Hindawi
Journal of Sensors
Volume 2023, Article ID 2951812, 14 pages
https://doi.org/10.1155/2023/2951812

Research Article

Continuous Human Motion Recognition Based on FMCW Radar and Transformer

Liubing Jiang,[1,2,3] **Minyang Wu**,[2,3] **Li Che**,[1,2] **Xiaoyong Xu**,[2,3] **Yujie Mu**,[2,3] **and Yongman Wu**[2,3]

[1]*School of Information and Communication, Guilin University of Electronic Technology, Guilin 541004, China*
[2]*School of Computer and Information Security, Guilin University of Electronic Technology, Guilin 541004, China*
[3]*Key Laboratory of Wireless Broadband Communication and Signal Processing in Guangxi, Guilin University of Electronic Technology, Guilin 541004, China*

Correspondence should be addressed to Liubing Jiang; jlbgl@guet.edu.cn

Received 18 October 2022; Revised 4 December 2022; Accepted 13 December 2022; Published 24 January 2023

Academic Editor: Carlos Marques

Radar-based human motion recognition has received extensive attention in recent years. Most current recognition methods generate a heat map of features through simple signal processing and then feed into a classification-based neural network for recognition. Such an approach can only identify a single action. When a set of data contains information about multiple movements, it can also only be recognized as a single movement. Another point that cannot be overlooked is that continuous action recognition methods are able to recognize continuously changing actions but ignore the issue of whether continuous actions are legitimate or not (continuous actions obtained by stitching together multiple current actions do not conform to real time). In this paper, we propose a continuous action recognition method based on micro-Doppler features and transformer, which translates the micro-Doppler features of continuous actions into machine translation tasks and uses the idea of natural language processing (NLP) to identify continuous action. In order to judge whether the continuous action is legal or not, we also design the action state transition diagram as a constraint condition to strictly control the forward and backward actions. The experimental results show that the method proposed in this paper achieves good recognition accuracy for the recognition of a single action and can also effectively segment and recognize continuous actions.

1. Introduction

Human action recognition (HMR) has received great attention in many areas of daily life, especially surveillance and medical [1]. Traditional solutions have many problems. For example, the solutions relying on cameras have high requirements on ambient light and have hidden dangers of violating personal privacy; solutions relying on wearable sensors need to wear sensors on the human body, which may lead to inconvenient human activities and cannot be used in many scenarios [2, 3]. Thanks to the development of radar miniaturization in recent years [4], the commercial realization of radar-based human action recognition has become possible [5]. As an active detection technology, radar is not affected by external light and does not need to be in contact with the human body. People have carried out a lot of research on radar-based human action recognition. The radar-received echoes contain modulation information of different frequencies, which is called the micro-Doppler effect [6] caused by the micromotion of the human body. Obviously, the key to identifying human motion is to extract the micro-Doppler frequency from the radar-received signal.

The recognition of single actions (only one action exists in an observation time) has been extensively studied. In the early years, researchers extracted handcrafted features directly from radar raw signals [7, 8] or time-frequency signals after simple processing and used traditional machine learning algorithms such as KNN to achieve single action recognition. Molchanov et al. [9] proposed to perform discrete cosine transform (DCT) on the micro-Doppler radar echo signal, then use the DCT coefficients as the features

of moving objects, and finally use a multilayer perceptron (MLP) to classify them. Reference [10] studied the feasibility of using ultra-wideband (UWB) radar to classify different human activities, using UWB radar to collect 8 different types of activities of 8 different people, including walking, running, spinning, punching, jumping, standing, and sitting. Then, use the PCA method to capture the features from the UWB radar echo signal. The feature set includes PCA coefficients, PCA mean and variance, PCA, FFT transformation results, and speed of the target. Finally, SVM is used for classification and recognition, and the obtained classification accuracy exceeds 85%. However, the solution of manually extracting features mainly relies on the expertise and experience of researchers, which greatly limits the accuracy and generality of recognition, because different people do the same action very differently. The identification method using machine learning also has great limitations because of the small feature parameters, which cannot effectively extract the features of the signal. In recent years, the development and maturity of deep learning [11] has provided a new research idea for radar-based human motion research. Compared with manual feature extraction and traditional machine learning, deep learning has stronger automatic learning and feature abstraction capabilities and has a wide range of applications in image processing [12] and natural language processing [13]. Kim and Moon [14] first proposed a deep learning network (DCNN) to process micro-Doppler time-frequency images to recognize human actions. The emergence of recurrent neural network (RNN) [15] can effectively utilize the timing information of micro-Doppler. Reference [16] regards radar spectrogram as a multichannel time series and proposes a deep learning model composed of one-dimensional convolutional neural network (1dcnns) and long short-term memory (LSTM) to meet the recognition accuracy requirements. At the same time, the complexity of 2D-CNN is reduced. Reference [17] extracts a time-varying range-Doppler image (TRDI) representing the time-varying motion characteristics by performing time-frequency analysis on the radar signal and then uses a principal component analysis (PCA) algorithm, or a pretrained convolutional autoencoder (CAE) extracts features from TRDI and feeds the extracted features into a temporal network for recognition.

However, the recognition of individual actions has many limitations. For example, when a person falls first and then stands up over a period of time, the single action recognition algorithm can only identify a single action performed by the target within a small time window, thus ignoring the meaning of continuous actions. For the above reasons, studies on single human actions have been extended to continuous human actions, which makes the study more challenging [18]. In a real environment, people always do one action after another in a row, and the duration of each action is different. This means that coherent actions of different durations represent different meanings, and it also brings challenges to data labeling. Reference [19] proposes a distance Doppler trajectory (DRDT) method to separate a single motion from a continuous motion and then process the single motion. Reference [20] used a superimposed gated recurrent unit network (SGRUN) to process the micro-Doppler time-frequency map to realize continuous action recognition. Inspired by CRNN [21], reference [22] proposes a network composed of multiscale compression and excitation network (MSENet), bidirectional long short-term memory (Bi-LSTM), and connectionist temporal classification (CTC) to identify the micro-Doppler time-frequency map. Reference [23] proposes a multimodal sensor fusion framework based on a multilayer Bi-LSTM network, which fuses the data collected by wearable sensors and the data collected by FMCW radar and input it to the Bi-LSTM network at the same time. In [24], Zhu et al. propose a hybrid classifier which combines both convolutional neural networks (CNNs) and recurrent neural networks (RNNs) for spatial-temporal pattern extraction. Guendel et al. [25] propose a coordinated network using five distributed pulsed ultra-wideband (UWB) radars for continuous activities of daily living recognition in an arbitrary movement direction. Continuous human motion recognition should include the following three elements: (1) what actions the target performs in an observation time window; (2) the start and end time of each action; and (3) the target in an observation time window whether the actions performed are logical (whether the two actions before and after conform to common sense in life). The above three elements require that the algorithm can accurately segment different actions in the time dimension and then identify them, and it is necessary to judge whether the consecutive actions before and after are legal.

In recent years, transformer [26] has achieved great success in the fields of natural language processing [13], image processing [12], and point cloud [27]. In this paper, we propose a continuous action recognition algorithm based on transformer and micro-Doppler features. The framework makes full use of the transformer's mutual attention mechanism, which can well identify human actions that are continuously changing and of unequal duration.

Our contributions are as follows:

(1) A dataset of continuous motion based on micro-Doppler features was collected. Use Texas Instruments AWR1843 and DCA1000 as the hardware platform for the continuous motion data of the human body. By performing signal processing on the raw data, a range-time map (RTM) was obtained, and then, the micro-Doppler time-frequency map within the specified distance range was extracted according to the RTM. We collected a total of 1660 sets of data, including (a) walking, (b) running, (c) squatting, (d) standing up, and (e) jumping

(2) This paper proposes a transformer-based human continuous action recognition network. The network uses the Inception structure block as the feature encoding layer which extracts features from the $256 \times 608 \times 3$ micro-Doppler time-frequency map and encodes them into 37×512-dimensional features that the Transformer can accept. Finally, the encoded features are fed into the Transformer network. The attention mechanism will be made

full use of to implement temporal segmentation and prediction of multiple actions

(3) Whether the existence of two consecutive actions before and after a continuous action conforms to social common sense, this paper proposes a constraint condition to judge the legitimacy of consecutive actions before and after. In the experiment, the network predicts five actions: (a) walking, (b) running, (c) squatting, (d) standing up, and (e) jumping. Constraints are designed to judge whether it is legal to do all consecutively (using the method of splicing multiple single actions to expand the dataset)

2. Signal Model and Process

2.1. The Signal Model of FMCW Radar. FMCW radar adopts the continuous wave system, has high bandwidth and strong range resolution, and can detect ultraclose targets, simple structure, and low cost. Therefore, it is very suitable for detecting human motion at a close range. Generally, FMCW radar system consists of a waveform generator, an antenna array with two transmitters and four receivers, a signal demodulator, and an analog-to-digital converter (ADC). A single chirp signal [28] of FMCW radar can be expressed as

$$x_T = A_T \cos\left(2\pi f_c t + \pi \frac{B}{T_c} t^2 + \varphi(t)\right), \tag{1}$$

where A_T is the transmit power, f_c is the start frequency of chirp, B is the bandwidth of chirp, T_c is the duration of chirp, and $\varphi(t)$ is the noise of phase.

The target will reflect back a delayed signal, which can be expressed as

$$x_R = \alpha A_T \cos\left[2\pi f_c (t - t_d) + \pi \frac{B}{T_c}(t - t_d)^2 + \varphi(t - t_d)\right], \tag{2}$$

where $t_d = 2R(t)/c$ which is used to represent the round-trip time between the signal and the target at a distance of $R(t)$ from the radar and α is the return loss coefficient and c is the speed of light.

The mixed signal $y(t)$ is obtained by mixing the transmitting signal x_T and the receiving signal x_R, which can be approximately expressed as

$$y(t) = A_R e^{j(2\pi f_b t + \varphi_b(t) + \Delta\varphi(t))}, \tag{3}$$

where A_R is the power of the received signal, $f_b = 2BR(T)/cT_c$, and $\varphi_b(t) = 2\pi f_c t_d + \pi B t_d^2/T_c$.

When detecting close targets, the residual phase noise $\Delta\varphi(t)$ can be ignored. Moreover, the value of $\pi B t_d^2/T_c$ is also very small and can be ignored. The final mixed signal can be expressed by the radar sampling data matrix as

$$Y[n, m] = A_R e^{j\left(2\pi f_b n T_f + \frac{4\pi}{\lambda} R\left(nT_f + mT_s\right)\right)}, \tag{4}$$

where n is the corresponding label on the fast time sampling axis, m is corresponding label on the slow time sampling axis, T_f is the ADC sampling time interval on the fast time sampling axis, and T_f is the sampling time interval on the slow time sampling axis.

The mixed signal can be obtained by the difference between the transmitted signal and the received signal. Therefore, the hardware processing is relatively simple, suitable for data acquisition and digital signal processing because of the low frequency mixed signal. Another thing that cannot be ignored is that high bandwidth ensures high range resolution of FMCW radar.

2.2. Signal Process. In order to obtain the speed and distance information of the target, we first add a Hanning window which can use sidelobe to eliminate high-frequency interference and leakage energy to the fast time dimension data of the original signal and then perform RangeFFT [29] to obtain the distance change information of the target. The distance information is selected according to the target, and the appropriate distance gate is selected to obtain the micro-Doppler features within the specified range gate.

Fast Fourier transform (FFT) is a commonly used signal analysis method. It mainly analyzes the components of stationary signals. It can only obtain the frequency components contained in a signal as a whole but cannot know the time when each component appears. The short-time Fourier transform (STFT) [30] splits the entire time-domain process of a signal into an infinite number of small processes of equal length and performs FFT on the small process, within which the signal is considered stationary. Therefore, the short-time Fourier transform can effectively extract the change of the micro-Doppler frequency in each time period. This makes it easier for deep transformer-based networks to extract features in tiny time slices. The formula for the short-time Fourier transform can be expressed as

$$\text{spectrogram}(t, f) = \left| \int_{-\infty}^{+\infty} \tilde{x}(\tau) W(t - \tau) e^{-j2\pi f \tau} d\tau \right|^2, \tag{5}$$

where $W(t)$ is the window function and $\tilde{x}(t)$ is the structured target motion signal.

3. Transformer-Based Continuous Human Motion Recognition

A micro-Doppler time-frequency diagram of a set of continuous motions is shown in Figure 1. Different from single human action recognition, continuous action recognition needs to recognize actions with different action durations. Continuous human action recognition needs to meet the following three elements: (1) what actions the target performs in an observation time window; (2) the start and end time of each action; and (3) in an observation time whether the action performed by the target in the window is logical. Optical character recognition (OCR) [31] needs to detect the location, extent, and layout of text. That is to say, the main problem solved by OCR is where there is text

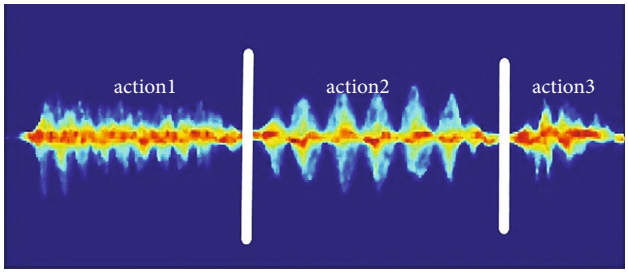


Figure 1: Micro-Doppler time-frequency plot of a group of continuous actions, where "action 1," "action 2," and "action 3" stand for running, walking, and jumping.

and how big is the range of the text. OCR can process the micro-Doppler time-frequency map of continuous motion very well and meets the three elements of continuous human motion recognition. Inspired by the paper [32], this paper proposes a continuous action recognition network consisting of Inception, max pooling layer, transformer, and fully connected layers. The network structure is shown in Figure 2.

3.1. Feature Extraction and Encoding. The traditional single CNN network can only use a fixed-size convolution kernel to extract features from the time-frequency map and obtain good results when processing images with a relatively single feature scale. However, continuous actions consist of multiple actions with different durations, which results in time-frequency maps containing different scales. Traditional single CNN networks are less effective in proposing time-frequency maps containing different scales. To address this issue, we use the Inception network as the backbone network to extract features.

The parameters of the feature encoding layer are shown in Table 1. Specifically, a single Inception block has three convolution kernels of different sizes (1×1, 3×3, and 5×5) to extract features of different scales. In addition, the 1×1 convolution kernel acts before the 3×3 and 5×5 convolution kernels to reduce the dimension of the data and reduce the amount of computation. A maximum pooling layer is connected behind each Inception block to compress the data so that the final output data is a dimension acceptable to the transformer. As shown in Figure 3, the input image is an RGB image of $608 \times 256 \times 3$. After the first and second convolution layers and pooling layers, the original data is compressed into a tensor of $38 \times 16 \times 128$. The lengths are, respectively, 4×1, 2×1, and 2×1 maximum pooling layers, and the original data is finally extracted into a $38 \times 1 \times 512$ feature sequence, which is then input into the transformer network for recognition. The feature extraction layer not only performs feature extraction on the original data but also encodes the original data to obtain the desired data dimension of the transformer.

3.2. Transformer. Transformers have achieved great success in NLP and CV. Inspired by this, we believe that continuous human action recognition based on FMCW radar has a lot in common with natural language processing in a sense. They are essentially translations of the original signal, translating the original signal into corresponding information that people can understand. Transformer consists of a decoder and an encoder, and the model structure is shown in Figure 4.

3.2.1. Multihead Attention. Transformers have achieved great success in NLP and CV largely because of the use of multihead attention mechanism. Scaled dot-product attention is the core component of multihead attention. Although the RNN algorithm has the advantage of using previous information to predict the current state, it has a big shortage. The accuracy of prediction depends heavily on the previous time step. Assume the time prediction and the next n step time. The longer the is, the lower prediction the is. LSTM alleviates the problem of strong time dependence by introducing a forgetting gate to discard useless information in past time steps, but it cannot fundamentally solve this problem. Scaled dot-product attention fundamentally solves this problem. Its structure is shown in Figure 5(a). The formula can be expressed as

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \tag{6}$$

where the softmax function is denoted as $p(i) = \exp(\theta_i^T x)/\sum_{k=1}^{K}\theta_i^T x$. Q, K, and V are short for Query, Key, and Value. d_k is the dimension. Assuming that the input sequence is "Deep Learning," $x1$ and $x2$ are the vectors after adding position encoding to "Deep" and "Learning," respectively; then, these two vectors can obtain the Query, Key, and Value vectors required by operating with the three matrices W^Q, W^K, and W^V. The specific process is shown in Figure 6, and the formula can be expressed as

$$\begin{aligned} Q &= x \times W^Q, \\ K &= x \times W^K, \\ V &= x \times W^V. \end{aligned} \tag{7}$$

Scaled dot-product attention takes the input through three different linear mappings to three intermediate variables Q, K, and V and then uses matrix multiplication to simulate the RNN calculation. In matrix multiplication, each column can be considered as a time step, and any two columns whose distance is the same in matrix multiplication will be calculated. Transformer essentially solves the problem of long-term timing dependence through scaled dot-product attention, but it also brings about the problem that matrix multiplication is disordered, so positional encoding is required to make the input orderly.

As shown in Figure 5(b), compared with scaled dot-product attention, multihead attention uses multiple sets of W^Q, W^K, and W^V to obtain multiple sets of Q, K, and V, and then, the obtained multiple sets of Q, K, and V are spliced together to get the final Q, K, and V matrix. Usually, there are 8 different sets of W^Q, W^K, and W^Vused in transformer. Obviously, multihead attention is essentially a

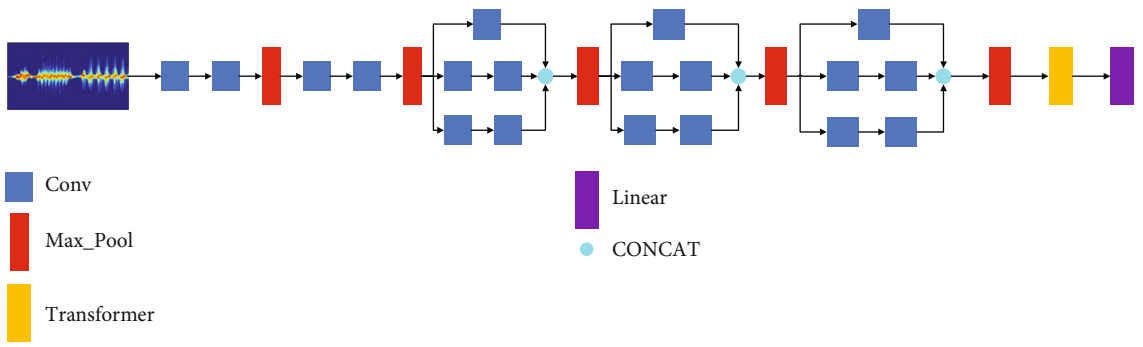
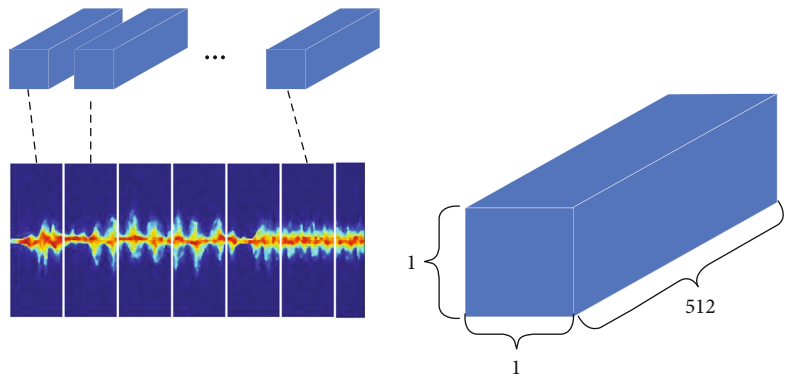


Figure 2: Structure of network for transformer-based continuous human action recognition network. The input is a micro-Doppler time-frequency map. The output is a set of sequences like "ffaaaffbbbfffccc," where "a," "b," and "f" represent "walking," "running," "blank (there is no action within the time window)."

Table 1: Feature encoding layer network parameters.

	Parameters (kernel, maps, stride)	Output(height, width, channel)
Input	—	(256,608,3)
Convolution	$\begin{bmatrix} 3\times3,64,1 \\ 3\times3,64,1 \end{bmatrix}$	(256,608,64)
Max pooling	$4\times4,[4,4]$	(64,152,64)
Convolution	$\begin{bmatrix} 3\times3,128,1 \\ 3\times3,128,1 \end{bmatrix}$	(64,152,128)
Max pooling	$4\times4,[4,4]$	(16,38,128)
Inception 1	$[1\times1,64]\begin{bmatrix} 1\times1,64 \\ 3\times3,128 \end{bmatrix}\begin{bmatrix} 1\times1,64 \\ 5\times5,64 \end{bmatrix}$	(16,38,256)
Max pooling	$4\times1,[4,1]$	(4,38,256)
Inception 2	$[1\times1,128]\begin{bmatrix} 1\times1,128 \\ 3\times3,320 \end{bmatrix}\begin{bmatrix} 1\times1,128 \\ 5\times5,64 \end{bmatrix}$	(4,38,512)
Max pooling	$2\times1,[2,1]$	(2,38,512)
Inception 3	$[1\times1,128]\begin{bmatrix} 1\times1,190 \\ 3\times3,256 \end{bmatrix}\begin{bmatrix} 1\times1,64 \\ 5\times5,128 \end{bmatrix}$	(2,38,512)
Max pooling	$2\times1,[2,1]$	(1,38,512)



(a) Image tensor after feature extraction (b) A single feature sequence

Figure 3: The final tensor dimension of the original data through the feature extraction layer.

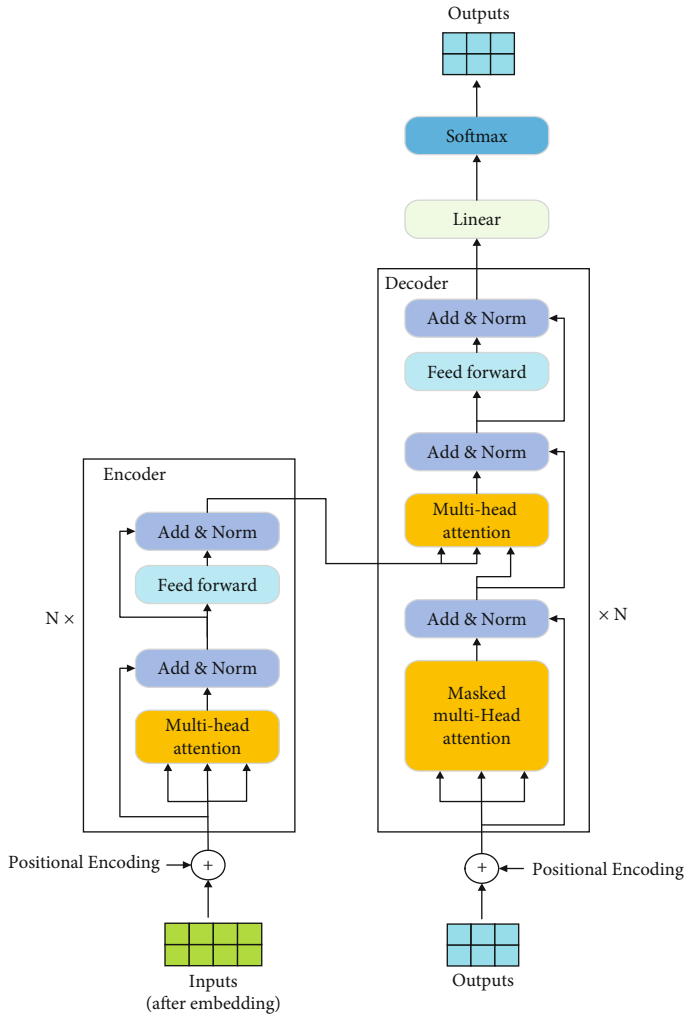
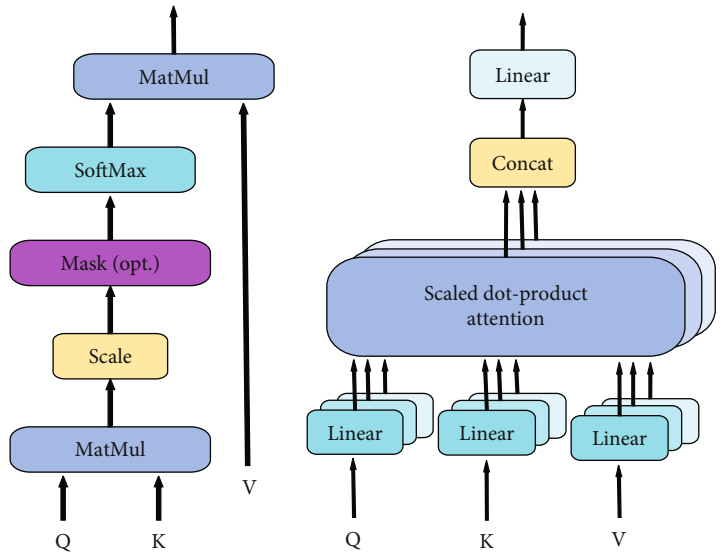


Figure 4: Schematic diagram of transformer model.



(a) Scaled dot-product attention (b) Multihead attention

Figure 5: The difference between two attention machines.

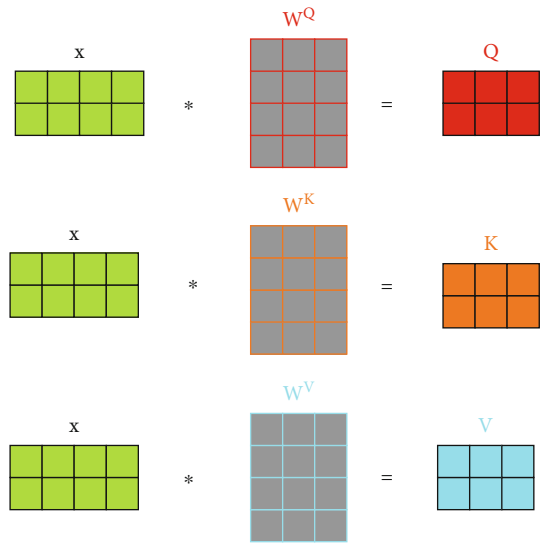


Figure 6: The specific process of obtaining Query, Key, and Value.

splicing of multiple scaled dot-product attentions. Multihead attention can be expressed as

$$\text{Multihead}(Q, K, V) = \text{Concat}(\text{head}_1, \cdots, \text{head}_N)W^o, \quad (8)$$

where $\text{head}_i = \text{attention}(QW_i^Q, KW_i^K, VW_i^V)$.

3.2.2. Positional Encoding. The transformer model does not contain recursion and convolution. In order to allow the model to effectively utilize the sequence order, position coding is introduced to mark the relative and absolute positions of the input sequence. The positional encoding can be expressed as

$$\begin{cases} \text{PE}(\text{pos}, 2i) = \sin\left(\dfrac{\text{pos}}{10000^{2i/d_{\text{model}}}}\right), \\ \text{PE}(\text{pos}, 2i+1) = \cos\left(\dfrac{\text{pos}}{10000^{2i/d_{\text{model}}}}\right), \end{cases} \quad (9)$$

where PE is position encoding, pos is position, and i is corresponding dimension. In addition, d_{model} represents the output size of the transformer, which is also equal to the embedded feature size l.

3.2.3. Position-Wise Feed-Forward Networks. With the exception of the attention sublayer, each layer in the encoder and decoder contains a fully connected feedforward network, which is applied identically to each position, respectively. This consists of two linear transformations with a ReLU activation function in between.

$$\text{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2. \quad (10)$$

Although linear transformations are the same in different locations, they use different parameters between different layers. Another way to describe it is to describe it as two convolutions with kernel size 1.

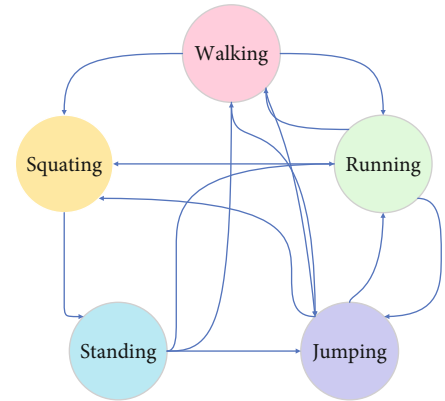


Figure 7: Action constraint state transition diagram.

3.3. Action Constraints. Inspired by the paper [33] on the study of continuous human motion, we propose a constraint to regulate the continuous action and judge whether the continuous action is legal or not (expanding the dataset by splicing multiple single actions). Specifically, as shown in Figure 7, the experiment identifies continuous actions consisting of (a) walking, (b) running, (c) squatting, (d) standing up, and (e) jumping. The three actions of jumping can be transformed into each other, because the same is true in real life. However, the next movement of the squat can only be to stand up, and then, it can be converted into the other three movements, because only then it is true. Another point that cannot be ignored is that walking, running, and jumping can also be converted into squatting, but the next action cannot be converted into standing up. The existence of action constraints can judge whether continuous actions conform to real-life facts, mainly because we will expand the dataset by splicing multiple single actions together, so it is likely to appear next to the squat action. Actions are three other illegal actions other than standing up.

4. Experiment

4.1. Lab Environment. We use the millimeter wave radar hardware experiment platform composed of AWR1843-BOOST radar sensor module and DCA1000EVM data acquisition module developed by Texas Instruments. The starting frequency of the radar is 77 GHZ, the bandwidth is 4 GHZ, the sampling point under a single chirp is 256, there are 128 chirps under a single frame, and the period of a single frame is 30 ms. 300 frames or 9 seconds of data was collected for a single action and two continuous actions, and 400 frames or 12 seconds of data was collected for 3 continuous actions.

The data of the experiment are collected indoors, and there are no other targets in the radar scanning sector except a single tester. The radar is placed on a bracket 0.8 m from the ground, and the tester performs the designated action on the spot 2 meters in front of the radar. The experimental environment is shown in Figure 8. A total of 10 volunteers participated in the data collection before and after, their weight was 41~80 kg, their height was 1.60~1.80 m, and their age was between 22 and 25. A total of 5 movements were

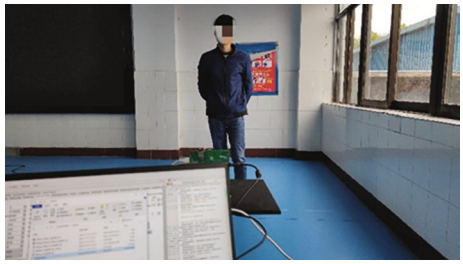


Figure 8: Lab environment.

collected in the experiment, namely, (a) walking, (b) running, (c) squatting, (d) standing up, and (e) jumping. The different action requirements are shown in Table 2. Schematic diagrams of different actions and their corresponding micro-Doppler features are shown in Figure 9.

4.2. Experimental Details. To allow our network to segment specific actions in time series, we output the raw data as a result consisting of 40 sequences. Assuming that a set of data is 12 seconds, then a sequence represents the action prediction within the 0.3-second time window. In this time window, we think that the action is unique. Too small a time window makes it impossible to accurately align the time when the real action occurs and ends with the predicted result. In addition, the annotation of the dataset cannot be used as a traditional machine translation, and only the character encoding after translation needs to be marked. It is necessary to roughly align the 40 sequences reflecting the time information with the real data in time. The time window before the action occurs is marked with the [begin] symbol (in this paper, it is represented by >), and the first time window after the action is marked with the [end] symbol (in this paper, it is represented by >). In order to reduce the difficulty of labeling the dataset, we regard these 40 sequences as valid arrays and do not use padding (represented by 0 in this paper) to fill the remaining time series, because this will lead to confusion in the generation of encode_mask, which will affect training. Instead, a transition symbol (represented by e in this paper) is used instead of the padding symbol. A set of data is labeled as Figure 10.

We collected a total of 1660 sets of data, including 800 sets for one exercise, 500 sets for two consecutive exercises, and 360 sets for three consecutive exercises. The different actions are composed as shown in Table 3.

The deep learning network is based on the Pytorch framework, the optimization function is Adam, the loss function is the KL divergence loss function (KLDivLoss), the learning rate is 0.0001, and the batch size is 16. The network training was performed on a server equipped with NVIDIA RTX3090 graphics processor (24G video memory), 64G memory, and CPU core i9. The input deep learning network image has a height of 256, a width of 608, and a number of channels of 3.

5. Experimental Results and Analysis

5.1. Single Human Motion Recognition. Our proposed method can recognize both single action and multiple actions. The confusion matrix for the recognition of 5 single actions is shown in Figure 11. Experiments show that our method achieves a good accuracy for the recognition of 5 single actions, among which the recognition accuracy for walking is the highest, reaching 97.23%. The recognition readiness rate for squatting is the lowest, only 85.19%, and the recognition accuracy for standing up is also relatively low at 89.25%. This is because squatting and standing are essentially the same in nature, but in different directions. Squatting is a downward movement, and standing is an upward movement, so the micro-Doppler characteristics of the two are very similar, and sometimes, the naked eye cannot distinguish the two movements very well.

To further investigate the performance of our proposed method, we compare the proposed method with those proposed in literatures [10, 14, 16, 17]. In [10], principal component analysis (PCA) is used to propose invalid components in UWB radar signals, retain valid components, and then input the processed signals into support vector machines (SVM) for classification. In [14], DCNN network is first used to identify radar micro-Doppler time-frequency maps. This method processes radar signals into micro-Doppler time-frequency maps and then inputs them into the DCNN network for identification. Reference [16] proposed a model composed of 1D-CNN and long short-term memory (LSTM) network. This network makes full use of the feature that 1D-CNN has fewer parameters than 2D-CNN and achieves the same accuracy with shorter time. Reference [17] collects time-range-Doppler map (TRDM) and treats multiple range-Doppler map (RDM) collected during the action window as a video stream. Firstly, the principal component analysis (PCA) method is used to reduce the dimension of the TRDM map, and then, it is input to the convolutional autoencoder (CAE) for pretraining. After pretraining, only the encoder part is input to the LSTM network for recognition. These methods can only recognize single actions because their structure is essentially determined by classification-based recognition methods.

The comparison results are shown in Table 4. DCNN has the highest recognition accuracy for a single action. The recognition accuracy for action (e) jumping is as high as 98.9%, and the recognition accuracy for action (c) squatting also reaches 92.1%. PCA_SVM has the lowest recognition accuracy for a single action, with an average accuracy of only 85.4%, and the recognition accuracy of the more confusing actions (c) crouching and (e) standing up is only 76.2% and 77.1%. The recognition accuracy of the 1D_CNN_LSTM network for actions (c) crouching and (e) standing up is higher than that of the other four actions, and the recognition accuracy for the other three actions also exceeds the 95% recognition accuracy. The recognition accuracy of TRDM for actions (c) crouching and (e) is only 87.0% and 86.1%. This is mainly because the method treats a series of RDMs as video streams input to the network for recognition, and the range-Doppler (RD) features of two actions are very similar. Compared with the other four methods, our method lacks in recognition accuracy but still achieves an acceptable accuracy. The main reason for the relatively low recognition accuracy is because the

Table 2: Description of 5 different human movements.

Letters	Motion category	Specific motion description
(a)	Walking	Alternate arm swings (arms cannot be bent) and legs kicked out
(b)	Running	Alternate arm swings (arms bent) and legs raised
(c)	Squatting	Slowly squat down and finally squat on the ground
(d)	Standing	Stand up slowly, transition from squatting to standing
(e)	Jumping	Jump up once

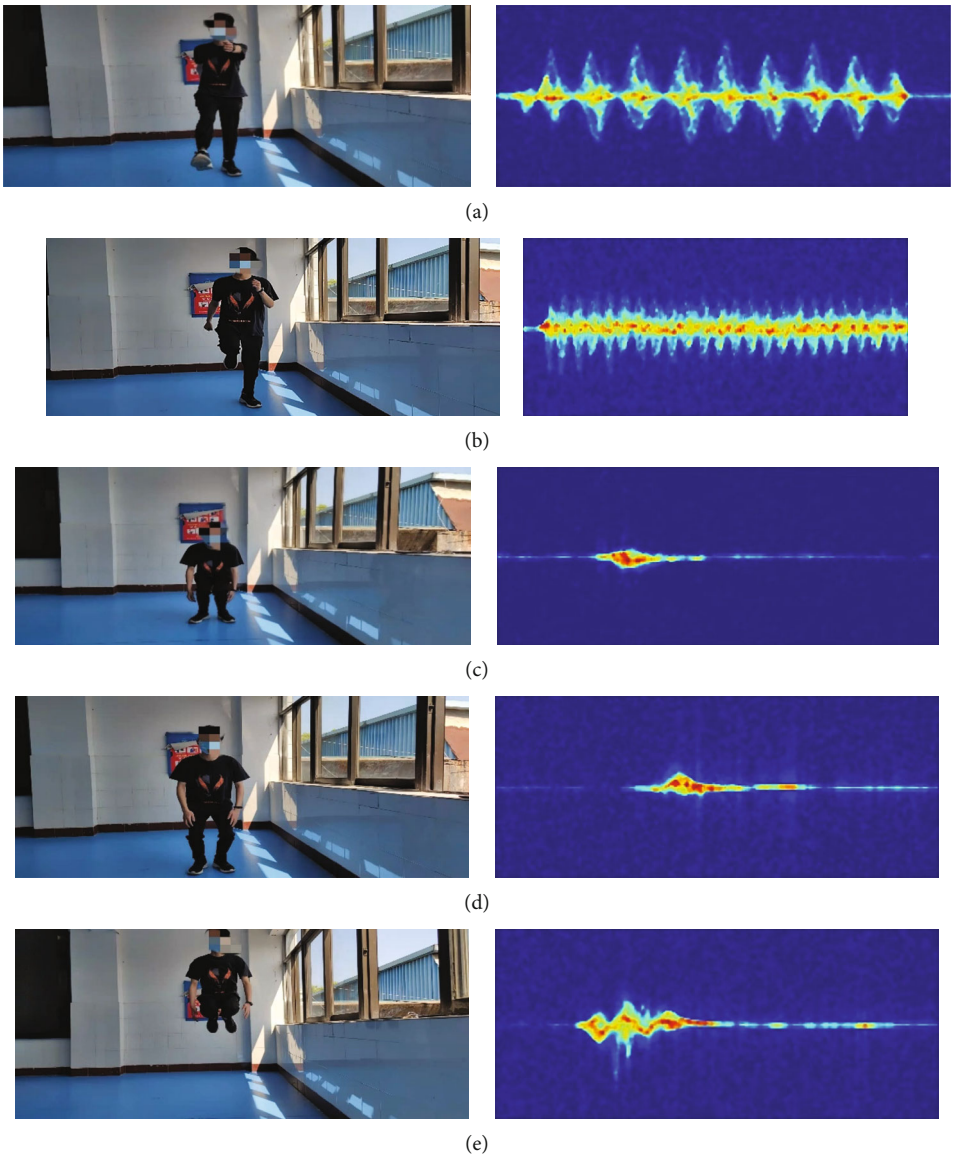


Figure 9: 5 different types of human motion measurement scenarios and micro-Doppler features: (a) walking; alternating arm swings (arms cannot be bent) and kicking; (b) running, alternating arms swinging (arms bent), and leg raising; (c) squat down, squat down slowly, and finally squat on the ground; (d) stand up, stand up slowly, and switch from squat to standing; and (e) jump and jump up once.

transformer has more parameters and our data is relatively small, resulting in the model's generalization ability is not as good as the other four methods.

5.2. Continuous Human Motion Recognition. In the experiment, the window time for collecting data for single and two actions is 300 frames and 9 seconds and the window time for collecting data for three actions is 400 frames and 12 seconds. Our method can identify continuous actions within 12 seconds. When training the network, the feature encoding layer of this paper does not use pretrained model but trained together with the transformer module. In order

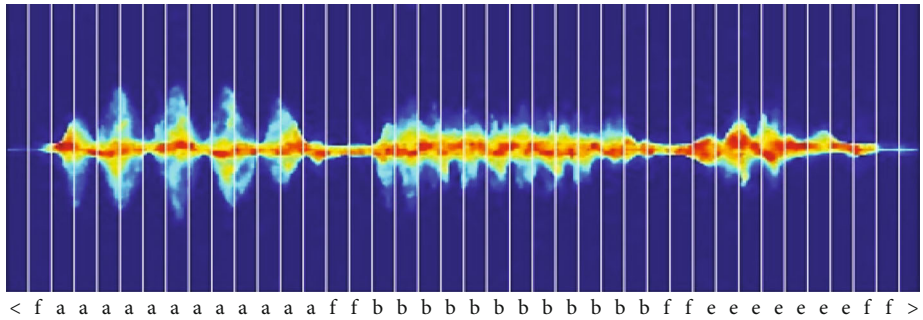


Figure 10: A typical set of data annotations. A set of data consisting of three actions is 12 seconds long. e divided the 12 seconds of data into 40 sequences, each representing 0.3 seconds, and contained only a single sequence within this sequence.

Table 3: Composition of different actions.

Dataset name	Composition
Walking	150
Running	150
Squatting	200
Standing	200
Jumping	100
Walking+running	40
Running+walking	40
Walking+squatting	40
Walking+jumping	40
Running+jumping	40
Jumping+walking	40
Jumping+running	40
Jumping+squatting	40
Squatting+standing	60
Standing+walking	40
Standing+running	40
Standing+jumping	40
Walking+running+jumping	30
Walking+jumping+running	30
Running+jumping+walking	30
Running+walking+jumping	30
Jumping+walking+running	30
Jumping+running+walking	30
Squatting+standing+walking	30
Squatting+standing+running	30
Squatting+standing+jumping	30
Walking+squatting+standing	30
Running+squatting+standing	30
Jumping+squatting+standing	30

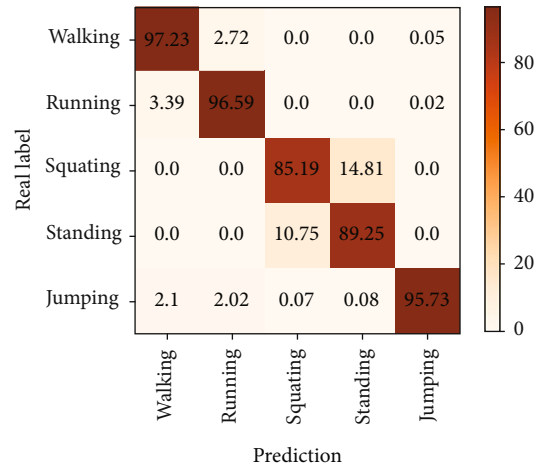


Figure 11: Confusion matrix of single action recognition.

Table 4: Comparison of our method with other method.

Method	Accuracy/%				
Motion	a	b	c	d	e
PCA_SVM	85.5	85.7	76.2	77.1	87.3
DCNN	98.1	97.6	92.1	95.4	98.9
1D_CNN_LSTM	96.8	97.3	92.3	91.9	97.1
TRDM	95.0	95.2	87.0	86.1	95.9
Ours	97.2	96.5	85.1	89.2	95.7

to better verify the advantages of transformer in self-attention, we extract the feature encoding layer in the network separately and then connects to the two-layer bidirectional long short-term memory network (Bi-LSTM) with 256 cell units. The network uses connectionism temporal classification (CTC) [34] as a loss function. At the same time, we also use the transformer network trained by the feature encoding layer pretrained by the above network to compare with the network proposed in this paper.

In tasks such as speech recognition and handwriting recognition, RNN-based networks are usually used for training and prediction. However, in order to train the RNN network, we need to accurately label each frame before there is no suitable loss function. It is difficult to collect and label a large amount of data in such a context, because the manual labeling cost will be very high. The emergence of the CTC algorithm solves this problem very well. The algorithm can automatically learn the relationship between the frame sequence and the label and can automatically align the frame sequence and the label. What we need to do is to let the network know the sequence of labels. If the sequence prediction is $y = (y_1, y_2, \cdots, y_N)$, the corresponding real label is $I = (I_1,$

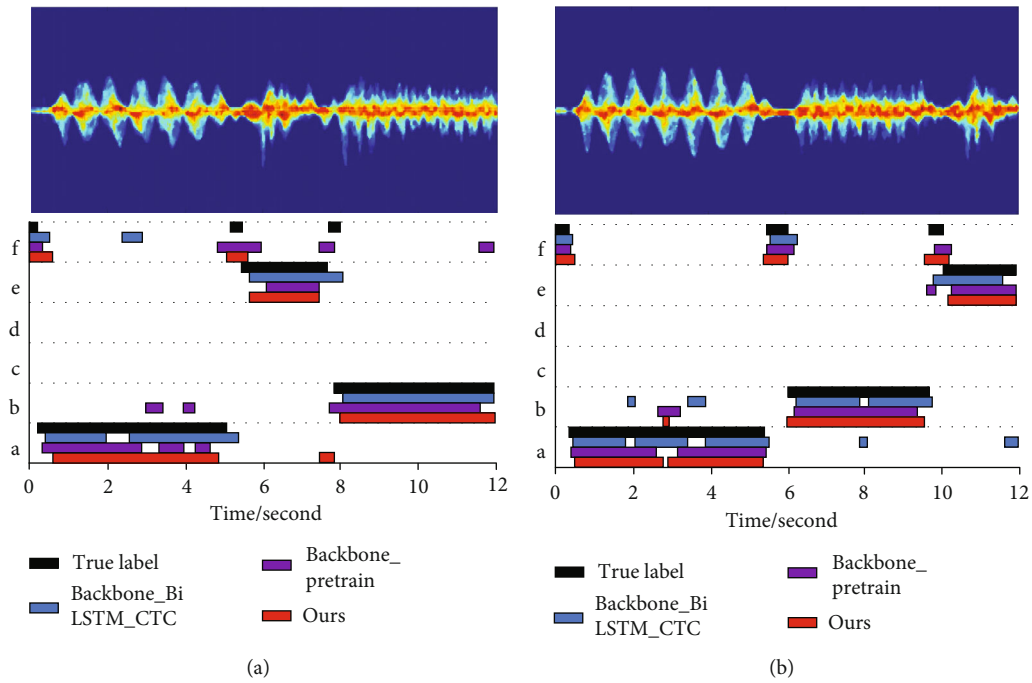


Figure 12: Comparison of two groups of 12-second data. Labels "a," "b," "c," "d," "e," and "f" stand for walking, running, squatting, standing up, jumping, and blank. (a) Data includes walking, jumping, and running. (b) Data includes walking, running, and jumping.

$I_2, \cdots, I_W$), where the sequence length is greater than or equal to the label length ($N \geq W$). Considering that there is a transition between different characters (there is a distance between two characters), the transition distance does not have any meaningful information, so it is necessary to define a blank as a blank and add it to the original label sequence to construct a new label sequence, that is, $L' = L \cup \{-\}$, where L' is the new label sequence and L is the original label sequence. Obviously, there are many prediction outputs corresponding to a real label. Assume that a prediction output sequence is $T = (t_1, t_2, t_3, t_4)$ and the true label sequence is label $= (l_1, l_2, l_3)$. Many paths such as $p(\pi^1) = (l_1, -, l_2, l_3)$, $p(\pi^2) = (l_1, l_2, -, l_3)$, and $p(\pi^4) = (-, l_1, l_2, l_3)$ can achieve the true label. Therefore, when the true label is y, the probability of the predicted label is I

$$p(I|y) = \sum_{\pi:B(\pi)=I} p(\pi|y), \tag{11}$$

where π is the output sequence of Bi-LSTM and $p(\pi|y)$ is the probability of path.

Since the predicted probabilities of each time series are independent of each other, the probability of the output sequence π at any time is calculated as follows:

$$p(\pi|y) = \prod_{t=1}^{N} y_t^{\pi_t}, \tag{12}$$

where $\pi_t \in L'$ is the predicted label under path π at time t and $y_t^{\pi_t}$ is the probability that the label is π_t at time t. The loss function can be defined as $L_{ctc} = -\ln p(I|y)$, and the gradient is updated by taking the derivation of each label at each step.

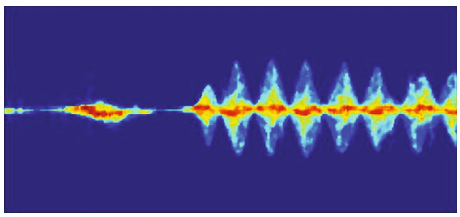
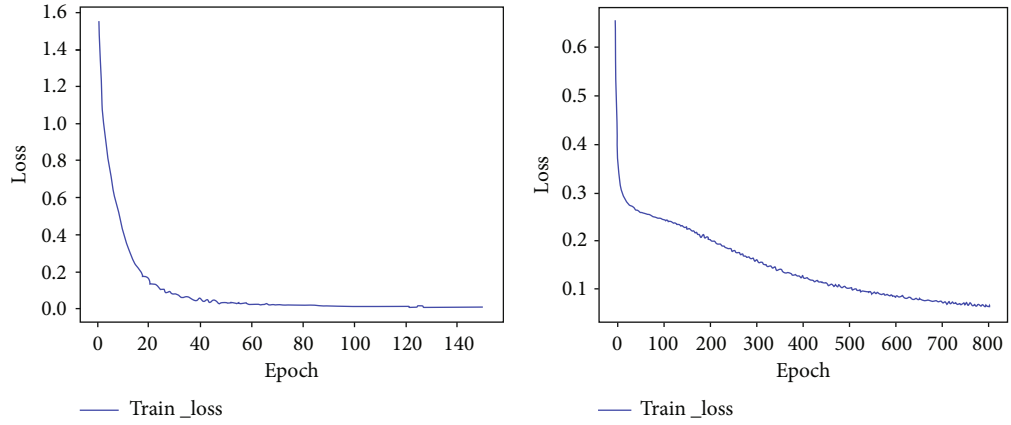


Figure 13: Concatenated illegal data.

We use two sets of 12-second data collected in the real environment containing three actions for test comparison, including (a) walking+jumping+running and (b) walking+running+jumping. After the feature encoding layer, a Bi-LSTM with 256 cells in two layers is connected. The training method using CTC as the loss function does not need to precisely align the labels with the output frame, because the CTC loss function can automatically learn the label sequence of the frame. This situation results in that the predicted label sequence cannot be accurately aligned with the real label sequence, and the prediction accuracy of continuous actions in time is not as high as the other two methods. Compared with the first method, the transformer-based network whose feature encoding layer which pretrained by Bi-LSTM network and CTC loss function improves the prediction accuracy of continuous actions in time. This is mainly because the input labels have been labeled relatively accurately in time. As shown in Figure 12, the network proposed in this paper can effectively segment continuous actions from time and recognize them. In terms of time, the method proposed



(a) Loss function trained by strategy (1)

(b) Loss function trained by strategy (2)

Figure 14: Training loss for policy (1) and policy (2). Strategy (1) only needs to predict the type of the action, so it only needs to be trained for 150 epochs. Strategy (2) not only needs to predict type of the action but also needs to predict the time of the action, so it needs to train 800 epochs.

in this paper will also have misjudgments in some time slices. This is mainly because the time series is manually annotated which makes labels and sequences not completely aligned and bring challenges to the network.

5.3. Experiment of Action Legitimacy. Illegal actions are difficult to collect in the display environment because there is no such action as squatting and then running or walking in the display life. Therefore, we use the method of splicing datasets to obtain illegal datasets. A set of illegal data is shown in Figure 13 (splicing). This paper proposes two interdependent strategies to determine the types of actions that exist in a set of data: (1) relabel the dataset to train the network proposed in this paper. The values in the data contain the information of the type of the action but not the time of the action. If a set of actions includes (a) walking, (b) running, and (e) jumping, these three actions only need to be marked as (a), (b), and (c). The output is two sets of information, one is the action information of the data in time, and the other is pure action type information. (2) Rules are used to constrain the actions predicted in time. In order to avoid judging that the action is illegal due to misjudgment of prediction, a simple judgment strategy is also used in this paper. Only when more than three consecutive time series ($t > 0.9$ s) are judged to be the same action, it is considered that the network has not misjudged. When predicting the three actions of (a) walking, (b) running, (e) jumping, only strategy (2) can be used because the duration of these three actions is relatively long. Misjudgment can also be corrected according to the strategy as long as the action duration of the misjudgment does not exceed three time series (0.9 s). When predicting (c) squatting and (d) standing up, these two actions with relatively short durations require strategy (1) and strategy (2) to cooperate with each other. The input image is only predicted and timed for the action. When the strategy (1) detects these two actions, it judges whether the prediction is accurate according to the prediction in time. According to the state transition diagram, when it is judged that the continuous actions before and after do not conform to the changes of human movements in reality, it is judged to be illegal.

The experimental results show that strategy (1) only needs to be trained for about 150 epochs to converge the loss to a very low level, and the recognition accuracy for a single action exceeds 94%, while strategy (2) needs to be trained for 800 epochs to converge the loss to a lower level. In addition to the prediction error of a single frame sequence, there is also a prediction error because of the long prediction sequence. The training loss graphs for strategy (1) and strategy (2) are shown in Figure 14. Obviously, strategy (1) converges much faster than strategy (2).

In addition, in most cases, strategy (1) can effectively identify the sequence of actions in a set of data. When the predictions about the sequence of actions before and after strategy (1) and strategy (2) are consistent, it can be judged that the action is legal according to the action state transition diagram. When the prediction results of strategy (1) and strategy (2) are different, the prediction result of strategy (2) will be preferred. Because strategy (2) is to predict a single frame, even if there are individual frame recognition errors in the middle, it may be filtered out decisively because of the duration. Usually, the goal of strategy (1) is the same as the OCR recognition task based on the attention mechanism. It only needs to predict the sequence of specific actions, while strategy (2) needs to recognize classification and time series at the same time.

6. Conclusion

For continuous actions, this paper proposes a method that can identify continuous actions and judge whether they are legal or not. The method consists of a transformer-based human continuous action recognition network and an action state transition graph. The transformer-based human continuous action recognition network is composed of a feature encoding layer, Inception maximum pooling layer,

and transformer. First, the input micro-Doppler time-frequency map is feature encoded and extracted into a high-dimensional space that the transformer can accept. Then, the transformer's unique multichannel attention mechanism is used to predict continuous actions in time. In addition, the network can also perform traditional OCR tasks, so it also trains a network that only predicts the sequence of actions without predicting the temporal information of actions. The information about predicting actions in time is combined with the information about only predicting the sequence before and after the action. Then, it is judged whether the continuous action is legal or not according to the action state transition diagram. In order to reflect the temporal information of the action, it is necessary to label a single frame relatively accurately when labeling the dataset which leads to a large workload of labeling work and the predicted label and the real label cannot be accurately aligned. The next stage is to use CTC as a loss function instead of training the network with the traditional KL divergence loss function (KLDivLoss) for training and prediction.

Data Availability

The datasets generated and/or analyzed during the current study are available from the corresponding author on reasonable request.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

This research was funded by the National Natural Science Foundation of China, grant number 61561010; Guangxi Innovation-Driven Development Special Fund, grant number AA21077008; Guangxi Key Laboratory of Wireless Wideband Communication and Signal Processing, grant numbers GXKL06220102 and GXKL06220108; and Innovation Project of GUET Graduate Education, grant numbers 2022YXW07 and 2022YCXS080. All work in this paper was completed in Guilin University of Electronic Technology. The experimental data was collected in Room 314, Teaching Building No. 3, Guilin University of Electronic Technology.

References

[1] S. Z. Gurbuz and M. G. Amin, "Radar-based human-motion recognition with deep learning: promising applications for indoor monitoring," *IEEE Signal Processing Magazine*, vol. 36, no. 4, pp. 16–28, 2019.

[2] S. C. Mukhopadhyay, "Wearable sensors for human activity monitoring: a review," *IEEE Sensors Journal*, vol. 15, no. 3, pp. 1321–1330, 2015.

[3] E. Cippitelli, F. Fioranelli, E. Gambi, and S. Spinsante, "Radar and RGB-depth sensors for fall detection: a review," *IEEE Sensors Journal*, vol. 17, no. 12, pp. 3585–3604, 2017.

[4] M. Pauli, B. Göttel, S. Scherr et al., "Miniaturized millimeter-wave radar sensor for high-accuracy applications," *IEEE Transactions on Microwave Theory and Techniques*, vol. 65, no. 5, pp. 1707–1715, 2017.

[5] M. E. Russell, C. A. Drubin, A. S. Marinilli, W. G. Woodington, and M. J. Del Checcolo, "Commercial radar technology," in *Record of the IEEE 2000 International Radar Conference [Cat. No. 00CH37037]*, pp. 819–824, Alexandria, VA, USA, 2000.

[6] V. C. Chen, F. Li, S. S. Ho, and H. Wechsler, "Micro-Doppler effect in radar: phenomenon, model, and simulation study," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 42, no. 1, pp. 2–21, 2006.

[7] R. J. Javier and Y. Kim, "Application of linear predictive coding for human activity classification based on micro-Doppler signatures," *IEEE Geoscience and Remote Sensing Letters*, vol. 11, no. 10, pp. 1831–1834, 2014.

[8] D. P. Fairchild and R. M. Narayanan, "Classification of human motions using empirical mode decomposition of human micro-Doppler signatures," *IET Radar, Sonar & Navigation*, vol. 8, no. 5, pp. 425–434, 2014.

[9] P. Molchanov, J. Astola, K. Egiazarian, and A. Totsky, "Ground moving target classification by using DCT coefficients extracted from micro-Doppler radar signatures and artificial neuron network," in *2011 Microwaves, Radar and Remote Sensing Symposium*, pp. 173–176, Kiev, Ukraine, 2011.

[10] J. D. Bryan, J. Kwon, N. Lee, and Y. Kim, "Application of ultra-wide band radar for classification of human activities," *IET Radar, Sonar & Navigation*, vol. 6, no. 3, pp. 172–179, 2012.

[11] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[12] L. Jiao and J. Zhao, "A survey on the new generation of deep learning in image processing," *IEEE Access*, vol. 7, pp. 172231–172263, 2019.

[13] L. Hang, "Deep learning for natural language processing: advantages and challenges," *National Science Review*, vol. 5, no. 1, pp. 24–26, 2018.

[14] Y. Kim and T. Moon, "Human detection and activity classification based on micro-Doppler signatures using deep convolutional neural networks," *IEEE Geoscience and Remote Sensing Letters*, vol. 13, no. 1, pp. 8–12, 2016.

[15] J. Craley, T. S. Murray, D. R. Mendat, and A. G. Andreou, "Action recognition using micro-Doppler signatures and a recurrent neural network," in *2017 51st Annual Conference on Information Sciences and Systems (CISS)*, pp. 1–5, Baltimore, MD, USA, 2017.

[16] J. Zhu, H. Chen, and W. Ye, "A hybrid CNN–LSTM network for the classification of human activities based on micro-Doppler radar," *IEEE Access*, vol. 8, pp. 24713–24720, 2020.

[17] Y. Wang, J. Zhou, J. Tong, and X. Wu, "UWB-radar-based synchronous motion recognition using time-varying range–Doppler images," *IET Radar, Sonar & Navigation*, vol. 13, no. 12, pp. 2131–2139, 2019.

[18] R. Zhao, X. Ma, X. Liu, and J. Liu, "An end-to-end network for continuous human motion recognition via radar radios," *IEEE Sensors Journal*, vol. 21, no. 5, pp. 6487–6496, 2021.

[19] C. Ding, H. Hong, Y. Zou et al., "Continuous human motion recognition with a dynamic range-Doppler trajectory method based on FMCW radar," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 9, pp. 6821–6831, 2019.

[20] M. Wang, G. Cui, X. Yang, and L. Kong, "Human body and limb motion recognition via stacked gated recurrent units network," *IET Radar, Sonar & Navigation*, vol. 12, no. 9, pp. 1046–1051, 2018.

[21] B. Shi, X. Bai, and C. Yao, "An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 11, pp. 2298–2304, 2017.

[22] R. Zhao, X. Ma, X. Liu, and J. Liu, "Continuous human motion recognition using micro-Doppler signatures in the scenario with micro motion interference," *IEEE Sensors Journal*, vol. 21, no. 4, pp. 5022–5034, 2021.

[23] H. Li, A. Shrestha, H. Heidari, J. Le Kernec, and F. Fioranelli, "Bi-LSTM network for multimodal continuous human activity recognition and fall detection," *IEEE Sensors Journal*, vol. 20, no. 3, pp. 1191–1201, 2020.

[24] S. Zhu, R. G. Guendel, A. Yarovoy, and F. Fioranelli, "Continuous human activity recognition with distributed radar sensor networks and CNN–RNN architectures," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–15, 2022.

[25] R. G. Guendel, M. Unterhorst, E. Gambi, F. Fioranelli, and A. Yarovoy, "Continuous human activity recognition for arbitrary directions with distributed radars," in *2021 IEEE Radar Conference (RadarConf21)*, pp. 1–6, Atlanta, GA, USA, 2021.

[26] A. Vaswani, N. Shazeer, N. Parmar et al., "Attention is all you need," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[27] M. H. Guo, J. X. Cai, Z. N. Liu, T. J. Mu, R. R. Martin, and S. M. Hu, "Pct: point cloud transformer," *Computational Visual Media*, vol. 7, no. 2, pp. 187–199, 2021.

[28] A. Ahmad, J. C. Roh, D. Wang, and A. Dubey, "Vital signs monitoring of multiple people using a FMCW millimeter-wave sensor," in *2018 IEEE Radar Conference (RadarConf18)*, Oklahoma City, OK, USA, 2018.

[29] M. Song, J. Lim, and D. J. Shin, "The velocity and range detection using the 2D-FFT scheme for automotive radars," in *2014 4th IEEE International Conference on Network Infrastructure and Digital Content*, pp. 507–510, Beijing, China, 2014.

[30] L. Durak and O. Arikan, "Short-time Fourier transform: two fundamental properties and an optimal implementation," *IEEE Transactions on Signal Processing*, vol. 51, no. 5, pp. 1231–1242, 2003.

[31] R. Mithe, S. Indalkar, and N. Divekar, "Optical character recognition," *International Journal of Recent Technology and Engineering (IJRTE)*, vol. 2, no. 1, pp. 72–75, 2013.

[32] M. Li, T. Lv, L. Cui et al., "Trocr: transformer-based optical character recognition with pre-trained models," 2021, https://arxiv.org/abs/2109.10282.

[33] M. G. Amin and R. G. Guendel, "Radar human motion recognition using motion states and two-way classifications," in *2020 IEEE International Radar Conference (RADAR)*, pp. 1046–1051, Washington, DC, USA, 2020.

[34] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd International Conference on Machine Learning*, pp. 369–376, 2006.