STPOINTGCN: SPATIAL TEMPORAL GRAPH CONVOLUTIONAL NETWORK FOR MULTIPLE PEOPLE RECOGNITION USING MILLIMETER-WAVE RADAR

*Chunyu Wang[1,3], Peixian Gong[1], Lihua Zhang[*1,2,4]*

[1]Academy for Engineering and Technology, Fudan University, China
[2]Ji Hua Laboratory, Foshan, China
[3]Engineering Research Center of AI and Robotics, Ministry of Education, China
[4]Engineering Research Center of AI and Unmanned Vehicle Systems of Jilin Province, China

ABSTRACT

Gait recognition is a new biometric technology, which aims to identify people by their walking posture. Compared with fingerprint recognition, face recognition and other technologies, gait recognition usually has the characteristics of long-distance non-contact and difficulty in camouflage. And compared with the camera-based method, using millimeter-wave radar for gait recognition is immune to light and weather conditions. Moreover, due to the non-invasive feature of millimeter-wave radar, we can design products without privacy risk. In this paper, we propose an end-to-end STPointGCN structure, which can extract and aggregate the features of sparse point clouds collected by millimeter-wave radar from the dimensions of space and time. In order to verify our method, we collect and disclose our own gait recognition dataset based on millimeter-wave radar. After comparing with the existing mainstream algorithms, we find that our method is superior to the existing mainstream methods for single-person scenarios and multi-person co-existing scenarios.

***Index Terms*—** Millimeter-wave radar, graph neural network (GNN), gait recognition

1. INTRODUCTION

Gait recognition is a biometric recognition method. Each person's gait has its unique feature, so gait recognition can be achieved. And because human gait has characteristics of long-distance recognition and difficulty in camouflage, it has great potential and wide application prospects, such as monitoring system [1], criminal investigation [2], evidence collection, etc..

At present, there are many methods to achieve gait recognition, which are mainly divided into two categories: *(i)* Gait recognition methods based on computer vision using image or video [3, 4]; *(ii)* Gait recognition methods based on sensor system, such as inertial sensor [5, 6], pressure sensor [7] and millimeter-wave radar [8, 9, 10, 11]. Millimeter-wave radar can not only be used in the all-weather environment, but also protect human private information. Moreover, it can provide accurate 3D information which is able to overcome low-light conditions and body shade especially for multi-person scenarios. Because of millimeter-wave radar's advantages of robustness, high privacy and penetrability, it has attracted extensive attention of researchers.

MID [11] is the first work to use millimeter-wave radar for multiple people gait recognition, and the mmGait dataset proposed in work [10] is the first published gait dataset based on millimeter-wave radar. MmGait uses two millimeter-wave radars in single-person scenarios and multi-person scenarios and uses CNN to achieve gait recognition. For multi-person scenarios, DBSCAN clustering algorithm [12] is used to segment point cloud clusters, and the Hungarian algorithm [13] is used to track the point cloud clusters of each person's route. Different from the multi-stage design of mmGait and mID in multi-person scenarios, our proposed model is an end-to-end algorithm. In order to verify the advancement of our algorithm, we compare it with existing algorithms on mmGait dataset. At the same time, we also use millimeter-wave radar to collect gait data attached RBG images, then evaluate the performance of our model.

Compared with mmGait using CNN for point cloud feature extraction, we design STPointGCN structure to extract the feature of sparse point cloud in space and time dimensions. Graph is a concise, abstract and intuitive mathematical expression of objects and their relation. The concept of GNN was first proposed in work [14] and further clarified in work [15] as well as graph convolutional networks (GCN) [16], which has been continual developed and improved recently. Although the point cloud collected by millimeter-wave radar is unstructured and non-grid data, a lot of previous works are still based on CNNs [10, 17, 18]. GNNs also have good feature extraction ability for non-grid domain data. PointGNN [19] is the first algorithm to process large-scale point cloud by GNN, which has inspired subsequent research. Especially

This work is supported in part by Shanghai Municipal Science and Technology Major Project (2021SHZDZX0103)

in work [20], the researchers use GNN to process the sparse point cloud collected by millimeter-wave radar, which shows the good performance of GNN in millimeter-wave radar related tasks.

In this paper, our STPointGCN achieves the best performance on mmGait dataset and our own dataset. The contributions of this paper lie in three aspects:

1. We propose an end-to-end model for co-existing multiple people recognition using a millimeter-wave radar.
2. We design the spatial-temporal graph convolutional network (STPointGCN) for feature extraction from point clouds.
3. We collect and publish a multiple people co-existing dataset generated by millimeter-wave radar.

2. METHODOLOGY

In this section, we elaborate on the proposed model STPointGCN for multiple people gait recognition.

2.1. Graph Construction

As our proposed model's input is a time sequence of point clouds, we consider all the points from T frames as a larger graph, and then define edges on it. Formally, we define a total of N points consisting of all T frames as a set of $P = \{p_1, p_2, \ldots, p_N\}$, where $p_i = (t_i, x_i, s_i)$ denotes a point with time index $t \in \mathbb{N}$, 3D coordinates $x_i \in \mathbb{R}^3$, and state features $s_i \in \mathbb{R}^m$. s_i is a m-length vector representing the point property such as velocity, snr, etc.. Given a point cloud P consisting of all T frames, we construct a graph $G = (P, E^s, E^t)$ with both spatial edges E^s and temporal edges E^t using P as vertices and define edges as:

$$\begin{aligned} E^s =& \{(p_i, p_j) \mid \|x_i - x_j\|_2 < r^s \ and \ t_i = t_j\} \\ E^t =& E^{t,+1} \cup E^{t,-1} \end{aligned} \tag{1}$$

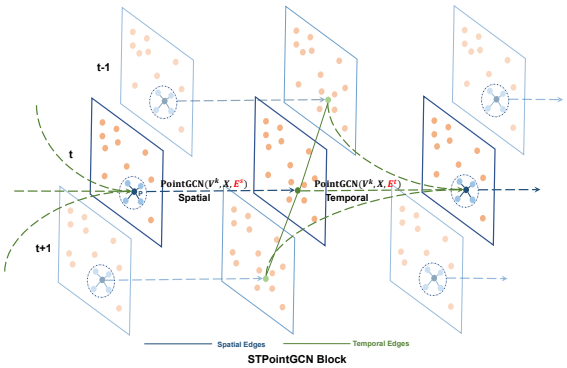
where,

$$\begin{aligned} E^{t,-1} =& \{(i,j) \mid \|x_i - x_j\|_2 < r^t \\ & and \ j = \arg\min_j \|x_i - x_j\|_2 \ and \ t_j = t_i - 1\} \\ E^{t,+1} =& \{(i,j) \mid \|x_i - x_j\|_2 < r^t \\ & and \ j = \arg\min_j \|x_i - x_j\|_2 \ and \ t_j = t_i + 1\} \end{aligned} \tag{2}$$

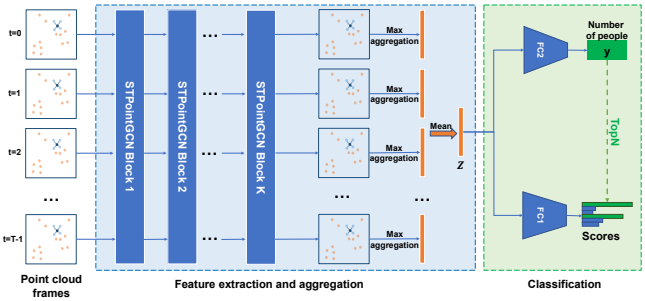
In the spatial dimension, each point is connected to its neighbors within a fixed radius r^s in a certain frame t. While in the temporal dimension, each point p_i is connected to its closest neighbor p_j within a fixed radius r^t, and p_j belongs to the previous frame $t-1$ or next frame $t+1$.

2.2. STPointGCN

Generalizing the convolution operator to irregular domains such as graph can be typically expressed as a neighborhood



(a)



(b)

Fig. 1: (a) STPointGCN block structure; (b) Architecture of proposed model STPointGCN for multi-person gait recognition.

aggregation scheme[19]:

$$\begin{aligned} v_i^{k+1} &= g^k\left(\rho\left(\left\{e_{ji}^k \mid (j,i) \in E\right\}\right), v_i^k\right) \\ e_{ji}^k &= f^k\left(v_i^k, v_j^k\right) \end{aligned} \tag{3}$$

where $v_i^k \in \mathbb{R}^F$ denotes vertex features of node i in layer k, $e_{ji} \in \mathbb{R}^D$ denotes edge features from node j to node i, $\rho(\cdot)$ denotes a differentiable, permutation invariant function, e.g., sum, mean or max which aggregates the edges features for each vertex, and $f(\cdot)$ and $g(\cdot)$ denote differentiable functions such as MLPs (Multi Layer Perceptrons) for vertex and edge features embedding.

In order to adapt to the characteristics of point clouds, we use point state property s_i to initialize vertex feature v_i^0 at the first layer. And in each iteration, relative coordinate from vertex j to i concatenating with original vertex feature v_j constitutes edge feature e_{ji}, and then use MLP to update the edge feature:

$$e_{ji}^k = \mathrm{MLP}_f^k\left(\mathrm{cat}\left(x_i - x_j, v_j^k\right)\right) \tag{4}$$

After updating edges features, we adopt max aggregation to aggregate edge features to vertex feature and then update it by MLP with residual connection:

$$v_i^{k+1} = \mathrm{MLP}_g^k\left(\mathrm{Max}\left(\left\{e_{ji}^k \mid (j,i) \in E\right\}\right)\right) + v_i^k \tag{5}$$

Different from PointGNN [19], we didn't adopt the *auto-registration* mechanism. On the other hand, we add self-loops

to adjacency edges, i.e., $E \coloneqq E \cup \{(i,i) \mid p_i \in P\}$. If the feature dimension of v_i^{k+1} is different from v_i^k, a linear layer is added additionally on the shortcut to ensure the feature residual connection. In order to distinguish our method from PointGNN, this iteration layer is named PointGCN. Its inputs consist of 3D coordinates X, vertices features V^k and edges E, and output is newly updated vertices features V^{k+1}:

$$V^{k+1} \coloneqq \mathrm{PointGCN}(V^k, X, E) \tag{6}$$

For time series of point clouds, we consider all the points from T frames as one graph and generate spatial edges E^s as well as temporal edges E^t as described in Sec.2.1. Our proposed STPointGCN block consists of two phases:

$$\begin{aligned} V^{k+1} &\coloneqq \mathrm{PointGCN}(V^k, X, E^s) \\ V^{k+1} &\coloneqq \mathrm{PointGCN}(V^{k+1}, X, E^t) \end{aligned} \tag{7}$$

In a STPointGCN block, we start with vertices features embedding and aggregation on spatial dimension using spatial edges E^s, and then on temporal dimension using temporal edges E^t, as shown in Fig.1(a) and Formula (7).

After K layers of STPointGCN blocks, we use max aggregation within each frame followed by mean aggregation among frames to get feature vector Z, as shown in Fig.1(b). Subsequently, we have two branches for the number of people prediction and classification.

The classification branch FC1 computes a multi-label probability $[p_1, p_2, \ldots, p_C]$ for co-existing multiple people recognition, where C is the total number of people in the dataset. The activation function of the last fully connected layer is sigmoid function so that we can use C binary classifiers to predict each person's presence.

The prediction branch FC2 regresses the number of people in the point clouds. The activation function of the last fully connected layer is ReLU function to guarantee the nonnegativity of output y. In practice, we take $\mathrm{round}(y)$ as the number of people $\tilde{y}$. If it is greater than C it will be set to C. Thus we don't have to manually set the probability threshold. The indices of top $\tilde{y}$ largest probability among $[p_1, p_2, \ldots, p_C]$ are the final classification result.

2.3. Loss

For multiple people recognition, the ground truth is encoded as $[l_1, l_2, \ldots, l_C]$ for each sample, where $l_i = 1$ if i^{th} person is present in point clouds otherwise $l_i = 0$. For a total of M samples, we adopt binary cross entropy loss as the classification loss:

$$\begin{aligned} \mathcal{L}_{cls} = -\frac{1}{M} \sum_{i=1}^{M} \sum_{j=1}^{C} \big(& l_j^{(i)} \log(p_j^{(i)}) \\ & + (1 - l_j^{(i)}) \log(1 - p_j^{(i)}) \big) \end{aligned} \tag{8}$$

For the number of people prediction, we adopt mean square error as the regression loss:

$$\mathcal{L}_{reg} = -\frac{1}{M} \sum_{i=1}^{M} (y^{(i)} - \hat{y}^{(i)})^2 \tag{9}$$

where $\hat{y}^{(i)}$ is the ground truth number of people in i^{th} sample. Our final loss function is the weighted sum of $\mathcal{L}_{cls}$ and $\mathcal{L}_{reg}$.

$$\mathcal{L} = \mathcal{L}_{cls} + \lambda \mathcal{L}_{reg} \tag{10}$$

3. EXPERIMENT AND RESULT

3.1. Experiment Setup and Data Collection

We deployed an easy-to-use commercial FMCW mmWave sensor IWR6843ISK which works in the 60-GHz to 64-GHz frequency range produced by Texas Instruments (TI) as shown in Fig.2. And it contains 4 receive (RX) 3 transmit (TX) antenna with 120° azimuth field of view (FoV) and 30° elevation FoV. These antennas are about 108 cm above the ground. The mmWave radar parameters used in our experiment are listed in Table 1. It tracked moving targets and detected its point clouds data then transmitted the data to our computer program through serial ports. Images captured by the RGB camera were also saved synchronously with the point clouds for 10 fps for observation and future research.

Table 1: mmWave Radar Waveform Configuration

Chirp Parameter (Units)	Value
Start Frequency (GHz)	60.6
Slope (MHz/us)	54.725
Samples per chirp	96
Chirps per frame	288
Frame duration (ms)	50
Sampling rate (Msps)	2.950
Bandwidth (MHz)	2249
Range resolution (m)	0.084
Max Unambiguous Range (m)	7.2
Max Radial Velocity (m/s)	8.38
Velocity resolution (m/s)	0.17
Azimuth resolution (deg)	14.5
Elevation resolution (deg)	58

Finally, we collected a total of 75 minutes of 3D point clouds data from 6 participants in three different environments: living room, hallway, and meeting room[1]. Participants were asked to walk for 3 to 5 minutes in single and multiple people scenarios. Each time sequence of point cloud data is labeled by a set of person ids. A series of collected data is shown in Fig.3.

3.2. Implementation Details

Our collected data was split into training and testing set with the ratio 80% and 20%. A sliding widow with window length

[1]https://github.com/FmmW-Group/STPointGCN_Dataset

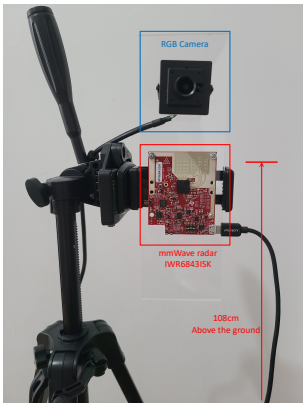


Fig. 2: Devices setup for recognizing human gait.

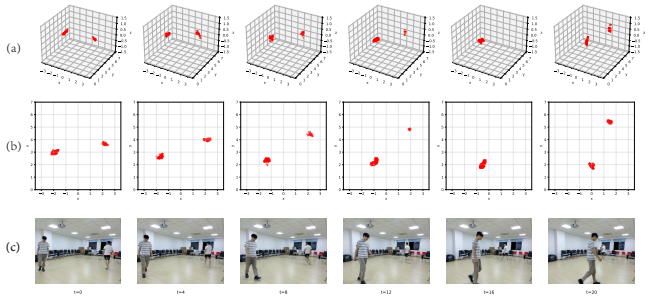


Fig. 3: Collected data visualization. There are 2 co-existing persons(id = 1,4) walking in this scenario. (a) 3D point cloud; (b) 2D point cloud on the xy plane; (c) RGB images.

T=20 and moving step s=20(T=30 and s=30 for mmGaitNet) was used to generate data which can adapt the input of the model. We used *z-score* to normalize the initial state vector s_i which consists of velocity and snr attributes. The neighborhood radius is $r^s = 0.3m, r^t = 0.3m$ when constructing graphs from point clouds frames.

Our proposed model used 7 layers of STPointGCN blocks, and the feature dimensions of each block are respectively [16,64,64,64,64,128,128]. We used batch normalization followed by LeakyReLU activation functions after each linear operation in MLPs. And we set the loss weight parameter $\lambda = 1$. We implemented our network in PyTorch and the optimizer is Adam with initial learning rate 1e-3.

3.3. Performance Evaluation and Discussion

We evaluated different benchmark algorithms as the backbone network including Pointnet [21] combined with LSTM(PL), Pointnet without T-net combined with LSTM(P-L), Pointnet++ [22] combined with LSTM(P+L), DR [10], Point-GNN [19] combined with LSTM(PGL) and mmGaitNet [10] on mmGait dataset and our own collected dataset for co-existing multiple people recognition.

Table 2 reports the accuracy obtained by benchmark algorithms and our proposed method STPointGCN on mmGaitNet (free-route) dataset for multiple people recognition. We have a 15% improvement over mmGaitNet method. MmGaitNet is difficult to separate each individual's point cloud by clustering method and establish associations among frames in the scenario of multiple people walking freely.

Table 3 reports the test performance of different benchmark algorithms and our proposed method STPointGCN on our own dataset. We use mean precision, mean recall and mean f1-score over all persons as the metrics because they are better suited in end-to-end multiple people recognition scenarios. The f1-score of STPointGCN outperforms other algorithms under both single-person scenarios and multi-person co-existing scenarios.

Compared to existing methods, STPointGCN is able to establish associations within point cloud frames of each individual in both spatial and temporal dimensions using spatial edges and temporal edges. Subsequently, the features of point cloud grouped by people are updated using message-passing through GCNs. As a result, STPointGCN stands out in gait recognition, especially in multi-person scenarios.

Table 2: Test accuracy on mmGait dataset(free-route).

Method	**PL**	**P-L**	**DR**	**mmGaitNet**	**STPointGCN (Ours)**
Accuracy	20%	20%	33%	45%	**60%**

Table 3: Test performance of different algorithms on our collected dataset. x% means multi-person recognition performance while (x%) means single-person recognition performance.

Method	**precision**	**recall**	**f1-score**
PL	67.71% (67.89%)	**80.71%** (67.70%)	73.09% (67.47%)
P+L	65.23% (57.92%)	77.88% (58.75%)	70.38% (56.78%)
PGL	65.99% (53.63%)	65.07% (55.80%)	65.31% (53.83%)
mmGaitNet	70.48% (67.72%)	72.48% (**69.76%**)	70.84% (67.66%)
STPointGCN(Ours)	**75.51%** (**68.88%**)	79.22% (68.57%)	**76.50%** (**68.31%**)

4. CONCLUSION

In this paper, we propose a novel end-to-end graph neural network named STPointGCN for multiple people gait recognition using mmWave sensing. Compared with existing methods, our approach shows higher recognition performance on mmGait dataset and our dataset under single-person scenarios and multi-person co-existing scenarios. Our future work will focus on building a high-quality open-source dataset with much more participants and research on models for large-scale people gait recognition through mmWave sensing.

5. REFERENCES

[1] W. Kusakunniran, "Review of gait recognition approaches and their challenges on view changes," *IET Biometrics*, 2020.

[2] D. Muramatsu, Y. Makihara, and Y. Yagi, "Gait recognition by fusing direct cross-view matching scores for criminal investigation," *IPSJ Transactions on Computer Vision and Applications*, vol. 5, pp. 35–39, 2013.

[3] X. Li, Y. Makihara, C. Xu, Y. Yagi, and M. Ren, "Gait recognition via semi-supervised disentangled representation learning to identity and covariate features," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[4] A Sg, A Mim, B Gmk, and A Fs, "Multi-view gait recognition system using spatio-temporal features and deep learning," *Expert Systems with Applications*, 2021.

[5] J. Permatasari, T. Connie, and T. S. Ong, "Inertial sensor fusion for gait recognition with symmetric positive definite gaussian kernels analysis," *Multimedia Tools and Applications*, vol. 79, no. 1, 2020.

[6] M. Ullrich, A. Kuederle, J. Hannink, S. D. Din, and F. Kluge, "Detection of gait from continuous inertial sensor data using harmonic frequencies," *IEEE Journal of Biomedical and Health Informatics*, vol. PP, no. 99, pp. 1–1, 2020.

[7] K. Ivanov, Z. Mei, M. Penev, L. Lubich, and L. Wang, "Identity recognition by walking outdoors using multimodal sensor insoles," *IEEE Access*, vol. PP, no. 99, pp. 1–1, 2020.

[8] Hajar Abedi, "Use of millimeter wave fmcw radar to capture gait parameters," *American Journal of Biomedical Science & Research*, vol. 6, pp. 122–123, 11 2019.

[9] Xin Yang, Jian Liu, Yingying Chen, Xiaonan Guo, and Yucheng Xie, "Mu-id: Multi-user identification through gaits using millimeter wave radios," in *IEEE INFOCOM 2020 - IEEE Conference on Computer Communications*, 2020.

[10] Zhen Meng, Song Fu, Jie Yan, Hongyuan Liang, Anfu Zhou, Shilin Zhu, Huadong Ma, Jianhua Liu, and Ning Yang, "Gait recognition for co-existing multiple people using millimeter wave sensing," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.

[11] Peijun Zhao, Chris Xiaoxuan Lu, Jianan Wang, Changhao Chen, Wei Wang, Niki Trigoni, and Andrew Markham, "Human tracking and identification through a millimeter wave radar," *Ad Hoc Networks*, vol. 116, pp. 102475, 2021.

[12] Martin Ester, Hans Peter Kriegel, Jrg Sander, and Xiaowei Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," *AAAI Press*, 1996.

[13] H. W. Kuhn, "The hungarian method for the assignment problem," *Naval Research Logistics*, vol. 52, no. 1-2, pp. 7–21, 2010.

[14] M. Gori, G. Monfardini, and F. Scarselli, "A new model for learning in graph domains," in *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.*, 2005, vol. 2, pp. 729–734 vol. 2.

[15] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini, "The graph neural network model," *IEEE Transactions on Neural Networks*, vol. 20, no. 1, pp. 61–80, 2009.

[16] Thomas N. Kipf and Max Welling, "Semi-supervised classification with graph convolutional networks," *CoRR*, vol. abs/1609.02907, 2016.

[17] A. Sengupta, F. Jin, R. Zhang, and S. Cao, "mm-pose: Real-time human skeletal posture estimation using mmwave radars and cnns," *IEEE Sensors Journal*, vol. PP, no. 99, pp. 1–1, 2020.

[18] F. Jin, R. Zhang, A. Sengupta, S. Cao, S. Hariri, N. K. Agarwal, and S. K. Agarwal, "Multiple patients behavior detection in real-time using mmwave radar and deep cnns," in *2019 IEEE Radar Conference (RadarConf19)*, 2019.

[19] Weijing Shi and Raj Rajkumar, "Point-gnn: Graph neural network for 3d object detection in a point cloud," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 1711–1719.

[20] Peixian Gong, Chunyu Wang, and Lihua Zhang, "Mmpoint-gnn: Graph neural network with dynamic edges for human activity recognition through a millimeter-wave radar," in *2021 International Joint Conference on Neural Networks (IJCNN)*, 2021, pp. 1–7.

[21] R. Qi Charles, Hao Su, Mo Kaichun, and Leonidas J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 77–85.

[22] Charles R. Qi, Li Yi, Hao Su, and Leonidas J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, Red Hook, NY, USA, 2017, NIPS'17, p. 5105–5114, Curran Associates Inc.