IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 8 April 2024, accepted 10 June 2024, date of publication 17 June 2024, date of current version 24 June 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3415359

RESEARCH ARTICLE

Human Activity Recognition Based on Self-Attention Mechanism in WiFi Environment

FEI GE[1], ZHIMIN YANG[1], ZHENYANG DAI[1], LIANSHENG TAN[1,2], JIANYUAN HU[1], JIAYUAN LI[1], AND HAN QIU[1]

[1]School of Computer Science, Central China Normal University, Wuhan 430070, China
[2]School of Technology, Environments and Design, University of Tasmania, Hobart, TAS 7001, Australia

Corresponding authors: Fei Ge (feige@ccnu.edu.cn) and Zhimin Yang (zhiminyang@mails.ccnu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 62173157.

ABSTRACT In recent years, the use of WiFi Channel State Information (CSI) for Human Activity Recognition (HAR) has attracted widespread attention, thanks to its low cost and non-intrusive advantages. Previous research mostly used models based on Convolutional Neural Networks (CNN) or Recurrent Neural Networks (RNN) for activity recognition. However, these methods fail to achieve good parallelism while learning global features and fine-grained features, so they often cannot achieve the ideal recognition effect or training speed. In light of this, we propose an ensemble deep learning model based on CNN and Transformer, ConTransEn. Specifically, we first use CNN to extract spatial features of the sequence, and then use Vision Transformer (ViT) to further extract temporal features. The Transformer introduces self-attention mechanism, enabling the model to fully consider information from other positions in the sequence, rather than being limited to the current input. Furthermore, due to the increased parallelism, Transformer has an advantage in training speed over RNN. In order to further improve the accuracy and robustness of the model, we adopt a bagging ensemble learning strategy, integrating the prediction results of multiple homogeneous base models using a soft voting method to obtain the final classification result. This ensemble learning method reduces the risk of model overfitting, and improves the overall accuracy and reliability of the model. We extensively evaluated the model on two publicly available datasets, and achieved excellent recognition results, indicating its good performance and robustness. The average recognition accuracy on the UT-HAR dataset reached 99.41%, surpassing existing solutions.

INDEX TERMS Attention, channel state information (CSI), convolutional neural networks, human activity recognition.

I. INTRODUCTION

Human Activity Recognition (HAR) is not only applicable in the realm of basic security surveillance, but also has broad application prospects in fields such as smart homes and healthcare. By discerning human gestures, gait, and even respiratory patterns, HAR offers a revolutionary interactive approach to elevate personal well-being and enhance the home automation experience. HAR refers to the process of recognizing specific human activities through the implementation of machine learning and pattern recognition algorithms on data obtained from sensors or similar technologies. Traditional HAR systems primarily utilize acoustical [1] or vision-based devices [2] and wearable sensors [3], [4], but these technologies have inherent limitations. Acoustic-based approaches have very limited coverage and therefore cannot be scaled up to large numbers of users. Vision-based methods can achieve relatively high recognition accuracy, but they require expensive equipment, suffer from field of view limitations, and have serious privacy issues, and more private scenes like smart homes, the requirements for privacy protection will be relatively high. Although the use of wearable devices for activity recognition can achieve real-time monitoring and provide more personalized services, it is not

The associate editor coordinating the review of this manuscript and approving it for publication was Jie Tang.

© 2024 The Authors. This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License.
For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

possible to achieve a good user experience because it relies on specific devices, so its application scope is limited.

In recent years, an increasing number of researchers have started using commercial WiFi devices to extract channel state information (CSI) [5] for human activity recognition. The principle behind this is that when human activities occur in a WiFi-covered environment, it causes changes in the propagation path of WiFi signals. These changes can be captured through WiFi CSI, which includes the amplitude and phase information of the WiFi signal. By analyzing the changes in CSI, different types of human activities, such as walking, falling, or making specific gestures, can be identified. WiFi-based human activity recognition overcomes many limitations of traditional methods and has the characteristics of low cost, privacy protection, and robustness in adverse lighting conditions. Existing methods for human activity recognition based on WiFi can be mainly categorized as model-based and learning-based methods. Model-based methods rely on physical models (such as Fresnel zone) to describe the propagation of WiFi signals, but this method is only suitable for handling periodic or one-time movements, such as breathing or falling. For relatively complex human activities, model-based methods may be inadequate. In contrast, learning-based methods can achieve relatively significant performance in complex sensing tasks by inputting a large amount of data to the network [6]. Many researchers have conducted various human perception tasks based on CSI signals, such as vital sign monitoring [7], fall detection [8], activity recognition [9], [10], [11], indoor positioning [12], gesture recognition [13], and identity recognition [14]. Many of these tasks have used deep learning models, including MLP [15], CNN [16], and CNN-LSTM [17], [18].

However, in actual WiFi environments, the CSI signal is influenced by various factors such as object obstruction, multipath effects, environmental noise, etc., which makes the processing of CSI signals complex. Simple convolutional neural network models may struggle to extract key features from the signals. Additionally, the continuous actions' CSI signals involve a large amount of time series data, which may contain long-distance dependencies. However, the existing CSI-based activity recognition work mainly uses models based on stacked convolutional kernels and pooling to extract features [19], [20], [21]. These models have limited ability to handle long-distance dependency relationships. While recurrent neural network (RNN) is suitable for handling time series data, the original RNN model may encounter the problem of vanishing gradients during backpropagation. Improved RNN models such as LSTM (Long Short-Term Memory) and GRU (Gated Recurrent Unit) have alleviated the vanishing gradient problem, but these RNN-based models often lack efficient parallel computing capabilities, resulting in high computational overhead. The Transformer model was first proposed in a paper by Google [22] to improve the efficiency of machine translation. Compared to RNN models, the Transformer model completely eliminates recursion and convolution, utilizing the self-attention mechanism to increase model parallelization, thereby improving training speed. Additionally, the Transformer can increase model depth by stacking encoding and decoding modules, fully exploiting the characteristics of deep neural network models to enhance model accuracy. Due to its superiority in handling image data compared to traditional models, the Vision Transformer (ViT) based on the Transformer has been widely applied in the field of computer vision. In order to effectively extract the spatial features of CSI data in WiFi environments, and capture the long-term dependencies of the data, thus achieving the ideal accuracy and generalization capability for activity recognition, we propose a deep learning model based on CNN and ViT, and employ the bagging algorithm to integrate the classification results of multiple homogeneous basic models.

We summarize the main contributions of our work as follows:

- A classification model based on self-attention mechanism, ConTransEn, is proposed, which first utilizes CNN to extract spatial features of the sequences, and then uses a ViT module that only includes the encoder to capture the dependency relationships between different positions in the sequence, effectively capturing the time series features in the CSI data, which is crucial for accurate activity recognition.
- To improve the robustness of the model, we further adopted the bagging algorithm. By integrating the outputs of multiple homogeneous base models to obtain the final classification results, we reduced the risk of model overfitting while enhancing the overall accuracy and reliability of the model.
- Extensive experiments were conducted on a publicly available dataset containing seven different activities, and the results show that our model achieves an average recognition accuracy of over 99%. Additionally, the model was evaluated using a gesture dataset containing multiple scenarios and users, and also achieved quite ideal recognition performance.

The remainder of this article is organized as follows. Section II elaborates on related knowledge in detail, including an introduction to the principles and basic processes of human activity recognition based on CSI, as well as a detailed explanation of related findings in this field. Section III provides an overview of the proposed model's overall structure and detailed explanations of the individual parts of the model. Section IV describes the experimental setup, as well as the presentation and analysis of the experimental results. Finally, in Section V, the paper concludes and proposes directions for future research.

II. RELATED WORKS

This section will first explain CSI and the principle and basic process of CSI based human activity recognition, followed by an introduction to related work in this field. With the advantage of using WiFi signals for human activity recognition,

many effective methods have been proposed. The methods for human activity recognition under WiFi environment can be divided into model-based methods and learning-based methods. The introduction to the related work will be presented from these two aspects.

A. HUMAN ACTIVITY RECOGNITION WITH CSI

Due to obstacles and other interfering factors, wireless signal propagation experiences reflection, refraction, and diffraction, resulting in signals reaching the receiver through different paths. This phenomenon is known as the multipath effect in wireless sensing. In order to describe the information of each path in wireless signal propagation, the propagation channel is usually represented as a time-linear filter, called the channel impulse response (CIR). It can be represented by the following equation:

$$h(\tau) = \sum_{i=1}^{n} a_i e^{-j\theta_i} \delta(\tau - \tau_i), \quad (1)$$

where n is the number of propagation paths, a_i, θ_i, and τ_i are the amplitude attenuation, phase offset, and time delay of the i-th path, respectively, and $\delta(\tau)$ is the Dirac pulse function. The channel frequency response (CFR) describes the signal's decay at different frequencies, and it is the Fourier transform of the CIR. In the IEEE 802.11 wireless LAN standard, two important physical layer technologies, Orthogonal Frequency Division Multiplexing (OFDM) and Multiple Input Multiple Output (MIMO), effectively address signal attenuation and interference caused by multipath effects by utilizing multiple antennas and orthogonal subcarrier frequency division, thus achieving higher data transmission rates within limited spectrum resources. In WiFi systems, CFR can be used for optimizing subcarrier selection, power allocation, and frequency domain equalization during OFDM modulation and demodulation processes. Channel State Information characterizes the wireless channel by presenting the amplitude and phase information of each OFDM subcarrier and represents the discrete samples of CFR.

CSI reflects the attenuation and interference experienced by wireless signals during multipath propagation, describing the characteristics of the channel during signal propagation, including multipath effects and phase shifts. CSI can be collected using tools such as the Intel 5300 network card [5] and Atheros CSI Tool [23]. In WiFi-based device-free sensing scenarios, CSI can be represented as the superposition of static signal components (combination of static object multipath reflections) and dynamic signal components reflected by moving human bodies. The received signal at the receiver can be represented by the following formula:

$$\mathrm{CSI} = A_{\mathrm{noise}}(f,t)\, e^{-j\theta_{\mathrm{offset}}(f,t)} \left(H_s(f) + H_d(f,t)\right), \quad (2)$$

where f is the carrier frequency, t is time, A_{noise} is the uncertainty of amplitude power amplification, θ_{offset} is the random phase offset, $H_s(f)$ and $H_d(f,t)$ are static vector components and dynamic vector components, respectively. In general, CSI

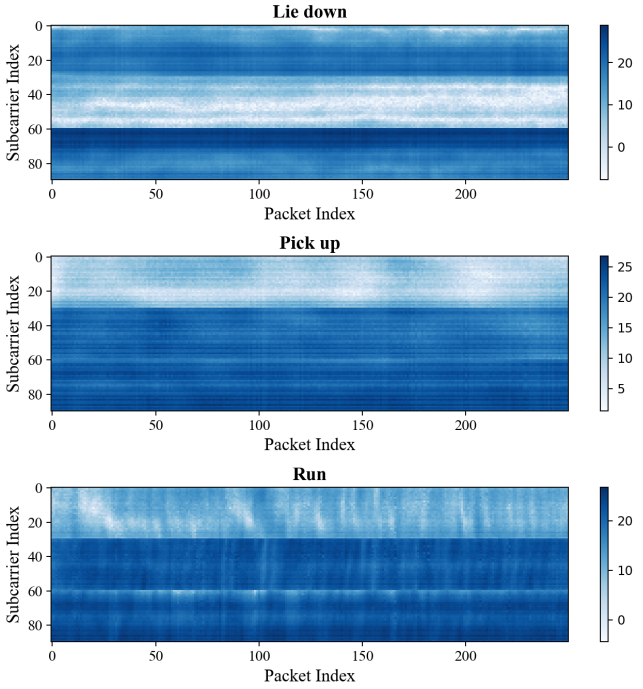

FIGURE 1. The CSI samples of three human activities in UT-HAR.

is a matrix composed of multiple complex values, where each value describes the channel response of a specific subcarrier on a specific antenna pair, including attenuation and phase shift information of the signal. Fig. 1 shows the CSI sample visualizations of the "Lie down," "Pick up," and "Run" activities in the UT-HAR dataset [24]. In this dataset, the CSI data consists of three dimensions: antenna count, subcarrier, and packet number (i.e., action duration). From the figure, it can be observed that each action is divided into three segments from top to bottom. This is because there are three receiving antennas, and each segment represents the CSI data collected on a receiving antenna. Assuming the number of transmit antennas and receive antennas are X_T and X_R, and the number of subcarriers is X_S, then at a specific time t, the received CSI is a complex matrix of size $X_T \times X_R \times X_S$, which can be seen as a "CSI image" reflecting the surrounding environment at time t. Therefore, collecting CSI at very short time intervals can create a "CSI video" that captures subtle human activities [6].

As mentioned above, CSI describes the channel characteristics of wireless signals during transmission, including parameters such as phase, amplitude, and frequency, providing rich information about the signal transmission process. Human activities can affect the multipath effects of wireless signals, causing changes in CSI. Therefore, it is possible to establish a mapping relationship between the changes in CSI signals and various human activities, in order to achieve recognition and analysis of different activities. The raw CSI data contains interference and phase noise, so in order to improve the accuracy and reliability of human activity recognition in a WiFi environment, it is typically necessary to perform denoising on the raw data. Common denoising

methods include linear filtering, mean filtering, and wavelet denoising [25]. In addition, if the surrounding environment is different, or even the relative position of the human body to the antenna is different, the interference received during WiFi signal propagation will also be different, resulting in different CSI signals. Therefore, even for the same activity, the CSI signals obtained in a WiFi environment will be different. To address this challenge, Widar3.0 [26] proposes a new environment-independent feature called the body-coordinate velocity profile (BVP). BVP is a three-dimensional feature that uses the human body as a reference coordinate system, calculated based on DFS power distribution. It quantifies the speed characteristics of human gestures by capturing the velocity patterns generated by different body parts at different action stages, thereby eliminating environmental dependencies. In this paper, we selected a commonly used CSI dataset with only amplitude and a BVP gesture dataset which is domain independent to evaluate the performance of model. Then, signal processing and machine learning algorithms such as Support Vector Machine (SVM), deep learning, etc. can be used to extract features related to human activities. Finally, the extracted features are trained and recognized. By building and training models, accurate recognition of different human activities can be achieved.

B. MODEL-BASED METHODS

Model-based WiFi sensing primarily involves extracting various information from the CSI signal, including Doppler Frequency Shift (DFS) [27], time of flight (ToF) [28], angle of arrival (AoA) [29], and then using physical models to map this information to discrete activities. For example, WiDance [30] extracted the Doppler frequency shift caused by motion and modeled the relationship between Doppler frequency shift and motion direction for activity recognition. WiTraj [31] reduced signal noise by using the ratio of the CSI from two antennas on the same receiver, significantly improving the quality of DFS estimation, and robustly reconstructed human walking trajectories using DFS extracted from multiple receivers. Although the work mentioned above, which used individual information extracted from CSI for modeling, could achieve good tracking or recognition accuracy, there were still many limitations, such as larger errors in multiperson scenarios and accumulating errors in static states. Therefore, some researchers choose to simultaneously extract multiple types of information to establish more robust mapping relationships. For example, IndoTrack [32] proposed a probabilistic, space-time joint trajectory estimation method, combining DFS and AoA spectrum estimation of target reflection paths to precisely locate humans indoors. Widar2.0 [33] used AoA, ToF, and DFS information simultaneously to establish a model and designs effective algorithms for their joint estimation to predict human positions.

C. LEARNING-BASED METHODS

Deep learning is a new research direction in the field of machine learning, the learning-based methods are mainly to input a large amount of CSI signal data into the deep learning model for feature extraction, and optimize the model parameters through backpropagation to reduce loss. Finally, the trained model establishes a mapping relationship between CSI features and human activities. Table 1 summarizes and compares recent research on activity recognition and other tasks using deep learning methods. Convolutional Neural Network (CNN) is one of the most commonly used deep learning models, which primarily extracts data features through convolution and pooling operations. For example, WiADG [34] adopted CNN as a classifier and designed an Unsupervised Adversarial Domain Adaptation scheme to reduce the domain difference between the source domain and the target domain (new environment), thereby reducing the interference of environmental elements. However, due to the limited convolutional kernel size of the CNN model, it cannot capture the long-term dependency of CSI data, so it often fails to achieve ideal results when processing time series data. Recurrent Neural Network (RNN) not only considers the input of the previous moment, but also gives the network a memory function for the previous content, making it very effective for time series data. The LSTM model, based on RNN, incorporates a gate mechanism through input gates, forget gates, and output gates to control information flow and memory retention. This enables better handling of long-term dependencies and reduces the impact of gradient vanishing. Therefore, some studies have begun to use RNN-based models for activity recognition. For example, Yousefi et al. [24] utilized machine learning methods including random forest, Hidden Markov Model, and LSTM to classify activities, and the results indicated that LSTM achieved significantly higher classification accuracy compared to other methods. Shang et al. [35] designed an LSTM-CNN model, which can simultaneously extract the temporal and spatial features of CSI data and effectively classify different activities. Chu et al. [36] proposed the C-MuRP system, designed a conditional recursive neural network, using two layers of CNN structure to extract spatial features, then using GRU to extract temporal features, and applying a fully connected layer for feature mapping to achieve human presence detection in multiple rooms.

However, models based on RNN may not only face the problem of gradient vanishing but also cannot implement parallel computing, thus resulting in high computational overhead. In 2017, after the proposal of the Transformer model, considering its advantages in performance and parallelism, many researchers began to try to incorporate Transformer or attention mechanisms into models for human activity recognition. For example, Chen et al. [37] proposed an attention-based BiLSTM model, using attention mechanisms to assign higher weights to more important features and time steps, thus achieving better recognition performance. Zhou et al. [38] proposed the MetaFi++ system, which performs WiFi-based human pose estimation and is used for virtual avatar simulation. The system uses a network structure based on shared convolutional modules and Transformer

TABLE 1. WiFi sensing research using deep learning approaches.

Method	Task	Model	Strategy	Dataset	Recognition Accuracy
Yousefi et al., 2017 [24]	Human Activity Recognition	RF, HMM, LSTM	Supervised learning	6 volunteers, 6 kinds of activities	RF: 65%; HMM: 73%; LSTM: 90.5%
WiADG, 2018 [34]	Gesture Recognition	CNN	Transfer learning	6 kinds of gestures, 2500 samples	Original environment:98%; new environment: increased by 25%
ABLSTM, 2019 [36]	Human Activity Recognition	Attention based BiLSTM	Supervised learning	6 kinds of activities	Over 95%
Sheng et al., 2020 [18]	Human Activity Recognition	CNN-BiLSTM	Transfer learning	5 volunteers, 4 kinds of activities	Average 98.38%
Shang et al., 2021 [35]	Human Activity Recognition	LSTM-CNN	Supervised learning	5 kinds of activities, 2800 samples	Average 94.14%
Widar3.0, 2022 [26]	Human Identification, Gesture Recognition	CNN-GRU	Supervised learning	22 kinds of gestures, 43652 samples	In-domain:92.7; cross-domain: 82.6%-92.4%
MetaFi++, 2023 [38]	Human Pose Estimation	CNN-Transformer	Supervised learning	22 volunteers in two different environments, 78408 samples	PCK@50: 97.30%

modules, learning the different importance of CSI from three pairs of antennas, thus obtaining location-aware features and achieving robust pose estimation. Zhou et al. [38] used CSI ratio instead of CSI as the basic signal, thus eliminating most of the signal noise. Then, they utilized self-attention modules to learn the fine feature representation related to gestures in the CSI ratio data, achieving excellent gesture recognition performance.

III. MODEL DESIGN

We use a self-attention-based model to extract features from the data, which consists of a CNN module and a ViT module that only includes the encoder. The schematic diagram of the entire model is shown in Fig. 2. We first perform multiple rounds of bootstrap sampling on the original training set to obtain multiple new training sets, and then input these training sets into the model for training separately. The data is first passed to the CNN module, which is used to extract spatial features of the data, while also altering the data dimensions to facilitate further feature extraction by subsequent modules, and reduce the risk of overfitting. The output of the CNN is then used as the input for the ViT module, where the input sequence is first subjected to positional embedding, and then passed to the encoder layer to assign different attention weights to different positions in the sequence, capturing the time dependencies of the sequence. Finally, the bagging soft voting algorithm is used as the final classifier to output the classification results.

A. SPATIAL FEATURE EXTRACTION

The dimension of each initial sequence in the UT-HAR [24] dataset is processed as $1 \times 250 \times 90$. In the CNN module, the initial sequence's dimensions are first down sampled from $1 \times 250 \times 90$ to $1 \times 63 \times 23$. Then, through a convolution layer, the data channel number is transformed to 64, preparing for input to the subsequent convolution layers. In order to ensure that the spatial features of the data are fully extracted while maintaining the performance and generalization ability of the model, we introduce skip connections, namely residual blocks, into the module. The module consists of 16 convolutional blocks with 3×3 size kernels, to stabilize the model during training and accelerate convergence, batch normalization is performed after each convolution. Specifically, for each channel of the input, the mean and variance of all samples in each channel are calculated, and then these means and variances are used to normalize each sample: subtract the mean from each sample, and then divide the variance. Additionally, learnable scaling parameters (gamma) and shift parameters (beta) are applied to scale and shift the normalized values. After batch normalization, a ReLU activation is applied to facilitate the model in learning complex features more effectively. Defining every four convolutional blocks as one layer, there are a total of four layers. After every two convolutions in each layer, a skip connection is made, adding the input features and the current intermediate features together as the input for the next convolutional block. The first convolutional block in the last three layers doubles the data channel number and their convolution step is set to 2. After a total of 16 convolutions, the output is then flattened, and then fed into a fully connected layer to turn the data into one dimension. To reduce the risk of overfitting, we perform a dropout operation on the data, and finally reshape the data back to three dimensions. The main structure of the convolutional module is shown in Fig. 3., where curved arrows

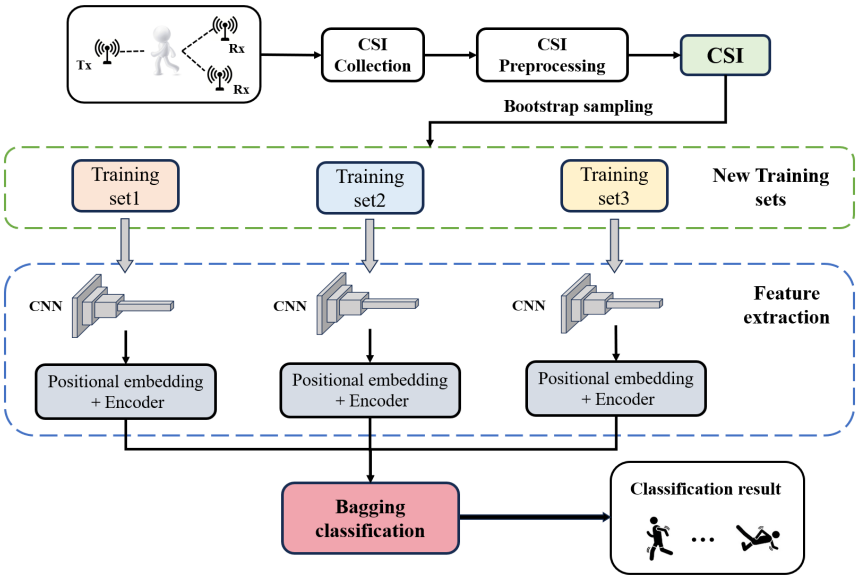

FIGURE 2. The overall framework of the classification model.

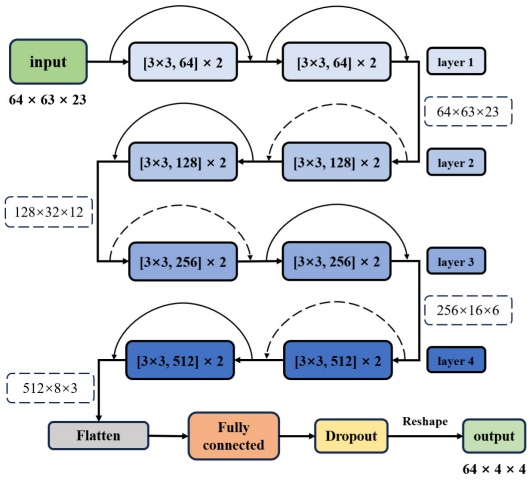

FIGURE 3. Main structure of CNN module.

indicate residual connections, the dashed arrows indicate that the channel number and size of features have changed, and the residual connection is also carried out, and the dashed box indicates the dimensions of the intermediate features. The output dimensions obtained from the CNN module are $64 \times 4 \times 4$, which are then used as the input for the ViT module to extract temporal features.

B. TIME FEATURE EXTRACTION

After completing the extraction of spatial features, we feed the output into the ViT module that only includes the Encoder for temporal feature extraction. The finite size of the convolution kernel in the CNN leads to an insufficient receptive field, thus failing to capture long-distance dependencies. Unlike CNN and RNN, Transformer is entirely based on the self-attention mechanism. This mechanism allows the model to assign higher weights to essential information when processing CSI signals, helping us effectively extract fine-grained information from time series data, thereby enhancing the model's activity recognition capabilities in complex environments. The process of extracting temporal features in the ViT module is shown in Fig. 4. The features outputted by the CNN module are first sent to the Position Embedding layer, which adds a vector representing the position information for each position in the input sequence, helping the model understand the relationships between different positions in the input time series. The Position Embedding layer generates a learnable position encoding matrix with the same dimension as the input features, and then adds the position encoding to the input features to form features containing position information. In order to reduce the risk of model overfitting and improve the stability of the model, a dropout operation is applied to the position embedding features before performing the multihead self-attention weight calculation in the Encoder block.

Each Encoder block contains multihead self-attention layers, a feed-forward neural network (MLP) layer, and residual connections. After obtaining the output of the Positional Embedding layer, it is normalized to make the data distribution more uniform, and then inputted into the multihead self-attention layer for weight calculation. Self-attention mechanism is the core of the Transformer, used to capture dependencies between features at different positions. The self-attention mechanism mainly involves calculating attention weights, first mapping the input sequence containing position information into three different matrices through linear transformation: query (Q), key (K), and value (V). Then, attention weights are calculated using these three matrices. The calculation process of attention weights can be

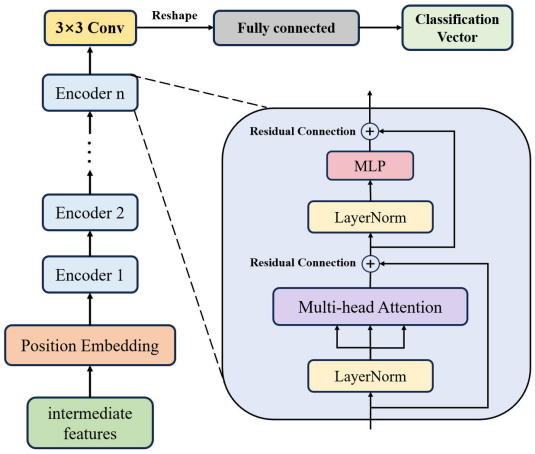

FIGURE 4. Temporal feature extraction process.

represented by the following formula:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{Q \cdot K^T}{\sqrt{d_k}}\right) \cdot V, \quad (3)$$

where $\sqrt{d_k}$ is a scaling factor used to control the size of attention weights, which helps to avoid gradient vanishing or exploding during training, and improves the training stability of the model. Multihead attention involves generating multiple sets of Q, K, and V matrices, performing multiple attention weight calculations, and finally the calculated attention weight matrices are spliced together. Then, a linear transformation is applied to the concatenated matrices as the output. Using multihead attention allows the model to better understand the relationships between different positions in the input sequence and enhances the model's representational capacity. After calculating the attention weight matrix, it is further subjected to a dropout operation to reduce overfitting of the model. Once the output of the multihead self-attention layer is obtained, it is connected with the original input, normalized, and then input to the MLP layer. Finally, a residual connection is applied to obtain the output of the Encoder block. In the ViT, multiple Encoder blocks are stacked, with the output of the previous Encoder block serving as the input for the next Encoder block for hierarchical feature extraction, gradually enriching the representation of the input sequence and improving the model's understanding and generalization abilities. To obtain the classification results, we use a convolutional layer to reduce the number of output feature channels from 64 to 2, with a kernel size of 3×3. Then, a dimension transformation is performed to convert the data into two dimensions, and finally, a linear transformation is applied to reduce the number of channels from $2 \times 4 \times 4$ to the number of classes in the dataset, in order to obtain the classification vector.

C. BAGGING ENSEMBLE

Bagging is a widely used ensemble learning algorithm that has shown good performance, particularly in solving classification and regression problems. Its basic process involves first using the Bootstrap sampling method to randomly sample the original training set multiple times to obtain multiple new training sets. Each of these new sets is then used to train multiple base models. The predictions of all base models are then combined using a voting strategy to calculate the final classification result of the ensemble model.

To further improve the model's classification accuracy and generalization ability, we adopted the idea of Bagging algorithm to ensemble the models. We use the Bootstrap sampling method to randomly sample the original training set containing n samples with replacement, meaning each time a sample is randomly selected from the original training set and copied into a new training set, then the sample is returned to the original training set, ensuring that the sample is still possible to be selected in the next sampling. This process is repeated n times, conducted in three rounds of sampling, resulting in three new training sets of the same size as the original training set. These three training sets are then used to train three homogeneous CNN + ViT models proposed in this paper, respectively. During the testing phase, the predictions of the three models are combined using the soft voting method, averaging the predicted probabilities for each class from the three models, and finally selecting the class with the highest average probability as the predicted result. The pseudocode for integrating multiple base models using the Bagging ensemble learning algorithm and obtaining the prediction results of the ensemble model on the test set using the soft voting method is shown in Algorithm 1.

Algorithm 1 Bagging Ensemble Learning Algorithm Using Soft Voting Method

Input: original training set D with n samples, Test set S, basic model ConTrans, Number of basic models N
Output: predicted results R
1: ensemble_model = []
2: for t = 1 to N:
    Dt = randomly_select_samples(D)
    Train a basic model ConTrans using Dt: ft = ConTrans(Dt)
    ensemble_model + = [ft]
3: def soft_voting_predict(ensemble_model, x):
    predictions = [model(x) for model in ensemble_model]
    Calculate the average probability for each category:
    avg_probs = average (predictions, axis = 0)
    return argmax(avg_probs)
4: Apply soft voting for classifying the test samples: R = soft_voting_predict (ensemble_model, S)

IV. EXPERIMENTS AND EVALUATION

A. DATA DESCRIPTION

The first dataset used for evaluation, UT-HAR [24], is a publicly available CSI dataset for human activity recognition, containing a total of seven common activities of daily life. It was collected using an Intel 5300 network card with 3 pairs of antennas, each pair recording 30 subcarriers at a sampling frequency of 1kHz. All data were collected in the same indoor environment. During the data collection process, each

person performed each activity for 20 seconds, and a sliding window with a window size of 2 seconds was used for data segmentation.

In addition, to provide a more comprehensive evaluation of the model, we also used the BVP data from the Widar3.0 dataset [26]. This data was obtained by transforming CSI and eliminating the influence of environmental noise. The dataset consists of 43K samples collected using a 3 × 3 pair of antennas with an Intel 5300 network card and includes 22 gesture classes performed by 16 volunteers in various environments. The dimension of BVP data is 22 × 20 × 20, where the first dimension represents the length of time, and the second and third dimensions represent the velocity components of the gesture action along the x and y axes of the body coordinate at a certain moment.

B. EVALUATION ON UT-HAR DATA SET

In the experiment, we ensemble three homogeneous base models implemented on the PyTorch platform. When evaluating the model using the UT-HAR dataset, each base model was trained for 50 epochs using the Adam optimizer, with a batch size of 64 and a learning rate of 0.0001. To validate the effectiveness of our proposed model, we compared it with baseline methods based on CSI action recognition as well as several popular approaches. In [24], the authors introduced the UT-HAR dataset and used the commonly used LSTM model in deep learning for activity classification. Wang et al. [40] designed a Sparse Autoencoder network (SAE) for simultaneous location, activity, and gesture recognition. Sheng et al. [18] proposed a CNN-BiLSTM deep learning model, achieving high accuracy in cross-scene action recognition. Chen et al. [37] proposed an attention-based BiLSTM model (ABLSTM), which delivered superior performance in action recognition.

The confusion matrices for the five methods are shown in Fig. 5. In the confusion matrix, the x-axis represents the actual activity types, and the y-axis represents the predicted activity types, with diagonal elements indicating the recognition accuracy for each activity. It can be observed from these confusion matrices that there is a significant disparity in the recognition accuracy of different activities. The recognition performance of activities such as "Run", "Walk", and "Fall" is relatively good, likely due to their larger amplitudes, making their data features more distinct and easier to extract. Additionally, most methods have the lowest recognition accuracy for the activities of "Sit down" and "Stand up", possibly because the CSI features of these two activities are similar and can be easily confused with the activity of "Lie down".

It can be observed that these methods have all achieved high recognition accuracy. The average accuracy for activity recognition based on the SAE method is 86.25%. Due to considering the temporal dependency of sequences, the methods based on the LSTM model have achieved better recognition results compared to SAE. The basic LSTM method achieved an average accuracy of 90.5% for activity recognition, while

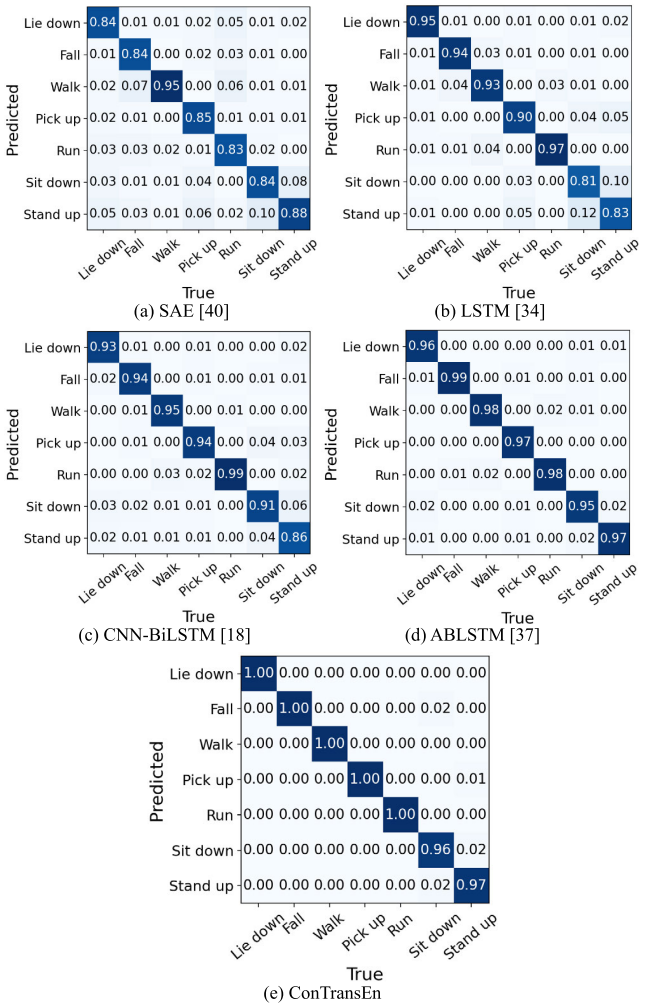

FIGURE 5. The confusion matrix obtained by five methods.

the CNN-BiLSTM method achieved an average accuracy of 93.08%, and the ABLSTM method achieved an average accuracy of 97.19%. The ConTransEn model we proposed utilizes CNN and ViT to extract spatial and temporal features of sequences, and finally improves the overall performance of the model through the bagging algorithm. In comparison, our proposed ConTransEn model achieved the highest average accuracy for activity recognition at 99.41%. The recognition accuracy for the activities of "Sit down" and "Stand up" both exceeded 95%, while the recognition accuracy for the other five activities surpassed 99.5%. Furthermore, from the graph, it can be seen that our method achieved the highest recognition accuracy for each activity among these methods, demonstrating the effectiveness of the CNN and ViT modules in extracting CSI sequence spatial and temporal features. The average recognition accuracy for the five methods is shown in Fig. 6.

Ablation Analysis: In order to evaluate the contributions of the different components of our proposed method to the overall performance, we analyzed the effectiveness of the

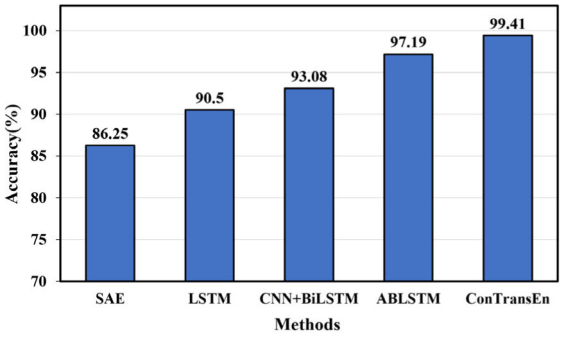

FIGURE 6. The average recognition accuracy of five methods.

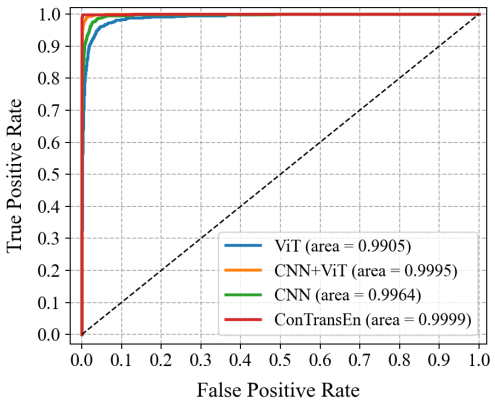

FIGURE 7. ROC curve comparison of CNN, ViT, CNN + ViT and ConTransEn.

combined CNN and ViT modules as well as the rationality of using the Bagging algorithm. As shown in Fig. 7, we plotted the receiver operating characteristic (ROC) curves for the individual CNN and ViT models, the combined CNN and ViT model, and our proposed ConTransEn model. The horizontal axis False Positive Rate (FPR) represents the proportion of negative samples incorrectly predicted as positive among all negative samples, and the vertical axis True Positive Rate (TPR), also known as recall, represents the proportion of positive samples correctly predicted as positive among all positive samples. From the graph, it can be observed that the CNN + ViT model has a significantly higher Area Under the Curve (AUC) value compared to the individual CNN and ViT models. This indicates that the CNN + ViT model is capable of extracting key features from the CSI signals more effectively than the standalone CNN or ViT models. This is because the CNN model can extract local features of the data through multiple convolutions, which are then further captured by the long-range dependencies in the ViT module through its attention mechanism. Additionally, the graph shows that the utilization of the Bagging algorithm leads to a slight improvement in recognition performance compared to the basic combined model. Although the improvement may not be very significant, the use of the Bagging algorithm plays a crucial role in enhancing the stability of the model. Bagging achieves this by performing bootstrap sampling on the training data to generate multiple subsets, training multiple models on these subsets, and then averaging the predictions of these models. This effectively reduces the impact of noisy in CSI data and decreases the variance of the model, thereby improving model performance. In our experiments on another Widar dataset, the use of the Bagging algorithm resulted in an average increase of 3.86% in recognition accuracy compared to not using it, further confirming that the Bagging algorithm can effectively improve model performance.

K-Fold Cross-Validation: In order to comprehensively evaluate the performance of the model, we employed K-fold cross-validation to conduct experiments on the basic model CNN + ViT. We chose K=5, then merged the training and testing sets, and randomly divided the merged dataset into five equally sized subsets for five rounds of training and validation. In each round, four subsets were used as the training set, with the remaining subset used as the validation set. The model was then trained on the training set and evaluated on the validation set. Finally, the performance metrics obtained from the five rounds were averaged to yield the final evaluation result. The results of the cross-validation are presented in Table 2. It can be observed from the table that the accuracy for each fold is very high, with the accuracy reaching 100% in the last three folds. The average accuracy across the five folds is 99.47%, indicating that the proposed basic model exhibits good generalization capabilities on different subsets of data. The precision and recall scores are also very high, with an average of over 98% across the five folds, indicating the model's strong ability to correctly identify positive instances and classify them accurately.

TABLE 2. Results of five-fold cross-validation on the base model.

Fold	1	2	3	4	5	Mean
Accuracy	0.9744	0.9989	1.0	1.0	1.0	0.9947
Precision	0.9683	0.9827	0.9881	0.9909	0.9929	0.9846
Recall	0.9643	0.9804	0.9867	0.9899	0.9920	0.9827

Impact of the Number of Encoder Layers and Attention Heads: Additionally, we analyzed the influence of the number of Encoder layers and the number of attention heads in the ViT module on recognition performance. The experimental results obtained using different numbers of Encoder layers and attention heads are shown in Table 3. A simple analysis of the data in the table reveals that recognition performance generally improves with increasing numbers of Encoder layers and attention heads. The model performs best when the Encoder has 2-6 layers and the attention heads are 7, 8, and 10-12. Increasing the number of attention heads and Encoder layers in ViT can enhance the model's ability to capture local and global information in CSI data sequences, aiding in learning more abstract and complex feature representations, thereby improving the model's representational capacity.

TABLE 3. Influence of the number of Encoder layers and attention heads on recognition accuracy.

Number of Heads	Number of Layers					
	1	2	3	4	5	6
1	99.38	98.89	99.37	99.38	99.41	99.43
2	99.27	99.36	99.38	99.48	99.40	99.51
3	99.39	99.37	**99.57**	99.28	99.09	99.42
4	99.45	99.47	99.48	**99.50**	**99.51**	99.42
5	99.39	99.34	99.48	99.37	99.22	99.41
6	99.38	99.25	99.28	99.38	99.39	**99.54**
7	99.38	99.26	99.38	99.37	**99.61**	99.51
8	**99.49**	99.38	99.48	99.38	**99.61**	99.41
9	99.38	**99.48**	99.38	99.48	99.41	99.41
10	**99.48**	**99.48**	99.38	**99.51**	98.90	**99.53**
11	99.38	**99.61**	99.46	99.49	**99.51**	99.41
12	99.27	99.41	**99.51**	**99.50**	99.50	99.41

However, a larger number of attention heads can also increase the computational complexity and the number of parameters, leading to increased training and inference time costs. Overly deep models may lead to the problems of vanishing or exploding gradients, as well as increased training time and resource consumption. Therefore, in order to maintain the model's recognition performance and stability while keeping the training cost reasonable, we chose to set the number of Encoder layers to 5 and the number of attention heads to 8.

Computational Cost and Model Parameters: The number of model parameters reflects the complexity of the model and determines the memory consumption during the training process, while Floating Point Operations (Flops) reflect the computational complexity of the model and determine the duration of the inference process. We calculated the model parameter quantity and Flops for the five methods, as shown in Table 4. From the table, it can be observed that the computational cost and model parameter quantity of the other three composite models are significantly higher than that of the SAE and LSTM models. This is because the SAE and LSTM models have relatively simpler structures with fewer layers, resulting in higher training efficiency and lower resource consumption, but their performance is relatively poorer. Additionally, our proposed method has a much larger number of parameters compared to the other four methods. This is mainly due to the positional encoding and multiple multi-head self-attention layers involved in the ViT module. Furthermore, the multiple convolution layers in the CNN module contribute to the increase in model parameters. Although a large number of model parameters increase the burden on memory and computational resources, they also enhance the model's expressive power, enabling it to learn and capture more subtle features, thereby better adapting to different tasks. From Table 4, it can also be seen that the CNN-BiLSTM model has the highest FLOPs, followed by ConTransEn. The CNN-BiLSTM method uses a double-layer BiLSTM structure, which involves a large amount of recurrent computation, leading to high computational cost. Similarly, ConTransEn has multiple convolution layers as well as computations with multiple heads of attention, resulting in high computational complexity. Therefore, during the model training process, we used the 'apex' library to improve training efficiency and reduce memory consumption through mixed-precision training. Although the training process takes relatively longer, it is an offline process as it needs to be performed only once to conduct online testing on the samples. Our proposed approach took a total of 3.14 seconds to test all 996 samples, which means that each sample's test time is approximately 0.0032 seconds. Hence, it can be said that ConTransEn can be used for real-time action recognition in a WiFi environment.

TABLE 4. Computational complexity and model parameters of five methods.

Method	Parameters (M)	Flops (M)
SAE	0.18	30.56
LSTM	0.25	61.70
CNN-BiLSTM	1.48	4844.99
ABLSTM	0.47	465.16
ConTransEn	73.32	3340.95

C. EVALUATION ON WIDAR DATA SET

Due to the limited number of samples in the UT-HAR dataset and the potential inherent noise in CSI data, to further validate the stability of the model, we also utilized the Widar3.0 dataset for evaluation. The dataset samples are collected from multiple volunteers in various environments and utilize environment-agnostic BVP data to eliminate the influence of environmental noise. The experiments were implemented on the PyTorch platform, integrating three basic models, with each model trained for 30 epochs. The models were optimized using the stochastic gradient descent with momentum (SGDM) algorithm, with a batch size of 32, a learning rate of 0.001, and a momentum of 0.9.

Our method achieved an average recognition accuracy of 85.09% on the Widar dataset, and the confusion matrix of various gestures are shown in Fig. 8. Each of the labels on the axes of the confusion matrix in the figure represents a type of gesture. Apart from the first four gestures, the word "drawing" is omitted before the other gesture labels. The letters 'H' in parentheses after the labels indicate that the gesture was performed in the horizontal plane, while the letter 'V' indicates that the gesture was performed in the vertical plane. From the figure, it can be observed that gestures such as "sweep," "drawing triangle," "drawing number 6,"

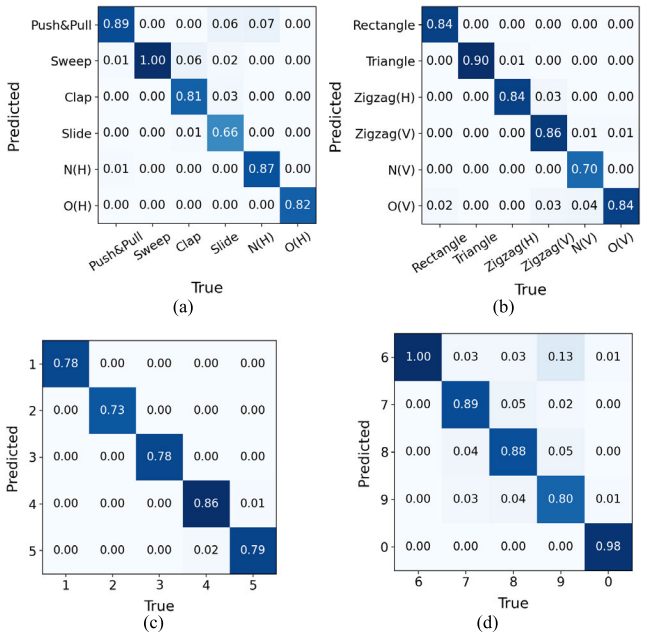

FIGURE 8. Confusion matrices of different gestures.

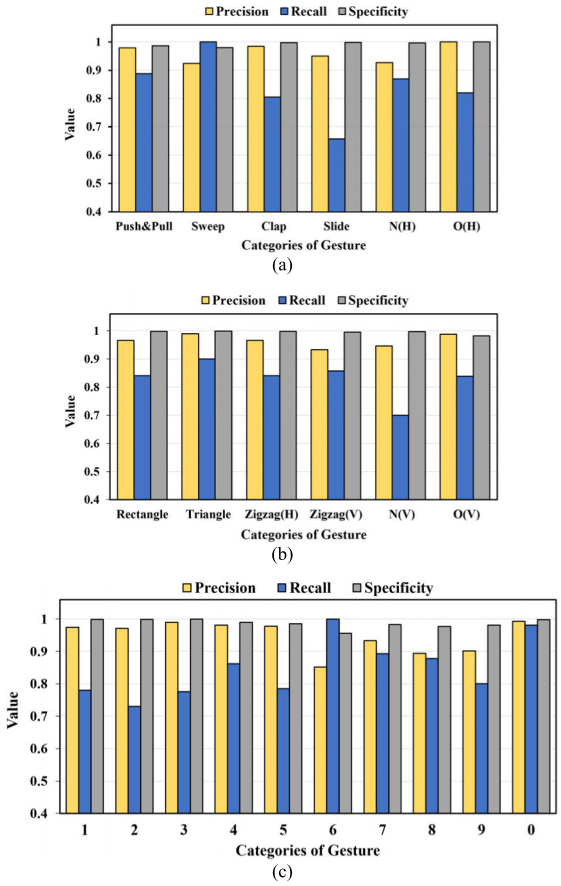

FIGURE 9. Bar chart of precision, recall, and specificity for gestures.

and "drawing number 0" have relatively high recognition accuracy, all reaching 90%. Therefore, the model achieved excellent recognition performance on the Widar 3.0 dataset, indicating its strong ability to learn different features from various data modalities and further validating its robustness. The gesture of "slide" had the lowest recognition accuracy of 66%, possibly due to WiFi signals being obstructed at certain angles when the gesture is performed in front of the subject's torso. Moreover, this gesture is relatively simple and lacks distinct features, making it challenging for the model to extract key information from the corresponding input sequences. The recognition accuracy for the "drawing N in the vertical plane" and "drawing number 2" gestures was relatively lower. The main reason is that these two gestures are quite similar and both performed in the vertical plane, making it difficult for the model to accurately capture the key features solely based on BVP data. Hence, the model tends to confuse them during classification.

The bar chart of precision, recall, and specificity for each gesture is shown in Fig. 9. From the graph, it can be seen that the specificity of all gestures reaches over 95%, indicating that the model has high classification accuracy and reliability. Among them, gestures such as "slide," "drawing N in the vertical plane," and "drawing numbers 1, 2" have high precision but relatively low recall. This suggests that other gestures are not easily misclassified as these gestures, but the model tends to classify these gestures as other gesture categories incorrectly. The possible reason is that there are significant differences between the samples of these gestures, making it difficult for the model to effectively extract the key features of these gestures, leading to confusion with other similar gestures. Additionally, the recall rate of the gesture "drawing number 6" is close to 1 but its precision is relatively low at 0.85. This indicates that other gestures are prone to be misclassified as this gesture. Moreover, the recall rate of the gesture "drawing number 9" is low, and these two gestures are similar, hence it is likely that the gesture "drawing number 9" is easily misclassified as the gesture "drawing number 6."

V. CONCLUSION

In this paper, we propose a deep learning model called ConTransEn, based on CNN and Transformer, and utilizing Bagging ensemble learning algorithm for human activity recognition in WiFi environments. ConTransEn learns spatial features from raw data sequences using CNN, and then uses an attention-based Vision Transformer module to allocate different weights to different positions of the sequence, thereby capturing long-term dependencies. We experimentally validate the effectiveness of our proposed method for action recognition and compare it with popular methods, including traditional deep learning methods such as SAE and LSTM, as well as combined model methods such as CNN-BiLSTM and ABLSTM. The results show that our proposed model outperforms these methods in recognition performance. Furthermore, we have demonstrated the effectiveness of combining CNN and ViT modules in our model

through ROC curves, as well as the rationality of using the Bagging algorithm. Afterwards, we further confirmed the stability of the base model through cross-validation, and analyzed the impact of the number of Encoder layers and attention heads in ViT on the recognition performance of the model. Additionally, we discussed the complexity and computational cost of the model. Finally, we evaluated the model using a gesture dataset in the form of BVP data, and it also achieved good recognition performance. The action samples in the different datasets were obtained under different environments and signal conditions. The model achieved good recognition performance on both datasets, demonstrating a certain degree of robustness.

However, our proposed method still has some limitations. For example, the model may have limited generalization ability for human activity recognition in different environments, as different environments introduce different noise and interference. In future work, we will try to apply our model to broader scenarios, such as environments with more obstacles or variable conditions, which will place higher demands on the adaptability and generalization ability of our model. Additionally, we will also further attempt to improve the model to extract more detailed features from different data modalities, thus avoiding misclassification between similar actions.

REFERENCES

[1] Y. Wang, J. Shen, and Y. Zheng, "Push the limit of acoustic gesture recognition," in *Proc. IEEE Conf. Comput. Commun.*, Jul. 2020, pp. 566–575, doi: 10.1109/INFOCOM41043.2020.9155402.

[2] N. Quader, J. Lu, P. Dai, and W. Li, "Towards efficient coarse-to-fine networks for action and gesture recognition," in *Computer Vision—ECCV*. Cham, Switzerland: Springer, 2020, pp. 35–51.

[3] T. McGrath and L. Stirling, "Body-worn IMU human skeletal pose estimation using a factor graph-based optimization framework," *Sensors*, vol. 20, no. 23, p. 6887, Dec. 2020, doi: 10.3390/s20236887.

[4] S. Jiang, B. Lv, W. Guo, C. Zhang, H. Wang, X. Sheng, and P. B. Shull, "Feasibility of wrist-worn, real-time hand, and surface gesture recognition via sEMG and IMU sensing," *IEEE Trans. Ind. Informat.*, vol. 14, no. 8, pp. 3376–3385, Aug. 2018, doi: 10.1109/TII.2017.2779814.

[5] D. Halperin, W. Hu, A. Sheth, and D. Wetherall, "Tool release: Gathering 802.11n traces with channel state information," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 41, no. 1, pp. 1–53, Jan. 2011.

[6] J. Yang, X. Chen, D. Wang, H. Zou, C. Xiaoxuan Lu, S. Sun, and L. Xie, "SenseFi: A library and benchmark on deep-learning-empowered WiFi human sensing," 2022, *arXiv:2207.07859*.

[7] X. Wang, C. Yang, and S. Mao, "On CSI-based vital sign monitoring using commodity WiFi," *ACM Trans. Comput. Healthcare*, vol. 1, no. 3, pp. 1–27, Jul. 2020, doi: 10.1145/3377165.

[8] Y. Chu, K. Cumanan, S. K. Sankarpandi, S. Smith, and O. A. Dobre, "Deep learning-based fall detection using WiFi channel state information," *IEEE Access*, vol. 11, pp. 83763–83780, 2023.

[9] J. Ding and Y. Wang, "WiFi CSI-based human activity recognition using deep recurrent neural network," *IEEE Access*, vol. 7, pp. 174257–174269, 2019, doi: 10.1109/ACCESS.2019.2956952.

[10] S. Duan, T. Yu, and J. He, "WiDriver: Driver activity recognition system based on WiFi CSI," *Int. J. Wireless Inf. Netw.*, vol. 25, no. 2, pp. 146–156, Jun. 24, 2018, doi: 10.1007/s10776-018-0389-0.

[11] Y. Wang, J. Liu, Y. Chen, M. Gruteser, J. Yang, and H. Liu, "E-eyes: Device-free location-oriented activity identification using fine-grained WiFi signatures," in *Proc. 20th Annu. Int. Conf. Mobile Comput. Netw.*, Sep. 2014, pp. 617–628, doi: 10.1145/2639108.2639143.

[12] J. Xue, J. Zhang, Z. Gao, and W. Xiao, "Enhanced WiFi CSI fingerprints for device-free localization with deep learning representations," *IEEE Sensors J.*, vol. 23, no. 3, pp. 2750–2759, Feb. 2023, doi: 10.1109/JSEN.2022.3231611.

[13] Y. Gu, X. Zhang, Y. Wang, M. Wang, H. Yan, Y. Ji, Z. Liu, J. Li, and M. Dong, "WiGRUNT: WiFi-enabled gesture recognition using dual-attention network," *IEEE Trans. Hum.-Mach. Syst.*, vol. 52, no. 4, pp. 736–746, Aug. 2022.

[14] J. Ding, Y. Wang, and X. Fu, "Wihi: WiFi based human identity identification using deep learning," *IEEE Access*, vol. 8, pp. 129246–129262, 2020, doi: 10.1109/ACCESS.2020.3009123.

[15] S. Liu, Y. Zhao, and B. Chen, "WiCount: A deep learning approach for crowd counting using WiFi signals," in *Proc. IEEE Int. Symp. Parallel Distrib. Process. Appl. IEEE Int. Conf. Ubiquitous Comput. Commun. (ISPA/IUCC)*, Dec. 2017, pp. 967–974.

[16] Y. Gu, H. Yan, M. Dong, M. Wang, X. Zhang, Z. Liu, and F. Ren, "WiONE: One-shot learning for environment-robust device-free user authentication via commodity Wi-Fi in man–machine system," *IEEE Trans. Computat. Social Syst.*, vol. 8, no. 3, pp. 630–642, Jun. 2021, doi: 10.1109/TCSS.2021.3056654.

[17] H. Zou, Y. Zhou, J. Yang, H. Jiang, L. Xie, and C. J. Spanos, "DeepSense: Device-free human activity recognition via autoencoder long-term recurrent convolutional network," in *Proc. IEEE Int. Conf. Commun.*, May 2018, pp. 1–6, doi: 10.1109/ICC.2018.8422895.

[18] B. Sheng, F. Xiao, L. Sha, and L. Sun, "Deep spatial–temporal model based cross-scene action recognition using commodity WiFi," *IEEE Internet Things J.*, vol. 7, no. 4, pp. 3592–3601, Apr. 2020.

[19] X. Ding, T. Jiang, Y. Zhong, S. Wu, J. Yang, and W. Xue, "Improving WiFi-based human activity recognition with adaptive initial state via one-shot learning," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*. China: IEEE, Mar. 2021, pp. 1–6, doi: 10.1109/WCNC49053.2021.9417590.

[20] Y. Wang, L. Guo, Z. Lu, X. Wen, S. Zhou, and W. Meng, "From point to space: 3D moving human pose estimation using commodity WiFi," *IEEE Commun. Lett.*, vol. 25, no. 7, pp. 2235–2239, Jul. 2021, doi: 10.1109/LCOMM.2021.3073271.

[21] J. Yang, X. Chen, H. Zou, D. Wang, Q. Xu, and L. Xie, "EfficientFi: Toward large-scale lightweight WiFi sensing via CSI compression," *IEEE Internet Things J.*, vol. 9, no. 15, pp. 13086–13095, Aug. 2022.

[22] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2017, *arXiv:1706.03762*.

[23] Y. Xie, Z. Li, and M. Li, "Precise power delay profiling with commodity Wi-Fi," *IEEE Trans. Mobile Comput.*, vol. 18, no. 6, pp. 1342–1355, Jun. 2019, doi: 10.1109/TMC.2018.2860991.

[24] S. Yousefi, H. Narui, S. Dayal, S. Ermon, and S. Valaee, "A survey on behavior recognition using WiFi channel state information," *IEEE Commun. Mag.*, vol. 55, no. 10, pp. 98–104, Oct. 2017, doi: 10.1109/MCOM.2017.1700082.

[25] J. Yang, H. Zou, H. Jiang, and L. Xie, "Device-free occupant activity sensing using WiFi-enabled IoT devices for smart homes," *IEEE Internet Things J.*, vol. 5, no. 5, pp. 3991–4002, Oct. 2018.

[26] Y. Zhang, Y. Zheng, K. Qian, G. Zhang, Y. Liu, C. Wu, and Z. Yang, "Widar3.0: Zero-effort cross-domain gesture recognition with Wi-Fi," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 11, pp. 8671–8688, Nov. 2021, doi: 10.1109/TPAMI.2021.3105387.

[27] W. Wang, A. X. Liu, M. Shahzad, K. Ling, and S. Lu, "Device-free human activity recognition using commercial WiFi devices," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 5, pp. 1118–1131, May 2017.

[28] S. Tan, L. Zhang, Z. Wang, and J. Yang, "MultiTrack: Multi-user tracking and activity recognition using commodity WiFi," in *Proc. CHI Conf. Hum. Factors Comput. Syst.*, May 2019, pp. 1–12, doi: 10.1145/3290605.3300766.

[29] X. Li, S. Li, D. Zhang, J. Xiong, Y. Wang, and H. Mei, "Dynamic-MUSIC: Accurate device-free indoor localization," in *Proc. ACM Int. Joint Conf. Pervasive Ubiquitous Comput.*, Sep. 2016, pp. 196–207, doi: 10.1145/2971648.2971665.

[30] K. Qian, C. Wu, Z. Zhou, Y. Zheng, Z. Yang, and Y. Liu, "Inferring motion direction using commodity Wi-Fi for interactive exergames," in *Proc. CHI Conf. Hum. Factors Comput. Syst.*, May 2017, pp. 1961–1972, doi: 10.1145/3025453.3025678.

[31] D. Wu, Y. Zeng, R. Gao, S. Li, Y. Li, R. C. Shah, H. Lu, and D. Zhang, "WiTraj: Robust indoor motion tracking with WiFi signals," *IEEE Trans. Mobile Comput.*, vol. 22, no. 5, pp. 3062–3078, May 2023, doi: 10.1109/TMC.2021.3133114.

[32] X. Li, D. Zhang, Q. Lv, J. Xiong, S. Li, Y. Zhang, and H. Mei, "IndoTrack: Device-free indoor human tracking with commodity Wi-Fi," *Proc. ACM Interact., Mobile, Wearable Ubiquitous Technol.*, vol. 1, no. 3, pp. 1–22, Sep. 2017, doi: 10.1145/3130940.

[33] K. Qian, C. Wu, Y. Zhang, G. Zhang, Z. Yang, and Y. Liu, "Widar2.0: Passive human tracking with a single Wi-Fi link," in *Proc. 16th Annu. Int. Conf. Mobile Syst., Appl., Services*. Germany: ACM, Jun. 2018, pp. 350–361, doi: 10.1145/3210240.3210314.
[34] H. Zou, J. Yang, Y. Zhou, L. Xie, and C. J. Spanos, "Robust WiFi-enabled device-free gesture recognition via unsupervised adversarial domain adaptation," in *Proc. 27th Int. Conf. Comput. Commun. Netw. (ICCCN)*, Jul. 2018, pp. 1–8.
[35] S. Shang, Q. Luo, J. Zhao, R. Xue, W. Sun, and N. Bao, "LSTM-CNN network for human activity recognition using WiFi CSI data," *J. Phys. Conf. Ser.*, vol. 1883, no. 1, Apr. 2021, Art. no. 012139, doi: 10.1088/1742-6596/1883/1/012139.
[36] F.-Y. Chu, C.-J. Chiu, A.-H. Hsiao, K.-T. Feng, and P.-H. Tseng, "WiFi CSI-based device-free multi-room presence detection using conditional recurrent network," in *Proc. IEEE 93rd Veh. Technol. Conf.*, Apr. 2021, pp. 1–5, doi: 10.1109/VTC2021-Spring51267.2021.9448848.
[37] Z. Chen, L. Zhang, C. Jiang, Z. Cao, and W. Cui, "WiFi CSI based passive human activity recognition using attention based BLSTM," *IEEE Trans. Mobile Comput.*, vol. 18, no. 11, pp. 2714–2724, Nov. 2019.
[38] Y. Zhou, H. Huang, S. Yuan, H. Zou, L. Xie, and J. Yang, "MetaFi++: WiFi-enabled transformer-based human pose estimation for metaverse avatar simulation," *IEEE Internet Things J.*, vol. 10, no. 16, pp. 14128–14136, Aug. 2023, doi: 10.1109/JIOT.2023.3262940.
[39] Y. Gu, H. Yan, X. Zhang, Y. Wang, J. Huang, Y. Ji, and F. Ren, "Attention-based gesture recognition using commodity WiFi devices," *IEEE Sensors J.*, vol. 23, no. 9, pp. 9685–9696, May 2023, doi: 10.1109/JSEN.2023.3261325.
[40] J. Wang, X. Zhang, Q. Gao, H. Yue, and H. Wang, "Device-free wireless localization and activity recognition: A deep learning approach," *IEEE Trans. Veh. Technol.*, vol. 66, no. 7, pp. 6258–6267, Jul. 2017, doi: 10.1109/TVT.2016.2635161.

FEI GE received the B.S. and M.S. degrees in measurement and automatic devices and the Ph.D. degree in communication and information system from Wuhan University, Wuhan, China, in 1997, 2001, and 2005, respectively. From 2008 to 2010, he was a Postdoctoral Research Fellow with the College of Physical Science and Technology, Central China Normal University, Wuhan. From 2010 to 2011, he was a Research Fellow with the Department of Electrical Engineering, City University of Hong Kong. He is currently an Associate Professor with the Computer Science Department, Central China Normal University. He has published more than 40 papers in international journals and conference proceedings, including IEEE, ACM, and Elsevier journals. His research interests include embedded systems, transmission control, the Internet of Things, wireless network communications, and data processing. He acts as a reviewer for international journals and a TPC member for conferences. His award includes Hubei Science and Technology Progress Award.

ZHIMIN YANG received the B.Eng. degree from Wuhan University of Science and Technology, in 2022. He is currently pursuing the degree with the School of Computer Science, Central China Normal University, Wuhan, China. His research interests include wireless sensing, wireless network communications, and machine learning.

ZHENYANG DAI received the B.S. degree in computer and software engineering from Nanjing University of Information Science and Technology, in 2021. He is currently pursuing the degree with the School of Computer Science, Central China Normal University. His research interests include algorithm optimization and machine learning.

LIANSHENG TAN received the Ph.D. degree in mathematical science from Loughborough University, U.K., in 1999. He was a Postdoctoral Research Fellow with the School of Information Technology and Engineering, University of Ottawa, Canada, in 2001. He was a Research Fellow with the Research School of Information Sciences and Engineering, The Australian National University, Australia, from 2006 to 2009. He was a Professor with the Department of Computer Science, Central China Normal University, China. He also held a number of visiting research positions at Loughborough University, the University of Tsukuba, the City University of Hong Kong, and The University of Melbourne. He is currently with the Discipline of ICT, School of Technology, Environments and Design, University of Tasmania, Australia. He has published over 130 papers in international journals and conference proceedings including over 20 in IEEE and ACM journals and two monographs with Elsevier and Taylor & Francis. His research interests include cloud computing, the Internet of Things, computer networks, and wireless sensor networks. He was an Editor of *Dynamics of Continuous, Discrete and Impulsive Systems* (Series B: Applications and Algorithms), from 2006 to 2008, and an Editor of *International Journal of Communication Systems*. He is the Editor-in-Chief of *Journal of Computers* and an Editor of *International Journal of Computer Networks and Communications*.

JIANYUAN HU received the B.Eng. degree from Jianghan University, in 2021. He is currently pursuing the degree with the School of Computer Science, Central China Normal University, Wuhan, China. His research interests include wireless sensor networks and computer networks.

JIAYUAN LI received the B.Eng. degree from Panzhihua University, Sichuan, China, in 2022. He is currently pursuing the degree with the School of Computer Science, Central China Normal University. His research interests include wireless networks and machine learning.

HAN QIU received the bachelor's degree from Hunan University, Changsha, China, in 2018. He is currently pursuing the degree with the School of Computer Science, Central China Normal University. His main research interests include applied software development, intelligent optimization algorithms, and machine learning.