STAPointGNN: Spatial-Temporal Attention Graph Neural Network for Gesture Recognition Using Millimeter-Wave Radar

Jun Zhang[1,3], Chunyu Wang[1], Shunli Wang[1], and Lihua Zhang[1,2,3,4(✉)]

[1] Academy for Engineering and Technology, Fudan University, Shanghai, China
junzhang22@m.fudan.edu.cn, {wangcy20,slwang19,lihuazhang}@fudan.edu.cn
[2] Jilin Provincial Key Laboratory of Intelligence Science and Engineering, Changchun, China
[3] Engineering Research Center of AI and Robotics, Ministry of Education, Shanghai, China
[4] Engineering Research Center of AI and Unmanned Vehicle Systems of Jilin Province, Changchun, China

Abstract. Gesture recognition plays a pivotal role in enabling natural and intuitive human-computer interaction (HCI), finding applications in diverse domains such as smart homes, robot control, and virtual reality. Thanks to advances in computer vision, the most popular method currently is to use the camera for gesture recognition. However, the camera struggles to function properly in poor lighting and inclement weather, and risks invading privacy. Due to the robust and non-invasive features of millimeter-wave radar, gesture recognition based on millimeter-wave radar has received extensive attention from researchers in recent years. In this paper, we propose a novel graph neural network named STA-PointGNN for gesture recognition using millimeter-wave radar. In order to better extract features in the spatial and temporal dimensions of point clouds collected by millimeter-wave radar, we designed a spatial-temporal attention mechanism based on graph neural network. We also propose a novel point flow embedding method to capture the motion features of the point clouds in adjacent frames. To verify the superiority of our method, we conduct experiments on two public millimeter-wave radar gesture recognition datasets. The results show that our model outperforms existing mainstream algorithms.

Keywords: Human-computer interaction · Millimeter-wave radar · Gesture recognition · Graph neural network · Attention mechanism

1 Introduction

Gesture recognition is the key to human-computer interaction (HCI) and has a wide range of applications, such as smart home [6], robot control [3] and virtual

H. Gao et al. (Eds.): CollaborateCom 2023, LNICST 562, pp. 189–204, 2024.
https://doi.org/10.1007/978-3-031-54528-3_11

reality [27], etc. Traditional work uses wearable devices [4,22] for gesture recognition, but it has the obvious disadvantages of not being readily available and uncomfortable to wear. Camera-based approaches [26,28] can avoid the above disadvantages, but the fatal drawback is the risk of privacy leakage, which will not be used in some scenarios with high privacy requirements. To realize natural HCI, researchers turn their attention to wireless sensing devices, such as Wi-Fi signals [10,40]. However, this solution is unable to recognize fine-grained gestures and is susceptible to interference from the surrounding environment. Beyond Wi-Fi signals, millimeter-wave radar as another wireless sensing device is gradually attracting extensive attention from researchers due to its unique advantages. Compared to other wireless sensors, millimeter-wave radar has better fine resolution of range and velocity, and has a certain penetration. Millimeter-wave radar can work in all-weather conditions, including rain, fog and low-light environments. Moreover, millimeter-wave signals, as the main venue of 5G technology, will be deployed on a huge number of IoT devices and smart home appliances, promising to become a ubiquitous sensing device.

There have been some studies on gesture recognition using millimeter-wave radar. MHomeGes [18] customizes a lightweight convolution neural network for millimeter-wave gesture recognition in smart homes. M-Gesture [19] proposes a person-independent real-time millimeter-wave gesture recognition solution and releases the MMGesture dataset. MTransSee [17] proposes a novel transfer-learning approach that can achieve decent recognition accuracy for new users using fewer training samples. However, all of the above tasks require manually designing the inputs to the network and ignore the 3D coordinate information, and more importantly they do not consider the connection between adjacent frames.

Compared with existing methods [17–19] that manually design network input and use convolutional neural network (CNN) processing, we retain the 3D coordinate of the original point and process data from the perspective of point cloud, which can make full use of spatial information. Unlike LiDAR, the point clouds captured by millimeter-wave radar in each frame are sparse and uneven, which increases the complexity of neural network structure design [8]. We use graph neural network (GNN) to extract point cloud features because graph constitutes a succinct, abstractive, and intuitively apprehensible mathematical representation delineating entities and their interconnections. Several studies [8,34] have demonstrated the effectiveness of adopting GNN to process sparse point clouds collected by millimeter-wave radar.

In this work, we propose a novel graph neural network named STAPointGNN for gesture recognition using millimeter-wave radar. Inspired by the success of attention mechanism in natural language processing [33] and image processing [21], we designed a spatial-temporal attention mechanism based on graph neural network, which better extracts features in the spatial and temporal dimensions of sparse point clouds collected by millimeter-wave radar. Moreover, since gesture movements are continuous, the information between adjacent frames has great potential for capturing motion features. Motivated by [12], we propose a novel point flow embedding method designed for millimeter-wave radar point clouds,

which can efficiently capture the motion features of the point clouds in adjacent frames. The proposed method is evaluated on the MMGesture dataset [19] and the MTransSee dataset [17], achieving the state-of-the-art accuracy.

In summary, the main contributions of this work are as follows:

- We propose a spatial-temporal attention graph neural network (STAPoint-GNN) for point cloud processing.
- We propose a novel point flow embedding method designed for millimeter-wave radar point clouds, which can efficiently capture the motion features of the point clouds in adjacent frames.
- We propose an end-to-end model for gesture recognition using millimeter-wave radar.

2 Related Work

2.1 MmWave and Wireless Sensing

With the rapid development of 5G, non-contact wireless sensing devices have gradually become a research hotspot. WiFall [37] implements a device-free fall detection system using WiFi signals to detect the fall of the elderly. WiTrack [1] is a WiFi-based system that enables precise 3D motion tracking through reflected radio signals. Chen *et al.* [5] uses FM broadcast radio signals for robust indoor localization.

Compared with other wireless sensing devices, millimeter-wave radar has attracted much attention due to its advantages of higher range and velocity resolution, penetrating capability, fine-grained and robust sensing ability. Millimeter-wave radar can be used for simultaneous localization and environment mapping, due to its high localization accuracy and obstacle detection capability [39]. MBeats [42] is a robot-mounted millimeter wave radar system for dynamic heart rate monitoring during diverse user activities. In addition, there are many other research areas of millimeter-wave radar such as human activity recognition [8,31], gesture recognition [17–19] and gait recognition [12,34].

2.2 Gesture Recognition

Gestures are a form of non-verbal communication that can be used in a variety of areas such as robot control, human-computer interaction and home automation. There are many sensing technologies used for gesture recognition. Data gloves as a wearable device are a common form of implementing gesture recognition [7,14]. Vision-based gesture recognition is a relatively mature technique that utilizes a camera to capture a scene containing a gesture, and then uses computer vision algorithms to recognize, extract, and classify the image [13,16,32]. WiG [10] utilizes WiFi signals for gesture recognition. In addition, there are researchers using surface electromyography [15] and ultrasound [11] for gesture recognition.

Gesture recognition through millimeter-wave radar has garnered noteworthy academic attention. Due to the diversity of data captured by millimeter-wave radar, current research can be broadly categorized into three categories. The

first way uses range doppler images, range angle image or doppler angle image [2,35,38]. However, this approach does not fully utilize the three-dimensional coordinate information. The second approach is to manually design the inputs to the model [18,19]. Nevertheless, hand-designed features need to be carefully designed by experienced domain experts, and the quality of the design features directly determines the model accuracy. The third method is to utilize point cloud data [20], which can fully leverage spatial information. The method proposed in this paper deals directly with point clouds and belongs to the third.

2.3 Graph Neural Network

CNN is suitable for processing images, recurrent neural network is suitable for processing sequential data. However, these two networks are not suitable for directly processing point clouds. Point clouds can be naturally viewed as graph structures and thus processed using graph neural networks. Point-GNN [29] is the first work to use graph neural networks to process point clouds. In millimeter-wave radar-based human activity recognition, RADHAR [31] uses CNN to process voxelized representations of point clouds, however, this is computationally prohibitive, and MMPoint-GNN [8] uses graphs to represent point clouds and achieves better results on the same dataset. In millimeter-wave radar-based gait recognition, STPointGCN [34] uses graph convolutional networks to extract features from point clouds superior to mmGait [23] using CNN. The temporal edges in STPointGCN are similar to the temporal attention module of our proposed method. A certain point in STPointGCN is only connected to the closest point in the adjacent frames. However, in our proposed method, a certain point is connected to all points in the adjacent frames and the connection is learned through the attention mechanism.

In this work, we propose a new graph neural network which is suitable for extracting spatial-temporal features of sparse point clouds collected by millimeter-wave radar. Specially, we designed a spatial-temporal attention mechanism based on graph neural network, which effectively integrates the attention mechanism into graph neural network. Moreover, we propose a novel point flow embedding method designed for millimeter-wave radar point clouds, which can efficiently capture the motion features of the point clouds in adjacent frames.

3 Method

3.1 Point Flow Embedding

The data generated by millimeter-wave radar is usually returned in the form of frames, with each frame containing several points and each point containing several features. Formally, we define a frame containing N points as a set $P = \{p_1, p_2, \ldots, p_N\}$, where $p_i = (x_i, s_i)$ denotes a point with 3D coordinates $x_i \in \mathbb{R}^3$ and state features $s_i \in \mathbb{R}^k$. In this work, s_i contains two properties, the doppler velocity v_i and the reflection intensity ϵ_i. So p_i can be represented as $p_i = (x_i, v_i, \epsilon_i)$.

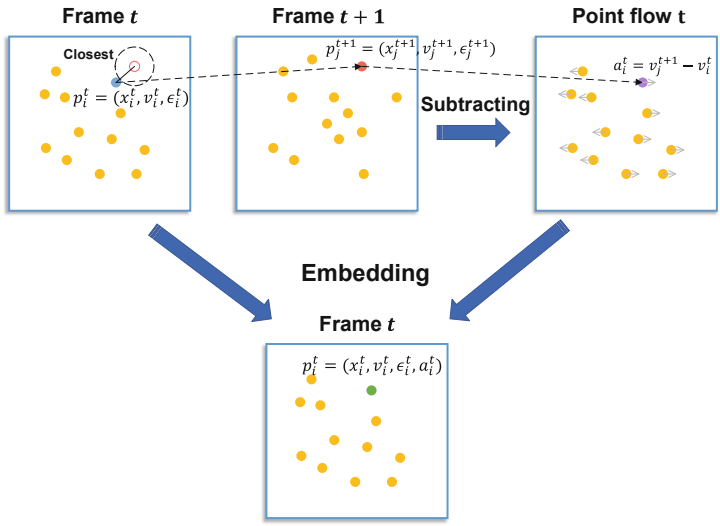

Fig. 1. Point flow embedding method.

Optical flow has been shown to be effective in the field of video based human action recognition [30]. It can effectively capture the features of motion. Gesture recognition as a subtask of human action recognition also has the above properties. Video based and millimeter-wave radar based gesture recognition are very similar, both consisting of many frames of data. The difference is that the former frame is an image, while the latter frame is a point cloud. On the other hand, millimeter wave radar senses the target at a certain sampling rate and therefore loses the information between two adjacent frames. Inspired by [12], We use point flow to capture motion information between adjacent frames.

Here is the specific description of point flow embedding method. As shown in Fig. 1, take two adjacent frames P^t and P^{t+1} as examples. For a certain point in the frame P^t, denoted as p_i^t, to find the motion relationship between adjacent frames, we find the point with the closest spatial distance to it in the frame P^{t+1}, denoted as p_j^{t+1}. Subtracting the doppler velocity of these two points, we obtain the characteristic of acceleration $a_i^t = v_j^{t+1} - v_i^t$. Then add acceleration as an additional property to the state features of the point cloud, so now the point p_i^t can be represented as $p_i^t = (x_i^t, v_i^t, \epsilon_i^t, a_i^t)$. Then we take the point clouds through point flow embedding method as the input of model.

3.2 STAPointGNN

The overall architecture of the proposed STAPointGNN is illustrated in Fig. 2. The whole network consists of three components: (a) spatial-temporal feature extraction, (b) temporal aggregation and (c) classification. Given several frames of raw point clouds, a module composed of PointGNN, spatial attention, and

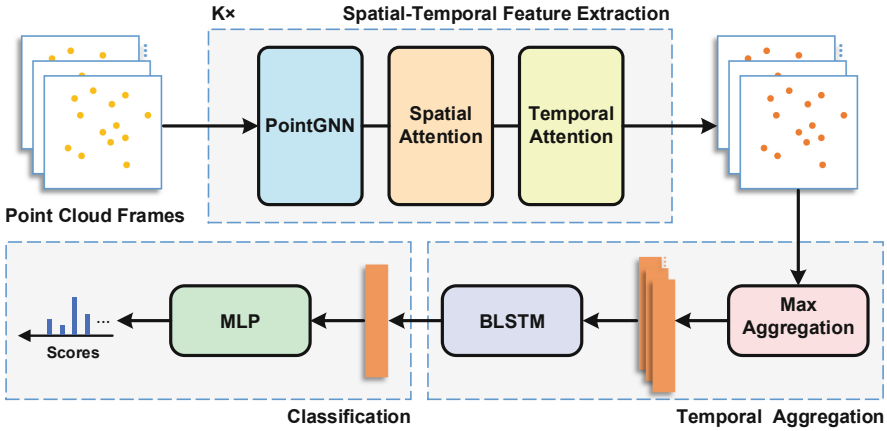

Fig. 2. The overall architecture of proposed STAPointGNN. It consists of three components: (a) spatial-temporal feature extraction, (b) temporal aggregation and (c) classification.

temporal attention extracts spatial-temporal features. After K iterations, we use max aggregation with each frame to obtain feature vectors representing each frame, then use bidirectional LSTM to further extract features from the time dimension. Finally, the output of bidirectional LSTM is processed through multi-layer perceptron (MLP) to obtain the prediction probabilities of different gestures. Next, we will introduce PointGNN and spatial-temporal attention in detail.

3.3 PointGNN

Unlike LiDAR, the points collected by millimeter-wave radar are sparse. We don't need to use voxel downsampling like [29] for the graph construction. The proposed method directly construct each point in the point cloud as each vertex in the graph. We use state features s_i as the initial vertex features. Then, we construct a fully connected graph $G(P, E)$ using P as vertices and define edges as:

$$E = \{(p_i, p_j) \mid i, j \in N\} \tag{1}$$

In an image, the convolution operation updates the current pixel value through adjacent pixel values. Similarly, message passing in a graph structure can be achieved by aggregating features along the edges [29]. Figure 3 takes the central vertex as an example to show the process of one iteration of PointGNN. In the $(k+1)^{th}$ iteration, the features of vertex are represented as follows:

$$\begin{aligned} v_i^{k+1} &= g^k(\rho(\{e_{ji}^k \mid (j,i) \in E\}), v_i^k) \\ e_{ji}^k &= f^k(v_i^k, v_j^k) \end{aligned} \tag{2}$$

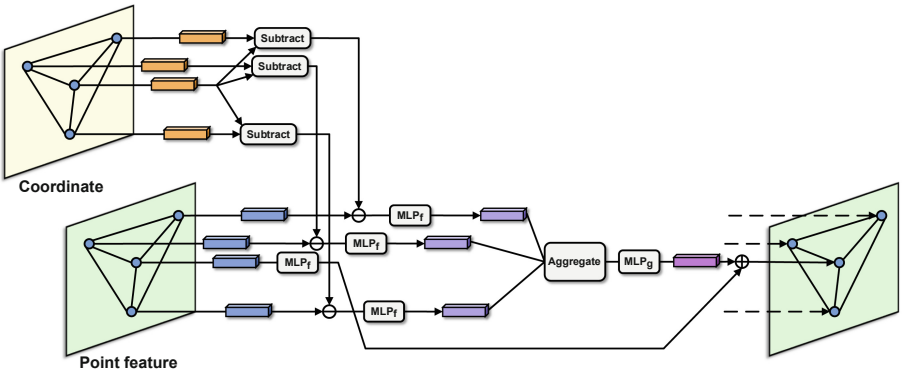

Fig. 3. Architecture of the proposed PointGNN. It takes the central vertex as an example to show the process of one iteration of PointGNN.

where v_i^k represents the vertex features of vertex i in the k^{th} iteration, e_{ji}^k represents the features of the directed edge from vertex j to vertex i in the k^{th} iteration. $f^k(\cdot)$ calculates the edge features between two vertices. $\rho(\cdot)$ is a permutation invariant function, which can be Max, $Mean$ or Sum operations, used to aggregate the features of edges for each vertex. $g^k(\cdot)$ updates vertex features using aggregated edge features.

We use relative coordinates to extract edge features and model $f^k(\cdot)$ using MLP. Specifically, the difference between vertex i coordinates x_i and vertex j coordinates x_j is concatenated onto vertex j features v_j, and then MLP is used to update the edge features:

$$e_{ji}^k = MLP_f^k(cat(x_i - x_j, v_j^k)) \tag{3}$$

We use the Max operation as $\rho(\cdot)$ to aggregate edge features, model $g^k(\cdot)$ using MLP to update vertex features, and add a residual connection in $g^k(\cdot)$:

$$v_i^{k+1} = MLP_g^k(Max(\{e_{ji}^k \mid (j,i) \in E\})) + v_i^k \tag{4}$$

3.4 Spatial-Temporal Attention

After PointGNN, the vertex features are further extracted through the spatial-temporal attention module. The spatial attention module is shown in Fig. 4.

Similar to natural language processing tasks, we treat a frame of point cloud as a sentence and each point as a token. For the spatial attention, we adopt the self-attention introduced in Transformer [33], as shown in Fig. 4. We define the dimension of the input feature and the dimension of the output feature as d_i and d_o, respectively. Let $\boldsymbol{Q}$, $\boldsymbol{K}$, and $\boldsymbol{V}$ represent *query*, *key*, and *value* respectively. They are respectively passed through a linear transformation by the input feature $\boldsymbol{F}_{in} \in \mathbb{R}^{N \times d_i}$ as follows:

$$
\begin{aligned}
&\boldsymbol{Q} = \boldsymbol{F}_{in} \cdot \boldsymbol{W}_q \\
&\boldsymbol{K} = \boldsymbol{F}_{in} \cdot \boldsymbol{W}_k \\
&\boldsymbol{V} = \boldsymbol{F}_{in} \cdot \boldsymbol{W}_v \\
&\boldsymbol{W}_q, \boldsymbol{W}_k, \boldsymbol{W}_v \in \mathbb{R}^{d_i \times d_m} \\
&\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V} \in \mathbb{R}^{N \times d_m}
\end{aligned}
\tag{5}
$$

where $\boldsymbol{W}_q$, $\boldsymbol{W}_k$, and $\boldsymbol{W}_v$ are shared learnable linear transformation, aiming to place semantically similar points closer together in the new space. d_m is the dimension of the query, key, and value.

We compute attention weights via matrix dot product using query and key:

$$
\tilde{\boldsymbol{A}} = \boldsymbol{Q} \cdot \boldsymbol{K}^T, \quad \tilde{\boldsymbol{A}} \in \mathbb{R}^{N \times N} \tag{6}
$$

Divide $\tilde{\boldsymbol{A}}$ by d_m, and apply softmax function for normalization:

$$
\boldsymbol{A} = softmax(\frac{\tilde{\boldsymbol{A}}}{\sqrt{d_m}}), \quad \boldsymbol{A} \in \mathbb{R}^{N \times N} \tag{7}
$$

We use the attention score and value to do dot product to get the attention output feature. In order to meet the requirements of the output dimension, we add MLP transformation with residual connection:

$$
\boldsymbol{F}_{out} = MLP(\boldsymbol{A} \cdot \boldsymbol{V}) + MLP(\boldsymbol{F}_{in}), \quad \boldsymbol{F}_{out} \in \mathbb{R}^{N \times d_o} \tag{8}
$$

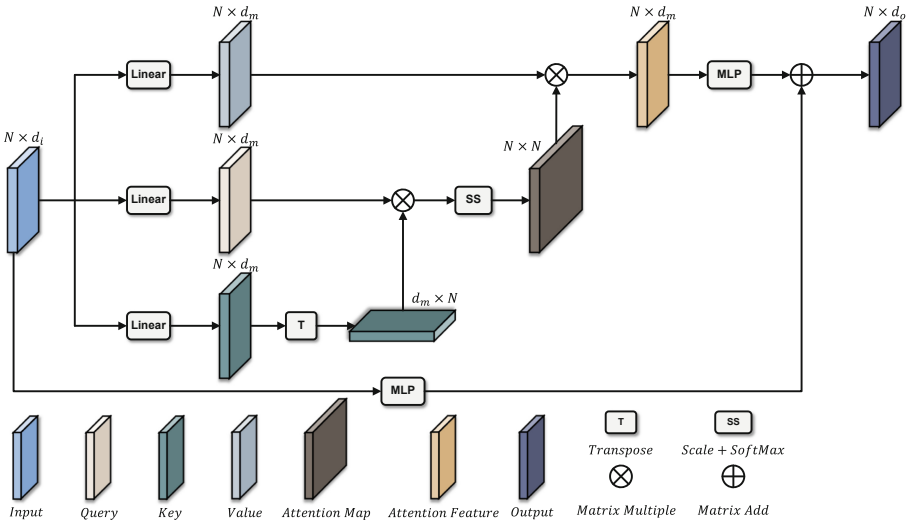

Fig. 4. Architecture of the proposed spatial attention module.

For the temporal attention module, the input is two adjacent frames, as shown in Fig. 5. For a certain point p_j^{t+1} in the frame P^{t+1}, all points features in the frame P^t are weighted and summed to update p_j^{t+1}.

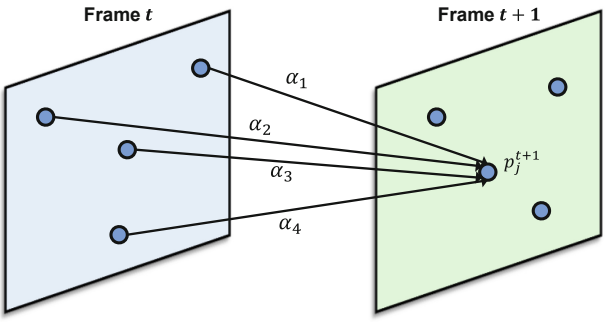

Fig. 5. Illustration of temporal attention module. For a certain point p_j^{t+1} in the frame P^{t+1}, all points features in the frame P^t are weighted and summed to update p_j^{t+1}.

In terms of specific implementation, we still use the spatial attention module, just set point features in frame P^{t+1} as *query*, point features in the frame P^t as *key* and *value*.

3.5 Loss Function

Gesture recognition is a multi-class classification task, softmax and cross entropy are adopted as the loss function:

$$\begin{aligned} loss(y, c) &= -\log\left(\frac{\exp\left(y[c]\right)}{\sum_{i=0}^{C-1}\exp\left(y[i]\right)}\right) \\ &= -y[c] + \log\left(\sum_{i=0}^{C-1}\exp\left(y[i]\right)\right) \end{aligned} \tag{9}$$

where C is the number of gesture categories, c is the ground truth label, $y = [y_0, y_1, \ldots, y_{n-1}]$ is the confidence vector predicted by the model.

4 Experiments

4.1 Datasets

We use two public gesture datasets MMGesture dataset [19] and MTransSee dataset [17] to evaluate our proposed STAPointGNN method.

MMGesture Dataset[1] MMGesture dataset is the first gesture dataset collected with millimeter-wave radar. It has 56420 gesture sample instances with a total duration of 1357 min involving 144 volunteers (64 men and 80 women). The dataset contains two scenarios, short range gesture (less than 0.5m) for interacting with accessory devices and long range gesture (between 2m and 5m) for interacting with smart homes. We use long range gestures because it provides point cloud data consistent with our method. There are 4 gestures, and the corresponding relationship between the gesture category name and its actual action is as follows: "knock" means dual knock, "rotate" means hand rotation, "lswipe" means left swipe, and "rswipe" means right swipe.

MTransSee Dataset[2] MTransSee dataset aims to control smart home appliances and contains 54080 samples, including 5 predefined gestures: draw a circle (CR), knock a virtual table twice (KO), pull a hand (PL), push a hand (PS) and lift up a hand (UP). The dataset considers the influence of different user habits and distances, involving 32 different volunteers and 13 different distances between 1.2m and 4.8m. In addition, the influence of the reflection of uneven objects on the data collected by the millimeter-wave radar is also considered. Volunteers perform gestures near different objects: such as chair, metal table, metal oven, TV, etc.

4.2 Implementation Details

We divide the dataset into training and testing set in proportion 80% and 20%. We use a sliding window with length $T = 20$ and moving step s = 20 to generate the data to fit the model input. Since the point flow is only calculated in adjacent frames, one input sample contains 19 frames.

Spatial-temporal feature extraction module repeats for 3 iterations, where the feature dimensions for each iteration are [16, 64, 128]. In the temporal aggregation module, the LSTM used is a bidirectional LSTM with 1 layer and 16 hidden units. The optimizer is Adam and the initial learning rate is 0.001. We implement the model in PyTorch.

4.3 Experimental Results

We evaluated different benchmark algorithms on two datasets including PointNet [24] combined with LSTM, PointNet++ [25] combined with LSTM, Point cloud transformer [9] combined with LSTM, Point Transformer [41] combined with LSTM and DGCNN [36] combined with LSTM. In Table 1, the results of methods [1–3] are baselines from [8], including PointNet combined with LSTM, Point-GNN combined with LSTM and MMPoint-GNN combined with LSTM.

Table 1 reports the accuracy of different algorithms on MMGesture dataset. The best baseline on MMGesture dataset is MMPoint-GNN + LSTM, achieving 92.67% accuracy. MMPoint-GNN has strong spatial feature extraction ability

[1] https://github.com/fengxudi/mmWave-gesture-dataset.
[2] https://github.com/mmTransGes/mTransSee_Dataset.

Table 1. Test accuracy of different algorithms on MMGesture dataset.

S. No	Method	Accuracy (%)
1	PointNet + LSTM	61.51
2	Point-GNN + LSTM	92.10
3	MMPoint-GNN + LSTM	92.67
4	PointNet++ + LSTM	89.62
5	Point cloud transformer + LSTM	87.09
6	Point Transformer + LSTM	91.66
7	DGCNN + LSTM	88.03
8	**STAPointGNN (Ours)**	**94.61**

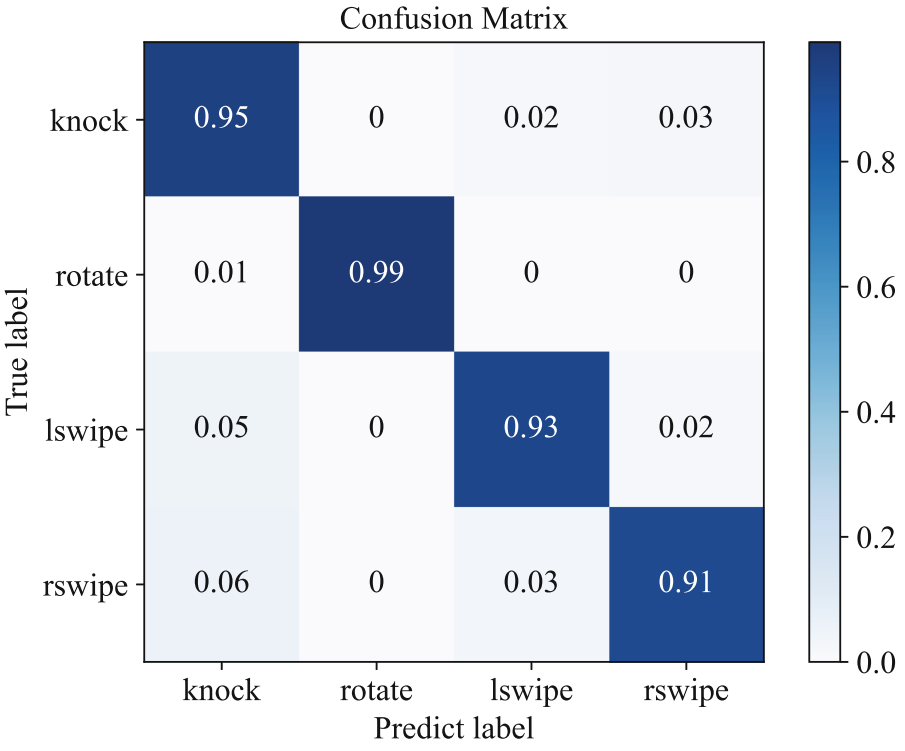

Fig. 6. Confusion matrix of STAPointGNN in MMGesture dataset.

Table 2. Test accuracy of different algorithms on MTransSee dataset.

S. No	Method	Accuracy (%)
1	PointNet + LSTM	95.45
2	PointNet++ + LSTM	95.96
3	Point cloud transformer + LSTM	93.57
4	Point Transformer + LSTM	96.67
5	DGCNN + LSTM	96.11
6	**STAPointGNN (Ours)**	**97.56**

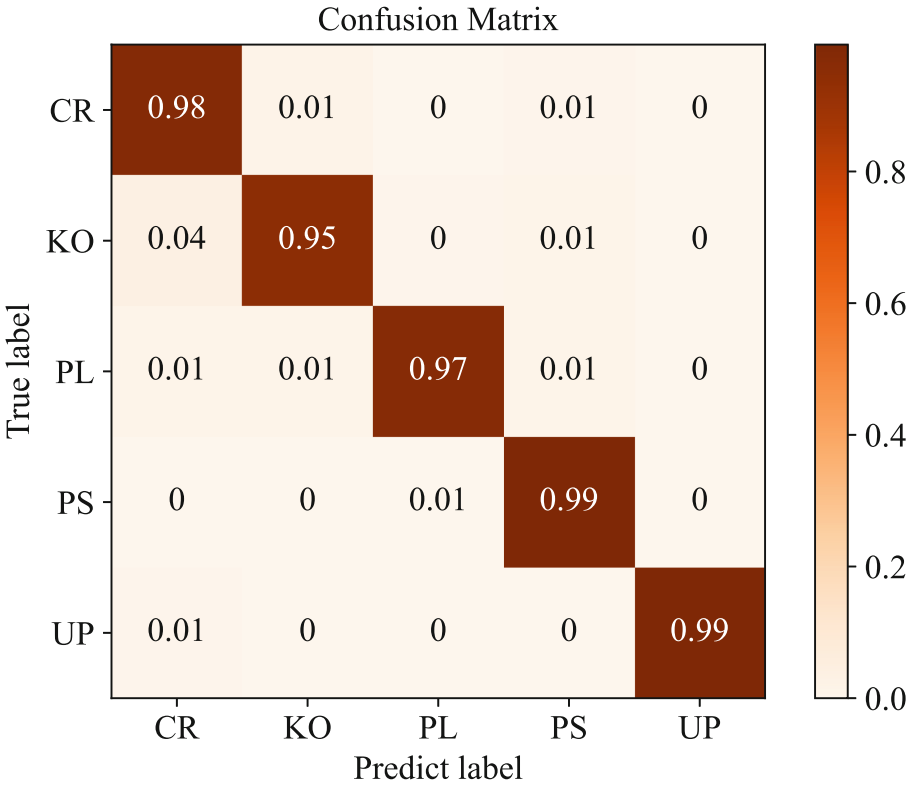

Fig. 7. Confusion matrix of STAPointGNN in MTransSee dataset.

due to its dynamic edge design [8]. Among other comparative algorithms such as PointNet++ + LSTM, Point cloud transformer + LSTM, Point Transformer + LSTM and DGCNN + LSTM, only Point Transformer + LSTM has an accuracy rate exceeding 90%, reaching 91.66%, but it is still inferior to MMPoint-GNN + LSTM. However, our method STAPointGNN achieves an accuracy of 94.61%,

surpassing MMPoint-GNN + LSTM by about 2%, which is currently the state-of-the-art method on MMGesture dataset. Figure 6 shows the confusion matrix of STAPointGNN in MMGesture dataset. It can be seen that our method can accurately distinguish various gestures. Rotate is the easiest action to distinguish because it is distinctly different from the other gestures. Left swipe and right swipe gestures are difficult to distinguish because they are similar except that they move in opposite directions. Moreover, left swipe and right swipe are easily recognized as dual knock.

Table 2 reports the accuracy of different algorithms on MTransSee dataset. The different algorithms perform well on this dataset compared to the previous dataset, all exceeding 93% accuracy. Mainly because the dataset has more samples and considers the influence of distance. The method PointNet + LSTM based on the set representation reached 95.45%, and the improved PointNet++ + LSTM based on PointNet + LSTM reached 95.96%. The method based on graph representation DGCNN + LSTM reached 96.11%, which shows that GNN is more suitable for the extraction of millimeter-wave radar point cloud features. The Point Transformer + LSTM of the attention-based method reaches 96.67%, which shows that the attention mechanism is effective. Our method STAPointGNN achieves 97.56%, surpassing the best baseline by about 1%. The confusion matrix of STAPointGNN in MTransSee dataset is shown in Fig. 7. Overall, although the number of gesture categories increases, the classification accuracy of gestures generally improves. PS and UP gestures are the most easily recognized gestures, while KO is easily recognized as CR.

Compared to existing methods, STAPointGNN incorporates graph neural networks and attention mechanisms to establish associations between adjacent frames, which can extract spatial and temporal features more efficiently. As a result, STAPointGNN excels in the field of millimeter-wave radar gesture recognition.

5 Conclusion

In this paper, we propose a novel graph neural network named STAPointGNN for gesture recognition using millimeter-wave radar. Our method based on spatial-temporal attention mechanism of graph neural network can extract spatial and temporal features of point clouds generated by millimeter-wave radar more efficiently. In addition, we propose a novel point flow embedding method to capture the motion features of the point clouds in adjacent frames. In the public datasets MMGesture and MTransSee, our method achieves leading accuracy compared to existing methods. In the future, we will consider real-time and multi-person gesture recognition issues to get closer to real scenarios.

Acknowledgement. This work was supported by National Key R&D Program of China (2021ZD0113502) and Shanghai Municipal Science and Technology Major Project (2021SHZDZX0103).

References

1. Adib, F., Kabelac, Z., Katabi, D., Miller, R.C.: 3D tracking via body radio reflections. In: 11th USENIX Symposium on Networked Systems Design and Implementation (NSDI 14), pp. 317–329 (2014)
2. Ali, A., et al.: End-to-end dynamic gesture recognition using mmWave radar. IEEE Access **10**, 88692–88706 (2022)
3. Van den Bergh, M., et al.: Real-time 3D hand gesture interaction with a robot for understanding directions from humans. In: 2011 Ro-Man, pp. 357–362. IEEE (2011)
4. Chen, L., Zhang, Y., Peng, L.: METIER: a deep multi-task learning based activity and user recognition model using wearable sensors. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies **4**(1), 1–18 (2020)
5. Chen, Y., Lymberopoulos, D., Liu, J., Priyantha, B.: FM-based indoor localization. In: Proceedings of the 10th International Conference on Mobile Systems, Applications, and Services, pp. 169–182 (2012)
6. Desai, S., Desai, A.: Human computer interaction through hand gestures for home automation using microsoft kinect. In: Modi, N., Verma, P., Trivedi, B. (eds.) Human computer interaction through hand gestures for home automation using microsoft kinect. AISC, vol. 508, pp. 19–29. Springer, Singapore (2017). https://doi.org/10.1007/978-981-10-2750-5_3
7. Fang, B., Sun, F., Liu, H., Liu, C.: 3D human gesture capturing and recognition by the IMMU-based data glove. Neurocomputing **277**, 198–207 (2018)
8. Gong, P., Wang, C., Zhang, L.: 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), pp. 1–7. IEEE (2021)
9. Guo, M.H., Cai, J.X., Liu, Z.N., Mu, T.J., Martin, R.R., Hu, S.M.: PCT: Point cloud transformer. Comput. Vis. Media **7**, 187–199 (2021)
10. He, W., Wu, K., Zou, Y., Ming, Z.: WiG: WiFi-based gesture recognition system. In: 2015 24th International Conference on Computer Communication and Networks (ICCCN), pp. 1–7. IEEE (2015)
11. Hettiarachchi, N., Ju, Z., Liu, H.: A new wearable ultrasound muscle activity sensing system for dexterous prosthetic control. In: 2015 IEEE International Conference on Systems, Man, and Cybernetics, pp. 1415–1420. IEEE (2015)
12. Huang, Y., Wang, Y., Shi, K., Gu, C., Fu, Y., Zhuo, C., Shi, Z.: HDNet: hierarchical dynamic network for gait recognition using millimeter-wave radar. In: ICASSP 2023–2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 1–5. IEEE (2023)
13. Indra, D., Madenda, S., Wibowo, E.P., et al.: Indonesian sign language recognition based on shape of hand gesture. Procedia Comput. Sci. **161**, 74–81 (2019)
14. Kakoty, N.M., Sharma, M.D.: Recognition of sign language alphabets and numbers based on hand kinematics using a data glove. Procedia Comput. Sci. **133**, 55–62 (2018)
15. Ketykó, I., Kovács, F., Varga, K.Z.: Domain adaptation for semg-based gesture recognition with recurrent neural networks. In: 2019 International Joint Conference on Neural Networks (IJCNN), pp. 1–7. IEEE (2019)
16. Lin, J., Ding, Y.: A temporal hand gesture recognition system based on hog and motion trajectory. Optik **124**(24), 6795–6798 (2013)

17. Liu, H., et al.: mTranssee: enabling environment-independent mmWave sensing based gesture recognition via transfer learning. Proc. ACM Interact. Mobile, Wearable Ubiquitous Technol. **6**(1), 1–28 (2022)
18. Liu, H., et al.: Real-time arm gesture recognition in smart home scenarios via millimeter wave sensing. Proc. ACM Interact. Mobile, Wearable and Ubiquitous Technol. **4**(4), 1–28 (2020)
19. Liu, H., et al.: M-gesture: Person-independent real-time in-air gesture recognition using commodity millimeter wave radar. IEEE Internet Things J. **9**(5), 3397–3415 (2021)
20. Liu, Yu., Wang, Y., Liu, H., Zhou, A., Liu, J., Yang, N.: Long-range gesture recognition using millimeter wave radar. In: Yu, Z., Becker, C., Xing, G. (eds.) GPC 2020. LNCS, vol. 12398, pp. 30–44. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-64243-3_3
21. Liu, Z., et al.: Video swin transformer. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3202–3211 (2022)
22. Lu, Y., Huang, B., Yu, C., Liu, G., Shi, Y.: Designing and evaluating hand-to-hand gestures with dual commodity wrist-worn devices. Proc. ACM Interact., Mobile, Wearable Ubiquitous Technol. **4**(1), 1–27 (2020)
23. Meng, Z., et al.: Gait recognition for co-existing multiple people using millimeter wave sensing. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 849–856 (2020)
24. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: PointNet: deep learning on point sets for 3D classification and segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 652–660 (2017)
25. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: PointNet++: deep hierarchical feature learning on point sets in a metric space. In: Advances in Neural Information Processing Systems, vol. 30 (2017)
26. Radu, V., Henne, M.: Vision2Sensor: knowledge transfer across sensing modalities for human activity recognition. Proc. ACM Interact., Mobile, Wearable Ubiquitous Technol. **3**(3), 1–21 (2019)
27. Sagayam, K.M., Hemanth, D.J.: Hand posture and gesture recognition techniques for virtual reality applications: a survey. Virtual Reality **21**, 91–107 (2017)
28. Sharp, T., et al.: Accurate, robust, and flexible real-time hand tracking. In: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, pp. 3633–3642 (2015)
29. Shi, W., Rajkumar, R.: 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, pp. 1711–1719 (2020)
30. Simonyan, K., Zisserman, A.: Two-stream convolutional networks for action recognition in videos. In: Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N., Weinberger, K. (eds.) Advances in Neural Information Processing Systems, vol. 27. Curran Associates, Inc. (2014)
31. Singh, A.D., Sandha, S.S., Garcia, L., Srivastava, M.: RadHAR: human activity recognition from point clouds generated through a millimeter-wave radar. In: Proceedings of the 3rd ACM Workshop on Millimeter-wave Networks and Sensing Systems, pp. 51–56 (2019)
32. Sun, J.H., Ji, T.T., Zhang, S.B., Yang, J.K., Ji, G.R.: Research on the hand gesture recognition based on deep learning. In: 2018 12th International Symposium on Antennas, Propagation and EM Theory (ISAPE), pp. 1–4. IEEE (2018)
33. Vaswani, A., et al.: Attention is all you need. In: Advances in Neural Information Processing Systems, vol. 30 (2017)

34. Wang, C., Gong, P., Zhang, L.: Stpointgcn: spatial temporal graph convolutional network for multiple people recognition using millimeter-wave radar. In: ICASSP 2022–2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 3433–3437. IEEE (2022)
35. Wang, S., Song, J., Lien, J., Poupyrev, I., Hilliges, O.: Interacting with soli: exploring fine-grained dynamic gesture recognition in the radio-frequency spectrum. In: Proceedings of the 29th Annual Symposium on User Interface Software and Technology, pp. 851–860 (2016)
36. Wang, Y., Sun, Y., Liu, Z., Sarma, S.E., Bronstein, M.M., Solomon, J.M.: Dynamic graph CNN for learning on point clouds. ACM Trans. Graph. (tog) **38**(5), 1–12 (2019)
37. Wang, Y., Wu, K., Ni, L.M.: WiFall: device-free fall detection by wireless networks. IEEE Trans. Mob. Comput. **16**(2), 581–594 (2016)
38. Yan, B., Wang, P., Du, L., Chen, X., Fang, Z., Wu, Y.: mmGesture: semi-supervised gesture recognition system using mmWave radar. Expert Syst. Appl. **213**, 119042 (2023)
39. Yassin, A., Nasser, Y., Al-Dubai, A.Y., Awad, M.: MOSAIC: simultaneous localization and environment mapping using mmWave without a-priori knowledge. IEEE Access **6**, 68932–68947 (2018)
40. Yu, N., Wang, W., Liu, A.X., Kong, L.: QGesture: quantifying gesture distance and direction with WiFi signals. Proc. ACM Interact. Mobile, Wearable Ubiquitous Technol. **2**(1), 1–23 (2018)
41. Zhao, H., Jiang, L., Jia, J., Torr, P.H., Koltun, V.: Point transformer. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 16259–16268 (2021)
42. Zhao, P., et al.: Heart rate sensing with a robot mounted mmWave radar. In: 2020 IEEE International Conference on Robotics and Automation (ICRA), pp. 2812–2818. IEEE (2020)