

# Device-free Human Activity Recognition Using Commercial WiFi Devices

Wei Wang   Alex X. Liu   Muhammad Shahzad   Kang Ling   Sanglu Lu

**Abstract**—Since human bodies are good reflectors of wireless signals, human activities can be recognized by monitoring changes in WiFi signals. However, existing WiFi based human activity recognition systems do not build models that can quantify the correlation between WiFi signal dynamics and human activities. In this paper, we propose CARM, a Channel State Information (CSI) based human Activity Recognition and Monitoring system. CARM is based on two theoretical models. First, we propose a CSI-speed model that quantifies the relation between CSI dynamics and human movement speeds. Second, we propose a CSI-activity model that quantifies the relation between human movement speeds and human activities. Based on these two models, we implemented CARM on commercial WiFi devices. Our experimental results show that CARM achieves recognition accuracy of 96% and is robust to environmental changes.

**Index Terms**—Device-free sensing, Human activities, WiFi, CSI

## I. INTRODUCTION

Recently, multiple WiFi-signal based human activity recognition systems have been proposed [18], [26], [28]. The key observation of these systems is that human bodies, which are mostly made of water, can reflect WiFi signals. Therefore, human activities introduce changes in the signal received by a nearby WiFi device. In consequence, we can recognize human activities by monitoring variations in the pervasive WiFi signals surrounding us. Compared to traditional human activity recognition systems, which use cameras [2], [6], radars [17], or wearable devices [8], [35], WiFi signal based systems have the following advantages. First, WiFi based solutions are device-free and the activity can be detected solely through the wireless signal reflected by the human body. Therefore, the targeting human users do not need to wear any sensors, which are inconvenient in many scenarios (*e.g.*, for elder care) and impossible for some scenarios (*e.g.*, for intrusion detection). Second, WiFi based solutions have better coverage than cameras. WiFi signals can penetrate walls, furniture, and doors, while cameras have limited viewing angles and require good lighting conditions. Furthermore, the recently developed 60 GHz radar solutions also have limited coverage of less than

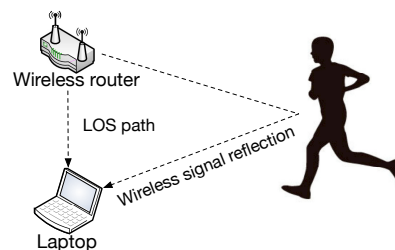


Figure 1. System structure of CARM

one meter due to their shorter wavelength [17]. Third, camera-based solutions have potential privacy issues because cameras unnecessarily capture other sensitive information, such as the face of the user, while performing activity recognition.

Existing WiFi based human activity recognition systems measure WiFi signal variations using the Channel State Information (CSI). However, these systems treat CSI based activity recognition as a black box. They use machine learning schemes to discover statistical CSI patterns related to specific human activities instead of building models to quantify CSI patterns. The lack of quantitative models that can connect the CSI dynamics and human activities limits the understanding on signal dynamics and further development of WiFi based human activity recognition technology. For example, without a good understanding of how human activities change the WiFi signal, it is difficult to separate useful signal changes from the random noises. Furthermore, without a guiding model, most activity recognition systems resort to trial-and-error methods when trying to optimize the system performance.

In this paper, we build a CSI based human Activity Recognition and Monitoring system (CARM) that is based on theoretical models of WiFi signal dynamics and human activities. Our system uses the WiFi link between Commercial-Off-The-Shelf (COTS) devices, as shown in Figure 1, to detect human activities. Consider the case where the transmitter (*e.g.*, the wireless router in Figure 1) is continuously sending WiFi frames to the receiver (*e.g.*, the laptop in Figure 1). When there are humans moving around, the WiFi signal reflected by the human body will interfere with the signals travelled through the Line-Of-Sight (LOS) path. By monitoring the wireless channel state, the receiver can measure small signal changes caused by human movements and use these changes to recognize the surrounding human activities.

The key contributions of CARM are the two models proposed in this paper: *CSI-speed model* and *CSI-activity model*. The CSI-speed model describes how the frequencies of CSI power variations are related to the human movement speeds.

Wei Wang, Alex X. Liu, Kang Ling, and Sanglu Lu are with the State Key Laboratory for Novel Software Technology, Nanjing University, China, (Email: ww@nju.edu.cn, alexliu@nju.edu.cn, lingkang@mail.nju.edu.cn, sanglu@nju.edu.cn).

Muhammad Shahzad is with the Department of Computer Science, North Carolina State University, Raleigh, NC 27695, USA, (Email: mshahza@ncsu.edu).

Alex X. Liu is the corresponding author of this paper.

The preliminary version of this paper was published in the Proceedings of ACM MobiCom, September, 2015 [27].

The CSI-activity model describes how the movement speeds of different human body parts are related to a specific human activity. Based on these models, we quantitatively connect CSI dynamics and human activities.

The model-based approach adopted by CARM has two advantages. First, our model is accurate enough to provide quantitative movement features using CSI measured by commercial WiFi devices. For example, based on our CSI-speed model, we can measure the movement distance with an accuracy of less than 3 cm. Such precise movement model gives us the capability to quantitatively measure human activities, which is important to many applications, including human activity recognition. Second, our model can guide the design of activity recognition systems and leads to new ways for removing interfering noises. Using the CSI-speed model and the CSI-activity mode, we discover that typical human activities only introduce CSI variations with frequency of lower than 300 Hz. Thus, we can safely remove high-frequency noises in the received signal. Furthermore, using our model, we find that the movement speed information can be fully captured by the amplitude of CSI so that we can safely ignore the highly noisy phase information of CSI measurements.

We need to handle the following technical challenges when building CARM. The first key challenge is that CSI values are too noisy to be directly used for human activity recognition. Low cost commercial WiFi cards introduce various types of noises in CSI measurements. For example, differences in the oscillator of the transmitter/receiver hardware introduce Carrier Frequency Offset (CFO), which leads to random variations in the phase of CSI [16], [32]. The amplitude of the CSI values also fluctuates due to the surrounding electromagnetic noises and the power adaptation process of the WiFi card. These fluctuations have a higher amplitude than the small changes introduced by human movements. Therefore, it is hard to extract useful information from these noisy CSI measurements. To address this challenge, we first remove the impact of CFO by ignoring the CSI phase as suggested by our models. We then observe from our models that the CSI variations introduced by human movements in different OFDM subcarriers are correlated. This leads to our Principal Component Analysis (PCA) based CSI denoising scheme, which discovers the correlation between OFDM subcarriers and amplifies the correlated changes using data from multiple OFDM subcarriers. Our denoising scheme is efficient so that we can reliably detect small CSI changes caused by a walking human at a distance of 12 meters.

The second challenge is to build a robust model that can capture the characteristics of human activities. As different people perform the same activity in slightly different ways, we need to discover a robust human activity model that can be used for different users. To address this challenge, we use the Hidden Markov Model (HMM) to describe the speed features of human activities. We choose the HMM model because of its capability to recognize the same activities that are performed in slightly different manners.

The third challenge is to build a system that is robust to environmental changes. As different indoor environments introduce different types of multipath conditions, the CSI

patterns measured by a single link could be distorted when the transmitter/receiver are placed at certain positions. To address this challenge, we propose to perform data fusion on multiple WiFi links to provide better robustness under different environments. Our results show that CARM achieves an average of 8% improvement in recognition accuracy when we combine the data from three WiFi links.

We implemented CARM on commercial WiFi devices, *e.g.*, WiFi routers and laptops equipped with Intel 5300 WiFi card. We built a CSI activity database with 1,400 samples for 8 different activities from 25 volunteers. CARM achieves an average activity recognition accuracy of 96% in ten-fold cross-validation on our CSI activity database. For a new environment and a new person that the system has never been trained on, CARM can still achieve a recognition accuracy of more than 85% when using data fusion on three links.

## II. RELATED WORK

Existing activity recognition systems can be classified into four categories: RSSI based, specialized hardware based, Radar based, and CSI based.

**RSSI Based:** Received Signal Strength Indicator (RSSI) based human activity recognition systems rely on the received signal strength changes caused by human activities [3], [22], [23]. However, as the RSSI is reported in a coarse granularity of 1 dB [23], RSSI based systems have limited accuracy. Although it is possible to improve the recognition accuracy to 72% by using software radios to capture RSSI with a higher resolution [22], the accuracy and coverage of RSSI-based systems are still much lower than CARM due to the lack of the frequency diversity that CSI provides.

**Specialized Hardware Based:** Software defined radios or specially designed hardware can report fine-grained signal measurements [12], [13], [15], [18]. For example, WiSee uses USRP to measure the Doppler shift in wireless signals and achieve an activity recognition accuracy of 95% [18]. Allsee proposes a short range (less than 2.5 feet) solution for gesture recognition by using a special low-power circuit to extract the envelope of the received signal [15]. Wision explores the possibility of reconstructing an image of the target using the wireless signal received by multiple antennas [12].

**Radar Based:** Radar can also be used for human activity recognition [4], [5], [25]. With the much higher bandwidth, *e.g.*, Frequency Modulated Carrier Wave (FMCW) radar can take up to 1.79 GHz bandwidth [4] while WiFi usually only uses 20 MHz bandwidth. Radar-based systems can extract the micro-Doppler information [25] and have higher distance resolution of approximately 20 cm [4], [5]. However, both the Radar-based and specialized hardware-based systems require specific hardware, while CARM runs on COTS WiFi devices.

**CSI Based:** Recently, CSI values provided by WiFi network interface cards (NICs) [10], [32] are widely used for human activity recognition [7], [28], [31]. Researchers have developed CSI based applications, including fall detection [11], presence detection [38], indoor localization [21], [34], and human crowd counting [31]. It is shown that CSI is also able to detect and recognize small movements such as lip movements

[26], keystrokes [7], and heartbeats [36]. To prevent potential privacy breaches, Qiao *et al.* use amplify-and-forward relays to obfuscate the CSI values [19]. Compared to the above CSI-based applications, CARM proposes a model-based approach which is more robust to environmental changes.

### III. CARM OVERVIEW

#### A. Channel State Information

Modern WiFi NICs measure the wireless channel for every received WiFi frame in order to decode the payloads of the frame [1]. Let  $X(f, t)$  and  $Y(f, t)$  be the frequency domain representations of transmitted and received signals at carrier frequency of  $f$  and time of  $t$ . The two signals are related by the expression  $Y(f, t) = H(f, t) \times X(f, t)$ , where  $H(f, t)$  is the complex-valued Channel Frequency Response (CFR). Note that the definition of CFR is based on the received signal  $Y(f, t)$  that contains the channel noise. Therefore, the channel noise is captured in the measured  $H(f, t)$ , which is defined as  $H(f, t) = Y(f, t)/X(f, t)$ . CFR values are reported by WiFi NICs in the form of CSI measurements, where each CSI measurement contains  $S$  matrices with dimensions of  $N_{Tx} \times N_{Rx}$ , where  $S$ ,  $N_{Tx}$ , and  $N_{Rx}$  are the number of OFDM subcarriers and the number of transmitting/receiving antennas, respectively. Each entry in the CSI matrix is a CFR value between a pair of transmitting/receiving antenna at a given OFDM subcarrier for one received WiFi frame. Usually, CSI are measured at  $S = 30$  subcarriers and we call the time-series of CFR values on a given subcarrier for a given antenna pair as a *CSI stream* in this paper.

#### B. System Overview

To recognize an activity based on CSI streams, CARM needs a CSI-activity model of that activity. Given the training CSI samples of the activity, it builds the CSI-activity model using HMM with the following three steps. The first and the most crucial step is *denoising CSI values*. CSI values reported by commercial devices contain high amplitude impulse noise, which makes it hard to extract small fluctuations caused by human movements. We observe that the signal fluctuations caused by human body movements in the OFDM subcarriers are correlated. Therefore, CARM can separate body movement signals from noise using Principal Component Analysis (PCA). The second step is *feature extraction* for classification. To extract features, CARM makes use of our CSI-Speed model that the frequency components in the CSI time series are related to the movement speeds of the body. CARM applies Discrete Wavelet Transform (DWT) on denoised CSI signals to extract both movement duration and speeds of the body and uses them as features for classification. The third step is *classifier training* to generate CSI-activity models. CARM builds an HMM model for each activity using the features extracted from the training samples of that activity. As human beings never do the same activity twice at exact same speeds, we chose to use HMM-based classifier because of its inherent capability to model activities performed at different speeds.

CARM recognizes an unknown activity using the models of all activities with the following five steps. The first step

is *denoising CSI values*, which is the same as in the training process. The second step is *activity detection* in which CARM uses an adaptive thresholding algorithm to monitor the energy of the denoised signals to detect the start and end of an activity. As soon as it detects the start of an activity, it proceeds to the third step, which is *feature extraction* as in the training process. The fourth step is *activity recognition*, where CARM calculates likelihood values using the extracted features on the HMM model of each activity. In the fifth step, CARM performs *data fusion* on multiple wireless links based on the likelihood of each activity. The data fusion algorithm declares the unknown activity to be the one whose HMM model has the highest likelihood. The pseudo code of our activity recognition system is shown in Algorithm 1.

---

#### Algorithm 1: CARM Activity Recognition Algorithm

---

**Input:** CSI measurements in 200 ms segments  
**Output:** Recognized activity  
**foreach** received CSI segment **do**  
    **foreach** subcarrier **do**  
        Preprocess the data by removing outliers;  
    Perform PCA based denoising using all CSI streams and extract five PCA components;  
    Calculate the activity indicator;  
    **if** activity indicator > adaptive threshold **then**  
        Perform DWT on five PCA components;  
        Extract the feature vector and add it into the feature vector buffer;  
        **if** feature vector buffer length > required vector size **then**  
            Calculate the log-likelihood of each activity using pre-trained HMM models;  
            Output the activity corresponds to the model with highest likelihood using the fusion algorithm;  
            Clear the feature vector buffer;  
    **else**  
        Output no activity;  
        Clear the feature vector buffer;

---

#### C. Ideal Phases of WiFi Signals

Radio signals may travel through multiple paths, including the LOS path and paths reflected by surrounding objects, from the transmitter to the receiver. In the case that there are  $N$  different paths,  $H(f, t)$  can be given as follows if we ignore the additive noise [24]:

$$H(f, t) = e^{-j2\pi\Delta f t} \sum_{k=1}^N a_k(f, t) e^{-j2\pi f \tau_k(t)}, \quad (1)$$

where  $a_k(f, t)$  is the complex-valued representation for both the attenuation and the initial phase offset of the  $k^{\text{th}}$  path,  $e^{-j2\pi f \tau_k(t)}$  is the phase shift on the  $k^{\text{th}}$  path caused by a propagation delay of  $\tau_k(t)$ , and  $e^{-j2\pi\Delta f t}$  is the phase shift caused by the Carrier Frequency Offset (CFO)  $\Delta f$  between the sender and the receiver. Note that we have ignored both the Symbol Timing Offset (STO) and the Sampling Frequency Offset (SFO) in Eq. (1), because the CFR phase shifts introduced by these offsets can be safely ignored in a similar way as the CFO, as we will show in later sections.

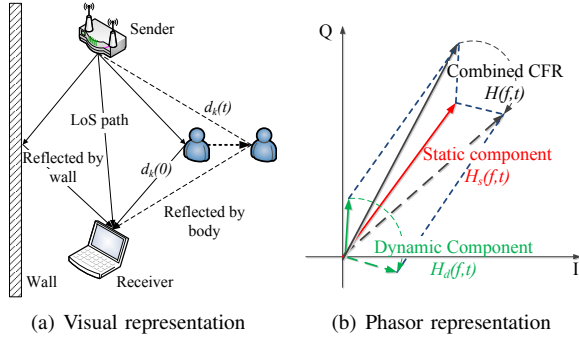


Figure 2. Multi-paths caused by human movements

Changes in the length of a path lead to the changes in the phase of the WiFi signal on the corresponding path. Consider the scenario in Figure 2(a), where the WiFi signal is reflected by the human body through the  $k^{\text{th}}$  path. When the human moves by a small distance between time 0 and time  $t$ , the length of the  $k^{\text{th}}$  path changes from  $d_k(0)$  to  $d_k(t)$ . The delay of the  $k^{\text{th}}$  path, denoted as  $\tau_k(t)$ , can be written as  $\tau_k(t) = d_k(t)/c$ , where  $c$  is the speed of light. The wavelength  $\lambda$  of the radio wave is related to the radio frequency  $f$  by the function of  $\lambda = c/f$ . Thus, the phase shift  $e^{-j2\pi f \tau_k(t)}$  can be written as  $e^{-j2\pi d_k(t)/\lambda}$ . Therefore, we observe that the receiver experiences a phase shift of  $2\pi$  on the given path whenever the path length changes by one wavelength.

Theoretically, it is possible to precisely measure the phase of the path when sender and receiver are perfectly synchronized, e.g., as in RFID systems [33]. However, commercial WiFi devices have non-negligible CFO, represented as the  $\Delta f$  in Equation (1), due to hardware imperfections and environmental variations [9]. IEEE 802.11n allows the carrier frequency of a device to drift by up to 100 kHz from the central frequency [1]. Such frequency drift leads to rapid phase changes in CSI values. With a transmission rate of 4,000 frames per second, which is around the maximum number of frames that the commercial device can continuously transmit due to the frame aggregation mechanism [1], the phase shift caused by the term  $e^{-j2\pi \Delta f t}$  in Equation (1) could be as large as  $50\pi$  between consecutive CSI measurements. Therefore, it is hard to measure the small phase shift in  $e^{-j2\pi d_k(t)/\lambda}$  under the interference introduced by the CFO.

#### D. CSI-Speed Model

To address the phase interference introduced by CFO, we measure the phase change of the path using the CFR power, i.e.,  $|H(f, t)|^2$ , while ignoring the phase of CFR. The principle behind our method is that when the lengths of multi-paths change, the CFR power varies according to the path length change. To understand the relationship between CFR power and the length change of a path, we first split CFR into two parts: dynamic CFR and static CFR. Dynamic CFR, represented by  $H_d(f, t)$ , is the sum of CFRs for paths whose lengths change with the human movement, and is given by  $H_d(f, t) = \sum_{k \in P_d} a_k(f, t) e^{-j2\pi d_k(t)/\lambda}$ , where  $P_d$  is the set of paths changed by human movements. Static CFR, represented by  $H_s(f)$ , is the sum of CFRs for static paths. Thus, the total CFR is given by the following equation:

$$H(f, t) = e^{-j2\pi \Delta f t} \left( H_s(f) + \sum_{k \in P_d} a_k(f, t) e^{-j\frac{2\pi d_k(t)}{\lambda}} \right). \quad (2)$$

The total CFR has time-varying power because the static component  $H_s(f)$  is a constant vector while the dynamic component  $H_d(f, t)$  is superposition of vectors with time varying phases and amplitudes, as shown in Figure 2(b). For example, let an object move at a constant speed such that the length of the  $k^{\text{th}}$  path changes at a constant speed  $v_k$  for a short time period, e.g., 100 ms. Let  $d_k(t) = d_k(0) + v_k t$  be the length of the  $k^{\text{th}}$  path at time  $t$ . The instantaneous CFR power at time  $t$  is as follows (detailed derivations are omitted due to space constraints):

$$\begin{aligned} |H(f, t)|^2 = & \sum_{k \in P_d} 2|H_s(f)a_k(f, t)| \cos\left(\frac{2\pi v_k t}{\lambda} + \frac{2\pi d_k(0)}{\lambda} + \phi_{sk}\right) \\ & + \sum_{\substack{k, l \in P_d \\ k \neq l}} 2|a_k(f, t)a_l(f, t)| \cos\left(\frac{2\pi(v_k - v_l)t}{\lambda} + \frac{2\pi(d_k(0) - d_l(0))}{\lambda} + \phi_{kl}\right) \\ & + \sum_{k \in P_d} |a_k(f, t)|^2 + |H_s(f)|^2, \end{aligned} \quad (3)$$

where  $\frac{2\pi d_k(0)}{\lambda} + \phi_{sk}$  and  $\frac{2\pi(d_k(0) - d_l(0))}{\lambda} + \phi_{kl}$  are constant values representing initial phase offsets.

Equation (3) provides a key insight: *the total CFR power is the sum of a constant offset and a set of sinusoids, where the frequencies of the sinusoids are functions of the speeds of path length changes.* By measuring the frequencies of these sinusoids and multiplying them with the carrier wavelength, we can obtain the speeds of path length change. In this way, we can build a CSI-speed model which relates the variations in CSI power to the movement speeds. By measuring the change in CFR power instead of the phase of CFR, we can safely ignore the phase noises introduced by CFO, STO, and SFO.

#### E. Model Verification

We use a simple moving object to verify our CSI-speed model in Equation (3). We move a steel plate with a diameter of 30 cm at a distance of 3 meters along the perpendicular bisector of the sender/receiver, similar to the scenario shown in Figure 2(a). Since flat steel objects serve as mirrors for radio waves [37], there is only one dominating path for the reflected signal and Equation (3) reduces to a single sinusoid wave plus a constant offset. Figure 3(a) shows the CSI waveform caused by movements of the steel plate.

We can calculate the movement distance by measuring the phase change of the signal using Hilbert Transform. We first remove the DC component that accounts for the static paths. We then use Hilbert Transform to derive the analytic signal from the real waveform. The unwrapped instantaneous phase of the analytic signal keeps track of the phase change and we can then multiply the phase change with the wavelength to get the path length change. Since the reflected signal goes through a round-trip from the reflector, the path length change is approximately two times of the movement distance of the reflector [30]. Figure 3(b) and 3(c) show the measurement results and the CDF of measurement error. The ground truth path length change is measured by a laser rangefinder with distance resolution of 0.1 cm. Our path length measurement has a maximal error of 5.87 cm and a mean error of 2.86 cm.

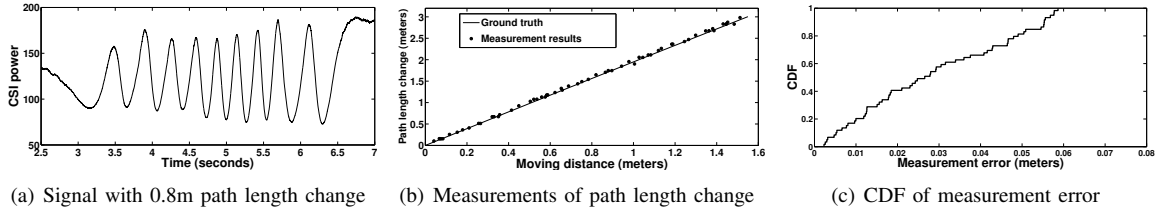


Figure 3. Experimental results with steel plates moving along a straight line.

#### IV. PCA BASED CSI DENOISING SCHEME

The first step for CARM is to preprocess the noisy CSI measurements. As shown in Figure 4(a), the raw CSI streams collected by the Intel 5300 NIC are extremely noisy and we can barely observe the small fluctuations in CSI power caused by surrounding movements. Thus, it is important to remove the noise in CSI before proceeding with further steps.

##### A. Noise in CSI Streams

CSI streams contain noises generated by the internal state transitions in the transmitter and the receiver, including transmission power changes, transmission rate adaptation, and internal CSI reference level changes. These internal state transitions introduce high amplitude impulse and burst noises in CSI streams.

Traditional filters such as low-pass filters or median filters do not perform well in removing these high amplitude impulse and burst noises. A low-pass filter, such as a Butterworth filter, smooths the signal by attenuating the high frequency components in the signal. However, due to the high energy and large bandwidth of impulse noises in CSI, the pass band for the low-pass filters usually needs to be less than one-twentieth of the sampling rate to suppress the residual energy of the noise [26]. When the cutoff frequency of the low-pass filter is not small enough, the residual noises can still distort the resulting stream. Figure 4(b) shows the output of a low-pass filter with a cutoff frequency of 100 Hz when applied to the CSI stream in Figure 4(a), which has a sampling rate of 2.5 kHz. The filtered stream is still severely distorted and further reducing the cutoff frequency will attenuate the energy of useful movement signals. Median filters, which is specifically designed to remove impulse noise, also do not work well on CSI streams because the density of the noises is very high. Figure 4(c) shows the output of a 5-point median filter, which is also severely distorted.

##### B. Correlation in CSI Streams

Our key insight for movement signal recovery is that the changes introduced by body movement are correlated in different CSI streams. The reason for such correlation is that CSI streams of different subcarriers are linear combinations of the same set of time-varying signals. Suppose that an object moves by a small distance between time 0 and time  $t$  that introduces path length change of  $\Delta_k(t)$ . In this case, we have  $d_k(t) = \Delta_k(t) + d_k(0)$ , where  $d_k(0)$  is the initial length of the path. When the initial phase offset is  $\phi_k$ , the phase at subcarrier  $s$  in Equation (3) seen by the receiver at time  $t$  is as follows:

$$\cos\left(\frac{2\pi d_k(t)}{\lambda_s} + \phi_k\right) = \cos\left(\frac{2\pi d_k(0)}{\lambda_s} + \phi_k\right) \cos\left(\frac{2\pi \Delta_k(t)}{\lambda_s}\right) - \sin\left(\frac{2\pi d_k(0)}{\lambda_s} + \phi_k\right) \sin\left(\frac{2\pi \Delta_k(t)}{\lambda_s}\right). \quad (4)$$

Note that the phase at subcarrier  $s$  is dependent on the wavelength  $\lambda_s$  of the subcarrier. However, the differences between wavelengths of subcarriers in the same WiFi channel are small. For example, in a 20 MHz WiFi channel, the lowest and highest subcarriers are separated by about 17 MHz, which leads to a wavelength difference of at most 0.34% in 5 GHz band. Such small difference in wavelengths usually does not change the number of multipaths across subcarriers. Considering two subcarriers with wavelengths  $\lambda_1$  and  $\lambda_2$ , two observations can be derived from Equation (4). First, the time-varying terms in the equation are approximately equal, *e.g.*,  $\cos\left(\frac{2\pi \Delta_k(t)}{\lambda_1}\right) \approx \cos\left(\frac{2\pi \Delta_k(t)}{\lambda_2}\right)$ , because  $\Delta_k(t)$  is small and  $\lambda_1$  and  $\lambda_2$  differ only slightly. Second, the constant terms in Equation (4), *e.g.*,  $\cos\left(\frac{2\pi d_k(0)}{\lambda_1} + \phi_k\right)$  and  $\cos\left(\frac{2\pi d_k(0)}{\lambda_2} + \phi_k\right)$ , are unequal for the two subcarriers. This is because path length  $d_k(0)$  is much greater than  $\Delta_k(t)$  so that it results in a non-negligible initial phase difference between the two subcarriers even though the wavelengths differ only slightly. For example, for a path length of 10 meters, a radio signal with wavelength of 5.150 cm traverses distance equal to 194.1 full wavelengths, while a radio signal with wavelength of 5.168 cm ( $= 5.15 \times 1.0034$ ) traverses distance equal to 193.5 wavelengths. Thus, there is an initial phase difference of  $(194.1 - 193.5) \times 2\pi = 1.2\pi$  between these two signals at the receiver. These two observations show that CFR for different subcarriers is a linear combination of the same set of time-varying waveforms with different initial phases. Therefore, the CSI power changes caused by movements in different subcarriers are correlated. Similar results can be obtained for CSI streams between different antenna pairs because the difference in positions of antennas only causes initial phases and attenuations for each multipath to be different.

Our measurements confirm the observation that CSI streams are correlated. Figure 5 plots the 180 CSI streams for a link with  $N_{Tx} = 2$  and  $N_{Rx} = 3$  when a human is walking around. We group the CSI streams in their transmission/receiving antenna pairs, *e.g.*, streams 1~30 are the 30 subcarriers for transmitting antenna 1 and receiving antenna 1. The amplitudes for CSI values are represented by the color, *i.e.*, red colors are “peaks” and blue colors are “valleys” in the curve. We have the following observations in the CSI streams. First, CSI streams are correlated. The “peaks” and “valleys” have similar shapes in all CSI streams across different antenna pairs and different subcarriers. Moreover, the phases of CSI



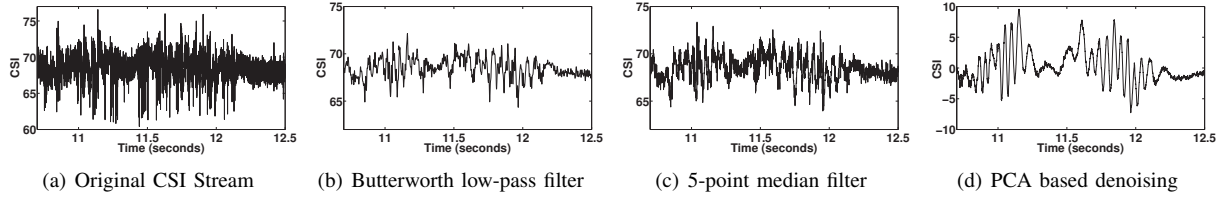


Figure 4. Denoising the time-series of CSI values

streams change smoothly across different subcarriers in the same antenna pair, e.g., streams 1~30 and 151~180, because the subcarriers on the same antenna pair only differ slightly in their frequencies. Second, there is no single “good” CSI stream. Although we can see clear “peaks” and “valleys” in stream 151~180 at time between 2.5~2.6 and 2.8~2.9 seconds, the changes in measurements are vague during the time 2.9~3 seconds for the same set of streams. However, we observe streams 1~30 give clear CSI fluctuations during the same time period of 2.9~3 seconds. This implies that we need to combine different streams to get optimal observations in the movements. Third, simply using weighted average over CSI streams [14] cannot provide good results. We see that the phase of different CSI streams are different so that if we add them up, they can cancel each other as the “peak” of a stream may be the “valleys” of other streams. Therefore, it is important to find a good way to combine CSI streams.

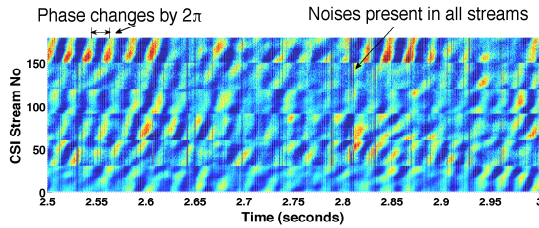


Figure 5. Correlation in CSI streams

### C. Principal Component Analysis

To address the challenges in combining CSI streams, we apply PCA to discover the correlations between CSI streams. With PCA, we can track the time-varying correlations between CSI streams, and optimally combine them to extract principal components of CSI streams. CARM applies PCA to CSI streams using the following four steps. (1) *Preprocessing*: In this step, CARM first removes the static path components from each CSI stream by subtracting the corresponding constant offsets from the streams. It calculates the constant offset for each stream through long-term averaging over that stream, i.e., average CSI amplitude for 4 seconds. After that, it cuts CSI streams into chunks that contain samples obtained in 1-second interval and arranges chunks of different CSI streams in columns to form a matrix of  $\mathbf{H}$ . We choose interval size to be 1 second so that the distance moved by the object is short and at the same time the number of samples is large enough to ensure accurate correlation estimation. (2) *Correlation estimation*: CARM calculates the correlation matrix as  $\mathbf{H}^T \times \mathbf{H}$ . The correlation matrix has a dimension of  $N \times N$ , where  $N$  is

the number of CSI streams. For the example in Figure 5, we have  $N = 180$ . (3) *Eigendecomposition*: CARM performs Eigendecomposition of the correlation matrix to calculate the eigenvectors. (4) *Movement Signal Reconstruction*: In this step, CARM constructs the principal components using the equation  $\mathbf{h}_i = \mathbf{H} \times \mathbf{q}_i$ , where  $\mathbf{q}_i$  and  $\mathbf{h}_i$  are the  $i^{\text{th}}$  eigenvector and the  $i^{\text{th}}$  principal components, respectively.

CARM discards the first principal component  $\mathbf{h}_1$  and retains the next five principal components to be used for feature extraction. As discussed in IV-A, noises caused by internal state changes present in all CSI streams, which are the vertical lines appear in Figure 5. Due to the high correlation, these noises are captured in  $\mathbf{h}_1$  along with the human movement signal. However, an interesting result is that all the information about the human movement signal captured in  $\mathbf{h}_1$  is also captured in other principal components, because by Equation (4), the phase of a subcarrier is a linear combination of two orthogonal components:  $\cos\left(\frac{2\pi\Delta_k(t)}{\lambda}\right)$  and  $\sin\left(\frac{2\pi\Delta_k(t)}{\lambda}\right)$ . Since the PCA components are uncorrelated, the first principal component only contains one of these orthogonal components and the other component is retained in the rest PCA components. Therefore, we can safely discard the first principal component without losing any information. The number of PCA components used for feature extraction is empirically selected to achieve a good tradeoff between classification performance and computational complexity. Figure 4(d) shows the second PCA component of our denoising scheme. We observe that our proposed method outperforms traditional filtering methods and does not contain the high-frequency noise.

## V. ACTIVITY DETECTION & FEATURE EXTRACTION

After denoising the CSI measurements, CARM first detect the start and end of an activity. Then, CARM extracts human movement features based on the CSI-activity model that quantifies the relation between human movement speeds and human activities.

### A. Activity Detection

CARM monitors the second eigenvector  $\mathbf{q}_2$  and the corresponding principal component  $\mathbf{h}_2$  to detect an activity. Our activity detection method is based on two key observations. First, in the absence of an activity, the eigenvector  $\mathbf{q}_2$  varies randomly over neighboring subcarriers because CSI streams contain uncorrelated values. However, in the presence of an activity, the CSI streams become correlated and  $\mathbf{q}_2$  varies smoothly over neighboring subcarriers. Second, in the absence of an activity, the principal component  $\mathbf{h}_2$  has smaller variance; whereas in the presence of an activity, it has higher variance.

CARM empirically calculates the variance,  $\mathbb{E}\{\mathbf{h}_2^2\}$ , of the time series  $\mathbf{h}_2$  and the mean of first difference of the eigenvector  $\mathbf{q}_2$  given by  $\delta_{\mathbf{q}_2} = \frac{1}{N-1} \sum_{l=2}^N |\mathbf{q}_2(l) - \mathbf{q}_2(l-1)|$ , where  $N$  is the number of CSI streams and  $|\mathbf{q}_2(l) - \mathbf{q}_2(l-1)|$  is the difference in coefficients for neighboring subcarriers. When there is an activity,  $\mathbb{E}\{\mathbf{h}_2^2\}$  has a higher value due to the variations introduced by human activity, whereas  $\delta_{\mathbf{q}_2}$  has a smaller value because the eigenvector becomes more smooth. Therefore, we define the *activity indicator* as  $\mathbb{E}\{\mathbf{h}_2^2\}/\delta_{\mathbf{q}_2}$ . Figure 6 plots  $\mathbb{E}\{\mathbf{h}_2^2\}$ ,  $\delta_{\mathbf{q}_2}$ , and the activity indicator over a period of about 10 seconds. We observe that activity indicator increase at 1.4 seconds and decreases at 9 seconds, which is the start and end times of the action, respectively. The activity indicator  $\mathbb{E}\{\mathbf{h}_2^2\}/\delta_{\mathbf{q}_2}$  has better detection performance because it has sharper edges than using single metrics of  $\mathbb{E}\{\mathbf{h}_2^2\}$  or  $\delta_{\mathbf{q}_2}$ .

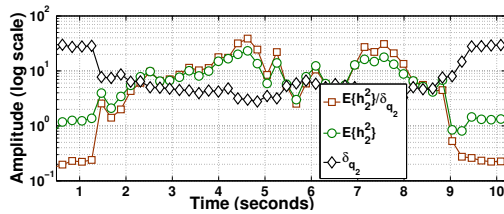


Figure 6. Activity detection indicators

To automatically detect the start or end of an activity, CARM compares the activity indicator with a threshold that it adjusts dynamically based on the background noise level. We use an Exponential Moving Average (EMA) algorithm to update the detection threshold. In the case of a sudden increase in noise level, CARM can incorrectly detect the start of an activity. To handle this, when CARM builds activity models, it also collects samples for the situation when there is no activity and builds a model for “no activity” using these samples. CARM classifies a detected activity using all activity models and decides whether a detected activity indeed has an activity in it or not. If it finds that there was no activity, it adjusts the detection threshold accordingly.

### B. Human Activity Characteristics

After detecting the activity, we extract features that can capture human activity characteristics from CSI. Human activities can be characterized by their specific moving speed patterns. For example, walking can be characterized by a constant moving speed of around 1 m/s and falling can be characterized by a sudden speed up within less than 0.5 seconds. These speed patterns can be captured by CSI measurements.

CFR power measurements are combinations of different reflection paths, which are reflected by different human body parts that may move at different speeds. Our key observation from Equation (3) is that the superposition of different paths are linear combinations, which do not change the frequencies of the components. Therefore, if two body parts move at different speeds, we will have two components with different frequencies in CFR power, respectively. In consequence, we can use Time-Frequency analysis tools, such as Short-Time Fourier Transform (STFT) or Discrete Wavelet Transform (DWT) to separate these components in the frequency domain.

Human activity can be modeled by profiling the energy of each frequency component derived from Time-Frequency analysis tools. As an example, Figure 7 illustrates the waveform and the corresponding STFT spectrogram for three human activities: walking, falling and sitting down. The spectrogram shows how the energy of each frequency component evolves with time, where high-energy components are colored in red. In the spectrogram for the walking activity, there is a high-energy band around 35~40 Hz frequency, as in Figure 7(d). With a wavelength of 5.15 cm, these frequency components represent 0.9~1.0 m/s movement speed after considering the round-trip path length change. This coincides the normal movement speed of human torso for walking [25]. Figure 7(e) shows the spectrogram of falling, which has an energy increase in the frequency range of 40~80 Hz between 1~1.5 seconds. This indicates a fast speed-up from below 0.5 m/s speed to 2 m/s, during a short time period of 0.5 seconds, which is a clear sign of falling. The activity of sitting down shown in Figure 7(f) is different from falling, as the speed for sitting down is much slower. Using the energy profile of different frequencies, we can build CSI-activity model, which quantifies the correlation between the movement speeds of different human body parts and a specific human activity.

### C. Robustness of Activity Speeds

It is well known that the path length change is determined by both the position of the sender/receiver and the movement directions [30]. Movements with the same speed may introduce different path length change speeds when movement directions are different. Furthermore, different people may perform the same activity with different speeds and the multi-path conditions may change under different environments.

Our experiments show that different human activities actually incur path length change speed with a significant difference, while the speed differences caused by different movement angles and the different ways to perform the same activity are small. To study the distribution of movement speeds, we collect more than 780 activity samples for three activities of walking, running, and sitting down. The activities are performed at different locations with different directions, e.g., we ask the volunteer to walk around a large table so that four different walking directions are captured. Figure 8(a) shows the estimated torso speed distribution for the three different activities.

Even with different movement directions, we observe that the three activities have different speeds in Figure 8(a). Such speed difference can be used for activity classification. As an example, we can achieve a classification accuracy of 88% for all three activities, when we divide the samples into three types with an estimated speed of 0~0.61 m/s, 0.61~1.0 m/s and above 1.0 m/s. Figure 8(b) shows the walking speed when the volunteers are asked to walk at different directions of 90, 60, and 30 degrees with respect to the sender and receiver. We observe that the speed distribution of walking at 90 degrees is similar to that of 30 degrees, while the distribution for 90 degrees is only slightly shifted towards higher speeds. Note that the speed distribution for walking back and forth

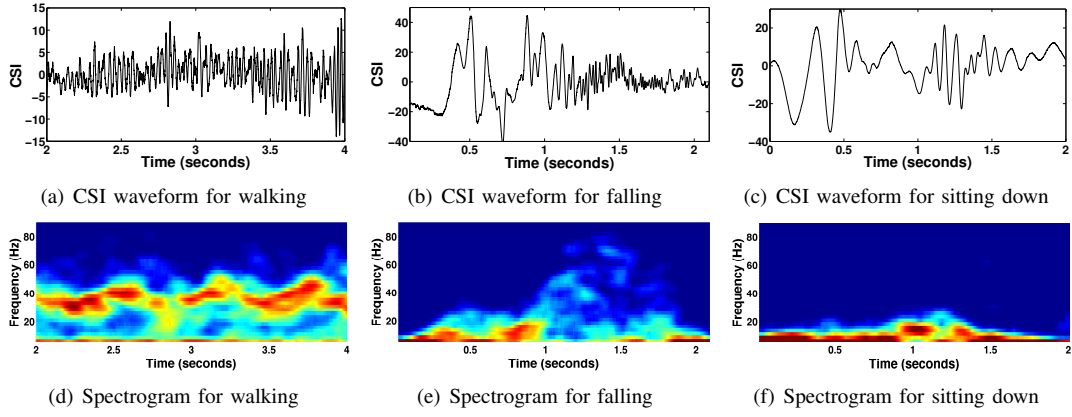


Figure 7. Waveforms and spectrograms for different activities.

in different directions in Figure 8(b) is slightly different from the distribution of walking in circles in Figure 8(a) due to the difference in walking patterns. Therefore, we use activity samples performed in different ways to generate a robust activity model.

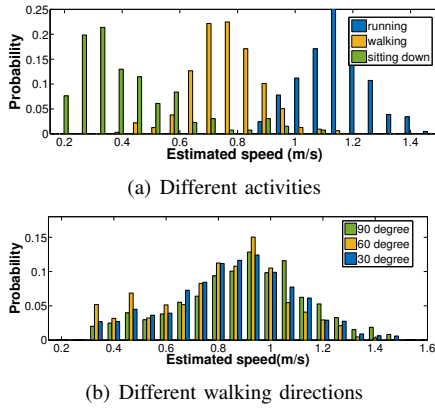


Figure 8. Histogram of speeds for activities

#### D. Feature Extraction

CARM extracts frequency components from different activities at different time scales as activity features. Our activity features capture both the duration and the frequency of the activity. Duration represents the time a person takes to perform an activity and frequency represents the speed of multi-paths due to body movements during the activity.

We use discrete wavelet transform (DWT) to extract frequencies at multiple resolutions on multiple time scales. DWT provides high time-resolution for activities with high frequencies in CFR signals and high frequency-resolution for activities with slow speeds. DWT calculates the energies in different levels at any given time in the CFR signals, where each level corresponds to a frequency range. The frequency ranges of adjacent DWT levels decrease exponentially. For example, if level 1 DWT represents a frequency range of 150~300Hz, which corresponds to 3.85~7.7 m/s movement speed in 5GHz band, then level 2 DWT represents a frequency range that is half of the frequency range for level 1, *i.e.*, 75~150Hz,

which corresponds to 1.925~3.85 m/s. The higher the energy in a DWT level is, the more likely there are movements with the speed in a range associated with the frequency of that level. The advantage of DWT compared to STFT is as follows: First, DWT has nice tradeoffs in time and frequency resolutions. DWT naturally groups frequencies that differ by several orders of magnitude into a few levels so that both high-speed movements and low-speed movements can be captured. Second, DWT reduces the size of data so that the classification algorithm can run in real time.

To extract features for classification from an activity sample, CARM applies DWT to decompose the PCA components into 12 levels that span the frequency range from 0.15Hz to 300Hz. The DWT for the five PCA components are averaged to capture the movement information present in different PCA components. From the output of DWT on each 200ms interval, CARM extracts a 27-dimensional feature vector that includes three types of features. 1). The energy in each level, which represents the intensity of movement in each speed range. 2). Differences in the energy of each level between consecutive 200 ms intervals, which represent rates of change of the speed for the activity. 3). Estimated torso and leg speeds using the percentile method introduced in Doppler radar [25].

#### VI. ACTIVITY RECOGNITION

Using the features extracted from CSI, we propose to use Hidden Markov Model (HMM) to build CSI-activity models that consist of multiple movement states. As an example, we observe that the action of falling comprises several states from Figure 7(e). The person first moves slowly, with most CSI energy on the low frequency (slow movement) components. Then, there is a fast transition to very high-speed movement where substantial energy is in high-frequency components. After that, there is a quick transition to the silent state, where the movement energy reduces to nearly zero. By looking at these transitions between different states, we can infer that the person is possibly falling. Similarly, other human activities also contain states with can be characterized by their movement speeds.

HMM is a suitable tool to build state transition models using time-dependent features. It has been extensively used



in several recognition applications such as speech recognition [20], handwriting recognition, and gesture recognition in videos [6]. Use of HMMs for activity recognition is based on the assumption that the sequence of observed feature vectors corresponding to an activity is generated by a Markov model, which is a finite state machine that changes state once every time unit. Each time a state is entered, a feature vector is generated from a probability density called *output probability density*. Furthermore, the transition from one state to another or back to itself is also probabilistic and is governed by a discrete probability called *transition probability*. Hidden Markov Models are called hidden because in practice, the sequence of feature vectors is known but the underlying sequence of states that generated those feature vectors is hidden. HMM can capture information from all training samples and thus works very well even when there is high within-class variance. Provided that a sufficient number of representative training samples of an activity are available, an HMM can be constructed that implicitly models all of the many sources of variability inherent in the activity. Compared to existing works which use statistical features along a long period [11], [28], HMM based models utilize the transitions within the activity that provide more details about the activity.

CARM constructs an HMM for each activity using the training samples of that activity. It also constructs an activity model for the situation when there is no activity in the room. To estimate the mean vector and covariance matrix corresponding to each state and the transition probabilities for the HMM, CARM uses the well-known Baum-Welch algorithm [29]. Baum-Welch algorithm needs a rough guess of these probabilities to start with. To guess the initial values, CARM first divides the sequence of feature vectors from each training sample equally amongst the states and then calculates the initial values for the mean vector and covariance matrix of each state using the feature vectors assigned to that state. CARM also calculates the initial transition probabilities by first counting the number of transitions between every pair of states from the sequence of feature vectors of all training samples divided equally amongst states and then dividing the counts by the number of transitions in all training samples. To decide the number of states, CARM iterates through different number of states and selects the number that provides highest cross-validation accuracy. To avoid overfitting of the Baum-Welch algorithm to a particular person or moving direction, in generating the model of an activity, we include samples of that activity from different people and different movement directions. Furthermore, we evaluate the models using both 10-fold cross-validation and separated testing samples collected in different environments to ensure that the models do not overfit on samples from specific scenarios.

## VII. DATA FUSION FROM MULTIPLE LINKS

### A. Limitation of Single Link Measurements

In theory, human activities can be identified using a single WiFi link, where there are one transmitter and one receiver (albeit with multiple antennas each). However, there are two limitations in the single link based recognition. First, the

dynamic component in Figure 2(b) is still affected by the multipath effects. This is because the dynamic component is a combination of WiFi signals reflected by different human body parts through different paths, *e.g.*, first reflected by the human body and then reflected by the wall before reaching the receiver. Under such complex multipath conditions, a particular WiFi link may not manifest clear variations in CSI signals due to human activities even after applying PCA based denoising process. Figure 9 shows the PCA results of three different links when observing the same sitting down activity instance at three different locations. Due to the different multipath effect experienced by these three links at different locations, the waveforms are quite different. We can observe that link C provides the best quality CSI signal, while link A only experiences small variations in CSI caused by the activity. Furthermore, depending on the environment and the location of the user, the signal received over a particular link may contain high levels of noises.

Second, the movement speed perceived by different links are different. The phase change of the reflected signal is determined by both the movement distance and the position of the transmitter/receiver relative to the target human [30]. Therefore, the movement speed measured by different WiFi links will be slightly different. From Figure 9(b) and 9(c), we observe that the CSI variations observed by different links have slightly different frequencies, even though the variation patterns are similar. Such small differences in CSI frequencies lead to different measurements in movement speeds. The error in movement speed measurements can lead to inaccurate activity patterns.

To overcome these two limitations, we propose to use CSI signals from all three links to observe the same activity. This increases the chances that at least one of the links will experience less interference and provide a high-quality activity signal. Similarly, measuring movement speed using information from multiple links rather than a single link will lead to a more accurate estimation of speed.

### B. Multi-link Data Fusion

We use multiple receiving devices to simultaneously measure the CSI of the signal coming from the same transmitter, as shown in Figure 10. We send unicast 802.11 frames to each of the receivers and get a CSI sampling rate of 1,500 samples per second on each link. We add timestamps to each CSI measurements so that samples from different receivers can be synchronized. As described in the previous section, the frequencies of CSI variations are different at different links. Therefore, there is little correlation between CSI variations across different links. Therefore, the PCA-based method used for combining CSI in different subcarriers cannot be used for combining data from different links. We propose three different methods to fuse data from multiple links: majority-voting fusion, likelihood fusion, and feature fusion.

*Majority-voting fusion:* In majority voting fusion, we treat the three links separately and use single link activity recognition algorithm on each link. After getting the recognition result of each link, we use majority-voting on the results from

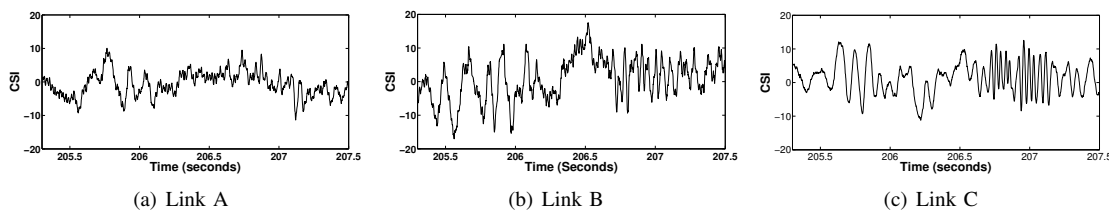


Figure 9. PCA results of the sitting down activity for three WiFi links

the three links to get the final decision. When there is a draw, *i.e.*, the three links give three different activity results, we randomly pick one output as the final decision.

**Likelihood fusion:** In likelihood fusion, we first use CSI measurements on each link to extract features and feed them separately into our HMM models. As our HMM model gives a log-likelihood result for each activity model on each link, we perform likelihood fusion by summing the log-likelihoods for the same activity on the three links. We then select the activity model that has the largest sum-log-likelihood as the final decision.

**Feature fusion:** In feature fusion, we first combine the extracted features before feeding them into the HMM models. We combine the features on different links using the standard maximal-ratio combining method. We first calculate the weight for each link using the SNR of their CSI variations. We then take the weighted average of each feature over the three links to get the combined feature sets. After that, the decision is made by feeding the combined features into the HMM model.

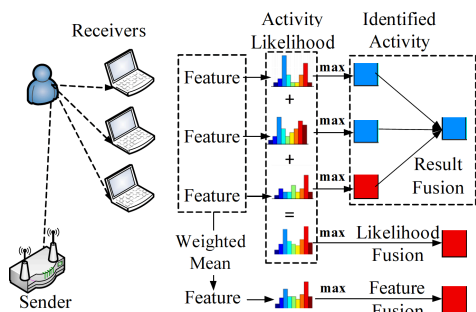


Figure 10. Leverage multiple links to enhance performance.

## VIII. IMPLEMENTATION & EVALUATION

### A. Implementation

We implemented CARM on commercial devices. We used Think-pad X200 laptops equipped with Intel 5300 WiFi card as the receivers and 802.11ac Access Points (APs), including NETGEAR JR6100 and TP-Link TL-WDR7500, as the transmitters. To obtain CSI values from regular data frames transmitted by the AP, we installed the CSI tool developed by Halperin *et al.* on the receivers [10]. All the experiments were performed in the 5GHz frequency band with channel bandwidth of 20 MHz. We chose 5GHz band for two reasons: first, the 5 GHz band has shorter wavelength, which leads to better distance resolution; second, it also has less wireless interferences than 2.4 GHz band. CARM acquired CSI

measurements from the CSI tool and processed it in real-time using MATLAB. During our experiments, there were other surrounding devices contending for the same channel used by CARM. We interpolated CSI measurements when there were missing CSI measurements due to WiFi frame loss.

### B. Data Collection

We collected training samples for eight different activities in the lab environment shown in Figure 11(a), which is 7.7m in length and 6.5m in width. Our activity database contains 1,400 samples performed by 25 volunteers who were 20 male and 5 female graduate/undergraduate students with ages in the range of 19–22. The activities in the database are listed in Table I. We collected training samples for each activity (except walking and running) at the location marked with star in Figure 11(a). For walking and running, our volunteers followed the path marked with a dashed line around the table in the center of the lab. Tx and Rx represent the locations of the transmitter AP and receiver laptop. While collecting the training data, we requested the volunteers to change their orientation to ensure the generality of the collected data. Triangles in Figures 11(a) and 11(b) represent the locations where our volunteers performed activities when evaluating accuracy of CARM. During the experiments, there were other persons sitting or using computers in the same room. CARM is robust to multiple persons coexisting in the same room given that they do not move at the same time. Recognizing simultaneous activities of multiple persons is left as future work of this paper.

### C. Performance Metrics

We evaluate the performance of CARM in three aspects: *detection accuracy*, *recognition accuracy*, and *efficiency*. For activity *detection accuracy*, we use two performance metrics: True Positive Rate (TPR) and False Alarm Rate (FAR). TPR is the ratio of the number of times for correctly detecting the presence of an activity to the total number of activity performed. FAR is the ratio of the number of times for incorrectly detecting the presence of an activity when actually there is no activity. For activity *recognition accuracy*, we evaluate the performance using the number of correctly recognized activity divided by the total number of activity performed. The activity recognition accuracy is evaluated in both the trained environments (the lab) and the untrained environments. We use four indoor scenarios as the untrained environments, including 1) a large open lobby area, which has a length, width, and height of 45m, 5.3m, and 4m, respectively; 2) a

small apartment, which has an area of  $70m^2$  as shown in Figure 11(b); 3) a small office, with a size of  $5.6m \times 3.4m \times 2.7m$ , 4) a large workshop region, with a size of  $14.8m \times 9.5m \times 2.7m$ . For *efficiency*, we evaluate CARM using three metrics: the time used for training, the time used for recognition, and the impact on transmission of other coexisting WiFi users.

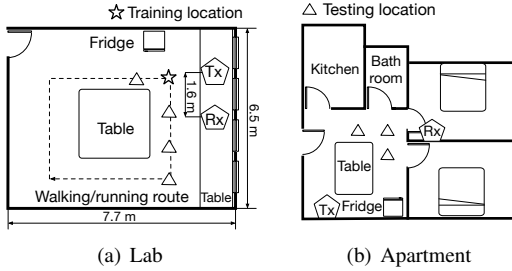


Figure 11. Floor plans for trained and untrained environments

Table I  
SUMMARY OF ACTIVITY DATASET

Activity	Samples	Training Time
(R) Running	205	16.38s
(W) Walking	315	26.84s
(S) Sitting down	266	14.49s
(O) Opening refrigerator	213	13.49s
(F) Falling	98	5.02s
(B) Boxing	75	4.88s
(P) Pushing one hand	72	7.00s
(T) Brushing teeth	96	7.35s
(E) Empty ( <i>i.e.</i> , no activity)	60	5.10s

#### D. Activity Detection

CARM detects small movements such as pushing hand and large movements such as walking with a TPR larger than 98% at distances of up to 5 meters and 12 meters, respectively. Figure 12 shows CARM's TPR for two different activities in the open lobby area as described in section VIII-C, where the sender and receiver were separated by 3.5 meters. Each TPR result was calculated from 20 samples of the given activity collected at three different locations at the same distance to the receiver. Compared to low-pass filtering, the PCA denoising method has a much longer detection range of 12 meters for walking. Low-pass filtering also cannot reliably detect small movements such as pushing at a distance of 2 meters. This shows that PCA is more efficient in extracting small changes in CSI values caused by human movements since the amplitude of CSI variations decreases as the distance increases. With a detection range of 12 meters, CARM achieves a large coverage area: with a single Tx/Rx pair, it reliably detects the presence of activities in a  $450m^2$  open region.

CARM has a low FAR of 1.4 false alarms per hour. To measure CARM's FAR, we recorded the activity log for 14 hours from 6:00 pm to 8:00 am when no one was around the sender and receiver. There were only 20 false alarms during these 14 hours. After checking the trace, we found that most false alarms were caused by sudden increase in noise levels, which lasted less than 2 seconds and can be easily filtered by the later recognition system.

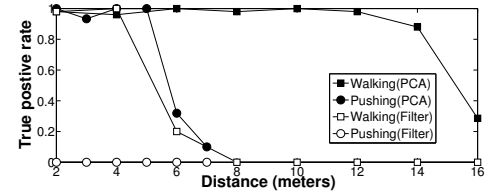


Figure 12. Detection range of CARM

#### E. Activity Recognition

CARM achieves an average cross-validation accuracy of 96.5% across all activities. Figure 13 shows the recognition accuracy of 10-fold cross-validation using the training data set collected in the “lab”. Except for “sitting down”, “opening fridge” and “falling”, CARM achieves close to 100% accuracy, under a sampling rate of 2,500 samples per second. The accuracy for “empty” is 100%, which means that if CARM incorrectly detects the presence of an activity when in reality there was no activity, it can correctly find out that there was no activity as soon as it evaluates the detected sample against the activity models.

The accuracy of CARM improves when CSI values are sampled at a higher sampling rate, but the increase is not significant beyond the sampling rate of 800 samples/second. Figure 13 gives recognition accuracy for each activity under four different sampling rates. We observe that with sampling rates of 800 samples/second and higher, the average cross-validation accuracy is over 94.8%, which is only a 1.7% drop compared to 96.5% achieved under a sampling rate of 2,500 samples/second. The sampling rate of 800 samples/second is very close to the Nyquist sampling rate because the frequency components in CFR may be as high as 300Hz due to high speed human movements, such as running. When sampling rate further reduces to 400 samples/second, the average accuracy reduces significantly to 87%. In comparison, the average recognition accuracy for low-pass filtering approach is only 73% with the sampling rate of 2,500 samples/second. In the low-pass filtering approach, the accuracies for activities of “falling” and small activities such as “brushing teeth” drop significantly because filtering cannot reliably preserve the high-frequency components and small fluctuations in CSI values. In our real-time experiments, CARM tolerates wireless interference caused by other devices that are using the same channel. When facing a high packet loss rate, CARM interpolates the missing CSI values and uses a lower sampling rate to achieve an acceptable recognition accuracy.

CARM achieves an accuracy of more than 72% for environments and persons that it has not been trained on when using a single link. Figure 14 shows the accuracy CARM achieves in different environments for each activity. In each environment, we randomly picked 3 to 6 locations and performed each activity for three times in each location to calculate the recognition accuracy. Two volunteers who were not in the training set performed these activities. Note that this figure does not have results for “opening refrigerator” in the lobby, office, and workshop, “running” in the apartment and office due to equipment and space constraints. In the lab, lobby,

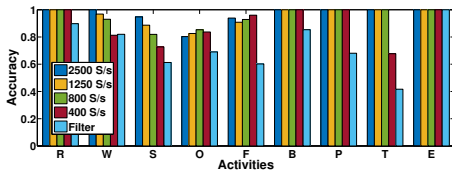


Figure 13. Impact of sampling rates on accuracy

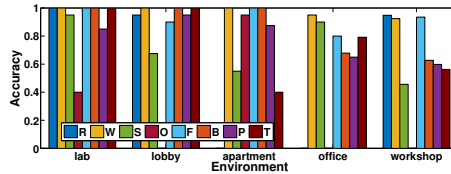


Figure 14. Accuracy in different environments

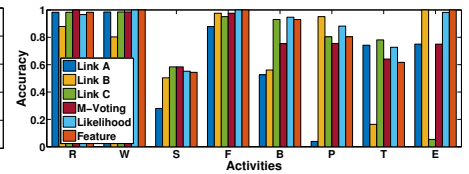


Figure 15. Different multi-link data fusion methods.

office, apartment, and workshop, CARM achieves an average accuracy of 90%, 93%, 83%, 80%, and 72%, respectively. For the environment “lab”, which is the same as the training environment, *i.e.*, the average accuracy reduces moderately to 90% with lower accuracy for a single activity, “opening refrigerator”, due to the different ways of performing the activity. For the more challenging case where CARM is applied to a new environment that it has no training data, the average accuracy drops to 72% in the worst case of “workshop”. Most of the recognition errors are in a few activities, *e.g.*, “sitting down” for the lobby, apartment, and workshop. Most errors are caused by the drastic change in multi-path conditions. For example, the workshop region has a reflective anti-static floor, which introduces complex multipath reflection from the floor.

*Multi-link data fusion improves the recognition accuracy of CARM in untrained environments by an average of 8%.* Figure 15 shows the recognition accuracy of CARM when using multi-link data fusion algorithms in the worst multi-path scenario of the previous experiment – the workshop. The average single link recognition accuracies for links A, B, and C are 64.8%, 72.3% and 75.8%, respectively. Some of the activities such as “brushing teeth” has very low accuracy ( $\sim 17\%$ ) on certain links. The recognition accuracies increase to 80.6%, 88.2%, and 86.0% when using the majority-voting, likelihood, and feature fusion algorithm. We observe that fusion on the lower level, *i.e.*, on likelihood or features, results in higher improvement in recognition accuracy compared to majority-voting based fusion.

#### F. Efficiency

*The training time for CARM is less than two minutes on COTS desktop computers.* Table I shows the training time of different activities on a desktop computer with an Intel i5-4285 CPU. The total training time for our activity database with 1,400 samples is 100.55 seconds. Note that our activities models have no location dependency. Once CARM is trained on the given training set, it can be directly applied to environments and persons that have not been included in the training set. Thus, CARM does not need on-site training data collection as for E-eyes [28]. Consequently, this single-pass training process can be done on a large training set using a data center. *CARM can also run in real-time on COTS laptops.* In our experiments, CARM takes 85.6 ms to process a 200 ms segment of CSI values on a Think-pad laptop with Intel i5-3320 CPU and 4GB RAM, when the sampling rate is 2,500 samples/second.

*CARM can reduce WiFi throughput of coexisting users by up to 20% when collecting data at 2,500 samples/second.* Figure 16 shows WiFi throughput of one to three coexisting users,

where each user is receiving UDP traffic at up to 20 Mbps. When CARM is not sending data, the WiFi channel can sustain a small packet loss rate of around 2% for three users, each with 20 Mbps offered load. When CARM sends measurement frames with a length of 100 Bytes at a rate of 2,500 frames/second, it causes to up to 20% packet loss to other coexisting users and the aggregated throughput drops from 58.8 Mbps to 47.7 Mbps. Similarly, for a single coexisting TCP user, the average throughput drops from 66.8 Mbps to 53.5 Mbps with CARM traffic. Note that CARM may reuse the data packets sent by other users for CSI measurements when it detects the existing traffic.

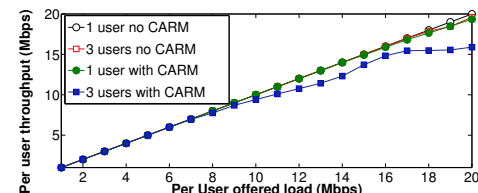


Figure 16. Throughput of coexisting WiFi users under CARM.

## IX. CONCLUSION

In this paper, we make the following three key contributions. First, we propose the CSI-speed model, which quantifies the correlation between CSI dynamics and the movement speeds, and the CSI-activity model, which quantifies the correlation between the movement speeds of different human body parts and a specific human activity. Second, we propose several signal processing techniques, such as PCA based denoising and DWT based feature extraction, for human activity recognition based on the two models. Third, we implement CARM using commercial WiFi devices and collect a human activity database that contains 1,400 samples. Our results show that CARM achieves an average accuracy of 96% and is robust to environmental changes.

## ACKNOWLEDGMENTS

We thank the students/volunteers in our lab who helped in collecting our dataset. This work is partially supported by the National Natural Science Foundation of China under Grant Numbers 61373129, 61472184, 61321491, and 61472185, the National Science Foundation under Grant Numbers CNS-1318563, CNS-1524698, CNS-1421407, CNS-1565609, and IIP-1632051, Collaborative Innovation Center of Novel Software Technology and Industrialization, and the Jiangsu High-level Innovation and Entrepreneurship (Shuangchuang) Program. Alex X. Liu is also affiliated with the Department of Computer Science and Engineering, Michigan State University, East Lansing, MI, USA.



## REFERENCES

- [1] Enhancements for higher throughput. IEEE Standard 802.11n, 2009.
- [2] Microsoft Kinect. <http://www.microsoft.com/en-us/kinectforwindows/>.
- [3] H. Abdelnasser, M. Youssef, and K. A. Harras. WiGest: A ubiquitous wifi-based gesture recognition system. In *Proc. IEEE INFOCOM*, 2015.
- [4] F. Adib, Z. Kabelac, and D. Katabi. Multi-person motion tracking via RF body reflections. In *Proc. Usenix NSDI*, 2015.
- [5] F. Adib, Z. Kabelac, D. Katabi, and R. C. Miller. 3D tracking via body radio reflections. In *Proc. Usenix NSDI*, 2013.
- [6] J. K. Aggarwal and S. R. Michael. Human activity analysis: A review. *ACM Computing Surveys*, 43(3), 2011.
- [7] K. Ali, A. X. Liu, W. Wang, and M. Shahzad. Keystroke recognition using WiFi signals. In *Proc. ACM MobiCom*, 2015.
- [8] E. Ertin, N. Stohs, S. Kumar, A. Raij, M. al'Absi, and S. Shah. AutoSense: unobtrusively wearable sensor suite for inferring the onset, causality, and consequences of stress in the field. In *Proc. ACM Sensys*, 2011.
- [9] J. Gjengset, J. Xiong, G. McPhillips, and K. Jamieson. Phaser: Enabling phased array signal processing on commodity WiFi access points. In *Proc. ACM MobiCom*, 2014.
- [10] D. Halperin, W. Hu, A. Sheth, and D. Wetherall. Tool release: Gathering 802.11n traces with channel state information. *ACM SIGCOMM CCR*, 41(1):53, 2011.
- [11] C. Han, K. Wu, Y. Wang, and L. M. Ni. Wifall: Device-free fall detection by wireless networks. In *Proc. IEEE INFOCOM*, pages 271–279, 2014.
- [12] D. Huang, R. Nandakumar, and S. Gollakota. Feasibility and limits of Wi-Fi imaging. In *Proc. ACM SenSys*, pages 266–279, 2014.
- [13] K. R. Joshi, S. S. Hong, and S. Katti. Pinpoint: Localizing interfering radios. In *Proc. Usenix NSDI*, pages 241–253, 2013.
- [14] B. Kellogg, A. Parks, S. Gollakota, J. R. Smith, and D. Wetherall. Wi-Fi backscatter: Internet connectivity for RF-powered devices. In *Proc. ACM SIGCOMM*, 2014.
- [15] B. Kellogg, V. Talla, and S. Gollakota. Bringing gesture recognition to all devices. In *Proc. Usenix NSDI*, 2014.
- [16] M. Kotaru, K. Joshi, D. Bharadia, and S. Katti. Spotfi: Decimeter level localization using wifi. In *Proc. ACM SIGCOMM*, 2015.
- [17] J. Lien, N. Gillian, M. E. Karagozler, P. Amihoud, C. Schwesig, E. Olson, H. Raja, and I. Poupyrev. Soli: ubiquitous gesture sensing with millimeter wave radar. *ACM Transactions on Graphics*, 35(4):142, 2016.
- [18] Q. Pu, S. Gupta, S. Gollakota, and S. Patel. Whole-home gesture recognition using wireless signals. In *Proc. ACM MobiCom*, 2013.
- [19] Y. Qiao, O. Zhang, W. Zhou, K. Srinivasan, and A. Arora. Phycloak: obfuscating sensing from communication signals. In *Proc. Usenix NSDI*, 2016.
- [20] L. R. Rabiner and B.-H. Juang. *Fundamentals of speech recognition*. Prentice Hall, 1993.
- [21] S. Sen, J. Lee, K.-H. Kim, and P. Congdon. Avoiding multipath to revive inbuilding WiFi localization. In *Proceeding of ACM MobiSys*, pages 249–262, 2013.
- [22] 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 Transactions on Mobile Computing*, 13(4):907–920, 2014.
- [23] 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. ACM MoMM*, 2013.
- [24] D. Tse and P. Viswanath. *Fundamentals of wireless communication*. Cambridge university press, 2005.
- [25] P. Van Dorp and F. Groen. Feature-based human motion parameter estimation with radar. *IET Radar, Sonar & Navigation*, 2(2):135–145, 2008.
- [26] G. Wang, Y. Zou, Z. Zhou, K. Wu, and L. M. Ni. We can hear you with Wi-Fi! In *Proc. ACM MobiCom*, 2014.
- [27] W. Wang, A. X. Liu, M. Shahzad, K. Ling, and S. Lu. Understanding and modeling of WiFi signal based human activity recognition. In *Proc. ACM MobiCom*, 2015.
- [28] Y. Wang, J. Liu, Y. Chen, M. Gruteser, J. Yang, and H. Liu. E-eyes: In-home device-free activity identification using fine-grained WiFi signatures. In *Proc. ACM MobiCom*, 2014.
- [29] L. R. Welch. Hidden markov models and the baum-welch algorithm. *IEEE Information Theory Society Newsletter*, 53(4):10–13, 2003.
- [30] N. J. Willis. *Bistatic Radar*. SciTech Publishing Inc., 2005.
- [31] 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. IEEE INFOCOM*, 2014.
- [32] Y. Xie, Z. Li, and M. Li. Precise power delay profiling with commodity wifi. In *Proc. ACM MobiCom*, 2015.
- [33] L. Yang, Y. Chen, X.-Y. Li, C. Xiao, M. Li, and Y. Liu. Tagoram: real-time tracking of mobile RFID tags to high precision using cots devices. In *ACM MobiCom*, pages 237–248, 2014.
- [34] Z. Yang, Z. Zhou, and Y. Liu. From RSSI to CSI: Indoor localization via channel response. *ACM Computing Surveys*, 46(2):25, 2013.
- [35] K. Yatani and K. N. Truong. Bodyscope: a wearable acoustic sensor for activity recognition. In *Proc. ACM UbiComp*, pages 341–350, 2012.
- [36] M. Zhao, F. Adib, and D. Katabi. Emotion recognition using wireless signals. In *Proc. ACM MobiCom*, 2016.
- [37] X. Zhou, Z. Zhang, Y. Zhu, Y. Li, S. Kumar, A. Vahdat, B. Y. Zhao, and H. Zheng. Mirror mirror on the ceiling: flexible wireless links for data centers. In *Proc. ACM SIGCOMM*, pages 443–454, 2012.
- [38] Z. Zhou, Z. Yang, C. Wu, L. Shanguan, and Y. Liu. Towards omnidirectional passive human detection. In *Proc. IEEE INFOCOM*, pages 3057–3065, 2013.



**Wei Wang** received his MS and Ph.D. degree from the ESE department of Nanjing University and the ECE department of National University of Singapore, in 2000 and 2008 respectively. He is currently an Associate Professor in the CS department of Nanjing University. His research interests are in the area of wireless networks, including Device-free Sensing, Cellular Network Measurements, and Software Defined Radio systems.



**Alex X. Liu** received his Ph.D. degree in Computer Science from the University of Texas at Austin in 2006. He received the IEEE & IFIP William C. Carter Award in 2004, a National Science Foundation CAREER award in 2009, and the Michigan State University Withrow Distinguished Scholar Award in 2011. He is an Associate Editor of IEEE/ACM Transactions on Networking and IEEE Transactions on Dependable and Secure Computing, and an Area Editor of Computer Communications. He received Best Paper Awards from ICNP-2012, SRDS-2012, and LISA-2010. His research interests focus on networking and security.



**Muhammad Shahzad** received his Ph.D. degree in computer science from Michigan State University in 2015. He is currently an Assistant Professor at the Department of Computer Science, North Carolina State University, USA. His research interests include design, analysis, measurement, and modeling of networking and security systems. He received the 2015 Outstanding Graduate Student Award, 2015 Fitch Beach Award, and 2012 Outstanding Student Leader Award at Michigan State University.



**Kang Ling** received his Bachelor degree in computer science from Nanjing University in 2014. He is currently a Ph.D. candidate in the Department of Computer Science and Technology, Nanjing University. His research interests include mobile sensing and device-free activity recognition.



**Sanglu Lu** received the BS, MS, and Ph.D. degrees from Nanjing University, China, in 1992, 1995, and 1997, respectively, in computer science. She is currently a professor in the Department of Computer Science and Technology, Nanjing University. Her research interests include distributed computing and pervasive computing. She is a member of the IEEE.