IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 22, 2019, accepted June 9, 2019, date of publication June 17, 2019, date of current version July 3, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2923574

Accurate Human Gesture Sensing With Coarse-Grained RF Signatures

HONGYU SUN[1,2], ZHENG LU[1,2], CHIN-LING CHEN[3,4,5], JIE CAO[6], AND ZHENJIANG TAN[1,2]
[1]Department of Computer Science, Jilin Normal University, China
[2]State Key Laboratory of Numerical Simulation
[3]Department of Computer Science and Information Engineering, Chaoyang University of Technology, Taiwan
[4]School of Information Engineering, Changchun Sci-Tech University, China
[5]School of Computer and Information Engineering, Xiamen University of Technology, China
[6]School of Computer Science, Northeast Electric Power University, Jilin, China

Corresponding authors: Chin-Ling Chen (clc@mail.cyut.edu.tw) and Zhenjiang Tan (tanzj@jlnu.edu.cn)

ABSTRACT RF-based gesture sensing and recognition has increasingly attracted intense academic and industrial interest due to its various device-free applications in daily life, such as elder monitoring, mobile games. State-of-the-art approaches achieved accurate gesture sensing by using fine-grained RF signatures (such as CSI, Doppler effect) while could not achieve the same accuracy with coarse-grained RF signatures such as received signal strength (RSS). This paper presents *rRuler*, a novel feature extraction method which aims to get fine-grained human gesture features with coarse-grained RSS readings, which means rought ruler could measure fine things. In order to further verify the performance of *rRuler*, we further propose *rRuler-HMM*, a hidden Markov model (HMM) based human gesture sensing and prediction algorithm which utilizes the features extracted by *rRuler* as input. We implemented *rRuler* and *rRuler-HMM* using TI Sensortag platforms and off-the-shelf (CTOS) laptops in an indoor environment, extensively performance evaluations show that *rRuler* and *rRuler-HMM* stand out for their low cost and high practicability, and the average gesture sensing accuracy of *rRuler-HMM* can achieve 95.71% in NLoS scenario and 97.14% in LoS scenario, respectively, which is similar to the performance that fine-grained RF signatures based approaches could achieve.

INDEX TERMS Gesture sensing, coarse-grained RF signatures.

I. INTRODUCTION

RF signature based human gesture sensing and prediction is the core technology that enables a wide variety of device-free applications such as fitness tracking, elders monitoring, smart homes and Human-Computer Interactions (HCI). Most RF signature-based human gesture sensing solutions utilize fine-grained RF signatures such as Doppler shifts [1]–[4] channel state information (CSI) [5]–[21] to achieve accurate human gesture sensing and recognition. However, these solutions require specific hardwares, customized modifications or multiple antenna systems such as WiFi technologies to fetch fine-grained RF signatures. For example, most of the current solutions use WiFi NIC such Intel 5300 [22] and Atheros Serious NICs (AR9580,AR9590 and etc..) [23] to fetch fine-grained CSI. Therefore, the adoption of the fine-grained RF signature-based approaches are limited with wireless devices and technologies.

In order to bring the human gesture sensing applications to all kinds of Internet of Things (IoT) devices and technologies such as ZigBee, BLE, LoRa, NB-IoT and etc., researchers propose to use coarse-grained received signal strength (RSS) to predict the human gestures [24]–[27]. It is possible to widely used since RSS is pervasively available in all kinds of wireless radios such as ZigBee, Bluetooth and WiFi. However, the sensing and prediction accuracy is relatively low with coarse-grained RSS, the comparisons of sensing accuracy of state-of-the-art works both use fine-grained and coarse-grained RF measurements are shown in Table 1. In order further improve the accuracy of the human sensing with coarse-grained RSS, this paper takes the first attempt to explore the feasibilities to achieve accurate (with the accuracy greater than 90%) human gesture sensing with coarse-grained RSS which extracted from the off-the-shelf Sensortag with ZigBee protocol, and our approach is easily to transplant in

The associate editor coordinating the review of this manuscript and approving it for publication was Mohammad Anwar Hossain.

TABLE 1. Sensing accuracy of state-of-the-art works.

Signal Features		Gesture Feature					
		Coarse-grained			Fine-Grained		
		reference	Accuracy		reference	Accuracy	
			LoS	NLoS		LoS	NLoS
Coarse Grained	RSS(WiFi/ZigBee/BLE)	[24], [25], [26]	88%	74%	–	–	–
	Droppler shift	[28], [1]	94%	–	[29]	91%	88%
Fine-Grained	CSI(WiFi)	[5], [6], [7], [8], [9], [10], [11] [12], [13], [14], [15], [16], [18], [19]	97.5%	90%	[30], [31], [32], [33], [34], [35], [17], [20]	98.5%	90%

other communication systems with WiFi, BLE, LoRa and NB-IoT protocols.

In order to achieve accurate human gesture sensing with coarse-grained RSS, we compared RSS with CSI to investigate potential solutions to increase the sensing accuracy with coarse grained RSS which pervasively available in all kinds of wireless radios. The main differences between them relies on four aspects: Compared with CSI i) the sampling rate of RSS is relatively lower (the sampling rate of CSI is 2500Hz, and the sampling rate of RSS varies from 100Hz to 2500Hz due to different IoT devices); ii) The resolution of RSS is extremely low (i.e., the resolution of RSS is $1dB$, while the resolution of CSI is calculated in mW unit with 64 bit or 128 bit due to different NICs); iii) the number of RSS samples with human gestures collected from IoT devices are single dimension data, while the CSI samples are multi-dimensions samples (e.g., AR9590 [23] has 3 antennas and each antenna has 56 groups of sub-carriers with 20MHz bandwidth. Therefore, its channel state information contains 56 matrices with 3×3 dimensions and each element in the matrix has 128 bits); and iv) the multi-path fading details carried by RSS and CSI are different, RSS only could reflect the comprehensive multi-path fading while CSI could deduce the details fading information from each signal propagation path.

To narrow down the differences between coarse-grained RSS and fine-grained CSI, we propose *rRuler*, a fine-grained feature extraction method which can utilize coarse-grained RSS to fetch fine-grained human gesture characteristics. Specifically, we use sampling rate enhancement method (*Sample-H*), resolution enhancement method (*Resolution-H*) and gesture duration extraction method (*gTime-ext*) respectively to narrow down the differences in sampling rate, resolution and detailed multi-path components between RSS and CSI. We also collect more samplings to increase the size of training sets for further improving the fine-grained gesture extraction accuracy. In order to verify the performance of *rRuler* and bring it to practical applications such as fitness tracking, health monitoring, we further propose *rRuler-HMM* to evaluate the human gesture sensing and prediction accuracy with coarse-grained RF signatures. In summary, the contributions of paper are as follows:

- To the best of our knowledge, this is the first work that explores how to narrow down the differences between coarse-grained RSS and fine-grained Channel State Information (CSI) to extract fine-grained human gesture features and achieve accurate human gesture sensing and predictions.
- We have proposed and implemented *rRuler*, a fine-grained human gesture feature extraction method by narrowing down the differences between RSS and CSI which contains five components: i) sampling rate enhancement module *Sample-H* (Detailed in Section IV-B), ii) resolution enhancement module *Resolution-H* (Detailed in Section IV-C), iii) gesture duration extraction module *gTime-ext* (Detailed in Section IV-D), iv) time-frequency analysis module (Detailed in Section IV-E) and v) k-means based feature dimensionality reduction module (Detailed in Section IV-F).
- We further proposed and implemented *rRuler-HMM*, a *rRuler* based human gesture sensing and prediction method, we analyzed how to mapping the features from *rRuler* to the parameters of Hidden Markov Models, how to divide the training sets and prediction set to optimize the parameters training and gesture sensing procedures (Detailed in Section V).
- We have conducted extensive real-world experiments with Sensortags in 802.15.4 mode, 7 different types of human gestures have been collected and sensed in a indoor environment. The average accuracy is approximately 95% for a signal stream of RSS readings in none-line-of-sight (NLoS) path. The RSS readings of each type of gestures are collected from 10 different persons which also verified that our system *rRuler-HMM* is robust to the same type of gestures with different amplitudes.
- We have also extensively revealed the insights by analyzing the similarities (Hamming distances) of the features among the same and different types of gestures, and also evaluated the factors (e.g. different up-sampling rate, different, different number of k-means clusters) that can potentially impact the human sensing accuracy. We also find that different channels does not have obvious impact to the human gesture sensing accuracy while the human gestures occurred in non-line-of-sight (NLoS) path do not have obvious impact to data packet receptions among different communication devices.

II. RELATED WORKS

Existing work on RF-based device-free human gesture sensing approaches could be divided into four categories:

Fine-grained RF signatures based, coarse-grained RF signature-based, Radar-based and acoustic-based approaches.

A. FINE-GRAINED RF-SIGNATURE BASED APPROACHES

Fine-grained RF signatures used to sense human gestures include doppler shifts, phase shifts and amplitude shifts of the received signals. WiSee captures WiFi OFDM signals and measures the doppler shift of the signals which reflected by different human gestures to distinguish a set of nine gestures with an accuracy of 95% [1]. AllSee uses a specially designed analog circuit to extract the amplitude shift of received signals to distinguish eight different gestures with an accuracy of 97% [7].

In practical WiFi networks, phase shift and amplitude shift of OFDM signals are measured by Channel State information (CSI). Currently, CSI values are exposed in some special commercial devices such as Intel 5300 [22] and Atheros serious network interface cards (NICs) [23]. Thus, CSI have been used for human gesture sensing [6], [8], [9], [16], [17], [20], [29], [31], [32], [35]–[39] and etc., Zhou et al. proposed to use CSI to detect the presence of a person in an environment [35]. WiFall detects a single human gesture (*falling*) [6]. E-eye senses a set of nine human gestures using CSI [9]. WiKey and Wigest proposed to sense fine-grained gestures such as keystroke or lip movement using CSI [29], [32]. CARM modeled the relation ship between CSI and the speeds and gestures when human moving [8]. QGesture measures the distance and directions of the gestures using CSI [37]. DeepBreath using FMCW radio to separate different persons' breathing when they are close to each other [20]. Duet estimates users' positions and identities with incomplete RF-data in smart homes [40], CrossSense proposes a novel methods for scaling up the RF-based sensing systems to new environment without re-trainings [16].

However, all these work are under an assumption that we can record the fine-grained RF signatures for gesture sensing. But in practical networks, the fine-grained Doppler shift and channel state information (CSI) do not expose to users, we need special hardwares or specific modifications of the hardware drivers. This paper propose *rRuler* and *rRuler-HMM* to sense the human gestures by coarse-grained RSS which could record by off-the-shelf IoT devices, we evaluate the performance and effectiveness of our approach with ZigBee protocol in Sensortags.

B. COARSE-GRAINED RF-SIGNATURE BASED APPROACHES

RSS is one of the most important coarse-grained RF-signatures which could be monitored easily and extensively in the propagation environment. Currently, the works uses RSS to sense the human gestures and activities includes [24], [25], [41], [42] and [27]. The sensing accuracy of DFAR [25] is range from 50% to 80% with different classification algorithm. [24] could achieve a sensing accuracy of 56% over 7 different gestures. Sigg et al. use software radio to improve the granularity of RSSI values and consequently improve the accuracy of gesture sensing and prediction to 72% for 4 gestures [42]. Wigest [41] and Harmony [26] achieves an accuracy of 87.5% and 88% by defining the gesture family in advance. Aryokee uses Convolutional Neural Networks (CNN) to extract different sources of gestures in the same environment [20], EAR [27] uses uncontrollable ambient RF signals in heterogeneous IoT devices to sensing different daily activities.

However, the state-of-the-art works which use coarse-grained RSS could not achieve accurate human gesture sensing with the sets contains multiple gestures. *rRuler* proposed in this paper aims to extract fine-grained features with coarse-grained RSS; and *rRuler-HMM* predict the gestures by utilizing the features extracted from *rRuler* which could achieve an average sensing accuracy of 95.7%.

C. RADAR-BASED APPROACHES

Radar technology is also used to recognize the human gestures [2], [20], [43]–[46]. WiTrack uses specially designed Frequency Modulated Carrier Wave (FMCW) signals to track human movements behind the wall with a resolution of approximately 20cm [43]. WiTrack2.0 could recognize the presence of 5 persons in the indoor environment with the accuracy of 11.7cm [44]. [47] uses backscatter to estimate the position of a person in smart homes. [46] uses radar signals for sleeping monitoring. Compared to the specially designed radar signals, *rRuler* and *rRuler-HMM* use Sensortag to extract RSS in ZigBee mode instead of making special hardware and specific modification of hardware drivers.

D. ACOUSTICS-BASED APPROACHES

Acoustics-based approaches be used to sense human gestures in recent works such as [38], [48], [49]. [48] proposes an ultrasound-based finger tracking approaches to enable the interface between human and AR/VR devices. [38] utilizes Channel Impulse Response to recognize the minor finger motions with 7 mm resolutions. HUG proposes a micro hand gesture system using ultrasonic active sensing which could achieve a recognition accuracy of 96.32%.

III. FEASIBILITIES AND POTENTIAL RESOLUTIONS

In order to analyze the feasibilities and potential resolutions to achieve accurate human gesture sensing by utilizing coarse-grained RSS, this section analyzed the similarities and differences between coarse-grained RSS and fine-grained channel state information (CSI) respectively. The similarities verify that it is feasible to fetch and achieve more accurate human gestures sensing in practical applications with coarse-grained RSS, the differences reveal the insights why coarse-grained RSS could not achieve the same performance that fine-grained CSI can do in current works, which could help us find potential resolutions by narrowing down the differences between RSS and CSI.

A. SIMILARITIES

The similarities between RSS and CSI is that both of them reflect the multi-path fadings in the typical indoor

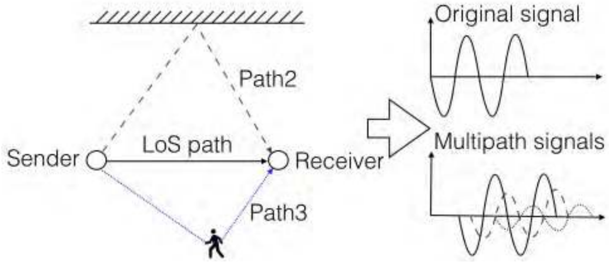


FIGURE 1. **The principles of multi-path effect in indoor environments.**

environment. The theory and principles of multi-path propagation are shown as Figure 1, the transmitted signal arrives at the receiving end through multiple paths, the different signal components arrived from different paths produce different amplitude offsets, frequency offsets and phase offsets. The example in Figure 1 assumes that there are three paths between the sender and the receiver, they are the Line-of-Sight path (Path 1, LoS path), the path reflected by the roof (Path 2) and the path reflected by human gestures (Path 3).

Based on the example showed in Figure 1, we introduce how RSS and CSI reflect multi-path fadings respectively.

1) HOW RSS REFLECTS THE MULTI-PATH FADINGS

the received signal is a superposition value of different arrival components from multiple paths. Therefore, the received energy could be calculated by Equation 1.

$$V = \sum_{i=1}^{N} \|V_i\| e^{-j\theta_i} \tag{1}$$

where V_i and $\boldsymbol{\theta}_i$ represent the amplitude and phase of the arrival signal components from the i^{th} path respectively, N represents the total number of the arrival signal components from different paths. RSS reflects the integrated situation of the multi-path effect, the relationship between RSS and the received signal from multi-path components can be represented as Equation 2.

$$RSS = 10\log_2(\|V\|^2) \tag{2}$$

where V represents all multi-path components arriving at the receiver. The unit of RSS is *dBm*. In the absence of human gestures, the fluctuated range of RSS in the static environment is 12 *dBm*, while if there exists human gesture impacts on RSS, the fluctuated range of *RSS* becomes to 520*dBm*, these abnormal fluctuation carries the features of different human gestures.

2) HOW CSI REFLECTS THE MULTI-PATH FADINGS

compared with RSS, CSI could deduce the detailed multi-path information (The amplitude shift and phase shift caused by each reflected path). The detailed response of multi-path effect which could be analyzed by CSI is shown as Figure 2. Where, the influence on the signal from propagation environment is defined as the channel impulse response (See Figure 2(a)), the amplitude shift is shown as Figure 2(b), and the phase shift is shown as Figure 2(c).

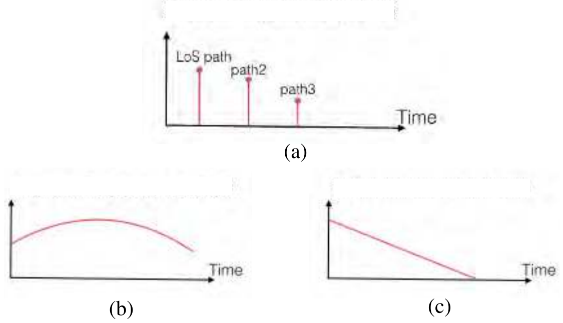


FIGURE 2. **The detailed response effected by human gestures. (a) Channel Impulse Response. (b) Amplitude-Frequency Response. (c) Phase-frequency Response.**

Figure 2 shows that the signal has different channel impulse responses from different paths, and the same channel impulse response will be differentiated for different frequencies of the signal according to the frequency selective attenuation characteristics. Simultaneously, different frequencies will also produce different phase attenuation. CSI is right the measurement used to describe channel impulse response which could deduce the amplitude fading and phase shift. Therefore, assume that the frequency of the carrier is f_k, the relationships between CSI and the amplitude and phase of the received signal are calculated by Equation 3.

$$H(f_k) = \|H(f_k)\|\, e^{j\sin(\angle H(f_k))} \tag{3}$$

Combined with the Equation 2, we can deduce that both V and $H(f_k)$ could describe multi-path propagation features in indoor environments. Therefore, the human gesture features should be attracted with coarse-grained RSS. In order to find potential solutions to increase the RSS-based human gesture sensing accuracy of RSS-based human gesture sensing and prediction applications, we also analyze the main differences between coarse-grained RSS and fine-grained CSI in Section III-B.

B. DIFFERENCES

The main significant differences between RSS and CSI rely on three aspects: i) different sampling rate; ii) different resolutions; and iii) different multi-path fading information reflected by RSS and CSI. In this section, we analyze the details of each difference between RSS and CSI, and propose the potential solutions based on these differences in Section IV.

- **Difference on sampling rate between RSS and CSI:** the sampling rate of RSS depends on the capabilities of different devices and communication protocols. For example, the sampling rate of ZigBee devices could reach 250Hz in a normal communication, and could be up to 1000Hz by minimizing the packet length. While, the sampling rate of WiFi devices could reach a maximum sampling rate of 2500Hz in a normal communication process.

TABLE 2. The number of samples collected from CSI in a normal communication process.

NIC name	Antenna Number	subcarrier numbers		samples	
		20MHz	40MHz	20MHz	40MHz
Intel 5300	3× 3	30	–	3×30	–
AR9590	3× 3	56	114	3×56	3×114
AR9580	3× 3	56	114	3×56	3×114
AR9565	1× 1	56	114	1×56	1×114
AR9462	2× 2	56	114	2×56	2×114
AR9380	3× 3	56	114	3×56	3×114
AR9382	2×2	56	114	2×56	2×114

Currently, CSI only can be read from a few network interface cards such as Intel 5300 and Atheros series NICs, which limits the adoption of CSI in all IoT devices. However, according to Shannon sampling theorem, RSS could record all the details of the human gestures as long as the sampling rate of RSS is twice fold of the gesture frequencies. Therefore, it is possible to attract fine-grained gesture features by up-sampling the RSS measurements (Detailed in Section IV-B).

• **Difference on resolutuon between RSS and CSI:** the differences on resolution between RSS and CSI are mainly reflected in two aspects: i) different measurement units; and ii) different number of the collected samples in a normal communication process.

For the first aspect, RSS is measured in *dBm* and changes in integers. Currently, the RSS resolution of Off-the-Shelf RFID device, ZigBee device, BLE device and WiFi NIC is 1 *dBm*. While CSI is a complex number, the complex number can deduce the amplitude and phase shifts of the received signals in milliwatt (mV) as unit, therefore, a small change of received amplitude could result in a larger fluctuation in the time serious, which helps to sense the human gestures moreaccurately. According to this feature, we propose to use RSS to extract fine-grained energy information to improve the accuracy of human gesture sensing applications (Detailed in Section IV-C).

For the second aspect, the number of samples collected in a normal communication process is different since the CSI could measure the amplitude and phase shifts of the received signal in each sub-carrier. Therefore, CSI obtains multiple samples through one acquisition, the number of samples currently available in NICs of Intel 5300, AR9590, AR9565, AR9462, AR9380, AR9382 are shown in Table 2. In order to narrow down this different, we increase the number of of RSS samples by collecting multiple times. The experimental results analyzed in SectionVI show that we can achieve an acceptable sensing accuracy when we collect 80 samples for one gesuture even though the number of samples is much lower than the CSI could get in the existing works.

• **Difference on the multi-path fading information reflected by RSS and CSI** According to the similarities between RSS and CSI, RSS reflects the comprehensive situation of multi-path effect while CSI reflects the detailed channel impulse response on different paths. In order to compensate for the shortcomings of RSS, this paper propose to add time dimension information to increase the resolution of the human gesture features for more accurate sensing (Detailed in Section IV-D).

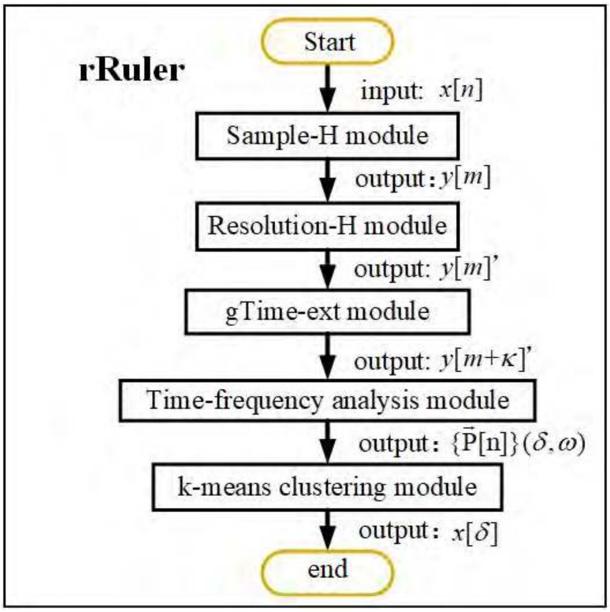


FIGURE 3. Feature extraction process of *rRuler*.

IV. RRULER: FINE-GRAINED FEATURE EXTRACTION FROM COARSE-GRAINED RSS

To narrow down the differences between RSS and CSI, this section proposes *rRuler*, a new feature extraction method for human gestures with coarse-grained RSS. We firstly introduces the overview of *rRuler* and follows with the detail design of each module of *rRuler*.

A. DESIGN OVERVIEW OF RRULER

According to the similarities and differences analysis of RSS and CSI in Section III, this section presents *rRuler*: a fine-grained human gesture feature extraction method with coarse-grained RSS. *rRuler* consists of five parts: i) sampling rate enhancement module (Sample-H), ii) resolution enhancement module (Resolution-H), iii) human gesture duration extraction module (gTime-ext), iv) frequency domain feature extraction module and v) k-means-based feature clustering module. The overview of *rRuler* is shown in Figure 3.

As shown in Figure 3, Sample-H module, Resolution-H module, and gTime-ext module belong to the data preprocessing part of *rRuler*. The purpose of the preprocessing part is to obtain fine-grained features for gestures from the coarse-grained RSS readings. Time-frequency domain analysis module is to extract the fixed frequency components which does not change with different time series data; k-means-based frequency domain feature clustering is to reduce the dimensions of time-frequency analysis results, which could further reduce the computational complexity of the gesture sensing process. The functions of each module are briefly described as follows:

- **Sample-H module**, the goal of this module is to improve the sampling rate of RSS without changing the characteristics of the original RSS time series measurements. The function of this module is used to compensate for the information loss that occurs during the time-frequency and k-means based clustering processes. The output sequence of Sample-H module is $y[m]$. The detailed design of the Sample-H module is introduced in Section IV-B.
- **Resolution-H module**, the goal of this module is to improve the resolution of the output sequence $y[m]$ of Sample-H module. This module converts the RSS value (in decibels) into the received energy (mW) to improve the resolution of the human gesture features. The output sequence of Resolution-H is $y[m]'$. The detailed design of Resolution-H module is introduced in Section IV-C.
- **gTime-ext module**, the goal of this module is to extract the durations of different gestures according to the RSS time series values, and integrated the duration information into the original RSS time series to help differentiate different gestures. The detailed design of gTime-ext module is introduced in Section IV-D.
- **Time-frequency analysis module**, as far as we know, the same gestures may introduce different RSS jitter changes when we collect the time domain RSS readings multiple times. However, the frequency domain features which represent the frequency of the same gestures are fixed. Therefore, this module uses Short-time Fourier Transform (STFT) method to extract the frequency domain features for accurate gesture sensing. The output of this module is a two-dimensional matrix $\vec{P}_{\delta,\omega}$, where δ indicates the number of time domain dimensions, while ω indicates the number of frequency domain dimensions. The detailed design of this module is introduced in Section IV-E.
- **k-means based feature clustering module**, the goal of this module is to reduce the dimension of the two-dimensional matrix into one-dimensional matrix again to overcome the overfitting problem in gesture prediction and sensing process. The output of this module is $x[\delta]$ which is a time domain series with frequency domain features. The detailed design of this module is introduced in Section IV-F.

B. SAMPLE-H: RSS SAMPLING RATE ENHANCEMENT APPROACH

According to differences between RSS and CSI, the sampling rate of CSI is 2500Hz, while the sampling rate of ZigBee, BLE and RFID is around 250Hz, and the maximum sampling rate of them can be increased to 1000Hz by changing the packet structure [26]. According to the Shannon formula, the RSS time series is theoretically able to record all the details of the gestures when RSS sampling frequencies is twice-fold of human gestures frequencies. However, the time-frequency analysis and k-means based feature extraction procedures may result in the loss of information,therefore, it is essential to increase the sampling rate of RSS time series to compensate for the data loss.

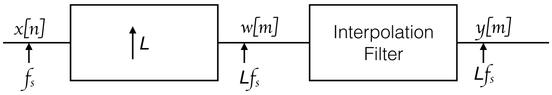
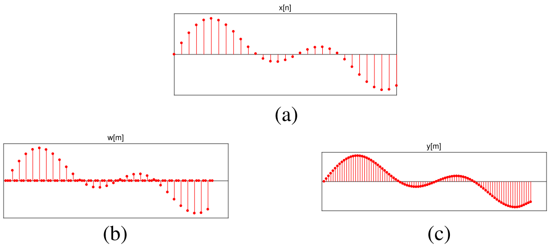


FIGURE 4. The structure of sampling rate enhancement module (Sample-H).



(a)

(b)

(c)

FIGURE 5. An example describes the principle of Sample-H module. (a) Original Signal. (b) Intermediate Signal. (c) Target Signal.

This section uses oversampling method to improve the sampling rate of RSS time series. The oversampling method could increase the resolution of the signal in time domain without destroying the original signal characteristics.

Assuming that the RSS time series which contains the human gestures collected by the network device is $x[n]$. We want to increase L-fold of the sampling rate compared with the original RSS sequence. The oversampling method mainly includes two steps: i) interpolation position acquisition; and ii) value calculation for each position. The structure of this module is shown in Figure 4.

Where $x[n]$ represents the input signal and L is the fold which we want to up-sampled, $w[m]$ represents the intermediate signal with the interpolated position, $y[m]$ is the target signal with the intermediate value which is estimated by the interpolation filter. Figure 5 shows an example of the original signal $x[n]$ (Figure 5(a)), the intermediate signal $w[m]$ (Figure 5(b)), and the target signal $y[m]$ (Figure 5(c)).

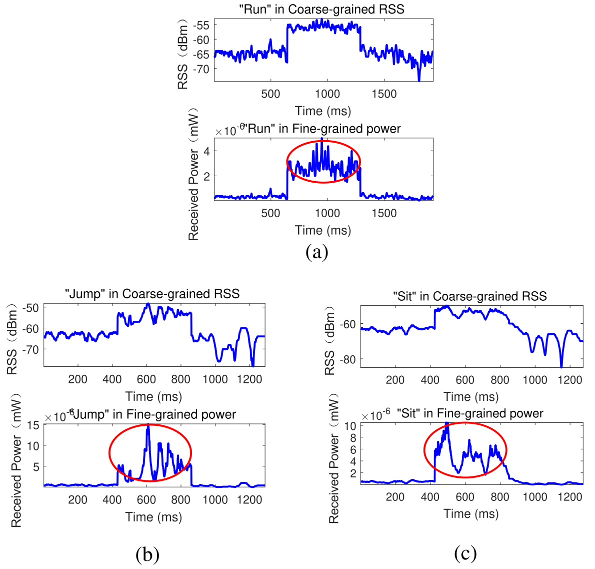
Where, the calculation method of the locations and values to be insert into the original signals is described in detail as follows.

- Interpolation position design: we mark the position that needs to be interpolated as 0, since the normal RSS readings are negtive, 0 could distinguish the original signal and the inserted data without ambiguity. Where the positions to be interpolated are determined by Equation 4.

$$w[m] = \begin{cases} x(m/L) & (m = nL) \\ 0 & (otherwise) \end{cases} \tag{4}$$

where n is the length of the original signal, m is the length of the intermediate signal, L is the up-sampling fold and $m = nL$.

- Interpolation filter design: after determining the interpolation positions, we use the interpolation filter algorithm to replace 0 to be interpolated value. The value is calculated by



(a)

(b)

(c)

FIGURE 6. The comparison of RSS time series and received power time series which contain the information of different gestures. (a) RSS contains gesture "Run". (b) RSS contains gesture "Jump". (c) RSS contains gesture "sit down".

Equation 5.

$$y[j+nL]=\sum_{k=0}^{K} x[n-k]\cdot h[j+kL], \quad j=0,1,2,\ldots,L-1 \tag{5}$$

where, $h[*]$ represents the channel impulse response, this paper uses low-pass filter instead of $h[*]$, K is the value variable k which makes $h[j+kL]$ to obtain non-zero value.

C. RESOLUTION-H: RSS RESOLUTION ENHANCEMENT APPROACH

In order to extract fine-grained human gesture features from the coarse-grained RSS time series, we collected RSS time series samples of three gestures under the experimental setup in Section VI-A, which are "Run", "Jump" and "sit down (Sit)". There RSS time series are shown in Figure 6, the jitter range of RSS is around $1dBm$, which caused by two reasons: i) caused by Human gestures; and ii) caused by other factors in the environment.

Through the experience of [50], RSS jitter will change in the range of $0-5dBm$ without any human gestures, while the jitter changes caused by human gestures are around $5-20dBm$. Therefore, the changes caused by different gestures on RSS series are not that significant, Figure 6(a) and Figure 6(b) are examples show that the RSS characteristics of gesture "jump" and the "sit" are difficult to differentiate in the unit of dBm.

In order to extract the fine-grained features of different gestures, this section converts RSS into received energy which is calculated by Equation 6.

$$P_r = 10^{\frac{RSS}{10}} \tag{6}$$

TABLE 3. Durations of different gestures.

Gesture Name	Duration time
Fall Forward	358ms
Fall Left	350ms
Fall Right	362ms
Run Vertically	809ms
Jump	163ms
sit down	204ms
walk	907ms

Assuming that RSS resolution is 1 dBm, therefore, the resolution of received energy is deduced by Equation 7.

$$\frac{P_{r1}}{P_{r2}} = \frac{10^{\frac{RSS1}{10}}}{10^{\frac{RSS2}{10}}} = 10^{\frac{RSS1}{10}-\frac{RSS2}{10}} = 10^{\frac{1}{10}} \tag{7}$$

Equation 7 shows that when the differences in decibel are 1dBm, 2dBm, 3dBm, 4dBm, 5dBm, the ratio of the received energies $P_{r1} : P_{r2}$ is $10^{\frac{1}{10}} \approx 1.26$, $10^{\frac{2}{10}} \approx 1.582$, $10^{\frac{3}{10}} \approx 2$, $10^{\frac{4}{10}} \approx 2.51$, $10^{\frac{5}{10}} \approx 3.16$. Therefore, when RSS produces a smaller jitter, the change in the received energy will change exponentially, which increases the resolutions of the RSS jitters. Figure 6(a), Figure 6(b) and Figure 6(c) show the different received energy patterns of different gestures. The feature resolutions of different gestures showed in Figure 6(a), Figure 6(b) and Figure 6(c) have been enhanced significantly, where the "jump" and "sit" gestures could be differentiated obviously. In summary, Resolution-H module has great potential to improve the resolution of gesture features and increase the accuracy of gesture sensing applications.

D. GESTURE DURATION EXTRACTION

RSS only reflects the comprehensive characteristics of multi-path effects, these complex features are sometimes affected by different propagation paths. For example, constructive multi-path effects increase the receiving intensity, and the non-constructive multi-path effect will weaken the receiving intensity. The CSI eliminates the effects of different multi-path effects on the gestures by analyzing the change in the received signal strength on each path. In order to compensate for this, this section extracts the duration characteristics of the gestures as a supplement to the original RSS sequence to enhance the resolution of the gesture features.

The main basis for this section is from the observation of the execution time of the different gesture samples. This section summarizes the average time of the gesture set collected in Section VI-A, and the average execution time of each gesture is shown in Table 3.

In order to identify the durations of the gestures as an additional feature, we first obtain the maximum length of the RSS sequence contains different gestures information. Secondly, we compare the length of each samples with the maximum value, it does not need to change the compared sample if its length is equal to the maximum value; while we add $10 \times \lg(-64)$ in the tail of the compared sample if its length is less than the maximum value which aim to make the sample has the same length with maximum value, it is obviously that the length of the tail reflects the durations of different gestures.

E. TIME-FREQUENCY ANALYSIS

As far as we know that, the time domain features varies with different samples in the training and prediction sets. To extract stable features, this section introduces time-frequency analysis to extract the frequency components of each gesture.

The reasons we have to extract the frequency components of the gestures attribute to the disadvantages of the received energy of the signal in time domain as follows:

- Time domain features of the same gesture extracted from different samples show a large vibration since different multi-path fading occurs when we conduct the same gesture in different locations or with different amplitudes. Therefore, it is difficult to recognize human gestures with time domain features.
- The length of time domain RSS increases with the upsampling folds in Sample-H module, which may result in over-fitting of the human gesture sensing algorithm (*rRuler-HMM* in Section V).

Due to the disadvantages of the time domain features, this section introduces time-frequency analysis to extract frequency domain features for accurate human gesture sensing and prediction. We use Short-time Fourier Transform (STFT) to extract the frequency domain features of gestures. Assume that $y[m]'$ is the time series of the received signal contains gesture i, the frequency components $\vec{P}[n]$ for $y[m]'$ is calculated by Equation 8.

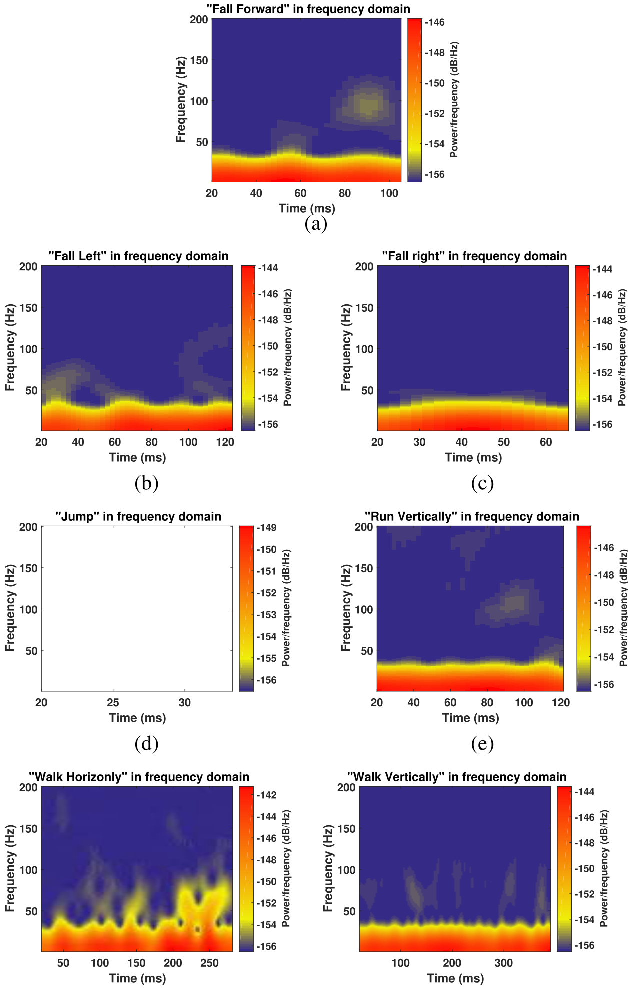
$$STFT\{Py[m']\}(\delta, \omega) \Leftrightarrow X(\delta, \omega) = \sum_{n=-\infty}^{\infty} y[m]'w[n-\delta]e^{-j\omega n} \tag{8}$$

where, $w[n]$ is the window size for STFT. Figure 7 shows the frequency domain features of the seven gestures which collected in section VI-A.

F. K-MEANS BASED FEATURE CLUSTERING MODULE

The original RSS time series are changed from one-dimensional time data to two-dimensional time-frequency energy distribution data after time-frequency analysis. The original energy sequence $y[m]'$ is divided into an energy distribution matrix whose time dimension length is δ and frequency dimension length is ω.

Assuming that the two-dimensional energy distribution data is $X(\delta, \omega) = \{\vec{X}_1, \vec{X}_2, \ldots, \vec{X}_\delta\}$, in which $\vec{X}_i (i = 1, 2, \ldots, \omega)$ is a vector. In order to reduce the computational complexity of the human gesture sensing algorithm, this section uses k-means to cluster all the gesture sets $\vec{X}_i$. Therefore, we convert any vector $\vec{X}_i$ into an integer which represents k-means cluster numbers, as a sequence the two-dimensional energy distribution matrix is reduced into one dimensional data again.



(a)
(b) (c)
(d) (e)
(f) (g)

FIGURE 7. Spectrograms of different gestures. (a) Fall Forward (FF). (b) Fall Left (FL). (c) Fall Right (FR). (d) Jump (JM). (e) Walk Horizontally (WH). (f) Walk Vertically (WV). (g) Run Vertically (RV).

The k-means based feature clustering process contains three phases as follows:

- Initialization phase: select k vectors $(\vec{X}_i(i = 1, 2, \ldots k))$ as the initial centers of each clusters, and we mark the initial centers as $m_1^{(1)}, \ldots, m_k^1$.
- Classification phase: for each $\vec{X}_i$, we calculate which clusters it should belongs to, and the calculation method is shown in Equation 9.

$$S_i^{(t)} = \{X_p \left\| X_p - m_i^{(t)} \right\|^2 \leq \left\| X_p - m_j^{(t)} \right\| \forall j, 1 \leq j \leq k\} \tag{9}$$

where t represents the the execution iterations of the k-means based clustering algorithm, $S_i^{(t)}$ represents the set of the vectors which belongs to cluster i.

- Cluster centers updating phase: repeat the second phase until the algorithm convergence or reach the requirements set by users. The new center calculation method is shown in

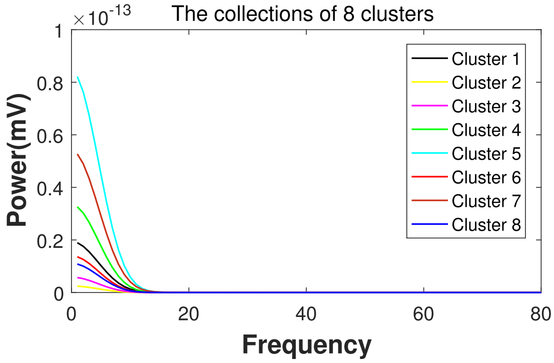


FIGURE 8. The centers of the *k*-means clusters.

Equation 10.

$$m_i^{(t+1)} = \frac{1}{\left|S_i^{(t)}\right|} \sum_{X_j \in S_i^{(t)}} X_j \tag{10}$$

Figure 8 shows the results of the seven gestures in Figure 7, which are divided into eight clusters. The figure shows the centers of each clusters.

V. GESTURE CLASSIFICATION AND SENSING

In order to further verify the performance of *rRuler* and bring it to practical applications, we propose *rRuler-HMM*, a Hidden Markov Model (HMM) based human gesture sensing and prediction algorithm in this section and implement it in Section VI. In this section we introduce: i) the overview of the *rRuler-HMM*, ii) how to divide the training set and prediction set of *rRuler-HMM*, iii) how to mapping the parameters from real word data sets to HMM, and iv) how to training the parameters of HMM and sensing the human gesture by utilizing the trained HMM Model.

A. *DESIGN OVERVIEW OF* RRULER-HMM

The objective of *rRuler-HMM* is to predict the type of different gestures, which take the output of *rRuler* as the training and prediction sets. The structure of *rRuler-HMM* contains two parts: i) parameter training part and ii) gesture prediction part. The overview of *rRuler-HMM* is shown in Figure 9. The functions of each part are as follows:

- **Parameter training part** is to use the initial HMM parameters and the data from training sets to calculate the HMM parameters iteratively according to the requirements of the practical applications. In this section, *rRuler-HMM* explores the classical *Baum-Welch* for parameters training, the procedures are detailed in Section V-D.2.
- **Gesture prediction part** is to predict gesture types for the input time series by utilizing the model trained in parameter training part. In this section we introduce classical *viterbi* algorithm and *maximum like-hood estimation* method to predict the type of different gestures, the procedures are detailed in Section V-D.3.

B. *THE DIVISION METHOD OF TRAINING SET AND PREDICTION SET*

According to the structure of Figure 9, this section divides the data set collected in section VI-A into training set and prediction set, where p is the proportion of the training set to the overall data set. In addition, we introduce k folding cross validation method for *rRuler-HMM*'s' parameters training to verify the stability of the model.

Assuming that the number of gestures to be predicted is m, the overall data set is S, the sub-data set for each gesture i is S_i. Therefore, $S = \left[S_1 \ S_2 \ \cdots \ S_n \right]^T$. In order to construct the training and prediction sets which meet the requirements of k-fold cross-validation, the overall data set is divided into k different parts according to different gesture types.

Assuming that the data set for each gesture is $S_i = \left[S_{i,1} \ S_{i,2} \ \cdots \ S_{i,|S_i|} \right]^T$, where $|S_i|$ is the size of the data set of gesture i. Therefore, if we divided the data set of each gesture i into k equal parts with the size of ${}^{|S_i|}/_{k}$, and the set $(S_{(i,j)})$ of each part j for gesture i is divided by the regular defined in Equation 11.

$$S_{(i,j)} = \begin{bmatrix} S_{j\times\frac{|S_i|}{k}-1} \\ S_{j\times\frac{|S_i|}{k}} \\ \vdots \\ S_{j-1\times\frac{|S_i|}{k}-2} \end{bmatrix} \tag{11}$$

Therefore, the overall data set S could be expressed as Equation 12.

$$S = \begin{bmatrix} S_{(1,1)} \\ S_{(1,2)} \\ \vdots \\ S_{(1,k)} \end{bmatrix} \cup \begin{bmatrix} S_{(2,1)} \\ S_{(2,2)} \\ \vdots \\ S_{(2,k)} \end{bmatrix} \cup \cdots \cup \begin{bmatrix} S_{(m,1)} \\ S_{(m,2)} \\ \vdots \\ S_{(m,k)} \end{bmatrix} \tag{12}$$

On the basis of Equation 12, the k-folding cross validation training set $T_s = \left[T_1 \ T_2 \ \cdots \ T_k \right]$ and perdition set $P_s = \left[P_1 \ P_2 \ \cdots \ P_k \right]$, are defined according to the following rules.

- For all gestures $i(i = 1, 2, \cdots, m)$, select $S_{(i,j)}$ as part of the training set if and only if $j = \alpha$. Therefore, the prediction set P_α is calculated by Equation 13.

$$P_\alpha = \exists_{j=\alpha} \forall_{i(i\in(1,2,\cdots m))} S_{i,j} \tag{13}$$

- For all gestures $i(i = 1, 2, \cdots, m)$, if the prediction set P_α has been selected, then the corresponding training set is $T_\alpha$is $S - P_\alpha$.

In this section, the k-fold cross-validation method is designed to verify the stability of the *rRuler-HMM*. All the experimental results shown in Section VI in the remainder of this paper are under the assumption of 10-fold cross-validation.

C. *HMM PARAMETERS MAPPING*

This section introduces how to map the fine-grained features extracted by *rRuler* to the parameters of *rRuler-HMM*.

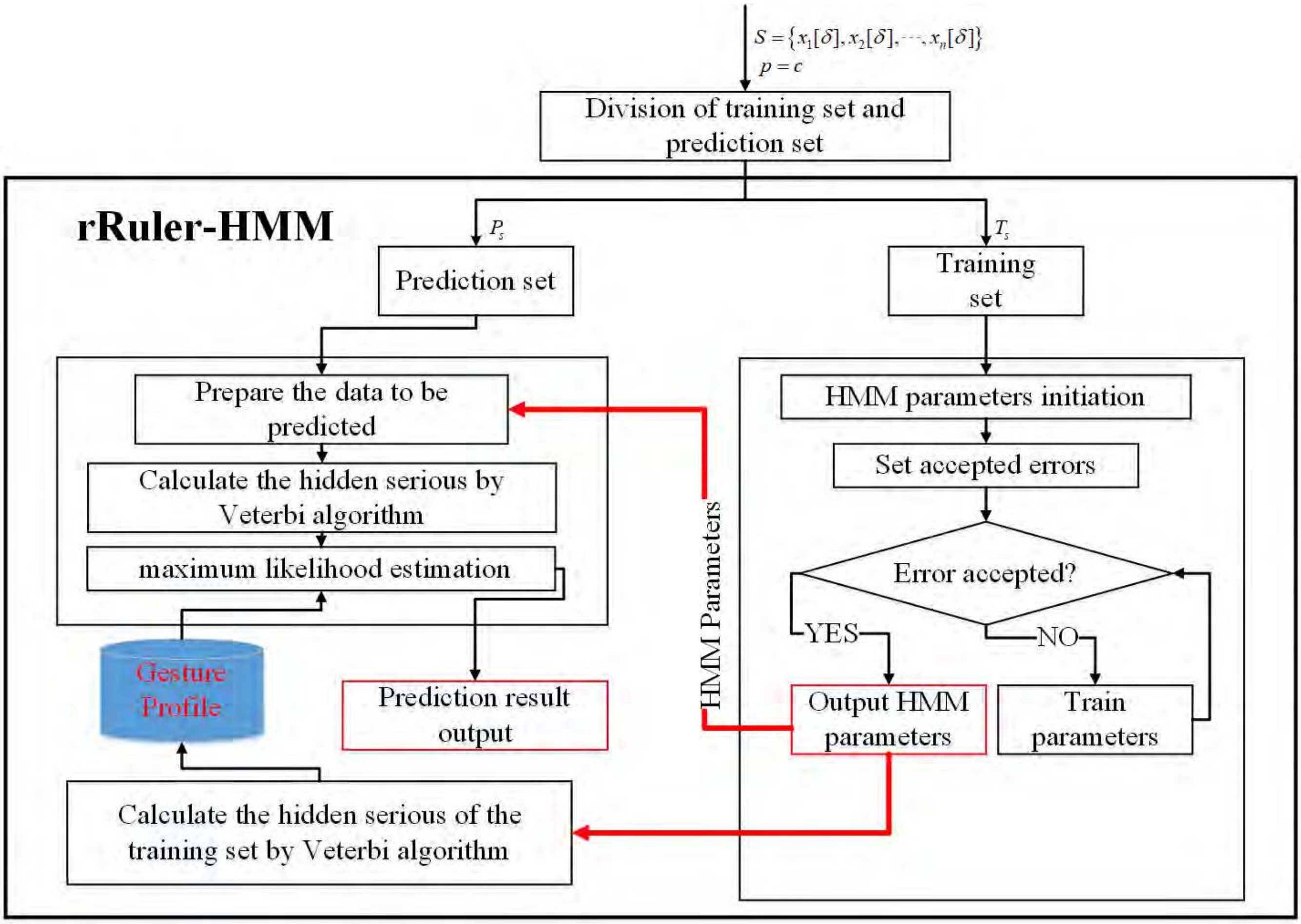

$S=\{x_1[\delta],x_2[\delta],\cdots,x_n[\delta]\}$
$p=c$
Division of training set and prediction set
rRuler-HMM
P_s
T_s
Prediction set
Training set
Prepare the data to be predicted
Calculate the hidden serious by Veterbi algorithm
maximum likelihood estimation
Gesture Profile
Prediction result output
HMM Parameters
HMM parameters initiation
Set accepted errors
Error accepted?
YES
NO
Output HMM parameters
Train parameters
Calculate the hidden serious of the training set by Veterbi algorithm

FIGURE 9. The overview of *rRuler-HMM*.

TABLE 4. Parameters mapping.

Parameters	The mapping with the feature extracted by rRuler
O	The observing states classified by k-means clustering algorithm (The output of the module introduced in Section IV-F)
hS	Implied state behind the observed states, for example, the implied states behind observed states may be the amplitude changing trends of the signals.
Tran_est	The transition probabilities among implicit states, which is defined by Equation 14
EMIS_est	The transition probabilities among observe states, which is defined by Equation 15
π	The initial state probability matrix

The parameters of HMM model are composed of five elements, including two states and three probability matrix, which are hidden States hS and Observed sequence O; initial state probability matrix π, the implicit state transition matrix $TRAN\_est$, and the observation state transition matrix $EMIS\_est$, respectively. Therefore, the HMM Model is $\lambda = \{TRAN\_est, EMIS\_est, \pi\}$. The mappings between HMM parameters and fine-grained features extracted in Section IV are shown in Table 4.

Suppose that $hS = \{hS_1\ hS_2 \cdots hS_n\}$, $n = |hS|$. The the dimension of $TRAN\_est$ is $n \times n$. Arbitrary element $TRAN\_est_{i,j}$ of matrix $TRAN\_est$ represents the probability of an implicit state hS_j occurs under the assumption that implicit state hS_i has already occurred. Therefore, $TRAN\_est_{i,j}$ could be calculated by Equation 14.

$$TRAN\_est_{i,j} = P(hS_j\,|hS_i) \tag{14}$$

Assuming that $O = \left\{O_1\ O_2 \cdots O_m\right\}$, $m = |O|$. The dimension of $EMIS\_est$ is $n \times m$. Arbitrary element $EMIS\_est_{i,j}$ of $EMIS\_est$ could be calculated by Equation 15.

$$EMIS\_est_{i,j} = P(O_i\,\left|hS_j\right.) \tag{15}$$

where $TRAN\_est$ and $EMIS\_est$ were trained by training set, which will be introduced in Section V-D.

D. HMM PARAMETERS TRAINING AND SENSING

This section introduces how to train the hidden state transition matrix $TRAN\_est$, observation state transition matrix $EMIS\_est$ and how to sense the human gestures by utilizing the trained *rRuler-HMM*. We introduce initial value settings, training method and sensing method respectively in the remaining of this section.

1) INITIALIZATION

The transition probabilities among the same hidden states are theoretically larger than that among different hidden states since human gestures have temporal locality features. Therefore, we set the initial values as Equation 16.

$$TRAN\_est_{(i,j)} = \begin{cases} 0.6(i = j) \\ 0.4/(n-1)(i \neq j) \end{cases} \tag{16}$$

where n is the number of hidden states.

The occurring probabilities of O_j under the assumption of hS_i are difficult to speculated. Therefore, we set the initial values according to the rulers defined by Equation 17 considering the fairness of the training process.

$$EMIS\_est_{i,j} = {}^{1}\!/_{m} \tag{17}$$

where m is the number of observed states.

In order to put it easily, we take $n = 5$, and $m = 10$ as an example, the initial matrix $TRAN\_ini$ and $EMIS\_ini$ are shown in the following matrices.

$$TRAN\_est = \begin{bmatrix} 0.6 & 0.1 & 0.1 & 0.1 & 0.1 \\ 0.1 & 0.6 & 0.1 & 0.1 & 0.1 \\ 0.1 & 0.1 & 0.6 & 0.1 & 0.1 \\ 0.1 & 0.1 & 0.1 & 0.6 & 0.1 \\ 0.1 & 0.1 & 0.1 & 0.1 & 0.6 \end{bmatrix}$$

$$EMIS\_est = \begin{bmatrix} 0.1 & 0.1 & 0.1 & 0.1 & \cdots & 0.1 & 0.1 & 0.1 \\ 0.1 & 0.1 & 0.1 & 0.1 & \cdots & 0.1 & 0.1 & 0.1 \\ 0.1 & 0.1 & 0.1 & 0.1 & \cdots & 0.1 & 0.1 & 0.1 \\ 0.1 & 0.1 & 0.1 & 0.1 & \cdots & 0.1 & 0.1 & 0.1 \\ 0.1 & 0.1 & 0.1 & 0.1 & \cdots & 0.1 & 0.1 & 0.1 \end{bmatrix}$$

2) TRAINING

In this section, the classical *Baum-Welch algorithm* is used to train *rRuler-HMM*. Where the training set is defined in Section V-B and the initial value of the training parameters are defined in Section V-D.1. The procedure of rRuler-HMM parameters training algorithm is shown as Algorithm 1.

As shown in algorithm 33, the 2^{nd} line is the initialization operations; 12^{th} to 13^{th} lines represent forward and backward algorithm; 15^{th} to 26^{th} lines represent the update process of parameters.

Algorithm 1 Parameter Training Procedures of rRuler-HMM

```
input  : $P_t$,$TRAN\_ini$,$EMIS\_ini$,$tol$,$maxiter$
output: $TRAN\_est$,$EMIS\_est$
1  Initialization;
2  $tol \leftarrow 1e-6$; $trtol \leftarrow tol$; $etol \leftarrow tol$; $maxiter \leftarrow 500$;
3  Abnormal judgment;
4  $[numStates, checkTr] \leftarrow size(guessTR)$;
5  $[checkE, numEmissions] \leftarrow size(guessE)$;
6  if $checkTr == numStates$ & $checkE == numStates$ then
7      $TR = zeros(size(guessTR))$; $pseudoTR = TR$;
       $converged = false$;$loglik = 1$;$E = zeros(numStates, numEmissions)$;
       $pseudoE = E$;$logliks = zeros(1, maxiter)$;
8      for $interation \leftarrow 1$ to $maxiter$ do
9          $oldLL \leftarrow loglik$; $loglik \leftarrow 0$;
           $oldGuessE \leftarrow guessE$; $oldGuessTR \leftarrow guessTR$;
10         for $count \leftarrow 1$ to $numSeqs$ do
11             Calculate the forward and backward probabilities;
12             $[\ , logPseq, fs, bs, scale] \leftarrow hmmdecode(seq, guessTR, guessE)$;
13             $loglik \leftarrow loglik + logPseq$; $logf \leftarrow log(fs)$;
               $logGE \leftarrow log(guessE)$; $logb \leftarrow log(bs)$;
               $logGTR \leftarrow log(guessTR)$; $seq \leftarrow [0\ seq]$;
14             Update the parameters;
15             for $k \leftarrow 1$ to $numStates$ do
16                 for $l \leftarrow 1$ to $numStates$ do
17                     for $i \leftarrow 1$ to $seqLength$ do
18                         $TR(k, l) \leftarrow TR(k, l) + exp(logf(k, i) + logGTR(k, l) + logGE(l, seq(i + 1)) + logb(l, i + 1))./scale(i + 1)$;
19                     end
20                 end
21             end
22             for $k \leftarrow 1$ to $numStates$ do
23                 for $i \leftarrow 1$ to $numEmissions$ do
24                     $pos \leftarrow find(seq == i)$;
25                     $E(k, i) \leftarrow E(k, i) + sum(exp(logf(k, pos) + logb(k, pos)))$;
26                 end
27             end
28         end
29         $guessE \leftarrow E./(repmat(sum(E, 2), 1, numEmissions))$;
           $guessTR \leftarrow TR./(repmat(sum(TR, 2), 1, numStates))$; if
           $converge == true$ then
30             return;
31         end
32     end
33 end
```

3) HUMAN GESTURE SENSING

rRuler-HMM verifies the accuracy of sensing and prediction by utilizing the prediction set defined in Section V-B, and the prediction algorithm is the classical *Viterbi* and *maximum like-hood estimation* algorithms.

VI. IMPLEMENTATION AND EVALUATION

In this section, we describe how to implement *rRuler* and *rRuler-HMM* in real word firstly, then we show the results of

TABLE 5. The descriptions of the samples collected from our scenarios.

Gesture Type	Gesture Name	Number of Samples (Samples/person×Number of persons)		Descriptions
		LoS scenario	NLoS scenario	
Accident	Fall Forward(FF)	80(8×10)	80(8×10)	"Fall Forward" is the gesture occurs at position n and fall towards to direction d (shown as Figure 11).
	Fall Left(FL)	80(8×10)	80(8×10)	"Fall Left" is the gesture occurs at position n and fall towards to direction c(shown as Figure 11).
	Fall Right(FR)	80(8×10	80(8×10)	"Fall Right" is the gesture occurs at position n and fall towards to direction a (shown as Figure 11).
Daily Life	Walk Horizontlly (WH)	80(8×10)	80(8×10)	The walking trace of "Walk Horizontally" is $a \rightarrow n \rightarrow c$ (shown as Figure 11).
	Walk Vertically(WV)	80(8×10)	80(8×10)	The walking trace of "Walk Veritically" is $b \rightarrow n \rightarrow d$ (shown as Figure 11).
Fitness	Jump(JM)	80(8×10)	80(8×10)	"Jump" represents the person jump at location n (shown as Figure 11).
	Run Vertically(RV) d	80(8×10)	80(8×10)	The run trace of "Run Vertically" is $b \rightarrow n \rightarrow d$ (shown as Figure 11).

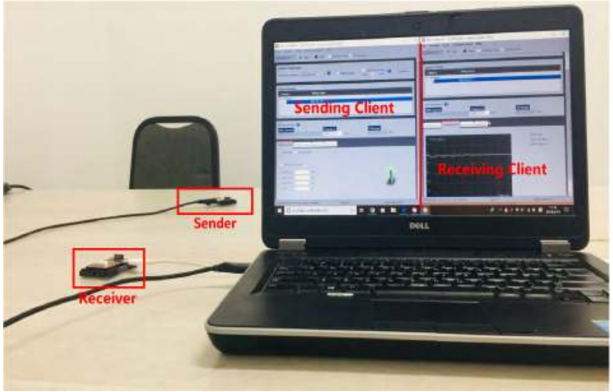


FIGURE 10. The implementation of *rRuler* and *rRuler-HMM* comprehensive system.

our comprehensive experiments including the performance of *rRuler*, the sensing accuracy of *rRuler-HMM*, and the factors that impact on human gesture sensing accuracy.

A. SYSTEM IMPLEMENTATION

In order to verify the performance of *rRuler* and *rRuler-HMM*, we implemented the system by utilizing *TI Sensortag* as data collection sub-system and a laptop as data analysis sub-system respectively. Specifically, the basic components of those two sub-systems are: i) data collection sub-system contains *TI Sensortag* in ZigBee mode and *Sensortag Debug Board (Devpack) modules*, and ii) data analysis sub-system contains a laptop installed with SmartRF Studios which is used to storage and analyze the RSS samples extracted from the data collection sub-system. The prototype of the system is shown in Figure 10.

For data collection sub-system, the software was implemented by TI development kit *SmartRF*, the protocol was set as ZigBee mode which is fully compatible with 802.15.4. The system uses totally two *Sensortags* to form a communication pair. The sender is responsible for sending packets to the receiver continuously, and the receiver is responsible for receiving the packets and record the RSS values.

For the data analysis sub-system, the feature extraction and gesture prediction functions are implemented by Matlab, we run the programs with a laptop which is shown in Figure 10.

The prototype implemented in this paper could collect and analyze RSS samples with different gestures in both LoS and NLoS scenarios. Due to the widely use of the NLoS scenarios in daily gesture sensing applications, we analyze the insight experimental results by utilizing the data collected from NLoS scenarios, we also show the optimized sensing accuracy in LoS scenarios to further verify the feasibility of our system.

B. DATA COLLECTION

In order to verify the performance of *rRuler* and *rRuler-HMM*, this section collects 7 different types of gestures in both LoS and NLoS scenarios under the system implemented in Section VI-A.

The data collection environment and detailed locations for LoS and NLoS scenarios are shown as Figure 11. The data collection environment is a laboratory with 7.7 meters long and 6.5 meters wide. Position 1 and Position 2 marked with red circle in Figure 11 are the locations for LoS and NLoS scenarios respectively. The attributes of the gestures such as name, type, the number of samples and descriptions are detailed in Table 5. In summary, we collect 560 samples for both LoS and NLoS scenarios respectively, which are from 10 different volunteers, where 7 of them are male volunteers, 3 of them female volunteers, they are all undergraduate students aged from 22 to 26.

C. PERFORMANCE MATRICES

In this section, we describe the performance matrices which use to evaluate both the feature extraction part *rRuler* and human gesture sensing part *rRuler-HMM*. For the feature extraction part, we measure the similarities among features to evaluate the accuracy of feature extraction performance, and we also measure the running time of *rRuler* to evaluate its time efficiency. For gesture sensing part, we evaluate

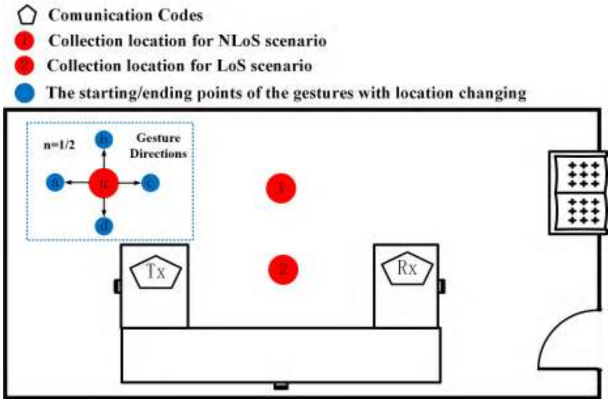

Comunication Codes
1 Collection location for NLoS scenario
2 Collection location for LoS scenario
The starting/ending points of the gestures with location changing
n=1/2
Gesture Directions
a
b
c
d
n
1
2
Tx
Rx

FIGURE 11. The laboratory structure and data collection locations for LoS and NLoS scenarios.

the human gesture accuracy for *rRuler-HMM*. Specifically, the performance matrices of *rRuler* and *rRuler-HMM* comprehensive system are defined as follows:

- **Feature Similarity (*hDist*)** We use hamming distance to evaluate the similarities among gestures, hamming distance $hDist(s, t)$ is calculated by Equation 18.

$$hDist(s,t) = 1 - \#(x[\delta]_{sj} \neq x[\delta]_{tj})\big/\delta \tag{18}$$

where, $x[\delta]_s$ represents the sample contains gesture s, $x[\delta]_t$ represents the sample contains gesture t, the length of the samples for each gesture is δ according to the output of *rRuler* which is calculated by k-means algorithm. Therefore, $\#(x[\delta]_{sj} \neq x[\delta]_{tj})$ represents whether it different or same in location j for gesture s and t. The similarity of gesture s and gesture t is larger if the cluster feature number is smaller, vice versa, the similarity of gesture s and t is smaller if cluster feature number is larger. Therefore, the similarity of gesture s and t is proportional to the value of $hDist(s, t)$.

- **Time Consumption:**

We uses the running time of our Matlab-based simulator to evaluate the computational overheads of *rRuler* and its individual parts.

- **Sensing Accuracy:**

We uses sensing accuracy to evaluate the performance of *rRuler-HMM* system, which also could further verify the efficiency of *rRuler*. Sensing accuracy is the ratio of the number of gestures which could be predicted correctly by *rRuler-HMM* and those could not be.

D. PERFORMANCES OF RRULER

In this section, we take the data collected from the NLoS scenario as an example to demonstrate the performances of *rRuler*, including the similarities among gestures and the computational overheads of each module in different parameter settings.

1) PERFORMANCE ANALYSIS UNDER SPECIFIC PARAMETERS

In this section, we evaluated the performance of *rRuler* under specific settings of the parameters. The parameters are set as: i) $SamplingRate = 250Hz$; ii) Up-sampling fold is 4, which is shortly marked as $UP - rate = 4$; and iii) the cluster number of k-means algorithm is 10, which is shortly marked as $ClusterNum = 10$. Table 6 and Table 7 show the similarities and the variance of the seven different gestures collected in Section VI-B respectively.

TABLE 6. The average values of the similarities(sampling rate = 250Hz, UP-rate = 4, ClusterNum = 10).

	FF	FL	FR	JM	RV	WH	WV
FF	**0.8213**	0.7637	0.7906	0.7447	0.7449	0.3608	0.1343
FL	0.7239	**0.758**	0.746	0.7366	0.7421	0.3722	0.1154
FR	0.7786	0.7456	**0.9**	0.859	0.7148	0.3668	0.1021
JM	0.7493	0.7097	0.8472	**0.9228**	0.7036	0.3755	0.0707
RV	0.7005	0.7111	0.6875	0.6971	**0.7455**	0.3732	0.0843
WH	0.3625	0.3806	0.3592	0.3725	0.375	**0.3888**	0.1815
WV	0.1004	0.1109	0.0693	0.054	0.086	0.134	**0.2968**

TABLE 7. Variances of the feature similarities (sampling rate = 250Hz, UP-rate = 4, ClusterNum = 10).

	FF	FL	FR	JM	RV	WH	WV
FF	0.0048	0.0012	0.0011	0.0004	0.0012	0.0011	0.0011
FL	0.0003	0.0038	0.0004	0.0005	0.0004	0.0007	0.0009
FR	0.0016	0.0012	0.0017	0.0008	0.0008	0.0011	0.0012
JM	0.0002	0.0005	0.0003	0.0004	0.0003	0.0009	0.0007
RV	0.0004	0.0004	0	0	0.0012	0.0009	0.0012
WH	0.0004	0.0006	0.0004	0.0004	0.0006	0.0057	0.0008
WV	0.0007	0.0005	0.0003	0.0001	0.0005	0.0012	0.0081

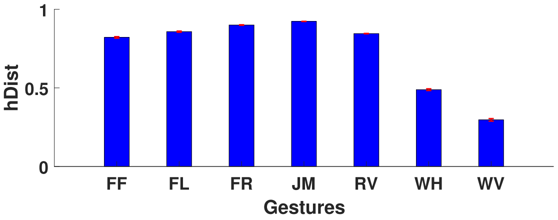


FIGURE 12. The comparison of the similarities of the same gesture (sampling rate = 250Hz, UP-rate = 4, ClusterNum = 10).

The results in Table 6 show that the similarities of the same gestures is higher than the similarities among different gestures for all the seven different types of gestures in our data set.

The similarity variances (Table 7) among different gestures are range from 0.01% to 0.48%, which show that *rRuler* has better stability and could extract specific features of different gestures effectively.

Figures 12 and Figure 13 show more details of the similarities among the gestures collected in Section VI-B when Sampling rate = $250Hz$, UP-rate = 4, ClusterNum = 10. Figure 12 shows the means and variances of the similarities for 7 gestures, 3 of them have a relative higher similarities which are 82%, 90% and 92% respectively; 2 of them reach 75% around; and another 2 gestures for "WH" and "WV" reach 38% and 29% respectively. Briefly, the similarities among the same gestures are all higher than those among different gestures (Detailed in Figure 13).

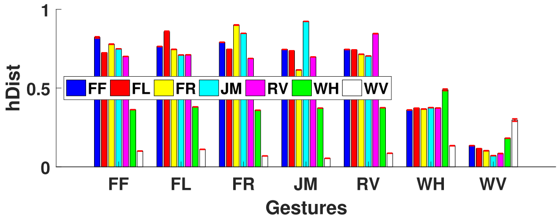


FIGURE 13. **The comparison of the similarities between different gestures (sampling rate = 250Hz, UP-rate = 4, ClusterNum = 10).**

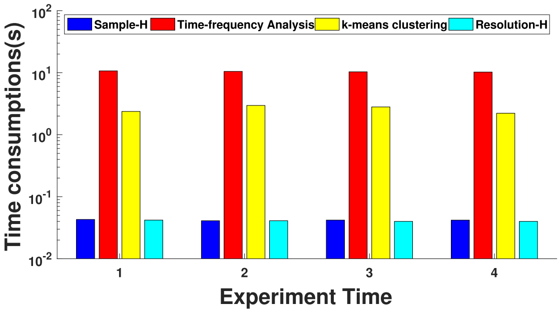


FIGURE 14. **Computational overhead of rRuler's modules.**

The similarities between the same gestures are not the only criterion to evaluate the performance of *rRuler*. The similarities between different gestures could fully reflect the performance of *rRuler*. Figure 13 show that the similarity of the gesture "Fall Forward" (FF) themselves is 82% among different samples, while the similarities of gesture "FF" compared with other six gestures are 76%, 79%, 74%, 36% and 13% respectively. Other gestures have the same trends with "FF" which are detailed in Figure 13.

Another important performance of *rRuler* is the computational overhead. This section uses the running time of Matlab-based simulator to evaluate the computational overhead of *rRuler*'s modules. The hardware and software configures of the Matlab-based simulator are as follows: i) CPU: Intel Core i7-4610M / 3.00GHz; ii) Memory: 8GB; and iii) Operating system: Windows 10 with 64-bit. The running time of each module under the above configures is shown in Figure 14.

Figure 14 shows the average running time of 4 times experiments with the same configurations. The statistical results show that the average execution time of Sample-H and Resolution-H is 0.043s and 0.040 s; the average execution time of Time-frequency analysis module is 11.70s and k-means based clustering module is 2.96s. Therefore, The overheads of Sample-H and Resolution-H module are extremely smaller than the traditional time-frequency analysis and k-means based clustering module, which demonstrate that the fine-grained feature extraction method proposed in this paper does not introduce much computational overheads.

2) PERFORMANCE WITH CLUSTER NUMBER

The analysis of the experimental results under specific parameters shown that *rRuler* has good reliability and stability. In order to further demonstrate the performances and their trends of *rRuler* under different parameter settings, we investigate the impact on *rRuler* under different cluster numbers (*ClusterNum*) in this section.

Figure 15 shows the trends of cross similarities among 7 gestures with cluster numbers (ClusterNum) of k-means. In order to save space, the coordinates "1", "2", "3", "4", "5", "6" and "7" of the X axis and Y axes in Figure 15 represent the gestures "FF", "FL", "FR", "JM", "RV", "WH", "WV" respectively. In addition, the red-like color represents a higher similarity while the blue-like color represents a lower similarity.

The results in Figure 15 show that it is difficult to identify the gestures from each other since the chaotic similarity patterns when the value of ClusterNum is small (for example, ClusterNum = 2). However, the differences between diagonal values and non-diagonal values in Figure 15 getting larger with the increasing of ClusterNum and achieves an acceptance results when ClusterNum = 10/12. The similarity patterns getting chaotic again when ClusterNum>12.

Figure 16 shows the execution overheads of *rRuler*'s modules with different ClusterNum values. As the ClusterNum increases, the execution time of the Sample-H, Resolution-H, and Time-frequency analysis modules are constant because the execution overheads of the Sample-H module are only related to the value of the UP-Rate and the length of the original RSS time series; The execution overheads of the Resolution-H module are only related to the length of the original RSS time series. The execution overheads of the Time-frequency analysis module are related to the length of the output of the Sample-H, Resolution-H and Time Feature Add modules. However, the overheads of k-means-based feature clustering module grows linearly with the increasing ClusterNum. Therefore, we should choose the appropriate value of ClusterNum to trade off the computational overheads and sensing accuracies in practical applications.

3) FEATURE EXTRACTION PERFORMANCE WITH UPSAMPLING RATE

Figure 17 shows the changing similarities of cross-gestures with the increasing UP-Rate.The similarities among the same gestures increase gradually with the increasing UP-Rate value when UP-Rate $\leq$ 4, and the similarities keep stable when UP-Rate>4. This is because the up-sampling module is to make up the loss of information generated by k-means dimensionality reduction method. Therefore, the similarities among the same gestures would not continuously increase if the up-sample fold could make up the information loss.

Figure 18 shows the execution overheads of the sample-H, Resolution-H, Time-frequency analysis, k-means clustering modules increase linearly with the increasing UP-Rate values. The reason attributes to the increase of the RSS series data, therefore, we should carefully select the UP-Rate value to trade off the computational overheads and sensing accuracy in practical applications.

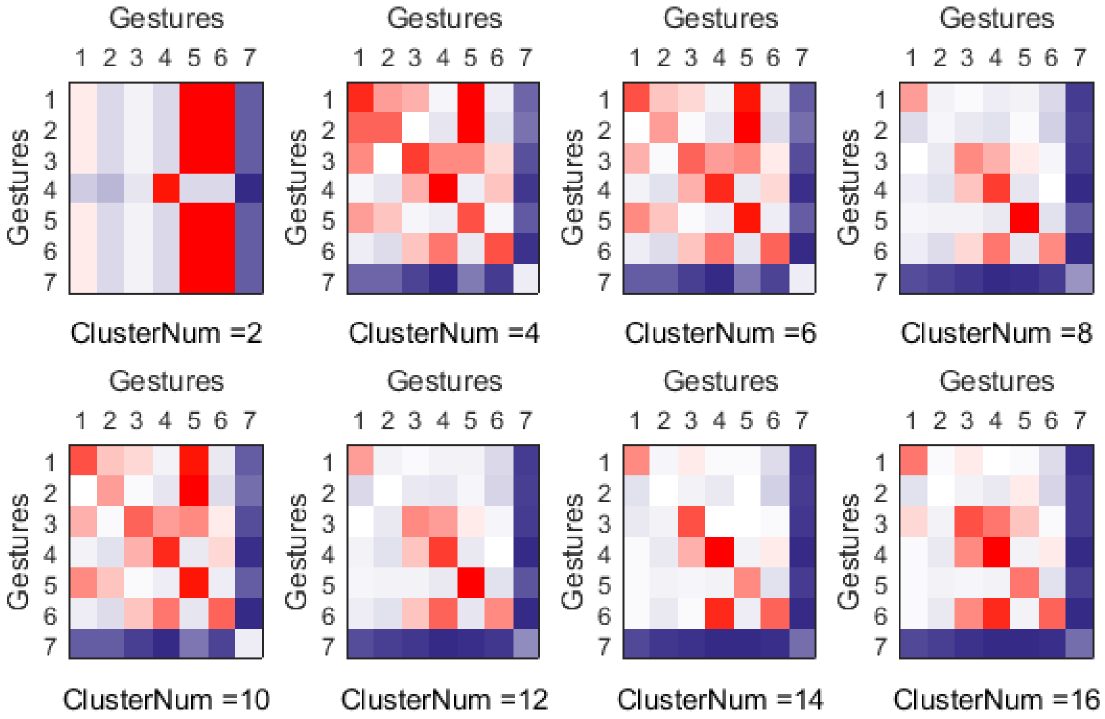


FIGURE 15. The trends of the similarities among gestures with ClusterNum (sampling rate = 250Hz, UP-rate = 4).

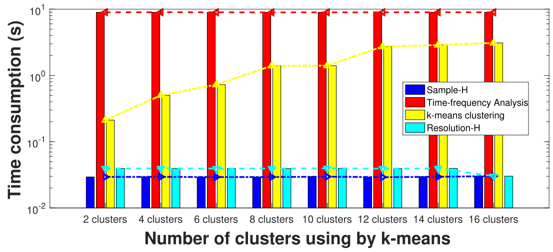


FIGURE 16. The trends of computational overheads with ClusterNum (sampling rate = 250Hz, UP-rate = 4).

4) COMPARISON WITH OTHER FEATURE EXTRACTION METHODS

In order to further improve the performance of *rRuler*, this section make some comparisons between our method and current works which without sample-H, Resolution-H, gTime-ext modules respectively, and the settings and performances of them are shown in Figure 19, and the results show that *rRuler* is more accurate than the other three methods, because the similarities of same gestures extracted by *rRuler* are obviously larger than those extracted by other three methods. Figure 20 shows that if there is no Sample-H module, "FL" is difficult to perceive since it mixed with other gestures together; if there is no Resolution-H module, "FF", "FL", "WH" and "FL" are judged to be "RV", and "WH" could also not be perceived. If there is no gTime-ext module, "FL" is judged to be "RV", "WH" can not be perceived correctly. Figure 20 is a detailed view of Figure 19 which would help the reader better study the performance comparisons between *rRuler* and other current methods.

E. THE ACCURACY OF GESTURE SENSING

According to the results of Section VI-D, when UP-Rate = 4 and ClusterNum = 12, the features of the same gestures are obviously different from those of the different gestures. In order to further prove the performance of *rRuler* proposed in Section IV. This section defines and implements *rRuler-HMM* to calculate the sensing accuracy base on the dataset collected in Section VI-B.

The experimental results in the remainder of this section are under the parameter settings defined in Table 8 in both LoS and NLoS Scenarios. As shown in Table 8, the window size of the fast Fourier transform in *rRuler* is 128, the overlaps among windows are 120, the frequency range is $1 - 200Hz$ (according to the results of Section VI-D, the frequency of gestures in our dataset is up to $200Hz$).

In order to verify the relationship between sensing accuracy and RSS sampling rate. *rRuler-HMM* collects more data in LoS and NLoS scenarios defined in Sections VI-B with a changing sampling rate of 250Hz, 500Hz, and 1000Hz respectively. We also collects more data from different channels (ZigBee Channel 15, 20 and 26) to perceive the relationship between sensing accuracy and different center frequencies.

In order to demonstrate the performance of *rRuler-HMM*, we show the sensing accuracies in both LoS and NLoS scenarios in Figure 21 with parameter settings: UP-Rate = 4;

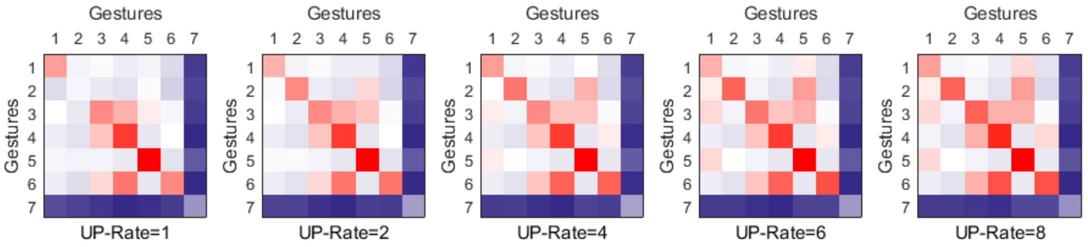


FIGURE 17. The trends of the similarities with up-sampling rate (sampling rate = 250Hz, ClusterNum = 12).

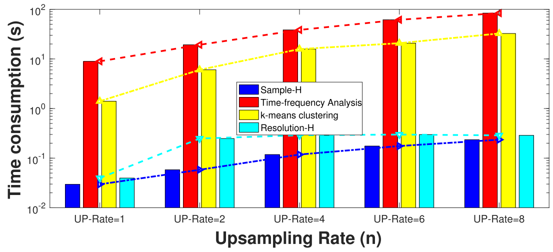


FIGURE 18. The changing trends of the computational overhead with up-sampling rate (sampling rate = 250Hz, ClusterNum = 12).

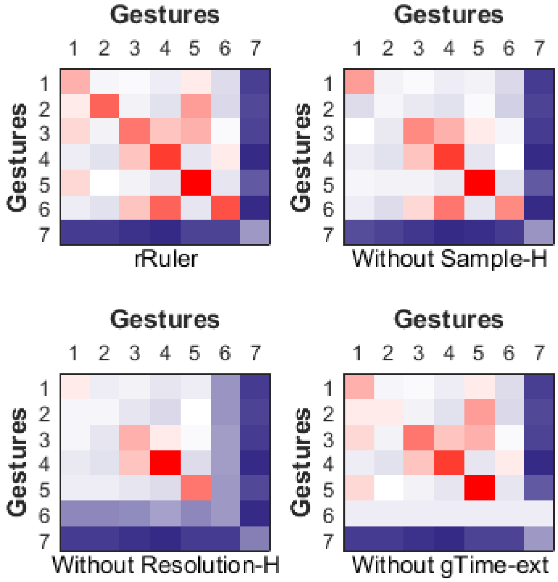


FIGURE 19. The comparison of *rRuler* and current methods (sampling rate = 250Hz, ClusterNum = 12, UP-Rate = 4).

ClusterNum = 12; STFT's parameters are set according to Table 8, sampling rate = 250*Hz*, and the data acquisition channel is ZigBee channel 15.

As shown in Figure 21, the accuracy of the *rRuler-HMM*-based human gesture sensing algorithm in LoS could reach an average value of 97.14% in LoS scenario and 95.71% in NLoS scenario respectively in above settings. The confusion matrix of the sensing results of LoS scenario and NLoS scenario are shown as Table 9 and Table 10.

As shown in Table 9 and Table 10, the sensing accuracies have the same trends in both LoS and NLoS scenarios.

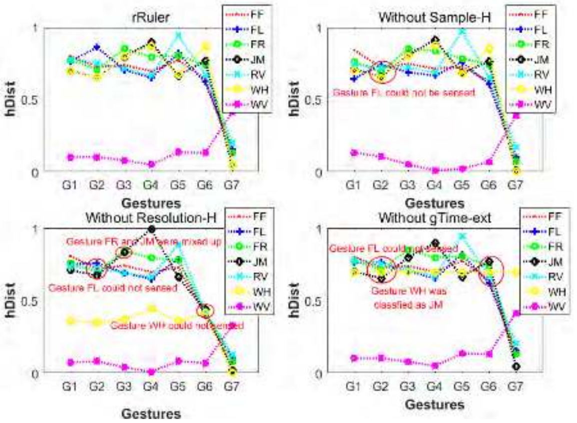


FIGURE 20. The feature extraction details of the datasets(sampling rate = 250Hz, ClusterNum = 12, UP-Rate = 4).

TABLE 8. Parameter settings *rRuler-HMM*.

Parameter Name		Parameter settings
data set		data collected from section X
the sampling rate of RSS		250Hz 500Hz 1000Hz
Channels		ZigBee channel 15 20 26
	the size of windows	128
STFT parameters	the size of overlaps	120
	frequency range	1-200Hz

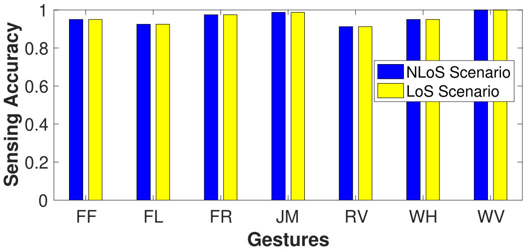


FIGURE 21. Sensing accuracy of *rRuler-HMM* (Sampling rate = 250Hz, UP-rate = 4, ClusterNum = 12, UP-rate × sampling rate = 1000Hz, ZigBee channel:15).

Gesture "FF" has a certain probability which is wrongly judged as "FR" and "JM". The reason is "FF", "FR" and "JM" are similar in frequency; Gesture "FL" has a certain probability of being erroneously judged as "FR" and "JM"; "JM" has a certain probability that wrongly judged as "FR" and "JM" "WH" has a certain probability of being erroneously judged as "FF"; "WH" has a certain probability wrongly judged as "FL"; the sensing accuracy pf "WH"

TABLE 9. Confusion matrix in LoS scenario (sampling rate = 250, UP-rate = 4, ClusterNum = 12 UP-rate × sampling rate = 1000, ZigBee channel:15).

	FF	FL	FR	JM	RV	WH	WV
FF	0.975	0	0.0125	0.0125	0	0	0
FL	0	0.95	0.0125	0.0125	0.025	0	0
FR	0	0	0.9875	0.0125	0	0	0
JM	0	0	0.0125	0.9875	0	0	0
RV	0.025	0.0025	0	0.0125	0.9375	0	0
WH	0	0.025	0	0	0	0.975	0
WV	0	0	0	0	0	0	1

TABLE 10. Confusion matrix in NLoS scenario (sampling rate = 250, UP-Rate = 4, ClusterNum = 12, UP-Rate × Sampling Rate = 1000, ZigBee channel:15).

	FF	FL	FR	JM	RV	WH	WV
FF	0.95	0	0.025	0.0125	0.0125	0	0
FL	0	0.925	0.0375	0.0125	0.025	0	0
FR	0.0125	0	0.975	0.0125	0	0	0
JM	0	0	0.0125	0.9875	0	0	0
RV	0.025	0.05	0	0.0125	0.9125	0	0
WH	0	0.05	0	0	0	0.95	0
WV	0	0	0	0	0	0	1

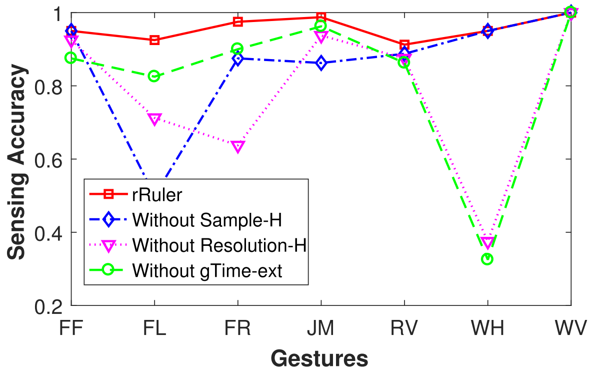


FIGURE 22. Contribution analysis of each modules in *rRuler* (Sampling Rate = 250, UP-Rate = 4, ClusterNum = 12, UP-Rate × Sampling Rate = 1000, ZigBee channel:15).

is 100%, the main reason is that the frequency of walking is lower and the directions are different from other gestures significantly.

In order to further analyze the contributions of each module proposed in *rRuler*, we take the data collected in NLoS scenario as example, the results are shown in Figure 22 and detailed in Table 11, Table 12, Table 13 respectively.

Figure 22 combines Table 11 show the contributions of Sample-H module, the results demonstrate that the sensing accuracy of gesture “FL” decrease 42% without Sample-H module due to the large loss of information in STFT phase; the accuracies of other gestures decrease 20%, 0%, 10%, 12.5% and 6.25% respectively.

Figure 22 combines Table 12 show the contributions of Resolution-H module, the results demonstrate that the sensing accuracies of gesture “FF”, “FL”, “FR”, “JM”, “RV”, “WH”, and “WV” decrease by 2.5%, 21.25%, 33.75%, 0%, 0%, 57.5%, 0% without Resolution-H module. The confusion matrix shown in Table 12 demonstrates gesture “FL” tends to be confused as gesture “FR”; gesture “FR” tends to be confused as “JM”; gesture “WH” tends to be confused as gesture “FF”, “FR”. The main reason is that the resolution of RSS is 1dB, therefore, the details of some gestures are mixed with each other, which results in reducing of the sensing accuracy.

TABLE 11. Confusion matrix (without sample-H module).

	FF	FL	FR	JM	RV	WH	WV
FF	0.95	0	0.025	0.0125	0.0125	0	0
FL	0.2	0.425	0.175	0.0875	0.0375	0	0
FR	0.0125	0	0.875	0.0625	0.05	0	0
JM	0	0	0.0375	0.8625	0	0.1	0
RV	0	0	0.025	0	0.975	0	0
WH	0	0	0.0125	0.0375	0	0.95	0
WV	0	0	0	0	0	0	1

TABLE 12. Confusion matrix (without resolution-H).

	FF	FL	FR	JM	RV	WH	WV
FF	0.925	0	0.0375	0.0125	0.025	0	0
FL	0.05	0.7125	0.1125	0.0375	0.0875	0	0
FR	0	0	0.6375	0.3625	0	0	0
JM	0	0	0.0125	0.9875	0	0	0
RV	0.025	0.05	0	0.0125	0.9125	0	0
WH	0.1	0.0625	0.15	0.1375	0.175	0.375	0
WV	0	0	0	0	0	0	1

TABLE 13. Confusion matrix (without gTime-ext).

	FF	FL	FR	JM	RV	WH	WV
FF	0.875	0	0.0625	0.0375	0.025	0	0
FL	0.0375	0.825	0.0375	0.025	0.075	0	0
FR	0.0125	0	0.9	0.0875	0	0	0
JM	0	0	0.0375	0.9625	0	0	0
RV	0	0.025	0.0125	0.0125	0.95	0	0
WH	0	0.05	0.0375	0.5875	0	0.325	0
WV	0	0	0	0	0	0	1

Figure 22 combines Table 13 show the contributions of gTime-ext module, the results demonstrate that 58% of gesture “WH” are confused as gesture “JM” since they occurs at the same location with similar frequencies. However, gTime-ext module introduce time domain features which could increase the sensing accuracies of those gesture with similar frequencies.

In summary, each module of *rRuler* makes specific contributions to increase sensing accuracies of the gestures in our dataset.

F. THE INTERACTIONS BETWEEN GESTURE SENSING APPLICATIONS AND ORIGINAL COMMUNICATION NETWORKS

This section validates the interactions between gesture sensing applications and the original communication networks. The goals are to validate: i) whether the gesture sensing accuracy affected by different communication channels? and ii) Does the gesture sensing applications affect the performance of existing communication networks (such as the Packet Reception Ratio?). Figure 23 shows the gesture sensing accuracy fluctuations with different communication channels (ZigBee channel 15, 20 and 26) are around

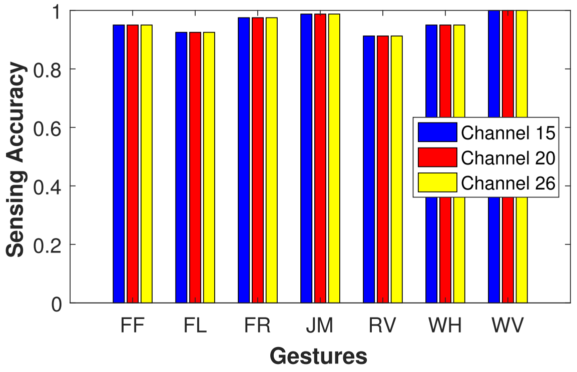


FIGURE 23. Gesture sensing accuracies under different communication channels.

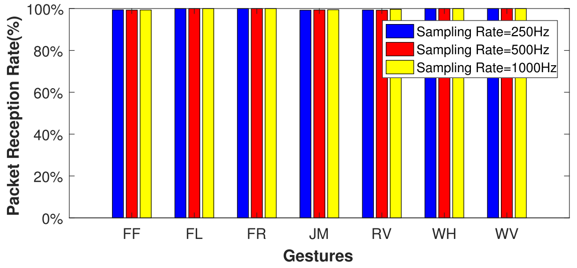


FIGURE 24. Packet reception rates under the environment with different gestures.

1-2%. Therefore, different communication channels have less influence on the accuracy of gesture sensing.

Figure 24 shows that the packet loss rate is below 1% under different gestures, and actually the packet loss rate is also due to other disturbances and noise factors in the propagation environment. Therefore, gesture sensing applications have little effect on the packet loss rate of the communication system. Therefore, we could conduct many applications such as human machine interaction / behavior monitoring without effecting the existing communication networks.

VII. CONCLUSION

RF-based human gesture sensing is an emerging technology in sensing systems. The basic of RF-based human gesture sensing is multi-path fading theory. In this paper, we aim to increase sensing accuracy of human gestures with coarse-grained RF measurement (RSS). The main contributions include: i) proposed a fine-grained human gesture feature extraction model (*rRuler*), which includes RSS sampling enhancement algorithm (Sample-H algorithm), RSS resolution enhancement algorithm (Resolution-H algorithm), gesture duration extraction algorithm (gTime-ext), frequency domain feature extraction algorithm and k-means based dimensionality reduction method; and ii) in order to further verify the accuracy of *rRuler*, this paper presents the *rRuler-HMM* for sensing accuracy evaluations. The experimental results show that the average accuracies of *rRuler-HMM* are around 95.71% in NLoS scenario and 97.14% in LoS scenario respectively, which means human gesture sensing technologies with coarse-grained RF measurements proposed in this paper are practical and could be widely adapted in various IoT technologies, such as ZigBee, Bluetooth/BLE, WiFi, cellular networks and other potential wireless protocols.

ACKNOWLEDGMENTS

This project is supported by CERNET Innovation Project NGII20180315. We also thank anonymous reviewers and their valuable comments.

REFERENCES

[1] Q. Pu, S. Gupta, S. Gollakota, and S. Patel, "Whole-home gesture recognition using wireless signals," in *Proc. 19th Annu. Int. Conf. Mobile Comput. Netw.* , Sep.Oct. 2013, pp. 27–38.

[2] B. Lyonnet, C. Ioana, and M. G. Amin, "Human gait classification using microDoppler time-frequency signal representations," in *Proc. IEEE Radar Conf.*, May 2010, pp. 915–919.

[3] P. Van Dorp and F. C. A. Groen, "Feature-based human motion parameter estimation with radar," *IET Radar, Sonar Navigat.*, vol. 2, no. 2, pp. 135–145, 2008.

[4] C. Li, J. Ling, J. Li, and J. Lin, "Accurate Doppler radar noncontact vital sign detection using the RELAX algorithm," *IEEE Trans. Instrum. Meas.*, vol. 59, no. 3, pp. 687–695, Mar. 2010.

[5] W. He, K. Wu, Y. Zou, and Z. Ming, "WiG: WiFi-based gesture recognition system," in *Proc. 24th Int. Conf. Comput. Commun. Netw. (ICCCN)*, Aug. 2015, pp. 1–7.

[6] C. Han, K. Wu, Y. Wang, and L. M. Ni, "WiFall: Device-free fall detection by wireless networks," in *Proc. IEEE INFOCOM*, Apr./May 2014, pp. 271–279.

[7] B. Kellogg, V. Talla, and S. Gollakota, "Bringing gesture recognition to all devices," in *Proc. NSDI*, Apr. 2014, pp. 303–316.

[8] W. Wang, A. X. Liu, M. Shahzad, K. Ling, and S. Lu, "Understanding and modeling of WiFi signal based human activity recognition," in *Proc. 21st Annu. Int. Conf. Mobile Comput. Netw.*, Sep. 2015, pp. 65–76.

[9] 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.

[10] Y. Wang, X. Jiang, R. Cao, and X. Wang, "Robust indoor human activity recognition using wireless signals," *Sensors*, vol. 15, no. 7, pp. 17195–17208, 2015.

[11] C. Wu, Z. Yang, Z. Zhou, X. Liu, Y. Liu, and J. Cao, "Non-invasive detection of moving and stationary human with WiFi," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 11, pp. 2329–2342, Nov. 2015.

[12] W. Xi, J. Zhao, X.-Y. Li, K. Zhao, S. Tang, X. Liu, and Z. Jiang, "Electronic frog eye: Counting crowd using WiFi," in *Proc. INFOCOM*, Apr./May 2014, pp. 361–369.

[13] D. Zhang, H. Wang, Y. Wang, and J. Ma, "Anti-fall: A non-intrusive and real-time fall detector leveraging CSI from commodity WiFi devices," in *Proc. Int. Conf. Smart Homes Health Telematics*, vol. 9102, May 2015, pp. 181–193.

[14] Z. Zhou, Z. Yang, C. Wu, Y. Liu, and L. M. Ni, "On multipath link characterization and adaptation for device-free human detection," in *Proc. IEEE 35th Int. Conf. Distrib. Comput. Syst.*, Jun./Jul. 2015, pp. 389–398.

[15] L. Wang, K. Sun, H. Dai, A. X. Liu, and X. Wang, "WiTrace: Centimeter-level passive gesture tracking using WiFi signals," in *Proc. 15th Annu. IEEE Int. Conf. Sens., Commun., Netw. (SECON)*, Jun. 2018, pp. 1–9.

[16] J. Zhang and Z. Wang, "CrossSense: Towards cross-site and large-scale WiFi sensing," in *Proc. 24th Annu. Int. Conf. Mobile Comput. Netw.*, Oct./Nov. 2018, pp. 305–320.

[17] M. Raja and S. Sigg, "RFexpress!—Exploiting the wireless network edge for RF-based emotion sensing," in *Proc. 22nd IEEE Int. Conf. Emerg. Technol. Factory Automat. (ETFA)*, Sep. 2017, pp. 1–8.

[18] D. Zhang, H. Wang, and D. Wu, "Toward centimeter-scale human activity sensing with Wi-Fi signals," *Computer*, vol. 50, no. 1, pp. 48–57, Jan. 2017.

[19] 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, p. 72, Sep. 2017.

[20] S. Yue, H. He, H. Wang, H. Rahul, and D. Katabi, "Extracting multi-person respiration from entangled RF signals," *Proc. ACM Interact., Mobile, Wearable Ubiquitous Technol.*, vol. 2, no. 2, p. 86, Jun. 2018.

[21] C. Gaom, X. Zhang, and Y. Li, ''Sensing human-object interaction through passive chipless WiFi tags,'' in *Proc. NSDI*, Apr. 2018, pp. 533–546.

[22] 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, p. 53, Jan. 2011.

[23] Y. Xie, Z. Li, and M. Li, ''Precise power delay profiling with commodity Wi-Fi,'' in *Proc. 21st Annu. Int. Conf. Mobile Comput. Netw.*, Sep. 2015, pp. 53–64.

[24] S. Sigg, U. Blanke, and G. Tröster, ''The telepathic phone: Frictionless activity recognition from WiFi-RSSI,'' in *Proc. PerCom*, Mar. 2014, pp. 148–155.

[25] S. Sigg, S. Shi, F. Buesching, Y. Ji, and L. Wolf, ''Leveraging RF-channel fluctuation for activity recognition: Active and passive systems, continuous and RSSI-based signal features,'' in *Proc. MoMM*, Dec. 2013, p. 43.

[26] Z. Chi, Y. Yao, T. Xie, Z. Huang, M. Hammond, and T. Zhu, ''Harmony: Exploiting coarse-grained received signal strength from IoT devices for human activity recognition,'' in *Proc. ICNP*, Nov. 2016, pp. 1–10.

[27] Z. Chi, Y. Yao, T. Xie, X. Liu, Z. Huang, W. Wang, and T. Zhu, ''EAR: Exploiting uncontrollable ambient RF signals in heterogeneous networks for Gesture recognition,'' in *Proc. 16th ACM Conf. Embedded Netw. Sensor Syst.*, Nov. 2018, pp. 237–249.

[28] L. Sun, S. Sen, D. Koutsonikolas, and K.-H. Kim, ''WiDraw: Enabling hands-free drawing in the air on commodity WiFi devices,'' in *Proc. 21st Annu. Int. Conf. Mobile Comput. Netw.*, New York, NY, USA, Sep. 2015, pp. 77–89.

[29] K. Ali, A. X. Liu, W. Wang, and M. Shahzad, ''Keystroke recognition using WiFi signals,'' in *Proc. MobiCom*, Sep. 2015, pp. 90–102.

[30] J. Liu, Y. Wang, Y. Chen, J. Yang, X. Chen, and J. Cheng, ''Tracking vital signs during sleep leveraging Off-the-shelf WiFi,'' in *Proc. 16th ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, Jun. 2015, pp. 267–276.

[31] S. Tan and J. Yang, ''WiFinger: Leveraging commodity WiFi for fine-grained finger gesture recognition,'' in *Proc. MobiHoc*, Jul. 2016, pp. 201–210.

[32] G. Wang, Y. Zou, Z. Zhou, K. Wu, and L. M. Ni, ''We can hear you with Wi-Fi!'' in *Proc. MobiCom*, Sep. 2014, pp. 593–604.

[33] J. Xiao, K. Wu, Y. Yi, L. Wang, and L. M. Ni, ''Fimd: Fine-grained device-free motion detection,'' in *Proc. ICDCS*, Dec. 2012, pp. 229–235.

[34] Y. Zeng, P. H. Pathak, C. Xu, and P. Mohapatra, ''Your AP knows how you move: Fine-grained device motion recognition through WiFi,'' in *Proc. 1st ACM Workshop Hot Topics Wireless*, Sep. 2014, pp. 49–54.

[35] Z. Zhou, Z. Yang, C. Wu, L. Shangguan, and Y. Liu, ''Towards omnidirectional passive human detection,'' in *Proc. IEEE INFOCOM*, Apr. 2013, pp. 3057–3065.

[36] Y.-C. Chen, L. Qiu, G. Xue, Z. Hu, and Y. Zhang, ''Robust network compressive sensing,'' in *Proc. 20th Annu. Int. Conf. Mobile Comput. Netw.*, Sep. 2014, pp. 545–556.

[37] N. Yu, W. Wang, A. X. Liu, and L. Kong, ''QGesture: Quantifying Gesture distance and direction with WiFi signals,'' *Proc. ACM Interact., Mobile, Wearable Ubiquitous Technol.*, vol. 2, no. 1, p. 51, Mar. 2018.

[38] K. Ling, H. Dai, Y. Liu, and A. X. Liu, ''UltraGesture: Fine-grained Gesture sensing and recognition,'' in *Proc. 15th Annu. IEEE Int. Conf. Sens., Commun., Netw.*, Jun. 2018, pp. 1–9.

[39] Y. Tian, G.-H. Lee, H. He, C.-Y. Hsu, and D. Katabi, ''RF-based fall monitoring using convolutional neural networks,'' *Proc. ACM Interact., Mobile, Wearable Ubiquitous Technol.*, vol. 2, no. 3, p. 137, Sep. 2018.

[40] D. Vasisht, A. Jain, C.-Y. Hsu, Z. Kabelac, and D. Katabi, ''Duet: Estimating user position and identity in smart homes using intermittent and incomplete RF-data,'' *Proc. ACM Interact., Mobile, Wearable Ubiquitous Technol.*, vol. 2, no. 2, p. 84, Jun. 2018.

[41] H. Abdelnasser, M. Youssef, and K. A. Harras, ''WiGest: A ubiquitous WiFi-based gesture recognition system,'' in *Proc. IEEE INFOCOM*, Apr./May 2015, pp. 1472–1480.

[42] S. Sigg, M. Scholz, S. Shi, Y. Ji, and M. Beigl, ''RF-sensing of activities from non-cooperative subjects in device-free recognition systems using ambient and local signals,'' *IEEE Trans. Mobile Comput.*, vol. 13, no. 4, pp. 907–920, Apr. 2014.

[43] F. Adib, Z. Kabelac, D. Katabi, and R. C. Miller, ''3D tracking via body radio reflections,'' in *Proc. NSDI*, 2014, pp. 317–329.

[44] F. Adib, Z. Kabelac, and D. Katabi, ''Multi-person localization via RF body reflections,'' in *Proc. NSDI*, May 2015, pp. 279–292.

[45] H. Khaloozadeh and A. Karsaz, ''Modified input estimation technique for tracking manoeuvring targets,'' *IET Radar, Sonar Navigat.*, vol. 3, no. 1, pp. 30–41, 2009.

[46] D. Vasisht, G. Zhang, O. Abari, H.-M. Lu, J. Flanz, and D. Katabi, ''In-body backscatter communication and localization,'' in *Proc. Conf. ACM Special Interest Group Data Commun.*, New York, NY, USA, Aug. 2018, pp. 132–146.

[47] C.-Y. Hsu, A. Ahuja, S. Yue, R. Hristov, Z. Kabelac, and D. Katabi, ''Zero-effort in-home sleep and insomnia monitoring using radio signals,'' *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, vol. 1, no. 3, p. 59, Sep. 2017. doi: 10.1145/3130924.

[48] K. Sun, W. Wang, A. X. Liu, and H. Dai, ''Depth aware finger tapping on virtual displays,'' in *Proc. 16th Annu. Int. Conf. Mobile Syst., Appl., Services*, Jun. 2018, pp. 283–295.

[49] Y. Sang, L. Shi, and Y. Liu, ''Micro hand Gesture recognition system using ultrasonic active sensing,'' *IEEE Access*, vol. 6, pp. 49339–49347, 2018.

[50] Z. Yang, Z. Zhou, and Y. Liu, ''From RSSI to CSI: Indoor localization via channel response,'' *ACM Comput. Surv. (CSUR)*, vol. 46, no. 2, p. 25, Nov. 2013.

HONGYU SUN received the Ph.D. from Jilin University, Changchun, China, in 2017. From 2015 to 2016, she was a Visiting Scholar with University of Maryland, Baltimore County. She is currently an Assistant Professor with Jilin Normal University. Her research interests include the Internet of Things (IoT), RF-based vision, wireless communications, and mobile computing. She has published over 20 articles in SCI/EI international conference proceedings and journals.

ZHENG LU received the master degree in computer science and technology form Jilin University, Changchun, China, in 2014. From 2015 to 2016, he is a Visiting Scholar with the University of Maryland, Baltimore County. From 2008 to 2015, he was a Senior Engineer with China Unicom Company, Ltd. From 2016 to 2019, he is currently an Assistant Professor with Jilin Normal University. His research interests include machine learning, network optimization, 5G, MIMO, and mobile computing.

CHIN-LING CHEN received the Ph.D. from National Chung Hsing University, Taiwan, in 2005. From 1979 to 2005, he was a Senior Engineer with Chunghwa Telecom Company, Ltd. He is a Professor. His research interests include cryptography, network security, and electronic commerce. He has published over 90 articles in SCI/SSCI international journals.

JIE CAO received the Ph.D. degree in computer science and technology from Jilin University, Changchun, China, in 2017. She is currently an Associate Professor and also a Master's Tutor with School of Computer Science, Northeast Electric Power University. Her research interests include computer network, machine learning, and power grid stability and security.

ZHENGJIANG TAN received the Ph.D. degree from Chinese Academy of Sciences, Changchun Institute of Optics, Fine Mechanics and Physics, in 2003. He is currently a Professor with Jilin Normal University. His research interests include network security, privacy protection, and network applications. He has published over 30 articles in SCI/EI international conference proceedings and journals.