2025 International Conference on Intelligent Computing
July 26-29, Ningbo, China
https://www.ic-icc.cn/2025/index.php

ST-PCN: A Dynamic Point Cloud Classification Network Based on the Spatiotemporal Attention Mechanism of Millimeter-Wave Radar

Zanqiang Wu[1] ,Qiaojuan Tong[2] and Hongbing Ma*(✉)

[1] Xinjiang University, Urumqi, China
[2] Tsinghua University, Beijing, China

hbma@tsinghua.edu.cn

Abstract. With the advent of the intelligent era, human activity recognition has become increasingly important in application scenarios such as intelligent monitoring and human-computer interaction. The application of millimeter-wave radar in human behavior recognition has also emerged as a research hotspot in recent years. The point cloud data of millimeter-wave radar can provide depth information in a three-dimensional space, which endows it with unique advantages in capturing spatial postures. This paper proposes a general method for human behavior recognition based on millimeter-wave radar point clouds. This method first processes the point cloud data of each frame in the spatial dimension to extract spatial features. Subsequently, it models the temporal dimension. By introducing attention mechanisms in both space and time, the model can focus on important features, thereby improving the accuracy of behavior recognition. Finally, the extracted features are classified through a multi-layer perceptron (MLP). By comparing with other methods on public datasets, the results show that the proposed ST-PCN network model outperforms other baseline models, verifying its effectiveness and superiority.

Keywords: Millimeter-wave Radar, Point Cloud, Behavior Recognition.

1 Introduction

1.1 A Subsection Sample

In daily life, application scenarios of smart homes, human-computer interaction, and intelligent monitoring can be seen everywhere [1]. These application scenarios are supported by human pose recognition technology. Therefore, pose recognition has always been a key focus highly anticipated by researchers. Currently, most methods use visual sensors [2] and wearable devices [3]. After years of continuous development, these solutions have undoubtedly achieved remarkable results. However, visual sensors are

prone to privacy leakage issues, and poor lighting conditions can affect the recognition effect [4]. For the wearable device solution, problems such as cost, wearing experience, and convenience have always been challenges [5]. In recent years, with the progress of electronics and information technology, radars have gradually been widely used in the market, especially in fields such as intelligent transportation, security monitoring, autonomous driving, and healthcare [6]. Millimeter-wave radars have unique advantages in action recognition compared to other sensors. They are not affected by lighting conditions and can work stably in dark or strong-light environments. Moreover, millimeter-wave radars do not capture facial or other sensitive biometric information, thus having significant advantages in privacy protection [7]. For these reasons, how to use millimeter-wave radars for accurate human pose recognition has become a research hotspot recently.

The commonly used method is to use the time-frequency analysis method to map the original radar echo signal to the time-frequency spectrum and observe the micro-Doppler effect of different behaviors [8, 9]. However, although traditional time-frequency spectrograms can display the frequency and time distribution of signals on a two-dimensional plane, they lack depth information and cannot fully capture the spatial pose changes of signals. In addition to providing time-domain and frequency-domain signal information, millimeter-wave radars can also generate high-precision point cloud data. Radar point cloud data contains key information such as the Cartesian coordinates of the target object and reflection intensity. This information can not only describe the spatial characteristics of the target object but also be used for further analysis of the object's motion state and behavior pattern. The addition of this depth information gives radar point cloud data unique advantages in fields such as human action recognition.

Radar point clouds have the characteristics of sparsity and inhomogeneity, and thus cannot be directly used as the input of a neural network. In previous works, it was common to transform radar point clouds into other forms for processing. For example, [10] transformed radar point clouds into three-dimensional voxel grids and utilized a three-dimensional convolutional neural network (3D CNN) to extract spatial features. [11] proposed a method based on graph neural networks (GNNs), which modeled the point clouds as a graph structure (where points serve as nodes and the relationships between points serve as edges), and extracted spatial and topological features through graph convolution operations for pose estimation. [12] projected radar point clouds onto a two-dimensional plane to generate depth images and used a convolutional neural network (CNN) for feature extraction and classification. These methods have their drawbacks. Although voxelization is suitable for 3D CNNs, it will lead to a loss of spatial resolution. In the scenario of sparse point clouds, a large of empty voxels will be introduced, increasing computational and memory redundancy, reducing efficiency, and losing precise geometric information, making it difficult to capture subtle pose changes. While graph structure modeling can preserve the topological relationships between points, the construction and calculation complexity of the graph structure are high, and the computational cost increases significantly when the amount of point cloud data is large. Projecting point clouds into depth images can handle occlusion problems, but in complex scenarios, it is difficult to accurately reflect the complete human pose, and the

three-dimensional spatial information of the point clouds will be lost. Although inputting time-series point clouds can capture the dynamic changes of actions, the computational cost of sequential processing is high. Since sequential methods have high requirements for time synchronization and data continuity, it is difficult to meet the real-time requirements.

Methods that convert point clouds into other forms make up for the disadvantage that point clouds cannot be directly used as the input of neural networks. However, the process of converting point clouds into other forms (such as graph structures, voxels, depth images, etc.) is a computationally intensive process. Therefore, there are still many problems in practical applications. To address these issues, this paper proposes a neural network named ST-PCN (Spatio-Temporal Dynamic Point Cloud Network) that is suitable for directly processing millimeter-wave radar dynamic point clouds for human pose classification and recognition. We verified our method on a public dataset and achieved a high accuracy rate.

In summary, the contributions of this paper are as follows:

- We design a model framework that incorporates spatio-temporal attention mechanisms for directly processing point clouds.
- We propose a method to address the non-uniformity of point clouds, enabling their direct use as neural network inputs.
- We introduce a data augmentation technique to increase the number of training samples for the model.

2 Related Work

2.1 Millimeter-wave Radar and Point Cloud Acquisitionse

The transmitted signal of an FMCW radar can be expressed as:

$$S_{TX}(t) = \exp(j2\pi(f_C t + \int_0^t f_T(\tau)d\tau)) \tag{1}$$

$$f_T(\tau) = \frac{B}{T_c}\tau \tag{2}$$

$$k = \frac{B}{T_c} \tag{3}$$

where f_c is the carrier frequency of the transmitted signal, $f_T(\tau)$ is the instantaneous frequency of the transmitted signal, B is the signal bandwidth, and T_c is the period of a chirp signal. The received signal of the radar is:

$$S_{RX}(t) = \exp(j2\pi(f_C(t-\delta) + \int_0^{t-\delta} f_T(\tau)d\tau)) \tag{4}$$

where δ is the time delay of the received signal:

$$\delta = \frac{2(R + V_r t)}{c} \tag{5}$$

where R is the target distance, V_r is the target radial velocity, and c is the speed of light. By mixing the transmitted and received signals and applying lowpass filtering, the intermediate frequency (IF) signal required for experiments is obtained:

$$S_{IF}(t) = \exp(j2\pi(f_c\delta - 0.5k\delta^2 + kt\delta)) \tag{6}$$

Substituting (5) into (6), we get:

$$S_{IF}(t) = \exp(j2\pi(f_c \frac{2(R + v_r t)}{c} - \frac{4k(R + v_r t)^2}{2c^2} + kt\frac{2(R + v_r t)}{c})) \tag{7}$$

Simplifying, we obtain:

$$S_{IF}(t) = \exp(j2\pi((\frac{2f_c v_r + 2kR}{c})t + \frac{2Rf_c}{c})) \tag{8}$$

The distance of the radar-detected target can be expressed as:

$$R = \frac{cT_c S_{IF}(t)}{2B} \tag{9}$$

where c is the speed of light, T_c is the duration of a chirp, and B is the bandwidth of the radar signal.

Based on the above theory, the target distance can be calculated by performing a Fast Fourier Transform (FFT) on the fast-time dimension, i.e., the sequence of echo chirp signals within a single cycle, to extract the frequency corresponding to the spectral peak. Subsequently, a Doppler FFT is performed on the IF signal to measure the phase changes and obtain the target velocity. Additionally, to determine the spatial coordinates x, y, and z of the target, angle estimation is required. The azimuth angle θ and elevation angle φ are calculated using the phase differences between antennas:

$$\theta = \sin^{-1}(\frac{\omega_z}{\pi}) \tag{10}$$

$$\varphi = \sin^{-1}(\frac{\omega_x}{\cos(\varphi)\pi}) \tag{11}$$

where ω_z and ω_x are the phase differences between the azimuth antennas and the corresponding elevation antennas, as well as the phase differences between consecutive azimuth antennas, respectively, after Doppler FFT processing.

The conversion from the radar coordinate system to the Cartesian coordinate system is shown in Figure 1, and the calculation is as follows:

$$\begin{aligned} x &= R \times \cos(\varphi) \times \sin(\theta) \\ y &= R \times \cos(\varphi) \times \cos(\theta) \\ z &= R \times \sin(\varphi) \end{aligned} \tag{12}$$

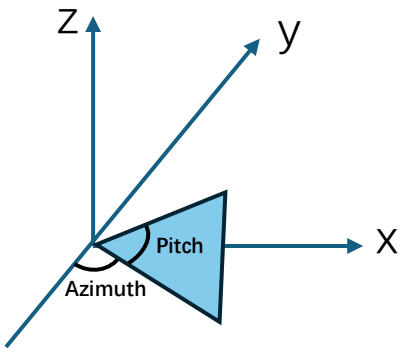


Fig. 1. Conversion diagram

2.2 Point cloud network

With the development of deep learning, the traditional Convolutional Neural Network (CNN) has achieved remarkable results in fields such as image processing and speech recognition. By virtue of local perception and weight sharing, it can effectively extract features from regular grid data (such as images) and perform task classification. However, with the popularization of devices such as lidar and depth cameras, point cloud data needs to be processed in application scenarios such as autonomous driving and pose recognition. Point cloud data has the characteristics of sparsity, disorder, and irregularity, which do not conform to the fixed input format (matrix or image grid structure) that traditional CNNs rely on. A point cloud is composed of a large of discrete points, and each point contains spatial coordinates (such as x, y, z) and other possible attributes (such as intensity, color). Unlike images, point cloud data has no fixed structure, and the relative positions between points have different degrees of importance for tasks, so it is difficult for traditional deep learning models to directly handle the spatial relationships of point clouds.

To solve this problem, point cloud networks have emerged as the times require. Their aim is to design deep learning models that can handle disordered and irregular point cloud data. For example, PointCNN [13] performs local convolution operations on

point clouds through the X-transformation method, overcoming the disorder and irregularity of point clouds. The demand for point cloud networks is driven by the need for processing three-dimensional spatial data in practical applications, especially in the field of recognition, where real-time and accurate understanding and analysis of the surrounding environment are crucial. Point cloud data is high-dimensional, sparse, and difficult to standardize, and traditional methods cannot meet these requirements. Point cloud networks have made innovations in feature extraction, spatial relationship modeling, and temporal dynamic processing, further promoting the application and development of deep learning technology in the processing of three-dimensional spatial data.

3 Design of ST-PCN

In this chapter, we will introduce the proposed ST-PCN model method, which is capable of utilizing raw millimeter-wave radar point clouds for human pose recognition. Our method consists of three main components: point cloud uniformization, data augmentation, and the ST-PCN model.

3.1 Point cloud homogenization

To ensure that the number of points in each frame of point cloud data is consistent while preserving the spatial characteristics and diversity of the data, we propose a point cloud augmentation strategy based on dynamically calculated offsets. This method combines the local statistical features of the point cloud data, avoiding unnecessary redundancy during data augmentation while enhancing data diversity. First, we set an appropriate threshold k. When the number of points in a frame exceeds the predefined threshold k,we randomly discard some points until the remaining number of points equals k. When the number of points is less than the threshold k,we augment the point cloud by adding points with locally perturbed offsets to the existing points. Specifically, we randomly select $k-n$ points from the original point cloud (where n is the number of points in the current frame) and apply perturbation offsets calculated based on the statistical features of the current frame to each added point. The method for calculating the perturbation offsets is as follows:

Calculate the Offset for Each Feature, for all n points in each frame, we compute the mean (μ_x、 μ_y、 μ_z、 $\mu_{\text{int}ensity}$) and standard deviation (σ_x、 σ_y、 σ_z、 $\sigma_{\text{int}ensity}$) of each feature (e.g., x, y, z, and intensity). Taking x and intensity as examples:

$$\mu_x = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{13}$$

$$\mu_{\text{int}ensity} = \frac{1}{n}\sum_{i=1}^{n} \text{int}ensity \tag{14}$$

$$\sigma_x = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \mu_x)^2} \tag{15}$$

$$\sigma_{\text{int}ensity} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\text{int}ensity - \mu_{\text{int}ensity})^2} \tag{16}$$

Calculate the Perturbation Offsets, for each newly added point, a small offset is generated based on the statistical features of the current frame. The offset is determined by the mean and standard deviation of the current frame's point cloud. The generated perturbations Δ_x、 Δ_y、 Δ_z、 $\Delta_{\text{int}ensity}$ are defined as:

$$\Delta_x = \mathrm{N}(0, \sigma_x/100) \tag{17}$$

$$\Delta_{\text{int}ensity} = \mathrm{N}(0, \sigma_{\text{int}ensity}/100) \tag{18}$$

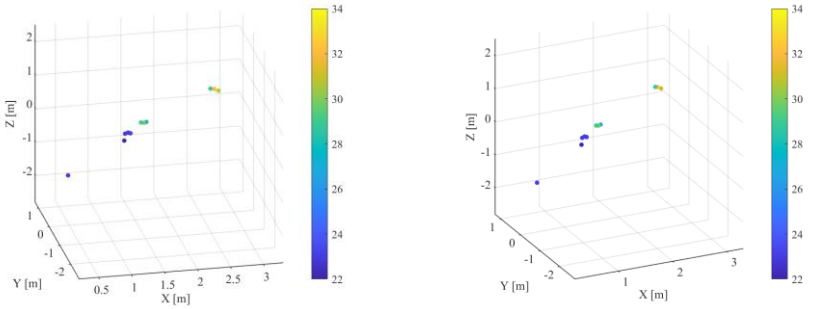
where $\mathrm{N}(0,\sigma)$ represents Gaussian noise with a mean of 0 and standard deviation σ. This ensures that the perturbations are dynamically adjusted according to the actual distribution of each frame's point cloud, introducing only minor adjustments without deviating from the overall distribution of the original point cloud.

Apply the Perturbations, for each newly generated point (x′,y′,z′,intensity′), the calculated perturbations are applied:

$$\begin{aligned} x_i^{'} &= x_i + \Delta x \\ y_i^{'} &= y_i + \Delta y \\ z_i^{'} &= z_i + \Delta z \end{aligned}$$

$$\text{int}ensity_i^{'} = \text{int}ensity_i + \Delta\,\text{int}ensity \tag{19}$$

By generating perturbation offsets based on the actual distribution of each frame's point cloud, we ensure that the newly generated points do not deviate from the distribution characteristics of the original point cloud, avoiding structural damage that might be caused by random noise. Figure 2(a) shows the distribution of a point cloud frame before uniformization, and Figure 2(b) shows the same frame after uniformization. It can be observed that the spatial distribution of the point cloud remains largely unchanged before and after processing. Therefore, this method allows us to ensure uniform point cloud counts per frame without disrupting the original spatial distribution of the point cloud.



(a) Point Cloud before uniform (b) Point Cloud after uniform

Fig. 2. Point cloud distribution before and after uniformization.

3.2 Data Augmentation

Radar sensors detect the distance, velocity, and shape of objects by emitting electromagnetic waves and receiving reflected signals [14]. In dynamic point cloud data, the motion of objects can be divided into two types: tangential motion and radial motion [15]. Tangential velocity refers to the velocity of an object along a direction parallel to the radar's line of sight. Tangential velocity primarily affects the position and geometric shape of the object in the radar image, causing changes in the reflected signal received by the radar, particularly in terms of angle and intensity [16]. Radial velocity, on the other hand, refers to the velocity of an object along the radar's line of sight, primarily affecting the distance between the object and the radar, thereby influencing the time delay and frequency changes of the radar's received signal.

The intensity of the signal received by the radar is related to the reflectivity, distance, and shape of the object. When rotating the point cloud, the reflected signal of the object does not change significantly due to variations in the rotation angle, especially when the object's reflectivity is uniform. Rotation only changes the relative position of the object without altering its distance from the radar (i.e., the radial distance), and thus does not cause significant changes in intensity. Therefore, rotation operations can effectively simulate radar data from different observation angles without introducing significant intensity disturbances

We use a rotation matrix R(θ) to transform the coordinates of each point to a new position. Assuming the coordinates of each point in the point cloud in three-dimensional space are (x , y , z), the rotated coordinates (x′,y′,z′) can be obtained using the following formula:

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = R(\theta) \cdot \begin{bmatrix} x \\ y \\ z \end{bmatrix} \tag{20}$$

where the rotation matrix R(θ) is defined as:

$$R(\theta)=\begin{bmatrix}\cos\theta & -\sin\theta & 0\\ \sin\theta & \cos\theta & 0\\ 0 & 0 & 1\end{bmatrix} \tag{21}$$

Excessive rotation angles may cause unnatural changes in the geometric structure of the point cloud and may indirectly affect the intensity feature. On the other hand, too small rotation angles may not effectively increase data diversity, resulting in insignificant data augmentation effects. Therefore, it is necessary to select an appropriate rotation angle to ensure that data diversity is increased while maintaining the geometric consistency of the point cloud and minimizing interference with the intensity feature.

3.3 Model Design

Point cloud data is divided into static point clouds and dynamic point clouds. Static point clouds are usually used to capture three-dimensional information of static scenes, such as buildings, terrain or industrial parts scanning. It does not contain a time dimension, and the data represents the spatial distribution at a certain moment, so it is suitable for 3D modeling, mapping and static scene analysis [17]. Dynamic point clouds introduce the time dimension and can capture the motion process of targets in the scene. Because factors such as object motion, occlusion, sensor viewpoint changes and sampling rate will change with the motion of targets, the number of point clouds captured in different frames will be different.

In a typical scenario, the complete process of a human body posture is a dynamic process. Usually, a complete posture from the start to the end is composed of many frames of point clouds, and we refer to this kind of point cloud as dynamic point cloud. In our previous work, we have already carried out homogenization processing on the point cloud data of each frame, so that the point cloud of each frame can be directly used as the input of the neural network. In this part, we will design a network model that is capable of handling the data structure of dynamic point clouds, which possess both spatial and temporal features. The overall framework is shown in Figure 3.

First is the spatial feature extraction stage. In this stage, the TNet method is introduced. The TNet method learns a transformation matrix through affine transformations such as rotation and translation of point cloud data to adjust the input point cloud so that it can be processed in a unified spatial framework, thus ensuring that the extraction of spatial features is not affected by unnecessary spatial transformations. We first process each frame of point cloud data through a TNet, and then pass the data through the first convolutional layer for preliminary feature extraction to obtain 64-dimensional features. To enhance the modeling ability of local geometric relationships, the model introduces a Spatial Deal module, and adds the output of this module to the original features in the form of residual connections. The enhanced features are then processed by the second TNet, and finally the feature dimension is gradually increased through two convolutional layers, and each frame of point cloud is compressed into a 1024-dimensional feature vector through global max pooling.

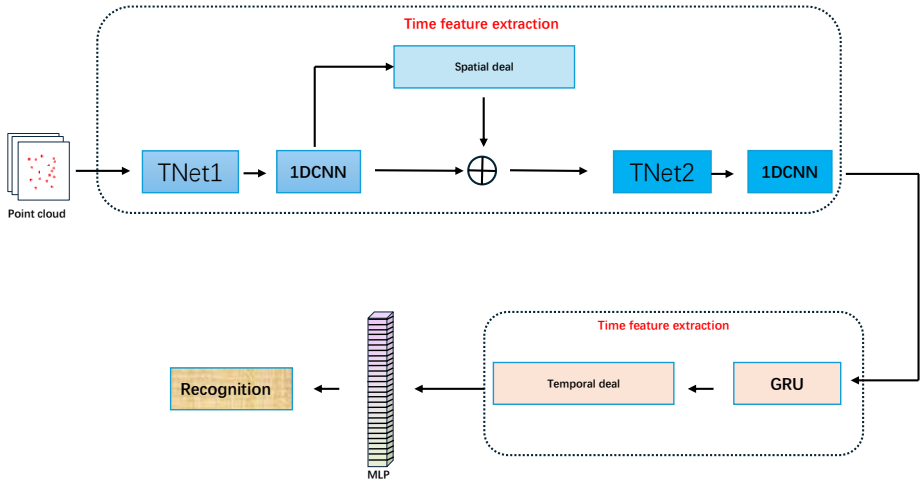


Fig. 3. Overall architecture of ST-PCN model

The core of the temporal feature extraction stage is the GRU-based feature extraction method. GRU consists of two gating units - the update gate and the reset gate, which control the flow of information and memory updates respectively, enabling GRU to decide how to update the hidden state at each time step based on the input and past memory states. Their mechanisms are as follows:

$$\begin{aligned} \mathrm{Zt} &= \sigma(W_z \cdot [h_{t-1}, x_t]) \\ r_t &= \sigma(W_r \cdot [h_{t-1}, x_t]) \\ \tilde{h}_t &= \tanh(W \cdot [r_t \odot h_{t-1}, x_t]) \\ h_t &= (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t \end{aligned} \tag{22}$$

where z_r and r_t are the reset gate and update gate respectively, h_{t-1} is the hidden state of the previous time step, x_t is the input of the current time step, and W_z , W_r ,W are the parameters of the gating operation. This gating mechanism of GRU enables it to gradually update the model's memory according to the temporal relationship when processing dynamic point cloud data, thereby capturing the motion and changes of objects in the time dimension. In the temporal feature extraction stage, the processed 1024-dimensional features of all frames are stacked in chronological order to form a temporal feature tensor. First, temporal modeling is performed through a two-layer GRU network, and then the obtained features are processed by introducing a Temporal Deal module. Finally, the last 128-dimensional feature is taken and mapped to the output through a fully connected layer.

Since static objects remain unchanged while dynamic activities have temporal persistence. For continuous actions such as human postures, the spatial position information of each frame of point cloud will change, and the point cloud data between frames has temporal continuity. Therefore, when extracting dynamic point cloud features, it is necessary to comprehensively consider the correlation of time series and

obtain global features in the spatial dimension. To this end, this paper designs a spatiotemporal attention feature extraction module, which aims to better model the spatiotemporal dependence in point cloud data through the spatiotemporal attention mechanism. This module consists of a Spatial Deal module and a Temporal Deal module. The former assigns different importance weights to each point in the point cloud in the spatial dimension to emphasize the global information of local spatial features, while the latter dynamically adjusts the weights between frames in the time series dimension to capture the temporal dependence between different time steps. For the spatial level, each frame of point cloud consists of N points, and each point in the point cloud has a d-dimensional feature vector $x_i \in R^d$. Its spatial relationship can be extracted by calculating the similarity of each point to other points in the point cloud. We calculate the Query (Q), Key (K) and Value (V) spaces:

where W_q、W_k、$W_v \in \Re^{C\times C}$ are learnable weight matrices and C is the feature dimension.

First, we calculate the attention score:

$$A = soft\max\left(\frac{QK^T}{\sqrt{C}}\right) \in \Re^{B\times N\times N} \tag{23}$$

where B is the batch size, N is the number of points.

Then perform weighted output:

$$F_{out} = AV \tag{24}$$

Finally, perform residual connection:

$$F_{\text{final}} = F_{\text{out}} + F \tag{25}$$

Similarly, for the temporal level, we take the output containing T frames of information after the spatial attention mechanism and temporal feature extraction as the input of the temporal attention mechanism. By calculating the similarity between frames, the model can assign different temporal weights α_t to each frame, thereby emphasizing the time steps related to the current frame and weakening the influence of irrelevant frames. Through the joint attention weights of space and time, the model can flexibly focus on the key spatial regions and time series in the dynamic point cloud data, thereby improving the classification accuracy in the dynamic point cloud classification task.

4 Experimentation

4.1 Datasets

The MMActivity dataset is one of the few publicly available datasets for human pose recognition research using millimeter-wave radar [18]. The dataset was recorded using the Texas Instruments IWR1443BOOST millimeter-wave radar and includes five types of actions: boxing, jumping jacks, jumping, squatting, and walking. Each point cloud contains spatial coordinates (x, y, z) and radar intensity information. Since the samples in the MMActivity dataset are common human poses and align with the application scenarios of this study, we use this dataset to evaluate the proposed dynamic point cloud recognition method.

4.2 Implementation Details

In the point cloud uniformization process, considering the dataset characteristics, we set the point cloud quantity threshold per frame to 40. Using data augmentation methods, each file in the dataset is rotated by ±10◦ clockwise and counterclockwise respectively to increase data volume, ensuring model generalization and robustness.

We employ an Adam optimizer with fixed learning rate decay (decay rate of 0.1 every 20 epochs) for model training, with initial learning rate set to 5×10−4. The ratio of training set to verification set is 8:2, using multi-class cross-entropy loss function. After 50 validation epochs, the model converges, achieving 97.37% accuracy on the test set. Figure 4 shows the corresponding confusion matrix.

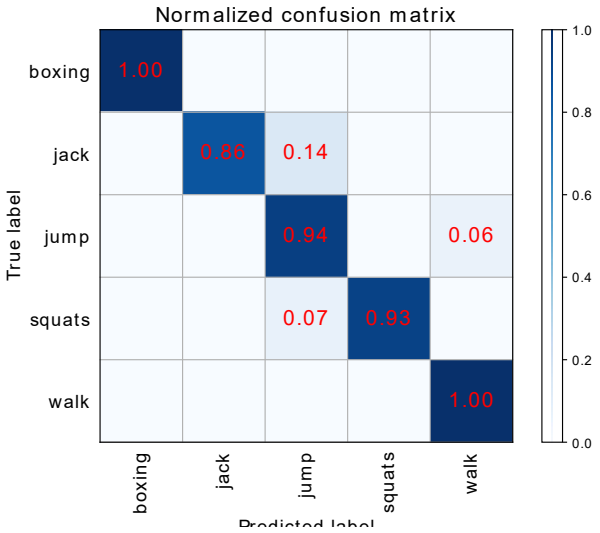


Fig. 4. Confusion matrix of ST-PCN model test set

To further validate model stability and generalization capability, we conduct 5-fold cross-validation and perform statistical analysis on the mean accuracy across different folds, with results shown in Table 1. The cross-validation results demonstrate relatively consistent performance across different data partitions with minimal overall fluctuation, indicating model robustness to random data splitting. Further t-test results show all p-values between different folds greater than 0.05, suggesting statistically insignificant accuracy differences between folds.

Table 1. Cross-validation results

Fold Comparison	t-statistic	p-value
Fold1 vs Fold2	0.32	0.74
Fold1 vs Fold3	0.55	0.57
Fold1 vs Fold4	0.33	0.74
Fold1 vs Fold5	0.03	0.97

4.3 Comparative Experiments

In this part, we compared the recognition effects of various baseline models on the MMActivity dataset, as shown in Table 2. Methods 1–4 are sourced from reference [18], including traditional machine learning algorithms such as Support Vector Machine (SVM) and Multi-Layer Perceptron (MLP), as well as deep learning models such as Convolutional Neural Network (CNN) and Long Short-Term Memory network (LSTM). These methods all process the point cloud data after voxelization (Voxel). Although they can extract certain spatial features, they have performance bottlenecks when dealing with sparse and irregularly structured raw point cloud data, especially in terms of temporal sequence modeling where their capabilities are limited. Method 5 [19] and Method 6 [20] belong to the modeling methods for point cloud sequences. They respectively introduce spatial feature extraction and temporal sequence modeling modules, which can better mine the dynamic information in the point cloud data, with recognition accuracies of 94.97% and 94.19% respectively. Although these methods perform excellently in dynamic point cloud modeling, they still lack effective mechanisms in aspects such as key frame selection and attention to key areas, making it difficult to fully focus on the core spatio-temporal features during the action process. The TD-PCN model proposed in this paper acts directly on the raw point cloud data, and introduces spatial attention and temporal attention mechanisms during the spatio-temporal feature extraction process, thus achieving accurate modeling of the key spatio-temporal features. In the recognition task, the accuracy of TD-PCN reaches 95.35%, which is significantly better than the above various baseline methods. Further ablation experiments show that the accuracy of the model with only the spatial attention mechanism introduced is 86.24%, and the accuracy of the model with only the temporal attention mechanism introduced is 92.47%. This indicates that the temporal attention mechanism plays a crucial role in capturing dynamic changes, enabling the model to more effectively model the temporal dependencies in the point cloud sequence; while the spatial attention mechanism enhances the model's perception ability of the local

spatial structure and improves the accuracy of point cloud representation. However, relying solely on the spatial attention mechanism makes it difficult to fully mine the motion patterns in the temporal dimension. After combining the two, TD-PCN achieves a further improvement in recognition accuracy, verifying the complementarity and synergistic effect between the modeling of spatial features and temporal features.

Table 2. Test results of different models

No	Input Format	Model	Accuracy
1	Voxels	SVM	63.74%
2	Voxels	MLP	80.34%
3	Voxels	Bidirectional LSTM	80.34%
4	Voxels	Time-distributed CNN + Bidirectional LSTM	90.47%
5	Point Cloud	PointLSTM	94.97%
6	Point Cloud	Pantomime	94.19%
7	Point Cloud	TD-PCN	95.35%
8	Point Cloud	TD-PCN (With temporal attention)	92.47%
9	Point Cloud	TD-PCN (With spatial attention)	86.24%

5 Conclusion

In this work, we propose a neural network model capable of processing dynamic point clouds based on millimeter-wave radar. This model eliminates the additional step of converting dynamic point clouds into other representations, as required by previous approaches. Furthermore, by incorporating a spatiotemporal attention mechanism during the processing of sparse point clouds, the model effectively extracts spatiotemporal features from dynamic point clouds while achieving a favorable balance between parameter quantity and model performance. Experimental results demonstrate that the TD-PCN model outperforms traditional methods in sparse point cloud classification tasks, showing significant performance improvements compared to baseline models. This study provides a novel approach for the efficient processing of millimeter-wave radar point clouds, particularly suitable for dynamic scene perception tasks that require direct extraction of pose information from sparse point clouds.

References

1. Qi, W.; Fan, H.; Xu, Y.; Su, H.; Aliverti, A. ASD-CLDNN Based Multiple Data Fusion Framework for Finger Gesture Recognition in Human-Robot Interaction. In Proceedings of the 2022 4th International Conference on Control and Robotics (ICCR), Guangzhou, China, 2 - 4 December 2022; pp. 383 - 387.
2. Y. Wang, Y. Xiao, F. Xiong, W. Jiang, Z. Cao, J. T. Zhou, and J. Yuan, "3dx- 3d dynamic voxel for action recognition in depth video," in IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 2020.

3. M. A. R. Ahad, A. D. Antar, and M. Ahmed, IoT Sensor-Based Activity Recognition - Human Activity Recognition, ser. Intelligent Systems Reference Library. Springer, 2021, vol. 173. [Online]. Available: https://doi.org/10.1007/978-3-030-51379-5
4. Zhang, W.; Wang, J.; Lan, F. Dynamic hand gesture recognition based on short-term sampling neural networks. IEEE/CAA J. Autom. Sin. 2020, 8, 110-120.
5. Zhang, Huafeng, Liu, Kang, et al. TRANS-CNN-Based Gesture Recognition for mmWave Radar. Sensors, 2024, 24(6).
6. M. Gu, Z. Chen, K. Chen, and H. Pan, "Ir-sit: A lightweight transformer network for human fall detection based on fmcw radar," IEEE Sensors Journal, 2023.
7. G. Wang, Y. Gu, and T. Inoue, "A Survey on Millimeter-Wave Radar for Human Activity Recognition: Technologies, Applications, and Challenges," IEEE Sensors Journal, vol. 21, no. 3, pp. 2528-2543, Feb. 2021.
8. Chen, X., et al. "Attention-Based Deep Learning for Radar-Based Human Activity Recognition." IEEE Transactions on Neural Networks, 2023.
9. Zhang, Y., et al. "Deep Learning for Micro-Doppler Classification: A CNN Approach." ICASSP, 2022.
10. Zhang, X., et al. "3D Human Pose Estimation from Radar Point Clouds Using Voxelization and Deep Learning." IEEE Transactions on Radar Systems, 2021.
11. Wang, Y., et al. "Graph Neural Networks for Radar-Based Human Activity Recognition." ICASSP, 2022.
12. Li, Z., et al. "Depth Image Generation from Radar Point Clouds for Action Recognition." IEEE Sensors Journal, 2021.
13. Li, Y., Bu, R., Sun, M., Wu, W., Di, X., & Chen, B. (2018). PointCNN: Convolution On X-Transformed Points. NeurIPS 2018.
14. A. Ouaknine et al., "Radar Signal Processing for Autonomous Driving: Challenges and Recent Advances," IEEE Signal Processing Magazine, 2023.
15. M. Schafer et al., "Motion Decomposition in Radar Point Clouds for Autonomous Vehicles," IEEE International Conference on Robotics and Automation (ICRA), 2022.
16. J. Lombacher et al. Radar-Based Motion Estimation Using Deep Learning for Autonomous Driving. IEEE Transactions on Intelligent Vehicles, 2021.
17. Y. Guo et al. A Review of 3D Point Cloud Processing and Learning for Autonomous Driving. IEEE Transactions on Intelligent Transportation Systems, 2022.
18. Zhang, Y., et al. “RadHAR: Human Activity Recognition from Point Clouds Generated through a Millimeter-wave Radar.” Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies (IMWUT), vol. 5, no. 3, 2021.
19. MIN Y., ZHANG Y., CHAIX, et al. An efficient point LSTM for point clouds based gesture recognition [C]//2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)
20. PALIPANAS, S., SALAMI D., LEIVALA, et al. Pantomime: mid-air gesture recognition with sparse millimeter-wave radar point clouds [J]. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, 2021, 5 (1): 1 - 27.