional neural networks for human action recognition using depth maps and postures. IEEE Trans. Syst. Man Cybern. Syst. **49**(9), 1806–1819 (2019)

11. Karambakhsh, A., Kamel, A., Sheng, B., Li, P., Yang, P., Feng, D.D.: Deep gesture interaction for augmented anatomy learning. Int. J. Inf. Manage. **45**, 328–336 (2019)
12. Kolotouros, N., Pavlakos, G., Daniilidis, K.: Convolutional mesh regression for single-image human shape reconstruction. In: Computer Vision and Pattern Recognition, pp. 4496–4505 (2019)
13. Meng, X., et al.: A video information driven football recommendation system. Comput. Electr. Eng. **85**, 106699 (2020)
14. Oberweger, M., Wohlhart, P., Lepetit, V.: Hands deep in deep learning for hand pose estimation. arxiv.org/abs/1502.06807 (2015)
15. Oberweger, M., Wohlhart, P., Lepetit, V.: Training a feedback loop for hand pose estimation. In: International Conference on Computer Vision, pp. 3316–3324 (2015)
16. Oberweger, M., Lepetit, V.: DeepPrior++: improving fast and accurate 3D hand pose estimation. In: International Conference on Computer Vision Workshops (ICCVW), pp. 585–594 (2017)
17. Oberweger, M., Wohlhart, P., Lepetit, V.: Generalized feedback loop for joint hand-object pose estimation. Pattern Anal. Mach. Intell. **45**, 1898–1912 (2020)
18. Sheng, B., Li, P., Zhang, Y., Mao, L.: GreenSea: visual soccer analysis using broad learning system. IEEE Trans. Cybern. 1–15 (2020)
19. Taylor, J., Shotton, J., Sharp, T., Fitzgibbon, A.: The Vitruvian manifold: inferring dense correspondences for one-shot human pose estimation. In: Computer Vision and Pattern Recognition, pp. 103–110 (2012)
20. Tompson, J., Stein, M., Lecun, Y., Perlin, K.: Real-time continuous pose recovery of human hands using convolutional networks. ACM Trans. Graph. (ToG) **33**(5), 169 (2014)
21. Yoo, C., Kim, S., Ji, S., Shin, Y., Ko, S.: Capturing hand articulations using recurrent neural network for 3D hand pose estimation. arxiv.org/abs/1911.07424 (2019)
22. Yuan, S., et al.: Depth-based 3D hand pose estimation: from current achievements to future goals. In: Computer Vision and Pattern Recognition, pp. 2636–2645 (2018)
23. Zhou, X., Wan, Q., Wei, Z., Xue, X., Wei, Y.: Model-based deep hand pose estimation. In: International Joint Conference on Artificial Intelligence, pp. 2421–2427 (2016)
24. Zhou, Y., Lu, J., Du, K., Lin, X., Sun, Y., Ma, X.: HBE: hand branch ensemble network for real-time 3D hand pose estimation. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) Computer Vision – ECCV 2018. LNCS, vol. 11218, pp. 521–536. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01264-9_31
25. Zhang, P., Zheng, L., Jiang, Y., Mao, L., Li, Z., Sheng, B.: Tracking soccer players using spatio-temporal context learning under multiple views. Multimedia Tools Appl. **77**(15), 18935–18955 (2018)