TRANS-CNN Based Gesture Recognition for mmWave Radar

Huafeng Zhang , Kang Liu * , Yuanhui Zhang

Posted Date: 5 February 2024

doi: 10.20944/preprints202402.0228.v1

Keywords: mmWave Radar; Dynamic Gesture Recognition; Multi-Head Self-Attention Mechanism; TRANS-CNN; Point Cloud

Article

TRANS-CNN Based Gesture Recognition for mmWave Radar

Huafeng Zhang, Kang Liu * and Yuanhui Zhang

College of Mechanical and Electrical Engineering, China Jiliang University, Hangzhou 310018,China;zhanghf@cjlu.edu.cn(HZ);zyh@cjlu.edu.cn(YZ);

* Correspondence: kang.liu@cjlu.edu.cn

Abstract: In order to improve real-time performance of gesture recognition by micro-Doppler map of mmWave Radar, the point cloud based gesture recognition for mmWave Radar is proposed in this paper.Two steps are carried out for mmWave Radar based gesture recognition. The first step is to estimate the point cloud of the gestures by 3D-FFT and the peak grouping. The second step is to train the TRANS-CNN model by combining the multi-head self-attention and the 1D-convolutional network, so as to extract the features in the point cloud data at a deeper level to categorize the gestures. In the experiments, TI mmWave Radar sensor IWR1642 is used as benchmark to evaluate the feasibility of the proposed approach. The results show that the accuracy of the gesture recognition reaches 98.5%. In order to prove the effectiveness of our approach, a simply 2Tx2Rx Radar sensor is developed in our lab and the accuracy of recognition reaches 97.1%. The results show that our proposed gesture recognition approach achieves the best performance in real-time with limited training data, in comparison with the existing methods.

Keywords: mmWave Radar; Dynamic Gesture Recognition; multi-head self-attention mechanism; TRANS-CNN; point cloud

1. Introduction

In the field of intelligent interaction [1], gesture recognition is used extensively in applications such as smart homes [2,3], robot control [4], autonomous driving [5], and AR/VR [6]. At present, gesture recognition technologies by wearable devices [7] or vision sensors [8] are well developed. However, these systems capture gesture signals through accelerometers and gyroscopes [9–11], which require users to wear many sensor devices, leading to a suboptimal user experience. On the other hand, gesture recognition technology by vision sensors often relies on depth cameras to capture depth images of gestures [12–14] for recognition. Nevertheless, depth cameras are susceptible to ambient light interference, and depth images may contain substantial user information, posing risks of privacy breaches [15]. Additionally, these sensors may suffer from high power consumption and susceptibility to environmental factors [16].

The rapid development of mmWave Radar sensors in recent years has provided new ideas for gesture recognition. MmWave Radar is characterized by high frequency, high resolution, and low power consumption [17], which can provide accurate detections, making it possible to perform non-contact sensing and recognition of gestures, and is relatively insensitive to common interfering factors such as occlusion, rain, and snow. Therefore, mmWave Radar has apparent advantages in gesture recognition. Compared with camera-based gesture recognition technology [18], it can avoid privacy leakage and the influence of light.

The most popular gesture recognition framework consists of two main parts: feature extraction and machine learning. The feature extraction could be range-Doppler map, micro-Doppler map and range-azimuth image. The most effective machine learning approaches are neural networks. The pioneer gesture recognition work by Google designed mmWave Radar(Soli) [19] utilizes the range-Doppler map as the training data. Lars et al. utilize the micro-Doppler map [20] to train the ResNet

 doi:10.20944/preprints202402.0228.v1

network, combined with migration learning to correct the original network to complete the gesture recognition [21]. Shrestha A. et al. train the micro-Doppler features by LSTM and Bi-LSTM networks [22]. The networks were validated with an accuracy of more than 90%. The work by K Alirezazad et al. [23] combines the range-Doppler map and the range-azimuth image to train a two-stream artificial neural network. An accuracy of 92.5% was achieved. The micro-Doppler map based features are cluttered by complex environments, and they have impacts on model training and robustness.

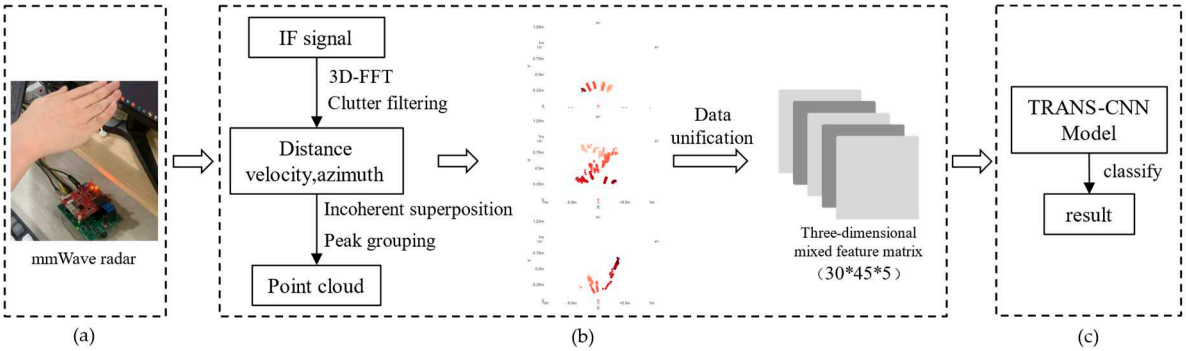
Recent years the Transformer Model [24] has made some achievements in natural language processing (NLP) [25], computer vision (CV) [26], and other fields. The transformer mode performs better gesture recognition than other neural networks for time series of Radar data. The research [27] in achieves better results by introducing a Transformer for sequence modeling of hand gestures. Biao Jin utilizes micro-Doppler maps to train the 2DCNN+Transformer model for gesture classification with 98% accuracy [28]. Kehelella K et al. combined the Transformer with a convolutional encoder to propose a vision converter-based HGR architecture and achieved 98.3% accuracy [29]. Song Y combined convolutional and attentional mechanisms to collect gesture echoes using the MMWCAS radar data to generate hybrid feature-time maps of distance-time, Doppler-time, azimuth-time, and elevation-time, which were inputted into a DenseNet-CBAM network to recognize 12 micromanipulation gestures with 99.03% accuracy [30].

The current feature extraction mainly adopts micro-Doppler map or range-Doppler map as the training data. However, this method suffers from the problems of massive data volume, too much redundant information, and insufficient feature extraction. One of our contributions is to use point clouds instead of the micro-Doppler features [31,32]. The second contribution is to embed the attention mechanism by combining the multi-head self-attention mechanism with 1D-convolutional networks [33,34] to extract features in time and spatial domains. In our paper, six features such as distance, velocity, azimuth, X-coordinate, Y-coordinate, and its time index, are used. Compared to micro-Doppler maps, the point clouds significantly reduce the training data size. This innovative approach fully considers the backward and forward correlation in the time series and effectively improves the robustness and generalization ability of the model in complex environments.

The remainder of the paper is divided into six parts. The second part introduces the gesture recognition system, including point cloud detection and network structure design. The third part presents feature extraction approach to build our train data. TRANS-CNN is described in the fourth part. The part five shows the experimental results and the performance is evaluated by comparing different approaches. The last part concludes the paper.

2. Gesture Recognition System

The gesture recognition system mainly consists of two modules: feature extraction and TRANS-CNN model training. Our hand gesture data is captured by using mmWave Radar. The features such as micro-Doppler, range-Doppler, point clouds are extracted as training data set for our TRANS-CNN model. The design of our TRANS-CNN model is to dig up the mechanism between time domain and spatial domain of the hand gesture data. Finally, the training process optimizes the mechanism and provides robust solutions for gesture recognition. The framework of the system is shown in Figure 1.



 doi:10.20944/preprints202402.0228.v1

Figure 1. Framework of TRANS-CNN based Gesture Recognition for mmWave Radar: (a) Data collection; (b) Radar echo signal processing; (c) Gesture recognition.

3. Feature Extraction of Point Cloud Data

As benchmarking, the TI Radar IWR1642 with 2Tx4Rx is used for data capture shown in Figure 2. The operating frequency range of the sensor is from 76GHz to 81GHz, with a maximum bandwidth of up to 4GHz.

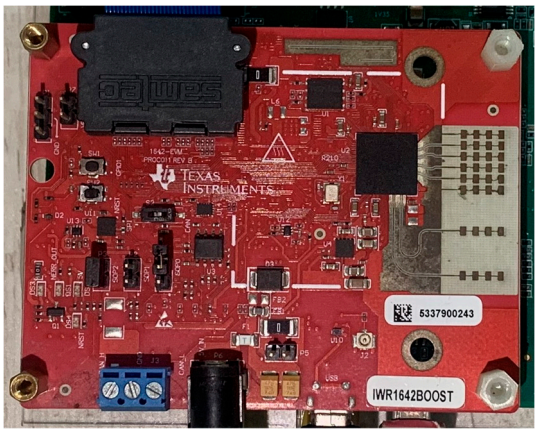


Figure 2. IWR1642 Radar Sensor.

The radar emits a Frequency Modulated Continuous Wave (FMCW):

$$s_\tau(t) = A_\tau \cos[\pi(2 f_0 t + St^2) + \phi_0] \tag{1}$$

Where: A_τ is the amplitude of the transmitted signal, f_0 is the center frequency of the carrier, $S = B / T_c$ is the FMCW slope, B is the bandwidth, T_c is the pulse width, and ϕ_0 is the initial phase.

The echoed signal received after being subjected to delay and Doppler shift caused by gesture movement is:

$$s_r(t) = A_r \cos\pi[2 f_0 + S(t - t_d)](t - t_d) \tag{2}$$

$$f_r = S(t - t_d) + \Delta f_d \tag{3}$$

Where: A_r is the amplitude of the received echo signal, f_r is the frequency of the received signal, $t_d = 2(R_0 + vt) / c$ is the delay time of the echo signal, and $\Delta f_d = -2vf_0 / c$ is the Doppler shift caused by the gesture movement.

The transmission signal and echoed signal are mixed through the radar's built-in mixer, and then undergo filtering by a low-frequency filter to remove irrelevant high-frequency components and noise, resulting in the extraction of a proper intermediate frequency (IF) signal:

$$s_{IF}(t) = s_\tau(t) * s_r(t) = \frac{1}{2} A \cos[2\pi(St_d - \Delta f_d) + 2\pi f_0 t_d] \tag{4}$$

Where: $A = A_r * A_\tau$ is the amplitude of the IF signal.

3.1. Clutter Removal

The captured IF Radar raw data is cluttered in a real-world environment. The environment includes stationary and moving objects.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 February 2024 doi:10.20944/preprints202402.0228.v1

3.1.1. Static Clutter Filter

To address the echo signals caused by static clutter, this study applies a Vector Mean-Cancellation algorithm to the results of 1D-FFT to filter out static clutter in the echo signals. The core idea is to calculate the mean of the intensity for each Chirp and subtract this mean from the original Chirp intensity. This process yields data with static clutter removed. The point cloud after clutter removal is shown in Figure 3, where the blue region in the left image represents targets generated by static objects.

$$R[rt,m,n] = D[rt,m,n] - \frac{1}{N}\sum_{i=1}^{N} D[rt,m,i] \tag{5}$$

Where: $D[rt,m,n]$ is the data containing clutter signal, m is the number of Chirp, and n is the number of sampling points for each Chirp.

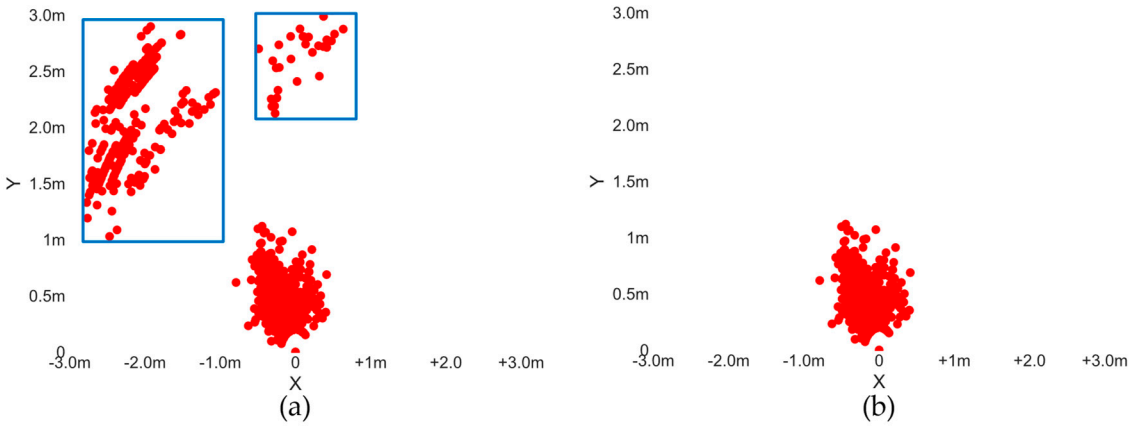


Figure 3. Comparison before and after static clutter filtering: (a) Accumulated point cloud; (b) Removal of static points in (a).

3.1.2. Dynamic Clutter Filter

Since the Radar data is noised by the rest body parts or other pedestrians while gesture capturing, the static filtering process cannot completely filter out dynamic clutter. Therefore, the Constant-False-Alarm-Rate algorithm (CA-CFAR) [32] is used for range-Doppler Radar images. Which adaptively adjusts the decision threshold according to the dynamic clutter in the echo signal. CFAR detection uses the reference unit in the data to estimate the clutter noise and other parameters to obtain the threshold based on the estimated density. If a value exceeds the threshold in the detected data, it is considered that there is a target.

$$H = \frac{K}{N}\left(\sum_{i=1}^{(N-M)/2} x_i + \sum_{j=(N+M)/2+1}^{N} x_j\right) \tag{6}$$

Where: H is the unit's threshold to be measured, K is the threshold for clutter estimation, N is the length of the distance dimension or velocity dimension, M is the length of the protection unit, and x_i / x_j is the distance dimension or velocity dimension input data.

After the clutter removal, the classical 3D FFT is used for estimating point cloud.

3.2. Point Cloud Refinement

In the actual processing of data, to fully demonstrate the effect of dynamic gestures, the false alarm rate of CFAR should not be set too small, which will therefore lead to some dynamic clutter

 doi:10.20944/preprints202402.0228.v1

still existing in the data after 2D CFAR, which is not conducive to the accurate recognition of gestures by the model. Therefore, the peak grouping algorithm [35] is used to group the internal target points of each gesture sample after CFAR. Target points that meet the requirements are retained, and those that do not are deleted. The specific process is outlined below:

(1) Following the distance dimension Constant-False-Alarm-Rate(CFAR), a threshold is set in the distance dimension for peak grouping. When the distance dimension CFAR produces output, and the peak value exceeds the threshold, the distance index of the target point is recorded.

(2) After CFAR processing in the velocity dimension, the same threshold is set in the velocity dimension. The algorithm checks whether there is a velocity dimension CFAR output at the distance index of the target point obtained in step 1 and if the peak value exceeds the threshold. If these conditions are met, the velocity index of the target is recorded.

(3) For each frame within the data, retrieve all eligible points indexed by the target velocity as the final point cloud for the current frame for output.

The effect of the gesture "Hand Wipe from left to right" to the right after grouping is shown in Figure 4.

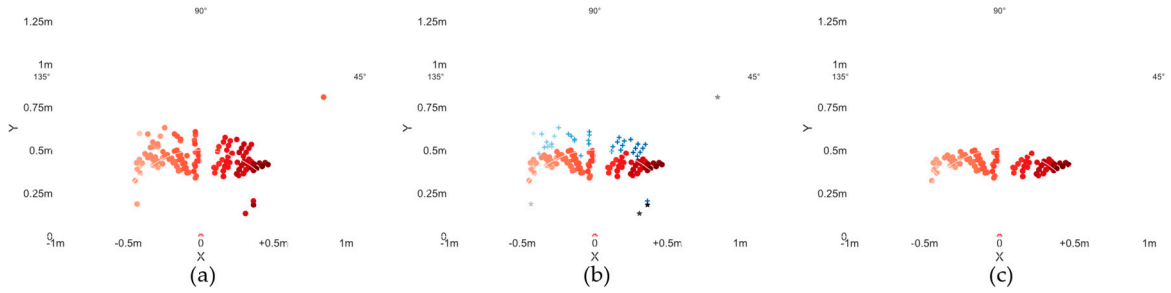


Figure 4. Clutter Removal Refinement by Peak Grouping for Gesture "Wipe from left to right": The colors from light to dark represent the point cloud occurring time (from frame zero to the last frame). A plus sign indicates the invalid points. Pentagrams indicate points that are out of the moving area. (a) Original cumulative point cloud; (b) Peak grouping; (c) Point clouds after peak grouping.

4. TRANS-CNN Model Training

4.1. Preprocessing of Training Data

When analyzing the collected point cloud data, it is found that the number of frames in each sample data and the number of target points in each frame are not the same, while the input of the model requires that the data format remains strictly consistent dimensions. The sample point cloud data is normalized in a matrix with 30 frames by 45 target points. Each point cloud data has 5 features: x, y, Range, Doppler, and Azimuth. The frame cycle is 100ms. If the sample data has smaller matrix, the rest of data matrix is padding with zeros; otherwise, sub-sampling is performed. Finally, the sample point cloud data represented as a 3D matrix with 30 Frames by 45 Points by 5 Features is obtained.

4.2. TRANS-CNN Model

In 2017, the attention mechanism was first noticed and applied in Transformer [24]. Today, the attention mechanism has been widely used in NLP, CV, and other fields [25,26] and has achieved great success. The characteristics of point cloud data are very similar to natural language; both are time series data and have spatial characteristics. Therefore, TRANS-CNN model for gesture recognition is constructed by combining the attention mechanism with a 1D convolutional (Conv1D), as shown in Figure 5. The model mainly consists of the attention module, the serial 1D convolution module, and the fully connected layer. This part reviews the TRANS-CNN model.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 February 2024 doi:10.20944/preprints202402.0228.v1

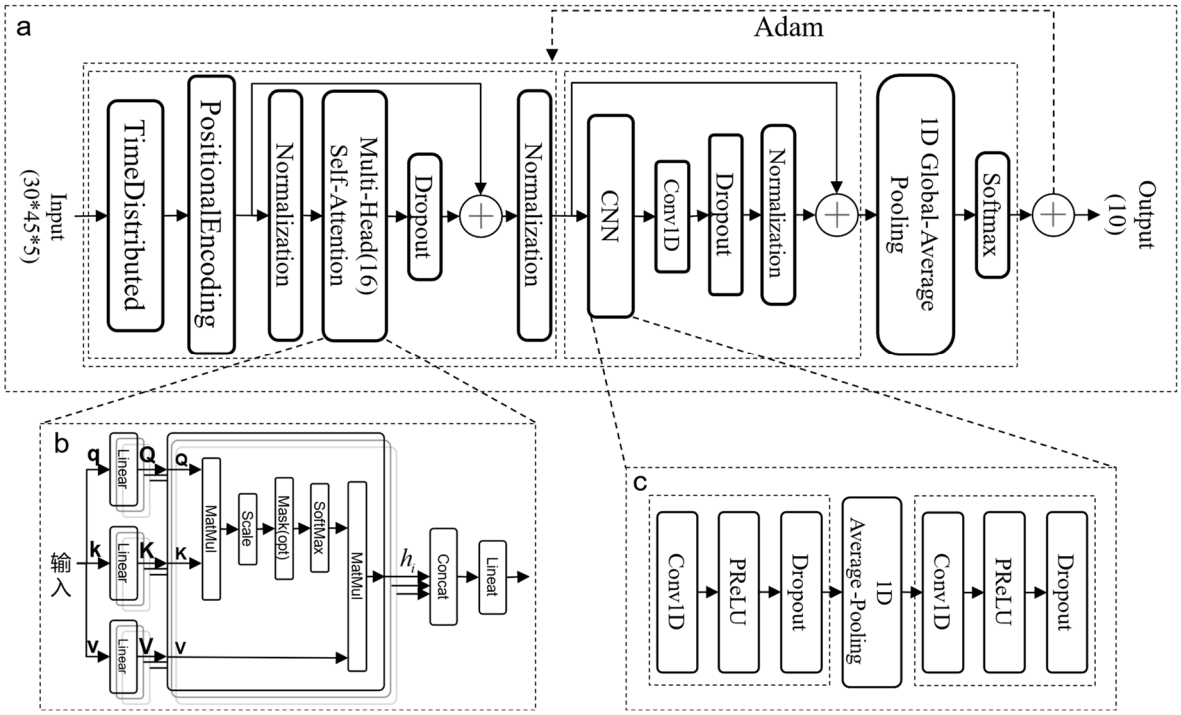


Figure 5. TRANS-CNN network structure: (a) overall network structure of gesture recognition; (b) multi-head self-attention mechanism based on Scaled Dot-Product; and (c) serial 1D convolution module composed of two 1D convolution layers.

4.2.1. Attention Module

The structure of the attention module network in this paper is mainly improved based on the Transformer encoder network. It is divided into two pieces: temporal position encoding and computation of dependencies between sequence targets.

The workflow of the time sequence position coding part is as follows: Firstly, the three-dimensional mixed feature tensor (30*45*5) is distributed and flattened in the time dimension through the Time Distributed function to ensure that all the data of each time frame participate in the calculation of the attention mechanism. Then, the Positional Encoding function is used to add nonlinear encoding values to the sequence using the standard position encoding method (sine function and cosine function combination) so that the model can better learn the relative relationship between different positions. Finally, it is copied and normalized by:

$$\tilde{X} = \frac{X_i - \bar{X}}{\sqrt{\sigma_X^2 + K}} \tag{7}$$

Where: $\tilde{X}$ is the data after normalization, X_i is the feature of the ith position of each frame, $\bar{X}$ is the mean of X_i, σ_X^2 is the standard deviation of X_i, and K is a constant.

Utilizing the attention mechanism based on Scaled Dot-Product as shown in Figure 5(b), continuous learning and computing the interdependence between different positions of target points in the target sequence are carried out. The variable A is duplicated and assigned to the query vector Q, the key vector K, and the value vector V, so as to realize the self-attention mechanism. The distinctiveness between different gestures is adequately extracted by:

$$h_i = \text{soft}\max\left(\frac{Q * K^T}{\sqrt{d_k}}\right) \bullet V \tag{8}$$

Where: h_i is the output of the ith attention header, and d_k is the length of K.

 doi:10.20944/preprints202402.0228.v1

Single-head attention has some limitations in extracting gesture features, and it is difficult to fully capture the gesture features in the sample data [34]. In contrast, in the multi-head attention mechanism, the Q, K, and V of each sub-network of the attention head are independent of each other, and different feature expressions can be learned in different subspaces. Therefore, this paper proposes a multi-head self-attention mechanism to concatenate the results of sub-networks of these single attention heads. Subsequently, a linear layer continuously learns the dependencies between different targets. The learned weight matrix is utilized to map the multi-head self-attention output to the input data's dimensions. Including the multi-head attention mechanism in the model effectively improves the expressive power of the model.

$$MH = MultiHead(Q,K,V) = Concat(h_1,h_2,\cdots,h_i)*W \tag{9}$$

Where: W is the weight matrix learned by the linear layer, $MH = MultiHead(Q,K,V)$ is the output of the attention module, and $Concat(\bullet)$ is the connection function.

4.2.2. Serialization of 1D Convolution Modules

In the process of multi-head self-attention computation, the focus is on the dependencies between targets at different positions within a sequence, but less attention is paid to the dependencies between different sequences. Therefore, after the attention module, this paper introduces a one-dimensional convolutional network and constructs a serial one-dimensional convolutional module, the structure of which is shown in Figure 5(c). It is used to capture the dependencies between different sequences.

A one-dimensional convolution operation is performed for each frame of data output from the attention module. The process utilizes Eq. 14 to extract features between sequences in different time spaces using appropriate step sizes and convolution kernels. Subsequently, the results are input into an activation function for nonlinear transformation, and the extracted features are fed into an average pooling layer for down-sampling. Next, the down-sampled results are inputted into the next one-dimensional convolutional layer to perform the same feature extraction operation.

$$\begin{aligned} x_{i,j}^{n} &= f\Big(\sum_{\substack{i\in 30\\ j\in 225}} x_{i,j}^{n-1} * w_{i,j}^{n} + b_{i,j}^{n}\Big) \\ y_{i,j} &= Down(x_{i,j-1}) \end{aligned} \tag{10}$$

Where: $f(\bullet)$ is the activation function (PReLU), $x_{i,j}$ is the input data and the output data, i, j is the position of the target point, n is the first convolutional layer, w^n is the weight matrix of the convolutional layer, b is the bias, $Down(\bullet)$ is the average pooling function, and $y_{i,j}$ is the pooled feature.

The output of the sequential one-dimensional convolutional module undergoes normalization through a one-dimensional convolutional layer and a regularization layer. Subsequently, a residual connection is employed to fuse the output with the input of the sequential one-dimensional convolutional module. Finally, the gesture is recognized by the global average pooling and a fully connected layer (Softmax).

5. Experimental Result

5.1. Radar Configuration

The IWR1642 mmWave radar from TI was used in this experiment, transmitting FMCW through time-division multiplexing mode (TDM-MIMO). The parameter configurations of the radar are detailed in Table 1. The distance resolution of the target was obtained from the calculations to be 0.039 m, and the radial velocity resolution was obtained to be 0.125 m/s.

 doi:10.20944/preprints202402.0228.v1

The experimental platform also includes a computer with an I9-12900H processor, an RTX3060 (6G) graphics card, and 16GDDR5 RAM. The TRANS-CNN model was built using the TensorFlow-gpu 2.7.0 and Keras 2.7.0 frameworks.

Table 1. Radar Parameters.

Radar parameter	value
Starting Frequency	77GHz
Bandwidth	3.85GHz
Frequency Modulation Slope	20MHz/us
Sampling Rate	2MHz
Frame Cycle	100ms
Sampling Points/Chirp	256
Chirp Number/Frame	32

5.2. Data Acquisition and Training Data Collection

While collecting data, the radar was placed horizontally, in the vertical direction, 30cm to 85cm (±10cm) away from the radar, and in the horizontal direction, -65cm to 65cm range of motion. Different palm shapes were used for data collection to increase the diversity of the dataset. Example of the test setup and the palm shapes for gestureis shown in Figure 6.

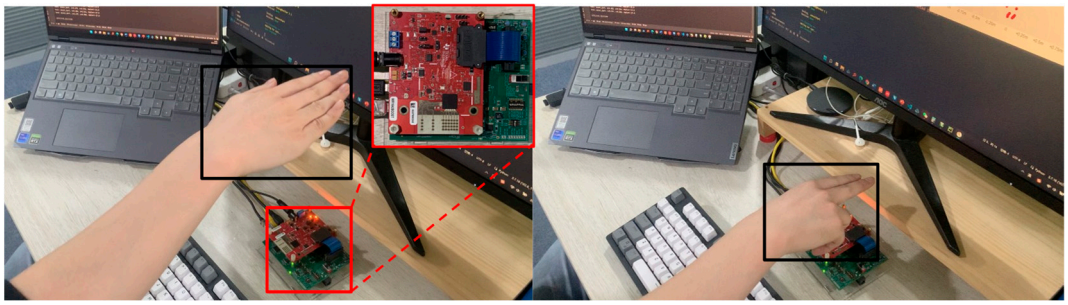


Figure 6. Two palm shapes used for hand gestures.

To demonstrate the concept of our gesture recognition system, ten typical gesture movements are defined in our tests, including wipe up (up), wipe down (down), wipe to the left (left), wipe to the right (right), rotating clockwise (CW), rotating counterclockwise (CCW), drawing a hook (√), drawing an X, drawing a Z, and drawing an S. Five volunteers (3 men and 2 women) are invited to participate in the data collection. A total of 10,000 sets of sample data are collected for 5 participants, 10 gestures, 2 palm shapes, and 100 groups for each palm shape. Each data set contains five significant features, resulting in a multi-dimensional point cloud dataset. A specific example of a gesture action and its corresponding multi-frame cumulative point cloud is shown in Figure 7. The colors from light to dark represent the movement direction of the gesture.

5.3. Recognition Results and Evaluation

The input data of the model is uniformly processed as 30*45*5, and the Epoch is set to 200. The Early Stopping function is used, and the model is developed to stop training and save the best model when the loss value of the model is no longer decreasing within 25 consecutive Epochs on the test set. The loss function uses the categorical cross entropy function, which is widely used in multi-objective classification applications. The Adam optimizer is selected and the learning rate was set dynamically adjustable. If the loss value is not decreasing within 10 Epochs on the test set, the learning rate is reduced to 20% of the current learning rate. The initial learning rate was set to 0.001. The data splitting ratio is 70:30, which means 70% of the data is for training and 30% for testing. All models in the experiment were trained using the above parameter settings.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 5 February 2024 doi:10.20944/preprints202402.0228.v1

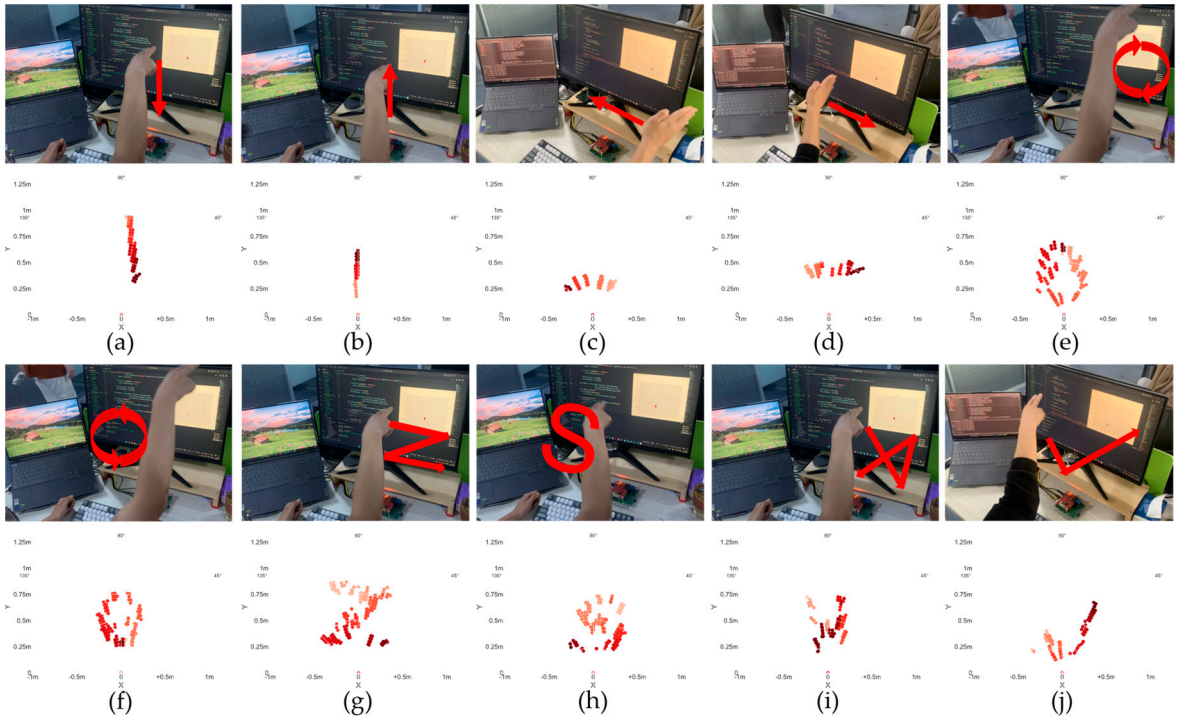


Figure 7. Gesture examples with corresponding cumulative point cloud: (a) wipe down (down), (b) wipe up, (c) wipe to the left, (d) wipe to the right, (e) rotating clockwise, (f) rotating counterclockwise, (g) drawing a Z, (h) drawing an S, (i) drawing an X, and (j) drawing a hook (√). For example, the Gesture Down indicates the hand moves from top to bottom facing the Radar sensor. The points are the detections by the Radar sensor.

To investigate the effect of multiple attention heads (Num_heads) on the model performance, 1, 4, 8, 16, and 32 attention heads were selected for the experiment, and the default Batch_size was used to train the model, and the model accuracy was obtained to be 85.4%, 91.8%, 92.43%, 97.91%, and 95.83%, respectively, as shown in Figure 8(a). It can be seen that the model accuracy achieves the maximum when the Num_heads is 16.

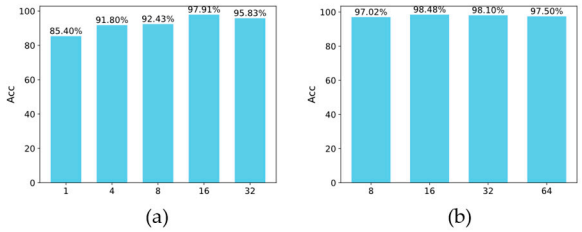


Figure 8. Model Accuracy for Different Num_heads and Batch_size: (a) represents the accuracy of different Num_heads corresponding models; (b) represents the model's accuracy corresponding to different Batch_size.

After setting Num_heads to 16, the effect of Batch_size on model performance is further explored; in the experiment, 8, 16, 32, and 64 are chosen to train the model, respectively, and the model accuracy are obtained as 97.02%, 98.48%, 98.1%, 97.50%, as shown in Figure 8(b). Among them, the highest model accuracy is obtained when the Batch_size is 16.

While the Num_heads and Batch_size is set to 16, the curves of accuracy and loss values are shown in Figure 9. The training results show that the model converges rapidly, reaching over 96% accuracy in just 20 Epochs, while the average accuracy of the final model stabilizes at around 98.5%.

 doi:10.20944/preprints202402.0228.v1

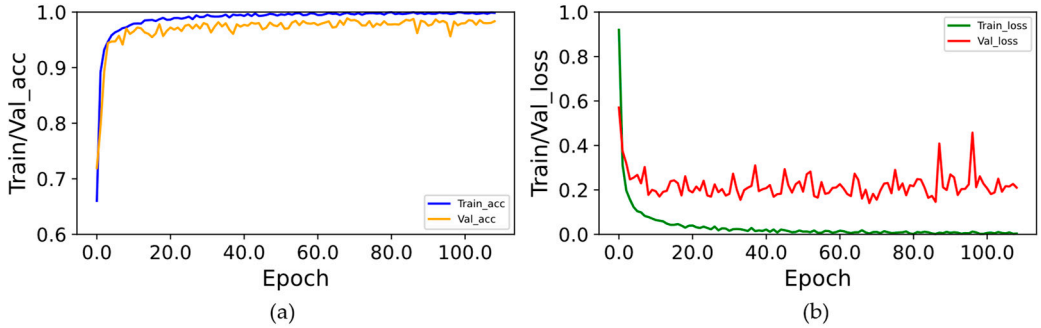


Figure 9. The Acc/Loss curve of the TRANS-CNN model: (a) model accuracy curve; (b) model error curve.

To evaluate the performance of the TRANS-CNN model, 1500 sample data are collected again, where 150 samples are for each gesture, the confusion matrix is drawn on the validation set at the same time. The statistics for each gesture are calculated by using Recall, Precision, and F1-Score, and the detailed results are shown in Table 2. Among them, the eight gestures of Up, Down, Left, CCW, draw S and draw √ have a precision of more than 97%, which shows that the TRANS-CNN model is effective for gesture recognition. The accuracy rate for gesture recognition of Right, CW, and X reaches more than 96%, while the gesture recognition accuracy of Z is 94%. The average recognition accuracy of the ten gestures reaches 98.4%.

Table 2. Confusion Matrix of 10 Gestures and their Evaluation Parameters.

True \ Predict	Up	Down	Left	Right	CW	CCW	Z	S	X	√	Recall (%)
Up	149	1	0	0	0	0	0	0	0	0	0.9917
Down	0	149	0	1	0	0	0	0	0	0	0.9917
Left	0	0	149	1	0	0	0	0	0	0	0.9906
Right	0	0	1	148	0	0	0	0	0	1	0.9867
CW	0	2	2	0	146	0	0	1	0	0	0.9750
CCW	0	0	0	0	0	148	0	2	0	0	0.9850
Z	0	1	1	3	0	0	144	1	0	0	0.9567
S	0	0	0	0	0	1	0	149	0	0	0.9933
X	0	0	0	0	0	0	0	0	150	0	0.9983
√	0	0	0	1	0	0	0	0	0	149	0.9787
Prec (%)	97.5	97.9	98.3	96.8	96.6	97.1	94.1	97.7	96.8	98.9	Acc=98.4%
F1(%)	98.8	98.4	97.6	97.7	98.5	96.8	97.3	98.5	97.8	98.7	

5.4. Comparison of Different Gesture Recogntion Approaches

5.4.1. TRANS-CNN compared with other models

Table 3 compares the TRANS-CNN model proposed in this paper with the Tesla model and the four models Self-Attention, 2DCNN-Transformer, 8HBi-GRU, and DenseNet-CBAM that use micro-Doppler maps as inputs, as well as the VGG16 model that uses the optical flow [18] to compute the trajectory of the gesture motions as input images. Although the accuracy rate reaches 99%, the data processing is troublesome, the data volume is significant, and it is impossible to avoid the effects generated by occlusion and illumination changes. Reference [36] used a self-attention mechanism to construct a model to capture micro-Doppler maps by radar to recognize eight types of gestures, but the accuracy rate is only 94.8%. Literature [28] introduced a two-dimensional convolutional network (Conv2D) before the Transformer model to recognize six types of gestures. Although there are only

 doi:10.20944/preprints202402.0228.v1

six types of gestures, the accuracy rate is increased by 3.2%, indicating that introducing a convolutional network helps improve the model accuracy, but it also increases the model complexity while improving the accuracy rate. Literature [33] recognizes 12 types of gestures through an 8-layer self-attention mechanism coupled with a GRU recurrent network. Compared with the 2DCNN-Transformer model of literature [28] and the Self-Attention model of literature [36], the accuracy is improved by 0.2% and 3.4%, respectively, and the training time of each Epoch is only 2.41s, which results in a fast model convergence and high accuracy. Reference [30] constructed micro-Doppler maps of gesture features by radar and designed a DenseNet-CBAM model for gesture recognition with an average accuracy of 99.03%. It shows that multi-head attention combined with recurrent networks can significantly help to improve the model accuracy.

Table 3. Comparison of Recognition Accuracy of Different Models.

Model	Dataset	Radar	Number of samples	Type of gesture	Iteration time/s	Acc/%
VGG16[18]	Trajectory image	-	2000	10	-	99
Self-Attention[36]	micro-Doppler maps	BGT60TR24B	1600	8	-	94.8
2DCNN-Transformer[28]		IWR1642	2700	6	-	98
8HBi-GRU[31]		AWR1243	7200	12	2.41	98.2
DenseNet-CBAM[30]		MMWCAS	-	12	-	99.03
Tesla[32]	Point Cloud	IWR1443	12097	5	-	97.5
TRANS-CNN		**IWR1642**	**10000**	**10**	**4.7**	**98.5**
		ADT6101	**480**	**4**	**1.3**	**97**

In contrast, the TRANS-CNN model in this paper uses point cloud data as input, making it easier to realize real-time gesture recognition than micro-Doppler maps as input data. In addition, the experiments demonstrate strong robustness and generalization ability, with a 1% improvement in accuracy. Compared with micro-Doppler maps, point cloud data reduces the size of training data set, which accelerates the model's training process and significantly improves the real-time performance in recognition.

5.4.2. Evaluation of Recognition on Simple Radar Sensor

Using the mmWave Radar ADT6101-4P with 2Tx2Rx developed in our lab shown in Figure 10, the data acquisition was carried out for the training and testing of the model. In this experiment, two volunteers were invited to perform data acquisition for four gestures, Up, Down, Left, and Right, with 120 groups for each gesture, totaling 480 groups of point cloud data. Subsequently, the dataset splitting ratio is 70:30 for TRANS-CNN model training and testing. The model converges quickly, and the specific test results are listed in Table 3. Using the ADT6101 radar for gesture recognition, it remains above 97%. The results indicate that the TRANS-CNN model can achieve significant performance on different radar devices.

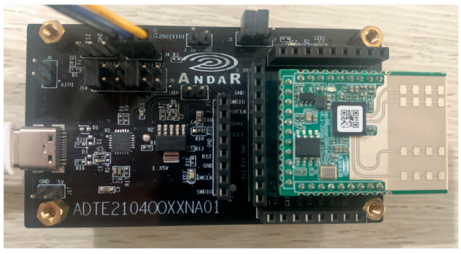


Figure 10. ADT6101 radar. 2Tx2Rx is designed for our simple Radar sensor to evaluate gesture recognition model.

 doi:10.20944/preprints202402.0228.v1

To more comprehensively evaluate the model's performance on different devices, the accuracy (Acc) variation curves of the IWR1642 and ADT6101 radar data are compared on the test set (see Figure 11 for details). The results show that the model exhibits high accuracy on both radars.

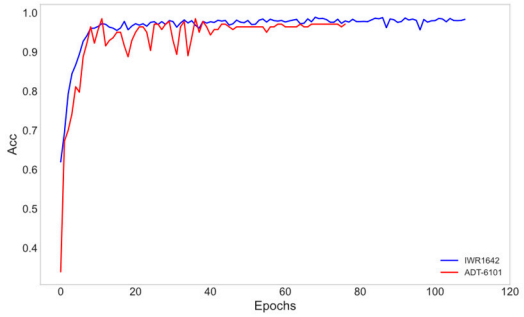


Figure 11. Acc change curves of IWR1642 (blue curve) and ADT6101 (red curve).

The confusion matrix for recognizing four gestures using the ADT6101 radar on the testing set is depicted in Figure 12. The confusion matrix illustrates that the recognition accuracy for both "up" and "down" gestures surpass 97%, while "left" and "right" gestures also approach 97%. Based on the actual gesture recognition performance, The TRANS-CNN model performs the gesture recognition task well in either radar.

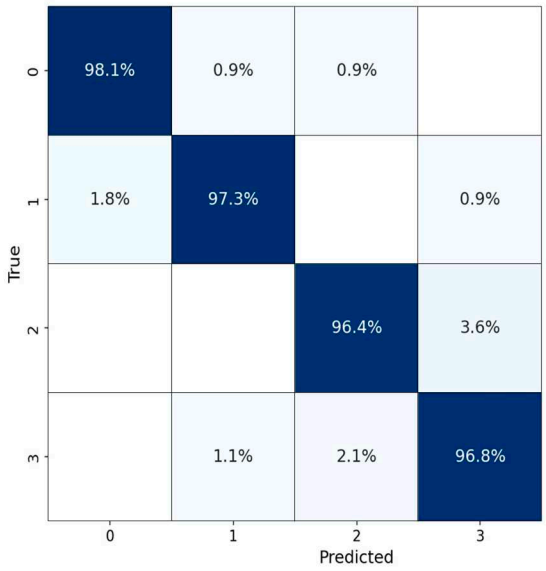


Figure 12. Confusion Matrix of ADT6101 Radar. The class is defined as 0:Up, 1:Down, 2:Left, 3:Right.

6. Conclusions

In this paper, the TRANS-CNN model constructed by using point clouds as training data is proposed to solve the problems of high redundancy of information in micro-Doppler maps, insufficient feature extraction, and large amounts of data, which significantly improves the speed of gesture recognition and the possibility of model deployment. The gesture features are fully extracted using the multi-head self-attention mechanism and one-dimensional convolutional network to realize ten gestures. The robustness and generalization ability of the model is improved. Regarding to effectiveness, real-time, generalization ability, and gesture recognition accuracy, our proposed TRANS-CNN gesture recognition is better performed than other gesture recognition approaches. In

the experiments, the proposed gesture recognition is evaluated on the benchmarking TI Radar sensor, and also adapted to the by our lab designed Radar sensor with the simple antenna configuration of2Tx2Rx. The results show that our approach has potential applications in modern human-computer interaction. Considering the accuracy and efficiency of the TRANS-CNN model, subsequent research will continue to investigate the situation in more complex and variable environments and multi-person gesture interference.

Supplementary Materials: The following supporting information can be downloaded at: https://youtu.be/WergAXIPS1M, Video: mmWave radar gesture recognition.

Author Contributions: Conceptualization, Huafeng Zhang, Kang Liu and Yuanhui Zhang; Data curation, Huafeng Zhang; Funding acquisition, Kang Liu and Yuanhui Zhang; Investigation, Kang Liu and Yuanhui Zhang; Methodology, Huafeng Zhang and Kang Liu; Project administration, Huafeng Zhang and Kang Liu; Resources, Kang Liu and Yuanhui Zhang; Software, Huafeng Zhang and Yuanhui Zhang; Supervision, Kang Liu; Validation, Huafeng Zhang; Writing – original draft, Huafeng Zhang; Writing – review & editing, Kang Liu. All authors will be informed about each step of manuscript processing including submission, revision, revision reminder, etc. via emails from our system or assigned Assistant Editor.

Funding: This work is supported by mmWave Radar Joint Lab Project NO 20222024-03103-21179. Part of the work is funded by Natural Science Foundation of Zhejiang Province of. China under Project NO. 21F010057 "Research on 3D FMCW-SAR Imaging of 77GHz mmWave Radar".

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data sets in this study will not be made public for legal/ethical reasons but may be made available upon reasonable request.

Acknowledgments: The authors would like to acknowledge the support from editors and comments from all the reviewers.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Qi W, Fan H, Xu Y, et al. A 3D-CLDNN Based Multiple Data Fusion Framework for Finger Gesture Recognition in Human-Robot Interaction. 2022 4th International Conference on Control and Robotics (ICCR). Guangzhou, China, 02-04 December 2022; pp. 383-387.
2. Verdadero M S, Martinez-Ojeda C O, Cruz J C D. Hand gesture recognition system as an alternative interface for remote controlled home appliances. 2018 IEEE 10th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM).Baguio City, Philippines, 29 November 2018-02 December 2018; pp. 1-5.
3. Wisener W J, Rodriguez J D, Ovando A, et al. A Top-View Hand Gesture Recognition System for IoT Applications. 2023 5th International Conference on Smart Systems and Inventive Technology (ICSSIT). Tirunelveli, India, 23-25 January 2023; pp. 430-434.
4. Qi W, Ovur S E, Li Z, et al. Multi-sensor guided hand gesture recognition for a teleoperated robot using a recurrent neural network[J]. IEEE Robotics and Automation Letters, 2021, 6(3), 6039-6045.
5. Kim K M, Choi J I. Passengers' gesture recognition model in self-driving vehicles: Gesture recognition model of the passengers' obstruction of the vision of the driver. 2019 IEEE 4th International Conference on Computer and Communication Systems (ICCCS). Singapore, 23-25 February 2019; pp. 239-242.
6. Chen T, Xu L, Xu X, et al. Gestonhmd: Enabling gesture-based interaction on low-cost vr head-mounted display[J]. IEEE Transactions on Visualization and Computer Graphics, 2021, 27(5), 2597-2607.
7. Yuan G, Liu X, Yan Q, et al. Hand gesture recognition using deep feature fusion network based on wearable sensors[J]. IEEE Sensors Journal, 2020, 21(1), 539-547.
8. Breland D S, Dayal A, Jha A, et al. Robust hand gestures recognition using a deep CNN and thermal images[J]. IEEE Sensors Journal, 2021, 21(23), 26602-26614.
9. Jiang S, Kang P, Song X, et al. Emerging wearable interfaces and algorithms for hand gesture recognition: A survey[J]. IEEE Reviews in Biomedical Engineering, 2021, 15, 85-102.
10. Ling Y, Chen X, Ruan Y, et al. Comparative study of gesture recognition based on accelerometer and photoplethysmography sensor for gesture interactions in wearable devices[J]. IEEE Sensors Journal, 2021, 21(15), 17107-17117.
11. Tong L, Ma H, Lin Q, et al. A novel deep learning Bi-GRU-I model for real-time human activity recognition using inertial sensors[J]. IEEE Sensors Journal, 2022, 22(6), 6164-6174.

12. Gavrilova M L, Wang Y, Ahmed F, et al. Kinect sensor gesture and activity recognition: New applications for consumer cognitive systems[J]. IEEE Consumer Electronics Magazine, 2017, 7(1), 88-94.
13. Zhang W, Wang J, Lan F. Dynamic hand gesture recognition based on short-term sampling neural networks[J]. IEEE/CAA Journal of Automatica Sinica, 2020, 8(1),110-120.
14. León D G, Gröli J, Yeduri S R, et al. Video hand gestures recognition using depth camera and lightweight cnn[J]. IEEE Sensors Journal, 2022, 22(14), 14610-14619.
15. Tang G, Wu T, Li C. Dynamic Gesture Recognition Based on FMCW Millimeter Wave Radar: Review of Methodologies and Results[J]. Sensors, 2023, 23(17), 7478.
16. Ahmed S, Kallu K D, Ahmed S, et al. Hand gestures recognition using radar sensors for human-computer-interaction: A review[J]. Remote Sensing, 2021, 13(3), 527.
17. Rao S. Introduction to mmWave sensing: FMCW radars[J]. Texas Instruments (TI) mmWave Training Series, 2017, 1-11.
18. Kavyasree V, Sarma D, Gupta P, et al. Deep network-based hand gesture recognition using optical flow guided trajectory images. 2020 IEEE applied signal processing conference (ASPCON). Kolkata, India, 07-09 October 2020; pp. 252-256.
19. Lien J, Gillian N, Karagozler M E, et al. Soli: Ubiquitous gesture sensing with millimeter wave radar[J]. ACM Transactions on Graphics (TOG), 2016, 35(4), 1-19.
20. Hazra S, Santra A. Robust gesture recognition using millimetric-wave radar system[J]. IEEE sensors letters, 2018, 2(4), 1-4.
21. Fhager L O, Heunisch S, Dahlberg H, et al. Pulsed millimeter wave radar for hand gesture sensing and classification[J]. IEEE Sensors Letters, 2019, 3(12), 1-4.
22. Shrestha A, Li H, Le Kernec J, et al. Continuous human activity classification from FMCW radar with Bi-LSTM networks[J]. IEEE Sensors Journal, 2020, 20(22), 13607-13619.
23. Alirezazad K, Maurer L. FMCW Radar-Based Hand Gesture Recognition Using Dual-Stream CNN-GRU Model. 2022 24th International Microwave and Radar Conference (MIKON). Gdansk, Poland, 12-14 September 2022; pp. 1-5.
24. Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[J]. Advances in neural information processing systems, 2017, 30.
25. Tetko I V, Karpov P, Van Deursen R, et al. State-of-the-art augmented NLP transformer models for direct and single-step retrosynthesis[J]. Nature communications, 2020, 11(1), 5575.
26. Li Y, Yao T, Pan Y, et al. Contextual transformer networks for visual recognition[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2022, 45(2), 1489-1500.
27. Kowdiki M, Khaparde A. Adaptive hough transform with optimized deep learning followed by dynamic time warping for hand gesture recognition[J]. Multimedia Tools and Applications, 2022, 1-32.
28. Jin B, Ma X, Zhang Z, et al. Interference-Robust Millimeter-Wave Radar-Based Dynamic Hand Gesture Recognition Using 2D CNN-Transformer Networks[J]. IEEE Internet of Things Journal, 2023.
29. Kehelella K, Leelarathne G, Marasinghe D, et al. Vision Transformer with Convolutional Encoder–Decoder for Hand Gesture Recognition using 24-GHz Doppler Radar[J]. IEEE Sensors Letters, 2022, 6(10), 1-4.
30. Song Y, Wu L, Zhao Y, et al. High-Accuracy Gesture Recognition using Mm-Wave Radar Based on Convolutional Block Attention Module. 2023 IEEE International Conference on Image Processing (ICIP). Kuala Lumpur, Malaysia,08-11 October 2023; pp. 1485-1489.
31. Palipana S, Salami D, Leiva L A, 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.
32. Salami D, Hasibi R, Palipana S, et al. Tesla-rapture: A lightweight gesture recognition system from mmwave radar sparse point clouds[J]. IEEE Transactions on Mobile Computing, 2022.
33. Zhao Y, Song Y, Wu L, et al. Lightweight micro-motion gesture recognition based on MIMO millimeter wave radar using Bidirectional-GRU network[J]. Neural Computing and Applications, 2023, 35(32), 23537-23550.
34. Chen Y S, Cheng K H, Xu Y A, et al. Multi-Feature Transformer-Based Learning for Continuous Human Motion Recognition with High Similarity Using mmWave FMCW Radar[J]. Sensors, 2022, 22(21), 8409.
35. Sieranoja S, Fränti P. Fast and general density peaks clustering[J]. Pattern recognition letters, 2019, 128, 551-558.
36. Hazra S, Santra A. Radar gesture recognition system in presence of interference using self-attention neural network. 2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA). Boca Raton, FL, USA, 16-19 December 2019; pp. 1409-1414.