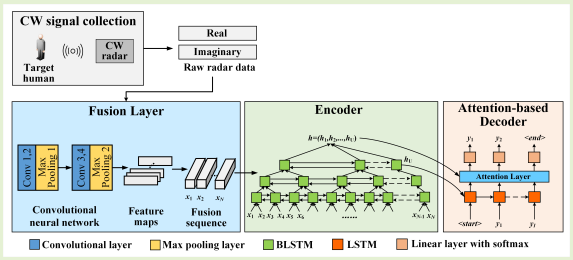
An End-to-End Network for Continuous Human Motion Recognition via Radar Radios

Running Zhao, Xiaolin Ma, Xinhua Liu, and Jian Liu

***Abstract*— Micro-Doppler-based continuous human motion recognition (HMR) has gained considerable attention recently. However, existing methods mainly rely on individual recurrent neural network or sliding-window-based approaches, which makes them hard to effectively exploit all the temporal information to predict motions. Additionally, they need to represent the raw radar data into other domains and then perform feature extraction and classification. Thus, the representation cannot be optimized, and its high computational complexity and independence from learning model make the network consume significant time. In this paper, to address these issues, we propose a new end-to-end network that uses radar radios to recognize continuous motion. Specifically, the fusion layer fuses the raw I & Q radar data without the need of representations, and it is integrated with subsequent networks in an end-to-end manner for jointly optimization. Moreover, the attention-based encoder-decoder structure encodes the fused data and selects useful temporal information for recognition, which guarantees the effective use of all the temporal information. The experiments show that in continuous HMR, the proposed network outperforms existing methods in terms of accuracy and inference time.**



***Index Terms*— Continuous human motion recognition, micro-Doppler, raw radar data, deep learning, end-to-end, attention-based encoder-decoder.**

I. Introduction

HUMAN motion recognition (HMR) has attracted significant research interest due to its wide range of applications. For example, HMR could help detect or prevent falls in residential care homes for elderly people [1]. It could also recognize different human body or hand gestures to enable more convenient human-computer interactions [2]. Various technologies have been investigated for HMR, ranging from camera-based technologies to wearable-sensor-based technologies. However, the approaches relying on cameras are susceptible to illumination, weather or obstruction, and would raise privacy concerns. Moreover, wearable-sensor-based solutions require users to attach multiple sensors on their bodies, resulting in a great inconvenience for many practical application scenarios [3]. In contrast, radar sensors are robust to various lighting conditions and they are non-intrusive [4]. Considering these attractive attributes of radar, significant efforts are recently made to explore radar-based HMR technologies that utilize micro-Doppler signatures extracted from radar radios to recognize human motion [5]–[7].

In the early stage, researchers focused on micro-Doppler-based single motion recognition, where an observation only includes one type of motion with fixed duration. Conventional methods mainly extract handcrafted features from the spectrogram generated by short-time Fourier transform (STFT) [8]–[10] or adopt linear predictive coding [11] and empirical mode decomposition [12] to extract features, and then use machine learning classifiers to perform feature-based classification. With the recent advancement of deep learning technologies, numerous related networks have been applied in radar-based HMR, and such networks yield reasonably good performance and generalization. Kim and Moon [13] pioneered the application of deep convolutional neural network (DCNN) to classify human activity. In [14], deep learning incorporated with spectrogram and range maps was applied for fall detection. In [15], an omnidirectional CNN was designed to classify human motions collected from different aspect angles. A parallel DCNN was proposed for gait classification based on multistatic radar micro-Doppler signatures [16]. Moreover, generative adversarial networks [17] and transfer learning [18] are considered when the amount of training data is insufficient.

In practice, human motion is an inherently dynamic time stream of actions [6], that is, people usually perform motions one after another with varying duration. Due to this reason, continuous motion recognition based on micro-Doppler signa-

Manuscript received x x, 2020; revised x x, 2020; accepted x x, 2020. Date of publication x x, 2020; date of current version x x, 2020. This work is supported by the National Natural Science Foundation of China under Grants no. 61772088 and 61502361. *(Corresponding author: Xiaolin Ma.)*

Running Zhao, Xiaolin Ma, and Xinhua Liu are with the Hubei Key Laboratory of Broadband Wireless Communication and Sensor Network, School of Information Engineering, Wuhan University of Technology, Wuhan 430070, China (e-mail: zhaorunning@whut.edu.cn; maxiaolin0615@whut.edu.cn; liuxinhua@whut.edu.cn).

Jian Liu is with the Department of Electrical Engineering and Computer Science, The University of Tennessee, Knoxville, TN 37996 USA (e-mail: jliu@utk.edu).

tures has become a topical area of research. In [19], a dynamic range-Doppler (R-D) trajectory method based on sliding window was proposed, which separates each single motion from R-D images and then extracts features to recognize continuous motion. Moreover, different types of recurrent neural networks (RNNs) are introduced to process the temporal series of continuous motion. Zhang *et al.* [20] developed the Latern system that combines CNN and RNN to recognize continuous hand gesture based on range-time image. In [21], Du *et al.* proposed an architecture composed of convolutional layer with multiple filter sizes and gated recurrent units (GRU), to recognize continuous motion through micro-Doppler spectrogram. Wang *et al.* [22] designed a stacked GRU network (SGRUN), which slides across the spectrogram like a sliding window to recognize continuous motion. In [23], Long Short-Term Memory (LSTM) and Bidirectional LSTM (BLSTM) were used for continuous activity classification, and the result showed the latter has the better performance. A framework based on BLSTM for multimodal sensors was investigated to classify continuous motion [24].

However, the aforementioned methods use either individual RNN or sliding-window-based approaches to perform continuous HMR, neither of which can effectively use temporal information for prediction. For instance, individual RNN uses all features that contain too much useless temporal information to predict continuous motion, and sliding window with fixed-length uses a part of temporal information at a time to perform prediction. Moreover, all these methods represent the raw radar data into other domains (e.g., T-F domain or R-D domain) and then perform feature extraction and classification, that is, the representation is independent of the learning model. This makes the representation cannot be optimized by back-propagating errors. Meanwhile, the high computational complexity of representation and the independence of representation and learning model make the network consume significant time [25], whereas most existing methods do not take this time consumption into consideration.

In this paper, we propose a new end-to-end network composed of fusion layer and attention-based encoder-decoder to address the above-mentioned issues. Specifically, the fusion layer consisting of 1D convolutional and max pooling layers is used to fuse the raw I & Q radar data without using any representations. It has a similar effect to representation, but compared with it, the fusion layer can be integrated with subsequent network in an end-to-end manner for jointly optimization. Moreover, the attention-based encoder-decoder structure encodes the fused data and selects useful temporal information to recognize continuous motion, which guarantees the effective use of all the temporal information. With these techniques, the proposed network can achieve higher accuracy and shorter inference time for continuous HMR. The contributions of our work are summarized as follows:

1) We propose a new end-to-end network that is capable of using raw radar data to recognize continuous motion without the need of additional representation. This is the first use of such end-to-end structure in micro-Doppler-based continuous HMR. All the existing works involving continuous motion recognition are based upon representation, and all the existing end-to-end network can only recognize single motion.
2) We leverage the fusion layer and attention-based encoder-decoder to construct the network. They enable the network to be jointly optimized and effectively use all the temporal information in raw radar data for recognition, achieving higher accuracy and shorter inference time.

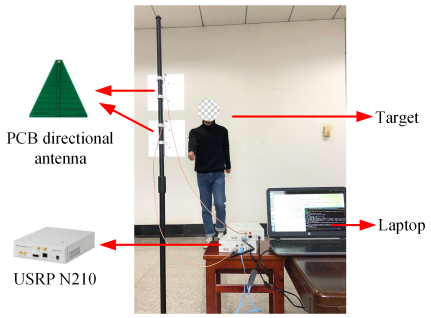


Fig. 1. Experimental scene and settings.

The remainder of the paper is organized as follows. Section II briefly introduces the signal model and data measurement. The details of the proposed end-to-end network are presented in Section III. Our experimental results are shown in Section IV and the paper is concluded in Section V.

II. Signal Model and Data Measurement

A. Signal Model

Continuous wave (CW) radar has been the first choice for radar-based HMR due to the simple hardware implementation. It transmits a sinusoidal signal $s(t)$ with the carrier frequency f_c, i.e., $s(t) = \exp(j2\pi f_c t)$. Suppose a point target located at a distance of R_0 from radar at time $t = 0$ moves with a velocity $v(t)$ towards the angle θ with respect to the radar line-of-sight. Then, the instantaneous distance $R(t)$ between the point target and the radar at time t is expressed as $R(t) = R_0 + \int_0^t v(u)\cos(\theta)du$. The radar receives the return signal scattered from the point target, which is given by:

$$x_p(t) = A(t)\exp(j2\pi f_c(t - \frac{2R(t)}{c})), \tag{1}$$

where $A(t)$ is the amplitude of the return signal and c is the speed of the electromagnetic wave propagation. The phase of the signal is $\varphi(t) = \frac{4\pi f_c R(t)}{c}$. In backscattering echoes, the micro-Doppler effect is caused by the vibration or rotation of human limbs, generating additional frequency modulation on the main Doppler shift induced by the torso [26]. Such Doppler and micro-Doppler signatures determine the features that underline different human motions.

Human motion is complex because it is composed of different movements of individual body parts. Among them, the torso motion is basically a translation with slightly body rocking and head movement. Whereas, the arms and legs swinging back and forth can be described as a partial vibration with a certain angle around a point [26]. By taking the time derivation of the phase, the Doppler frequency shift induced

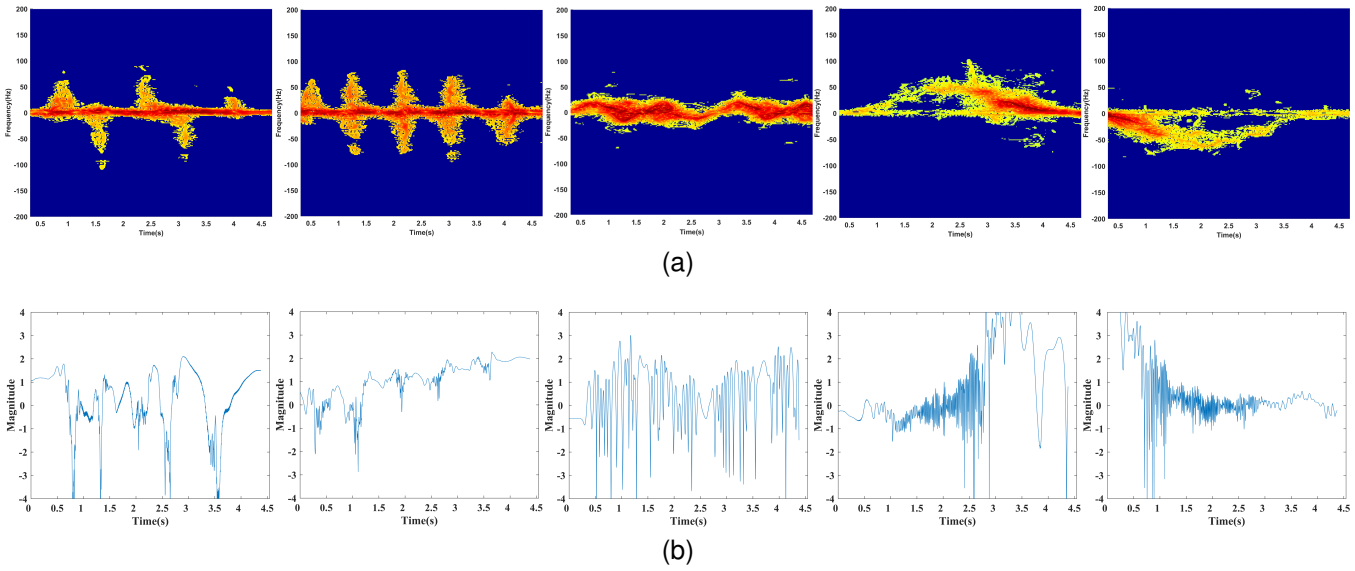


Fig. 2. Spectrograms and raw radar data waveforms of five single human motions. From left to right: one arm swinging, two arms swinging, squatting, running forward and running backward. (a) Spectrograms. (b) Raw radar data waveforms.

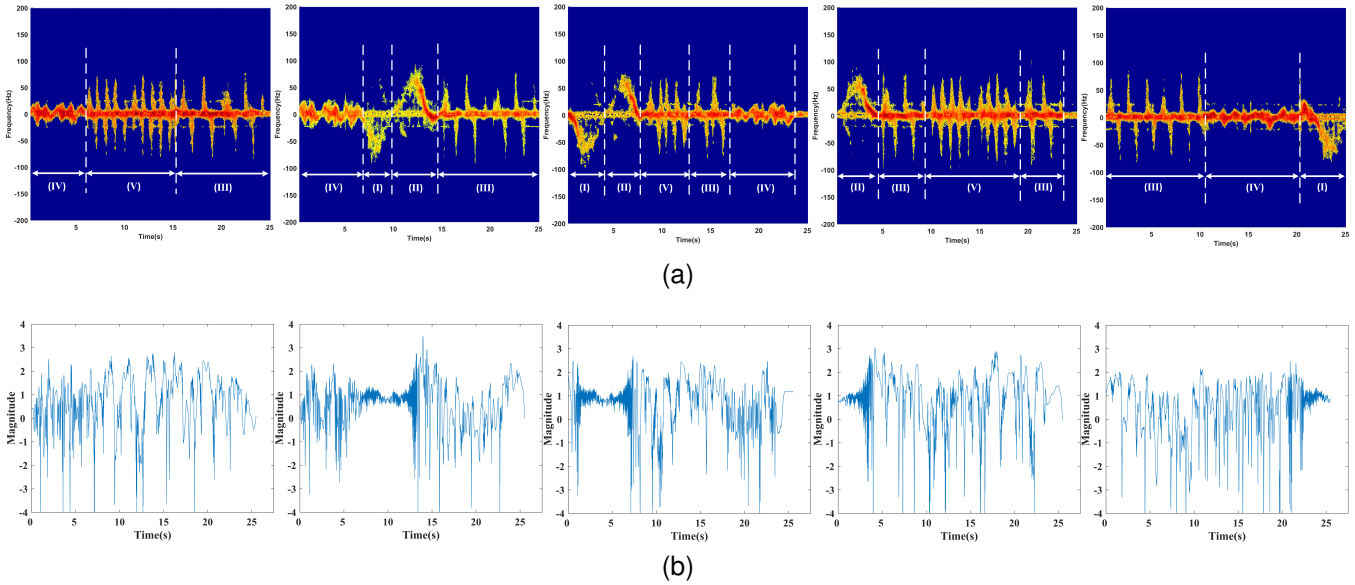


Fig. 3. Five examples of spectrogram and raw radar data waveform of continuous motion. (a) Spectrograms. (b) Raw radar data waveforms.

by the translation can be calculated as:

$$f_{translation}(t) = \frac{2v(t)\cos(\theta)f_c}{c}. \tag{2}$$

The micro-Doppler shift due to the vibration is:

$$f_{vibration} = \frac{4\pi f_c f_v D_v}{c}\cos(2\pi f_v t)S(\alpha,\beta), \tag{3}$$

where f_v is the vibration rate, D_v is the vibration amplitude, and $S(\alpha,\beta)$ is the function of azimuth angle. The Doppler and micro-Doppler frequencies corresponding to various parts of the body vary with time. Therefore, making full use of such time-varying properties is the key to recognizing motions.

B. Data Measurement

We built a continuous wave (CW) radar system using software radio equipment USRP N210 and open source software toolkit GNU Radio. The carrier frequency is 4.2 GHz and the instantaneous bandwidth is 120 MHz. Considering the signal redundancy, we set the sampling frequency to 420 Hz. The experimental scene and settings are shown in Fig. 1. We recruited 6 volunteers and each of them was asked to perform the following five motions: (I) running away from the antenna (running backward); (II) running towards the antenna (running forward); (III) walking while holding a subject (one arm swinging); (IV) standing and sitting (squatting); (V) walking (two arms swinging). The corresponding spectrograms and raw radar data waveforms of the above single human motions are shown in Fig. 2. Each volunteer repeated 140 times for each motion, and thus the total number of single motion samples is (5 motions)×(6 people)×(140 times), i.e., 4200. The collection lasted 5s. Moreover, we also directly collected 12000 continuous motion samples from 6 different volunteers

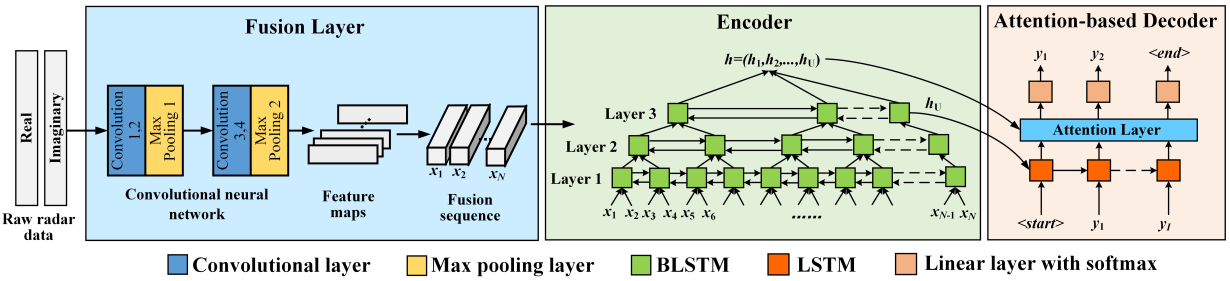


Fig. 4. Overall architecture of the proposed end-to-end network. The raw I & Q radar data is simply combined as two channels and each channel is a one-dimension time sequence. The BLSTM and LSTM blocks represent their corresponding cells. $\langle start \rangle$ and $\langle end \rangle$ are the special start-of-motion token and end-of-motion token, and $\boldsymbol{X}$, $\boldsymbol{h}$ and $\boldsymbol{Y}$ are fusion sequence, high-level features, and output sequence, respectively.

TABLE I
CONFIGURATION OF THE PROPOSED END-TO-END NETWORK. "H", "W", AND "C" INDICATE THE HEIGHT, WIDTH, AND CHANNEL OF THE INPUT SIGNAL. 'I' REPRESENTS THE LENTH OF OUTPUT SEQUENCE

Type	Output size	Configuration
Input	-	$1\times W\times 2$ $(H\times W\times C)$
Convolution 1	$1\times W\times 32$	kernel:$1\times 35\times 32$, stride:1×1
Convolution 2	$1\times W\times 32$	kernel:$1\times 35\times 32$, stride:1×1
Max Pooling 1	$1\times (W/2)\times 32$	kernel:1×2, stride:1×2
Convolution 3	$1\times (W/2)\times 64$	kernel:$1\times 35\times 64$, stride:1×1
Convolution 4	$1\times (W/2)\times 64$	kernel:$1\times 35\times 64$, stride:1×1
Max Pooling 2	$1\times (W/4)\times 64$	kernel:1×2, stride:1×2
BLSTM layer 1	$1\times (W/8)\times 128$	hidden units: 128
BLSTM layer 2	$1\times (W/16)\times 128$	hidden units: 128
BLSTM layer 3	$1\times (W/32)\times 128$	hidden units: 128
LSTM with attention	$1\times I\times 128$	hidden units: 128
Linear layer	$1\times I\times 7$	units: 7

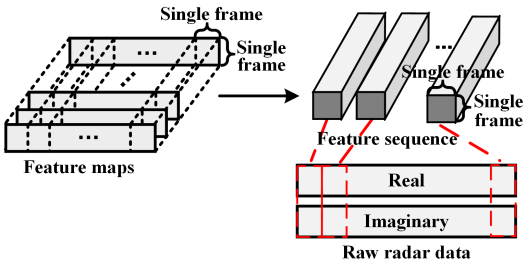


Fig. 5. Process of transforming feature maps into feature sequence, and relation between feature sequence and raw radar data.

in the same experimental scene (4800, 3600, 2400 and 1200 continuous motions with durations of 25s, 20s, 15s and 10s were collected, respectively). Each continuous motion contains several of the above five kinds of single motions, and the duration of each single motion is random. Five examples of spectrogram and raw radar data waveform of continuous motion are shown in Fig. 3. It can be found in Fig. 2 and 3 that spectrograms clearly show the change of micro-Doppler signatures over time and the difference of motions, whereas the waveform shapes of different motions have a high similarity. Therefore, using raw radar data for HMR is a challenge.

III. PROPOSED NETWORK

The overall architecture of the proposed end-to-end network is shown in Fig. 4, and it contains three components, including *fusion layer, encoder* and *attention-based decoder*. The input is the raw radar data whose real and imaginary parts are treated as two channels. The *fusion layer* fuses the raw radar data with any length into a fusion sequence. Then, the *encoder* maps the fusion sequence into high-level features, and the *attention-based decoder* selects useful temporal information from these features to make prediction. Though the proposed network is composed of different kinds of network architectures, it can be jointly trained in an end-to-end manner.

A. Fusion Layer

The fusion layer is used to directly process the raw radar data. Traditional methods mainly use spectrogram generated from raw radar data as input of deep network to perform image-based feature extraction and classification, which is not conducive to extracting temporal information. Moreover, such methods separate the whole process of recognition into two parts, including time-frequency representations (TFR) and image-based classification. Thus, the representation cannot be optimized by errors back-propagated from network, and its high computational complexity and independence from learning model make the network consume significant time. To overcome the above limitations, we use the fusion layer constructed by convolutional and max pooling layers to replace TFR, and incorporate the fusion layer into subsequent network to construct an end-to-end network. This is inspired by the success of end-to-end training in natural language processing and image classification. Compared with TFR, the fusion layer can be optimized by back-propagating errors and the end-to-end manner can greatly reduce the inference time. Therefore, it performs better than representation.

The fusion layer fuses the raw I & Q radar data into a fusion sequence with more dimensional information, and its configuration is shown in Table I. Before being input to the network, the raw radar data is of arbitrary length and does not require any preprocessing. Since the value of raw radar data is complex whereas the neural network can only process the real-value, we treat its real and imaginary parts (i.e., I and Q data of radar output) as two channels, each of which is a one-dimension time sequence. Thus, the input size is $1\times$W$\times 2$ (height$\times$width$\times$channel). Then, the convolutional layers fuse the data of two channels and expand them to more channels to obtain feature maps with multiple dimensional information. Specifically, two convolutional kernels are stacked together

for more nonlinear transformations. A large kernel size of 1×35 is chosen to fuse more related temporal information. Max pooling layers with 1×2 filter size and 1×2 stride are used to reduce the input length, resulting in an output of size 1×(W/4)×64. Excessive use of the max pooling layer would lose a lot of temporal information, and thus the result is poor, which is detailed in Section IV-D. Finally, the feature maps are transformed into the fusion sequence $X = (x_1, x_2, ..., x_N)$, which is the input of the encoder. As shown in Fig. 5, each fusion vector of the fusion sequence is generated from left to right on the feature map by column, and the width of each fusion vector is single frame. This means that each fusion vector can be regarded as a description of a segment of the raw radar data.

B. Encoder

In the encoder-decoder structure, encoder maps the input sequence with variable length to a fixed-size vector and then decoder maps the vector to the target sequence. Our encoder uses a BLSTM with a pyramid structure to reduce the length of fusion sequence (input time steps) and to transform fusion sequence $X = (x_1, x_2, ..., x_N)$ into high-level features $h = (h_1, h_2, ..., h_U)$. LSTM is a kind of RNN which can capture long-range dependencies. Since the information from both directions is useful in raw radar data, we adopt a BLSTM, which can learn the forward and backward information. The length of input raw radar data can be tens thousands of frames long. After fusion, the number of input time steps is still thousands. If BLSTM is directly used to encode the input time steps, the network would converge slowly and produce inferior results. This is presumably because the decoder has a hard time extracting the relevant information from a large number of input time steps [27].

We circumvent this problem by using a pyramid BLSTM, where the outputs are concatenated at successive time steps of each layer before being fed to the next layer, as shown in Fig. 4. In each successive stacked pyramid BLSTM, the number of time steps is reduced by a factor of two. If only pyramid BLSTM is used to reduce the length of input without using max pooling layer, the network would need to stack more layers, which greatly increases network complexity. In a traditional deep BLSTM, at the *t* th time step, the output hidden state h_t^j from the *j* th layer is calculated as follows:

$$h_t^j = BLSTM(h_{t-1}^j, h_{t+1}^j, h_t^{j-1}). \tag{4}$$

Whereas, in the pyramid BLSTM, the output hidden state h_t^j from the *j* th layer at the *t* th time step is calculated as:

$$h_t^j = pBLSTM(h_{t-1}^j, h_{t+1}^j, [h_{2t-1}^{j-1}, h_{2t}^{j-1}]). \tag{5}$$

Especially, when $j = 1$, the output hidden state h_t^1 from the first layer at the *t* th ($t \leqslant N/2$) time step is calculated as $h_t^1 = pBLSTM(h_{t-1}^1, h_{t+1}^1, [x_{2t-1}, x_{2t}])$.

As shown in Table I, in our encoder, we stack three pyramid BLSTMs with 128 hidden units to reduce the input time steps by $2^3 = 8$ times. Additionally, the encoder can learn nonlinear feature representations from input time steps to obtain high-level features (encoder hidden states) *h*. Subsequently, the hidden state h_U at the last time step in the last layer, and the high-level features are passed to the attention-based decoder as input.

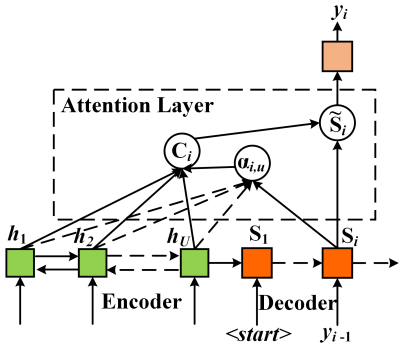


Fig. 6. The structure of attention mechanism, which selects useful information from encoder hidden states and propagates it to decoder for decoding. $\boldsymbol{h_u}$, $\boldsymbol{S_i}$, $\boldsymbol{\alpha_{i,u}}$, $\boldsymbol{C_i}$, $\boldsymbol{\widetilde{S}_i}$, and $\boldsymbol{y_i}$ represent encoder hidden state, decoder hidden state, attention vector, context vector, attentional hidden state, and output, respectively.

C. Attention-based Decoder

The decoder consists of attention-based LSTM, and the goal is to compute output sequence $Y = (\langle start \rangle, y_1, ..., y_I, \langle end \rangle)$ by maximizing the conditional probability $P(Y|X)$, where *X* is the fusion sequence produced by fusion layer, and $\langle start \rangle$ and $\langle end \rangle$ are the special start-of-motion token and end-of-motion token. In the traditional encoder-decoder structure, the encoder encodes all input time steps and then transmits the hidden state at the last time step to the decoder for decoding, whereas the transmitted hidden state contains too much useless temporal information. Similarly, the methods that use individual RNN or sliding window for continuous HMR also cannot effectively use temporal information. We solve this problem by introducing attention mechanism, which selects useful information from the hidden states of encoder and propagates it to decoder for decoding [28].

Specifically, the structure of attention mechanism is shown in Fig. 6. At each decoder time step *i*, the decoder hidden state S_i matches with each encoder hidden state h_u to obtain the score, which can be considered as a content-based function of $score(S_i, h_u) = S_i^\top h_u$. Among them, the decoder hidden state S_i is computed by LSTM based on the previous hidden state S_{i-1} and the input (previous output of decoder) y_{i-1}, i.e., $S_i = LSTM(S_{i-1}, y_{i-1})$. Then, the score is transformed into the attention vector $\alpha_{i,u}$ using a softmax function:

$$\alpha_{i,u} = \frac{\exp(score(S_i, h_u))}{\sum_u \exp(score(S_i, h_u))}. \tag{6}$$

It is used to compute the context vector C_i by linearly blending the encoder hidden state h_u, at different time steps:

$$C_i = \sum_{u=1}^{U} \alpha_{i,u} h_u. \tag{7}$$

The core of the attention mechanism is to learn the attention vector, and the purpose is to associate those parts of the sequence that contain more information with stronger

weights and vice versa. Subsequently, the decoder hidden state S_i and context vector C_i are combined by a simple concatenation layer to produce an attentional hidden state $\widetilde{S}_i = \tanh(W_c[C_i; S_i])$, where W_c is the weight matrix. The attentional hidden state $\widetilde{S}_i$ is then fed through a linear layer with softmax to produce the predictive distribution, the conditional distribution of output y_i over previous predictions $y_1, ..., y_{i-1}$ and input sequence X:

$$P(y_i|y_1, ..., y_{i-1}, X) = softmax(W_s\widetilde{S}_i + b_s), \quad (8)$$

where W_s is the weight matrix, and b_s is the bias. Finally, the probability distribution of output sequence is computed as:

$$P(Y|X) = \prod_{i=1}^{I} P(y_i|y_1, ..., y_{i-1}, X). \quad (9)$$

Considering the network complexity, we use a one layer LSTM with 128 hidden units in our decoder, as shown in Table I. After decoding, the network produces the prediction of continuous motion.

D. Training and Decoding

We want to optimize the network parameter by end-to-end training. Given the input sequence (i.e., raw radar data) $S = (s_1, s_2, \ldots, s_M)$ with the ground truth $Y^* = (\langle start\rangle, y_1^*, y_2^*, \ldots, y_I^*, \langle end\rangle)$ and the corresponding output sequence $Y = (\langle start\rangle, y_1, y_2, \ldots, y_I, \langle end\rangle)$, the sequence loss is calculated as follows:

$$E(\theta) = -\sum_{i=1}^{I} y_i^* \log P(y_i|y_i^*, ..., y_{i-1}^*, X; \theta), \quad (10)$$

where θ is the parameters to be trained and X is the fusion sequence produced by fusion layer from S. During training, the ground truth is always fed into the decoder for next step prediction. The network is trained to minimize the sequence loss over L training sequences in the training dataset $O = \{(S_1, Y_1^*), \ldots, (S_l, Y_l^*), \ldots, (S_L, Y_L^*)\}$:

$$Loss(\theta) = \frac{1}{L} \sum_{S_l, Y_l^* \in O} E_l(\theta), \quad (11)$$

where E_l is given in Equation (10). In the proposed network, the fusion layer, encoder, and attention-based decoder can be trained jointly on pairs of raw radar data and ground truth, namely end-to-end training.

During inference, for each input sequence X, we want to find the output sequence $\widehat{Y}$ that has the highest conditional probability defined in Equation (9):

$$\widehat{Y} = \arg\max_{Y} \log P(Y|X). \quad (12)$$

To do so, we use a simple left-to-right beam search algorithm [28]. It starts with the start-of-motion token $\langle start\rangle$. At each time step, only the B most likely hypotheses are kept according to the probability predicted by the network. When the end-of-motion token $\langle end\rangle$ is appended to the hypothesis, the whole algorithm is completed. In order to keep a balance between network accuracy and computational complexity, the beam size is set to 5 (i.e., B=5) as the network performance is optimal at this time.

TABLE II
CONFUSION MATRIX OF SINGLE MOTION TEST SET

True \ Pred	Running backward	Running forward	One arm swinging	Squatting	Two arms swinging
Running backward	**98.81%**	0	0	0	1.19%
Running forward	0	**96.43%**	0	1.19%	2.38%
One arm swinging	0	1.79%	**93.45%**	0	4.76%
Squatting	0	0	1.79%	**97.02%**	1.19%
Two arms swinging	0	2.98%	6.55%	0	**90.47%**

TABLE III
COMPARISON RESULTS OF DIFFERENT METHODS FOR SINGLE MOTION RECOGNITION

Model name	Average accuracy in LOSOCV scheme	Inference time / sample	Training time
STFT-DCNN [13]	88.53%	10.45 ms	**2256 s**
STFT-TResNet [18]	89.98%	10.66 ms	4865 s
STFT-IRNet [29]	87.73%	15.01 ms	23631 s
ID-1D-CNN [30]	92.87%	**0.18 ms**	9495 s
SGRUN [22]	91.58%	10.48 ms	14872 s
CRNN-CTC [20]	92.34%	11.56 ms	13652 s
Ours	**93.13%**	**1.21 ms**	**17546 s**

IV. Experimental Results

In section IV-A and IV-B, we evaluate the proposed network and other existing methods on single HMR and continuous HMR, while ablation studies and parameters discussion are conducted in section IV-C and IV-D. The proposed network was trained by adaptive moment estimation (Adam) optimizer with the batch size set to 64 and the learning rate set to 0.0001. We implemented and trained all networks with the open source toolkit Tensorflow v1.12. All experiments were performed on a workstation with NVIDIA Tesla K40c GPU (with a 12 GB memory) and 2.5 GHz Intel Xeon CPU E5-2650 v3.

A. Single Human Motion Recognition

For single motion samples, 80% of them were used as single motion training set and the rest were considered as single motion test set. We tested the proposed network using the single motion test set, which results in a high recognition accuracy of 95.23%. As shown in Table II, 'running backward' and 'squatting' achieve high accuracy of 98.81% and 97.02%, respectively. On the other hand, 'two arms swinging' has the lowest accuracy of 90.47%, as 6.55% and 2.98% of them are misrecognized as 'one arm swinging' and 'running forward'. This is because the waveform amplitude and frequency generated by these three motions are similar.

To further demonstrate the superiority of the proposed network in single motion recognition, we compared our network with other recent studies. Moreover, we introduced leave one subject out cross-validation (LOSOCV) scheme to compute the average accuracy of all methods, thereby evaluating their generalization performance over the unseen subjects. In this scheme, each subject's single motion data was used for testing once, while the rest data of the other subjects was used for

TABLE IV
CONFUSION MATRIX OF CONTINUOUS MOTION TEST SET

True \ Pred	Running backward	Running forward	One arm swinging	Squatting	Two arms swinging
Running backward	**95.96%**	0.88%	1.39%	0	1.77%
Running forward	0.95%	**90.28%**	1.04%	1.42%	6.31%
One arm swinging	0.88%	0	**94.69%**	0	4.42%
Squatting	1.42%	0	2.94%	**95.64%**	0
Two arms swinging	0	0.63%	2.24%	0	**97.13%**

TABLE V
COMPARISON RESULTS OF DIFFERENT METHODS FOR CONTINUOUS MOTION RECOGNITION

Model name	Average accuracy in LOSOCV scheme	Inference time / sample	Training time
SGRUN [22]	83.92%	1398.66 ms	**14872 s**
CRNN-CTC [20]	90.13%	35.47 ms	18631 s
Ours	**92.67%**	**2.31 ms**	**32756 s**

training. In [13], a three-layer CNN was proposed to classify human activities based on spectrogram, named as STFT-DCNN. The transferred ResNet-18 (STFT-TResNet) [18] and the Inception-ResNet (STFT-IRNet) [29] were also adopted for spectrogram-based HMR. In [30], a low-complexity network named ID-1D-CNN was proposed, which uses the raw radar data as input to perform HMR. As for CRNN-CTC, its architecture was derived from [20] (CTC is a loss function, avoiding the labor of labeling each frame of input), and we changed 3D-CNN to 2D-CNN to process CW radar data. The SGRUN [22] was also re-implemented for comparison.

The comparison results are shown in Table III. Among them, the inference time refers to the sum of the time of TFR and the time of network forward propagation, and the average TFR time of single motion is 9.51 ms. STFT-IRNet performs the worst in terms of inference time and training time for its high network complexity. As for the training time of the proposed network, it is in the same order of magnitude as other methods. Moreover, since TFR is not used, the inference time of ID-1D-CNN and ours is significantly shorter than others, and ID-1D-CNN has the shortest inference time. For average accuracy, STFT-IRNet has the lowest accuracy, whereas the proposed network achieves the highest accuracy of 93.13%, which is 0.26% to 5.4% higher than that of other methods. This verifies that the recognition ability and generalization ability of the proposed network are better than existing works. Overall, in single HMR, the proposed network achieves the highest accuracy, and its inference time is also at a low level.

B. Continuous Human Motion Recognition

In order to further verify the effectiveness of the proposed network, we performed a more complex experiment to recognize continuous human motion. In experiment, 80% of continuous motion samples were used for training and the rest samples were used as continuous motion test set. We used such test set to test the proposed network, and a high accuracy of

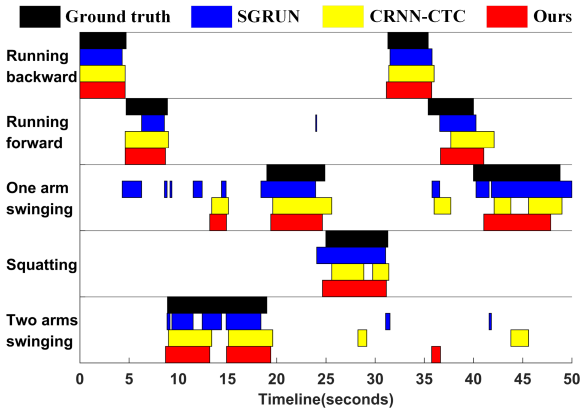


Fig. 7. Example timeline of continuous motion recognition for different methods.

94.74% was achieved. Table IV shows the confusion matrix of the proposed network. It can be seen that 'running forward' obtains the lowest accuracy of 90.28%, and 6.31% of them are misrecognized as 'two arms swinging'. This is because when people run forward, their arms also swing. The accuracy of 'two arms swinging' is the highest, which is 97.13%.

To show the superiority of the proposed network in continuous HMR, SGRUN and CRNN-CTC were constructed for comparison. LOSOCV scheme was also used to evaluate the generalization performance of them. SGRUN was trained by the single motion samples, whereas ours and CRNN-CTC were trained by the continuous motion samples, which is determined by their architecture. The comparison results are shown in Table V, and the average TFR time of continuous motion is 28.46 ms. Compared with other methods, the proposed network has a relatively long training time, but the training work mainly performs offline and can be improved by using high-performance servers. In addition, the inference time of SGRUN is the longest, lasting for 1398.66 ms, and the average accuracy of SGRUN is only 83.92%. This is because every time the window function slides one frame, the method needs to re-recognize the motion, which significantly increases the inference time. The average accuracy and the inference time of CRNN-CTC are 90.13% and 35.47 ms, respectively. This suggests that in addition to the use of sliding window, the use of TFR is also an important factor affecting inference time. This is because TFR has high complexity and it is independent of subsequent network. In contrast, the proposed network achieves the highest accuracy of 92.67% and the shortest inference time of 2.31 ms, demonstrating the superiority of the proposed network in terms of inference time, and generalization and recognition abilities. Therefore, in continuous HMR, the proposed network outperforms existing works.

We used a 49 s continuous motion sample to further test the recognition effect of continuous motion that lasts longer than training samples. The example timeline of the proposed network and compared methods for continuous motion recognition is presented in Fig. 7, where the recognition results of them are recorded along the time axis and various colors represent different methods and the ground truth. It can be seen that SGRUN is often confused at the time frame of motion

TABLE VI
RESULTS OF ABLATION STUDIES ON CONTINUOUS HMR. EACH COMPONENT OF THE PROPOSED NETWORK IS REPLACED BY 2D-CNN, BLSTM-CTC AND ED

Model name	Average accuracy in LOSOCV scheme	Inference time / sample
2D-CNN-AttentionED	91.12%	42.38 ms
1D-CNN-BLSTM-CTC	90.98%	10.90 ms
1D-CNN-ED	77.34%	**1.92 ms**
Ours	**92.67%**	**2.31 ms**

transition and its false recognition rate is high, showing that the sliding window-based method is difficult to distinguish the motions during motion transition. Moreover, the false recognition rate of CRNN-CTC is also high, especially for 'one arm swinging' and 'running forward'. This suggests that the use of representation is not conducive to recognizing long-lasting continuous motion. In contrast, the proposed network can accurately recognize all motions and it has more stable performance for the motion with long duration.

C. Ablation Studies

To verify the effectiveness of each component of the proposed network, we introduce ablation studies to analyze its individual components. Firstly, we substituted 1D-CNN in fusion layer with the traditional 2D-CNN to construct 2D-CNN-AttentionED, which uses the spectrogram generated by STFT as input instead of raw radar data. Then, to explore the utility of attention-based encoder-decoder, we replaced it by BLSTM-CTC (since traditional BLSTM cannot reduce input length, more 1D-CNN layers were used), and the corresponding network was named as 1D-CNN-BLSTM-CTC. Moreover, a network named 1D-CNN-ED was constructed, where attention mechanism is removed, to evaluate the importance of attention. The comparison results of these networks are listed in Table VI. Since the difference between these networks is not obvious in single HMR, we only compare them in continuous HMR. It can be observed that the proposed network outperforms 2D-CNN-AttentionED in terms of average accuracy and inference time, demonstrating using raw radar data is superior than using spectrogram. Moreover, the proposed network has better performance comparing 1D-CNN-BLSTM-CTC. This suggests that attention-based encoder-decoder structure also plays an important role in our network. When the attention mechanism is removed, the average accuracy is reduced significantly. This is presumably because the network without attention mechanism is difficult to converge, resulting in inferior performance. The results demonstrate that the components in our proposed network including the fusion layer, encoder-decoder and attention mechanism, are all imperative in helping improve recognition accuracy and reduce inference time.

D. Parameters Discussion

To explore the impact of parameters on recognition accuracy, we test the proposed network with four different parameters, including the number of convolutional layers,

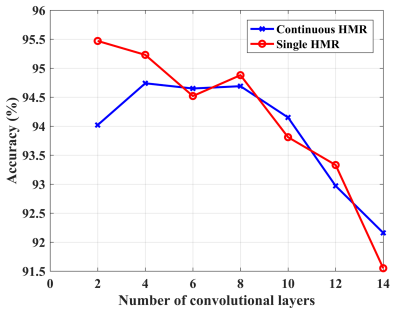


Fig. 8. Accuracy of the proposed network with different numbers of convolutional layers.

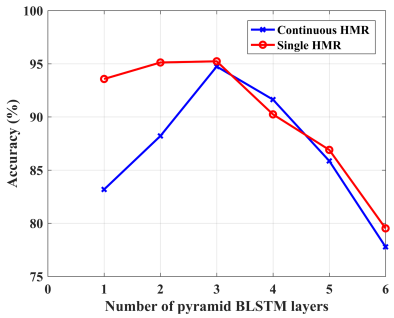


Fig. 9. Accuracy of the proposed network with different numbers of pyramid BLSTM layers.

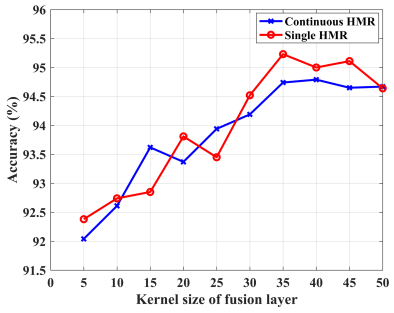


Fig. 10. Accuracy of the proposed network varying with the kernel size of fusion layer.

number of pyramid BLSTM layers, kernel size of fusion layer, and length of input signal. As shown in Fig. 8, as the number of convolutional layers increases from 1, the accuracy of single HMR gradually decreases. Moreover, when the number of convolutional layers is 4, the accuracy of continuous HMR has a maximum value. The convolutional layer is mainly used to fuse raw radar data instead of extracting temporal features. As the number of convolutional layers increases, the number of max pooling layers following convolution layer also increases, and thus more temporal information is lost, which is not conducive to the encoder to extract features.

The accuracy of the proposed network with different numbers of pyramid BLSTM layers is shown in Fig. 9. We observe that the accuracy of single HMR and continuous HMR achieves a maximum when the number of pyramid BLSTM layers is 3, and as the number continues to increase, the accuracy gradually declines. Although the pyramid BLSTM improves network performance by stacking BLSTM layers to reduce input length, excessively stacking loses temporal

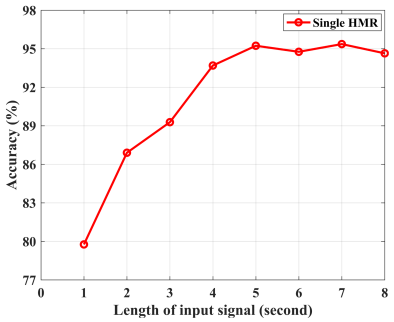


Fig. 11. Accuracy of the proposed network for single HMR varying with the length of input signal.

information and increases network complexity, resulting in inferior performance.

The kernel size of fusion layer has a similar effect as the window in the STFT that affects the recognition performance. As shown in Fig. 10, the accuracy of single HMR and continuous HMR gradually increases as the kernel size increases from 5. When the kernel size is 35, the accuracy of single HMR achieves the highest value. Similarly, the accuracy of continuous HMR reaches maximum when the kernel size is 40, but it is not much different from the kernel size of 35. Increasing the kernel size of fusion layer can promote the improvement of accuracy, while the excessive increase has limited improvement in network performance.

The accuracy of single HMR varying with the length of input signal is shown in Fig. 11. As the length of input signal increases from 1, the accuracy gradually increases until the length is 5, and then the accuracy becomes stable. As the length of input signal increases, more temporal information can be used for recognition, but excessive information may not help recognition. Therefore, the collection time for single motion is set to 5s, and that for continuous motion is determined by the number of single motions contained in a continuous motion.

V. Conclusion

In this paper, we propose a new end-to-end network using radar radios to recognize continuous human motion without the need of representation. The fusion layer is designed to fuse the raw radar data, avoiding the limitation of using representation. Moreover, the attention-based encoder-decoder structure is able to effectively use all the temporal information in raw radar data to perform recognition. The experimental results show that compared with existing works, the proposed end-to-end network achieves higher accuracy and shorter inference time. Meanwhile, it has better generalization. However, the encoder-decoder structure requires a large number of training samples, which is a challenge for continuous motion collection and labeling. In the future, we will develop an end-to-end network based on few-shot learning for micro-Doppler-based HMR.

References

[1] M. G. Amin, Y. D. Zhang, F. Ahmad, and K. C. D. Ho, "Radar signal processing for elderly fall detection: The future for in-home monitoring," *IEEE Signal Process. Mag.*, vol. 33, no. 2, pp. 71–80, Mar. 2016.

[2] Y. Lang, Q. Wang, Y. Yang, C. Hou, H. Liu, and Y. He, "Joint motion classification and person identification via multitask learning for smart homes," *IEEE Internet Things J.*, vol. 6, no. 6, pp. 9596–9605, Dec. 2019.

[3] S. C. Mukhopadhyay, "Wearable Sensors for Human Activity Monitoring: A Review," *IEEE Sensors J.*, vol. 15, no. 3, pp. 1321–1330, Mar. 2015.

[4] J. Liu, H. Liu, Y. Chen, Y. Wang, and C. Wang, "Wireless sensing for human activity: A survey," *IEEE Commun. Surv. Tutor.*, vol. 22, no. 3, pp. 1629–1645, 2020.

[5] W. Jiang *et al.*, "Towards environment independent device free human activity recognition," in *Proc. Annu. Int. Conf. Mobile Comput. Networking (ACM MobiCom)*, Oct. 2018, pp. 289–304.

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

[7] J. L. Kernec, F. Fioranelli, S. Yang, J. Lorandel, and O. Romain, "Radar for assisted living in the context of Internet of Things for Health and beyond," in *Proc. IFIP/IEEE Int. Conf. Very Large Scale Integration (VLSI-SOC)*, Oct. 2018, pp. 163–167.

[8] Y. Kim and H. Ling, "Human activity classification based on micro-Doppler signatures using a support vector machine," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 5, pp. 1328–1337, May 2009.

[9] L. Du, L. Li, B. Wang, and J. Xiao, "Micro-Doppler feature extraction based on time-frequency spectrogram for ground moving targets classification with low-resolution radar," *IEEE Sensors J.*, vol. 16, no. 10, pp. 3756–3763, May 2016.

[10] R. M. Narayanan and M. Zenaldin, "Radar micro-Doppler signatures of various human activities," *IET Radar, Sonar Navigat.*, vol. 9, no. 9, pp. 1205–1215, Dec. 2015.

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

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

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

[14] B. Jokanović and M. Amin, "Fall Detection Using Deep Learning in Range-Doppler Radars," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 54, no. 1, pp. 180–189, Feb. 2018.

[15] Y. Yang, C. Hou, Y. Lang, T. Sakamoto, Y. He, and W. Xiang, "Omnidirectional Motion Classification With Monostatic Radar System Using Micro-Doppler Signatures," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 5, pp. 3574–3587, May 2020.

[16] Z. Chen, G. Li, F. Fioranelli, and H. Griffiths, "Personnel Recognition and Gait Classification Based on Multistatic Micro-Doppler Signatures Using Deep Convolutional Neural Networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 5, pp. 669–673, May 2018.

[17] B. Erol, S. Z. Gurbuz, and M. G. Amin, "Motion Classification Using Kinematically Sifted ACGAN-Synthesized Radar Micro-Doppler Signatures," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 56, no. 4, pp. 3197–3213, Aug. 2020.

[18] H. Du, Y. He, and T. Jin, "Transfer learning for human activities classification using micro-Doppler spectrograms," in *Proc. IEEE Int. Conf. Comput. Electromagn. (ICCEM)*, Mar. 2014, pp. 1–3.

[19] C. Ding *et al.*, "Continuous human motion recognition with a dynamic range-Doppler trajectory method based on FMCW radar," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 9, pp. 6821–6831, Sep. 2019.

[20] Z. Zhang, Z. Tian, and M. Zhou, "Latern: Dynamic continuous hand gesture recognition using FMCW radar sensor," *IEEE Sensors J.*, vol. 18, no. 8, pp. 3278–3289, Apr. 2018.

[21] H. Du, T. Jin, Y. He, Y. Song, and Y. Dai, "Segmented convolutional gated recurrent neural networks for human activity recognition in ultra-wideband radar," *Neurocomputing*, vol. 396, no. 5, pp. 451–464, Jul. 2020.

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

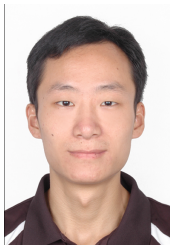
[23] A. Shrestha, H. Li, J. L. Kernec, and F. Fioranelli, "Continuous human activity classification from FMCW radar with Bi-LSTM networks," *IEEE Sensors J.*, to be published.

[24] 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 J.*, vol. 20, no. 3, pp. 1191-1201, Feb. 2020.

[25] J. Le Kernec *et al.*, "Radar signal processing for sensing in assisted living: The challenges associated with real-time implementation of emerging algorithms," *IEEE Signal Process. Mag.*, vol. 36, no. 4, pp. 29–41, Jul. 2019.
[26] V. C. Chen, F. Li, S. -S. Ho, and H. Wechsler, "Micro-Doppler effect in radar: phenomenon, model, and simulation study," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 42, no. 1, pp. 2–21, Jan. 2006.
[27] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP),* Mar. 2016, pp. 4960–4964.
[28] T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," in *Proc. Conf. Empir. Methods Nat. Lang. Process. (EMNLP),* Sep. 2015, pp. 1412-–1421.
[29] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Proc. AAAI Conf. Artif. Intell.,* Feb. 2017, pp. 4278—4284.
[30] H. Chen and W. Ye, "Classification of human activity based on radar signal using 1-D convolutional neural network," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 7, pp. 1178–1182, Jul. 2020.

Running Zhao received the B.E. degree in information engineering from Wuhan University of Technology, Wuhan, China, in 2018, where he is currently pursuing the M.S. degree in information and communication engineering.

His research interests include wireless sensing systems, signal processing, and deep learning.



Xiaolin Ma received the B.S. degree in electrical and information engineering, and the M.S. and PhD degrees in information and communications engineering from Wuhan University of Technology, Wuhan, China, in 2008, 2011, and 2014, respectively. From 2011 to 2013, he was as a visiting PhD student in the Department of Electrical and Computer Engineering at Dalhousie University, Halifax, Canada.

He is currently an Associate Professor in the Department of Information Engineering, School of Information Engineering, Wuhan University of Technology. He has authored or co-authored more than 30 journal and conference papers, and holds one Chinese patent. His current research interests include wireless sensing systems, wireless communications and networks, and machine learning.

Xinhua Liu received the B.S. degree in computer science from the University of South China, Hengyang, China, in 1997, and the M.S. and Ph.D. degrees in information and communications engineering from the Wuhan University of Technology, Wuhan, China, in 2006 and 2010, respectively.

He is currently a Professor with the Department of Communications Engineering, School of Information Engineering, Wuhan University of Technology. His current research interests include wireless communications and networks, and wireless sensor and actuator networks.

Jian Liu received the B.E. and M.S. degrees from the Department of Information Engineering, Wuhan University of Technology, Wuhan, China, and the Ph.D. degree in Wireless Information Network Laboratory (WINLAB) at Rutgers University, New Brunswick, NJ, USA.

He is currently an Assistant Professor in the Department of Electrical Engineering and Computer Science at the University of Tennessee, Knoxville. He is the recipient of the Best Paper Awards from IEEE SECON 2017 and IEEE CNS 2018. He also received Best-in-session Presentation Award from IEEE INFOCOM 2017, and two Best Poster Award Runner-up from ACM MobiCom 2016 and ACM MobiCom 2018. His current research interests include mobile sensing and computing, cybersecurity and privacy, intelligent systems and smart healthcare.