

FingerDraw: Sub-wavelength Level Finger Motion Tracking with WiFi Signals

DAN WU, RUIYANG GAO, YOUNWEI ZENG, JINYI LIU, and LEYE WANG, Peking University, China

TAO GU, RMIT University, Australia

DAQING ZHANG*, Peking University, China and Institut Polytechnique de Paris, France

This paper explores the possibility of tracking finger drawings in the air leveraging WiFi signals from commodity devices. Prior solutions typically require user to hold a wireless transmitter, or need proprietary wireless hardware. They can only recognize a small set of pre-defined hand gestures. This paper introduces FingerDraw, the first sub-wavelength level finger motion tracking system using commodity WiFi devices, without attaching any sensor to finger. FingerDraw can reconstruct finger drawing trajectory such as digits, alphabets, and symbols with the setting of one WiFi transmitter and two WiFi receivers. It uses a two-antenna receiver to sense the sub-wavelength scale displacement of finger motion in each direction. The theoretical underpinning of FingerDraw is our proposed CSI-quotient model, which uses the channel quotient between two antennas of the receiver to cancel out the noise in CSI amplitude and the random offsets in CSI phase, and quantifies the correlation between CSI value dynamics and object displacement. This channel quotient is sensitive to and enables us to detect small changes in In-phase and Quadrature parts of channel state information due to finger movement. Our experimental results show that the overall median tracking accuracy is 1.27 cm, and the recognition of drawing ten digits in the air achieves an average accuracy of over 93.0%.

CCS Concepts: • Human-centered computing → Ubiquitous and mobile computing systems and tools.

Additional Key Words and Phrases: Channel Quotient, WiFi, Channel state information (CSI), Finger-draw gesture

ACM Reference Format:

Dan Wu, Ruiyang Gao, Youwei Zeng, Jinyi Liu, Leye Wang, Tao Gu, and Daqing Zhang. 2020. FingerDraw: Sub-wavelength Level Finger Motion Tracking with WiFi Signals. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 4, 1, Article 31 (March 2020), 27 pages. <https://doi.org/10.1145/3380981>

1 INTRODUCTION

In recent years, finger gesture has gained increasingly attentions in facilitating advanced human computer interaction. Different from touch-based finger input which requires traditional hardware such as a keyboard and

*This is the corresponding author

Authors' addresses: Dan Wu, dan@pku.edu.cn; Ruiyang Gao, gry@pku.edu.cn; Youwei Zeng, ywzeng@pku.edu.cn; Jinyi Liu, liujinyi@pku.edu.cn; Leye Wang, leye@pku.edu.cn, Key Laboratory of High Confidence Software Technologies (Ministry of Education), School of Electronics Engineering and Computer Science, Peking University, Beijing, China; Tao Gu, Computer Science and School of Science, RMIT University, Melbourne, Australia, tao.gu@rmit.edu.au; Daqing Zhang, Key Laboratory of High Confidence Software Technologies (Ministry of Education), School of Electronics Engineering and Computer Science, Peking University, Beijing, China , Telecom SudParis, Institut Polytechnique de Paris, Evry, France, dqzhang@sei.pku.edu.cn.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2020 Association for Computing Machinery.

2474-9567/2020/3-ART31 \$15.00

<https://doi.org/10.1145/3380981>

touch screen, finger gesture input offers drawing in the air in a contactless and free manner. It facilitates natural interaction for small form-factor devices or devices which are lack of input interface, and provides many benefits especially for older adults and disabled people. Traditional finger gesture techniques leverage on peripheral devices and hardware systems, e.g., Kinect [29] and Leap Motion [9]. Recent radio signal based finger gesture sensing has the virtue of contact-free, transparent and non-intrusive, and it is an ideal input solution for smart applications. For instance, a user can perform finger gestures to control home appliances or manipulate characters in online gaming. The success of finger interaction depends much on effectively tracking of finger movements and recognition of finger gestures.

Tracking and recognizing finger gestures have been well studied in the literature. Early solutions instrument finger with wearable sensors [7, 13], but they are rather burdensome to be applied in real home scenarios. Other solutions use cameras/infrared sensors [6, 14, 16], but they do not work well under non-line-of-sight (NLoS) conditions. Acoustic-based solutions leverage on speaker and microphone to measure the range information of fingers, thus achieve high precision ($6 \sim 8$ mm) with no accumulative errors [23, 30, 42]. However, it often suffers from environmental noise and privacy concerns (i.e., unattended recording of voice). Radio wave based solutions such as RFID systems [36, 39] and radar [11] need dedicated hardware, which are relatively expensive.

WiFi-based sensing has recently attracted much attention due to its low deployment cost, always-on communication links, and rich subcarrier-level sensing information obtained from Channel State Information (CSI). With the availability of CSI information, WiFi-based sensing has shown great progress in various applications ranging from intrusion detection [19, 43, 46], human activity recognition [4, 22, 28, 40, 41], fall detection [24, 38], indoor localization [47], walking direction and trajectory estimation [10, 18, 26, 44], to vital sign monitoring [20, 21, 37, 50, 52].

Existing WiFi-based systems mainly leverage on CSI waveform patterns to train a machine learning classifier for recognizing hand gestures [1, 32, 53]. Patterns are matched based on features extracted from waveform shapes [32] or waveform frequency distribution [1]. The pattern-based approaches assume there exists a fixed mapping between gesture and signal pattern. However, according to the study in this paper, the same finger-drawings performed at different locations or with different drawing sizes would lead to very different CSI waveforms. Previous learning-based gesture recognition approaches are sensitive to the drawing position and drawing sizes. It's difficult to collect the CSI waveforms of different people with all gesture-drawing positions and gesture-drawing sizes for training, thus the gesture recognition performance cannot be guaranteed. Apart from the pattern inconsistency problem, learning-based approaches require intensive training to recognize a predefined set of hand gestures, not for arbitrary finger-draws.

Another set of solutions do not rely on training but make use of the physical model between signal and sensing target for motion tracking. Signal Angle-of-Arrival has been used to track a hand's motion by sensing its occlusion directions with a dense deployment of WiFi devices [31]. It gives a rather coarse trajectory estimation. Doppler shifts in WiFi signals have also been explored to sense the motion speed of human body [18, 27]. The Doppler-based solutions are designed for tracking the coarse motion of large body parts such as arm and leg rather than finger, the best can only achieve submeter-level motion tracking [18]. Fully capturing the tiny movement of a finger require precise CSI amplitude and phase information. As the commodity Wi-Fi raw CSI phase contains random offsets, the existing phase sanitation methods could only stabilize CSI phase offsets to some extent, but not able to cancel the random phase offset perfectly to obtain the precise phase [15, 48]. Sensing tiny motions of a finger also require high signal-to-noise ratio (SNR). The captured CSI signal variations of tiny finger motions are likely to be overwhelmed by amplitude impulse noises. We need methods to greatly increase the SNR in CSI signal in order to capture the tiny finger motions.

In this paper, we design and implement a tracking system named FingerDraw to track centimeter-level finger drawings with commodity WiFi devices in a contactless manner. Limited by the relatively low bandwidth (20 MHz), the WiFi-based technologies can only achieve a low ranging resolution of 15 m. We demonstrate that with carefully designed signal processing techniques, small motion displacement within a wavelength (5.8 cm

in a typical 5.2 GHz frequency band) of WiFi signal can be accurately extracted. We present two key enablers underpinning the basis of FingerDraw—the CSI-quotient model for sensing tiny motions and the techniques for finger-draw segmentation and reconstruction.

We begin with exploiting the capability of CSI signals obtained from commodity WiFi transceivers in a typical indoor setting. CSI signals are complex in nature. There exist both hardware and environmental noise in CSI amplitude [41], extremely limiting signal resolution which is essential in tracking and reconstructing small finger motions. There are also random offset noise in CSI phase which is hard to compensate [45, 48], limiting the use of its fully available orthogonal information for moving direction and displacement sensing. In FingerDraw, we aim to improve CSI signal resolution. As illustrated in Fig.1, we deploy two antennas for each receiver to obtain two CSI signals reflected from the finger. We then propose a novel CSI-quotient model which computes the quotient of these two CSI signals. The newly transformed signal will be used as the basis for tracking. The proposed model is able to cancel random phase offsets which are identical among the antennas of the same receiver based on the study in [18, 38]. In addition, it effectively removes uncertain impulse noise in CSI amplitude based on our experimental studies. Essentially, the CSI-quotient model maximizes SNR so that the resolution of CSI signals can be significantly improved. With most noise being removed, we make the orthogonal information of newly constructed CSI signals fully available, similar to the In-phase and Quadrature signals used in radar systems [8]. It is thus possible to track tiny finger motions and reconstruct its trajectories.

With significantly improved CSI signal resolution, we are able to quantify the correlation between the reflected path length and the phase change of the CSI quotient in a sub-wavelength level granularity. However, reconstructing the complete drawing trajectory is still not a trivial task. In principle, tracking and reconstructing a 2D trajectory requires information from two dimensions. We therefore deploy one WiFi transmitter and two WiFi receivers. We discover that the two WiFi receivers are best placed orthogonally. With this setting, we collect CSI signals from the two receivers in two orthogonal directions. We then design a method to segment the finger drawing into pieces, recover and combine the displacement for each piece from each pair of transceivers to finally track and reconstruct the trajectory of finger drawing.

We made the following contributions in this work.

- We propose a CSI-quotient model leveraging on two antennas of a WiFi receiver to achieve finger motion sensing. With this model, CSI signal noise can be largely eliminated because the two antennas on the same receiver share very similar hardware noise and phase offsets. To the best of our knowledge, it is for the first time that sub-wavelength level finger motion tracking can be achieved with commodity WiFi devices.
- We analyze how finger drawing affects the CSI-channel-quotient on two antennas based on our proposed CSI-quotient model, and develop algorithms to sense sub-wavelength level finger motions from CSI data and reconstruct complete finger trajectory from finger displacement information.
- We implement FingerDraw on commodity WiFi devices and conduct real-world user studies. The system can reconstruct the trajectory of finger drawings in the wild. Evaluation results with 20 people in three different environments show that the median tracking error is less than 1.27 cm and the trajectory of 10 digits can be recognized with Microsoft Azure OCR service at an average accuracy of greater than 93.0%.

The remaining of the paper is organized as follows. We survey the state-of-the-arts in Sec. 2, then we provide the background in Sec. 3, and explain why prior works fail in sub-wavelength level subtle displacement sensing. In Sec. 4, we propose a novel CS-quotient model and explore its property for motion sensing at high precision. We show our basic idea of finger tracking in Sec. 5 and detail the implementation of FingerDraw in Sec. 6. In Sec. 7, we evaluate the performance in various settings. Sec. 8 discusses the limitation and opportunities. Finally, we conclude our work in Sec. 9.

2 RELATED WORK

Gesture recognition and motion sensing have been broadly studied in the literature. It can be classified in three categories: Non-WiFi-based gesture recognition, WiFi-based gesture recognition and WiFi-based micro-motion tracking.

Non-WiFi-based gesture recognition Non-WiFi based gesture recognition techniques usually require specialized hardware devices to capture predefined signals to recognize gestures.

Imaging-based systems (such as Xbox Kinect [29], and leap motion [9]) use monochromatic infrared cameras and LEDs to build body-depth imaging. These solutions are all based on the line-of-sight (LOS) operation. The required specialized equipment are rather expensive.

Sonar-based systems use one or more active sonars and sound receiver for fine-grained finger tracking. LLAP [42] uses the phase change of the continuous wave sound signal for motion sensing. FingerIO [23] transmits specially modulated OFDM signals and locates finger based on the change of the echo profiles. Strata [49] combines the frame-based approach and the phase-based approach. All the above sonar based systems provide reasonable tracking accuracy. Like camera-based solutions, sound-based systems have privacy concerns, since they record all the sounds in a room, including human conversations.

TV and cellular transmissions have been used for recognizing coarse-grained hand, arm and leg gestures [3, 54]. Radar based systems such as Google Soli [11] show the ability to track minor finger movement by constructing Doppler profile using 60GHz radar signals. WiSee [25] uses specialized devices (USRP) with OFDM modulated signals to extract the Doppler shift of the movement to recognize gestures. WiTrack [2] also uses custom radar transmissions to detect pointing gestures. Such systems, however, require embedded chips to capture and process radar signals.

RFID based techniques have been widely used to sense micro-motions of fingers. RF-IDraw [39] traces a finger with RFID tag attached using multiple antenna arrays to enable a virtual touch screen application. RF-finger [36] enables device-free finger tracking using specially arranged tag arrays with one RFID antenna.

In a nutshell, the above systems often require dedicated hardware which are not cost-effective, and they may work on specific occasions, but their further deployment is limited.

WiFi-based gesture recognition Recent WiFi-based gesture recognition approaches are learning-based. They use CSI values obtained from widely deployed commodity WiFi devices for gesture recognition.

Most learning-based gesture recognition systems are trained using CSI waveforms. They assume that each gesture has its own unique CSI signal variation pattern. The signal patterns are compared either in the time-domain or frequency-domain with the reference patterns in the training set. WiFinger [32] extracts patterns in time domain using principal component identification and compares waveform shapes with Dynamic Time Warping (DTW) to identify different gestures. WiGest [1] and WiAG [35] extract frequency distributions using Discrete Waveform Transform (DWT) to recognize different hand gestures. WiMu [34] uses frequency features of short time Fourier transform (STFT) and generate virtual samples to enable multi-user gesture recognition. Mudra [53] uses the “cancellation” between two antennas in a receiver which is sensitive to finger movements. It then compares the patterns in frequency distribution for recognizing different gestures. WiAG [35] and Widar 3.0 [55] propose orientation-independent gesture recognition solutions. WiAG uses gesture translation function to generate virtual samples of different orientations to increase the training set. Widar 3.0 use multiple WiFi device pairs to build cross-domain body-coordinate velocity profile from Doppler information for gesture recognition.

The above systems essentially leverage on patterns, they can only recognize a pre-defined gesture set, not for arbitrary finger motion tracking.

WiFi-based micro-motion tracking The systems in this category exploit WiFi signals for motion tracking without training. They track objects by tracking changes in signal arriving angles or variations of signal propagation path length.

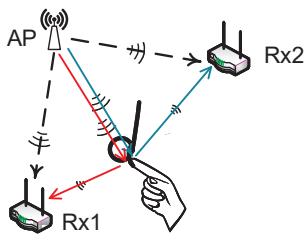


Fig. 1. illustration of FingerDraw design. An AP is transmitting WiFi signals, two WiFi receivers each equipped with two antennas are placed orthogonally for the touchless finger-draw tracking.

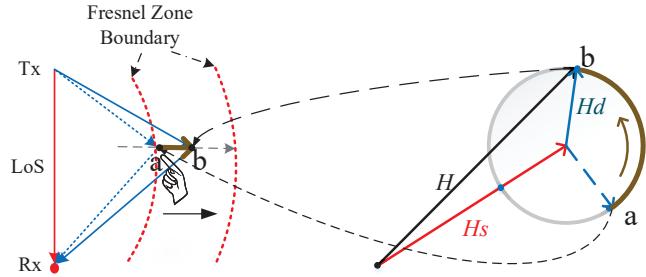


Fig. 2. illustration of the CSI dynamics. CSI signal is composed of signals from static and dynamic paths. As a finger moves from a to b , the CSI signals form an arc on the complex plane.

WiDraw [31] shows signal Angle-of-Arrival (AoA) can be used to track hand motions. It senses hand/arm occlusions of incoming wireless signals from 20-30 known WiFi devices by detecting signal drops in certain direction. QGesture [48], WiDance [27] and Doppler-MUSIC [18] sense target motion by estimating the length change of the reflected path with both CSI amplitude and phase information. Due to the added random offset in CSI phase[15, 33, 45, 48], they need to compensate phase offset before use. QGesture [48] propose a method to estimate the Carrier Frequency Offset (CFO) and Sampling Frequency Offset (SFO) and stabilize CSI phase for the movement distance and direction estimation of human hands. WiDance [27] and Doppler-MUSIC [18] use conjugate multiplication of two CSIs from two receiver antennas to cancel phase offset and extract Doppler speed information of a moving human target. The above methods also suffer from noise in raw CSI amplitude and render low SNR in the extracted Doppler speed profile. They also require target to move at least several wavelengths for the motion to be correctly extracted.

In summary, existing WiFi-based systems are either pattern-based for recognizing a pre-defined set of hand gestures, or model-based for tracking coarse-grained motions that span multiple wavelengths. None of them has the capability of tracking centimeter-level finger motions.

3 BACKGROUND & CHALLENGES

To understand why existing WiFi sensing approaches fail to sense sub-wavelength level finger motions, we first introduce the mapping between finger motion and CSI signal variation in the context of Fresnel Zones. Then we use an example to illustrate the sensing limitation with CSI waveform patterns used in most of the existing gesture recognition systems.

3.1 Sensing Finger Motions with CSI Signals

As illustrated in Fig. 2, a finger moves within the sensing range of a pair of WiFi transceivers. The received signal strength is the superposition of all the path components from the static Line-of-Sight (LoS) propagation and environmental reflections, and the dynamic reflection path of a finger. For a radio wave with a wavelength of λ , when it travels along a reflected path with a length of d , its phase is shifted by $2\pi d/\lambda$. As a finger moves, the reflected path changes in length, and the phase of the dynamic component changes accordingly.

We study CSI signal variations with the Fresnel Zone model [37, 44]. As illustrated in Fig. 2, a WiFi transceiver pair is placed at a fixed location and a finger moves in the Fresnel Zones for a short distance. CSI is the superposition of components from all the paths a radio wave travels through, which can be divided into static and dynamic

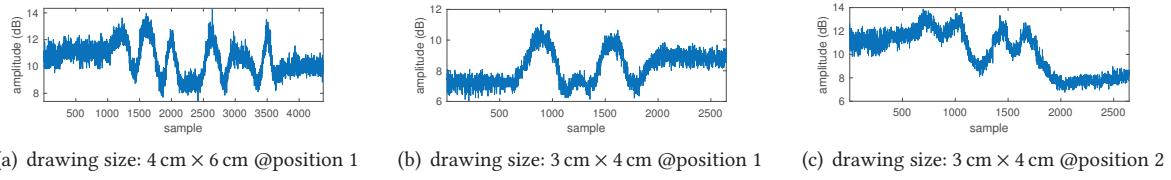


Fig. 3. the recorded CSI waveforms for the same drawing of letter ‘d’ in three cases , we can see clearly the waveform patterns are inconsistent. There are noticeable noises in the CSI waveforms.

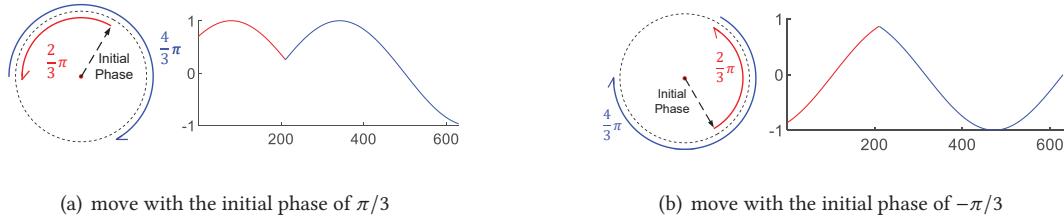


Fig. 4. the same move with different initial position lead to different initial phase of the dynamic phasor, thus the different waveforms

components. Apparently, the static component is affected by the surrounding environment and the LoS of transceivers while the dynamic component is determined by the reflection path from the finger. Mathematically, the total CSI can be denoted as follows [37, 41, 44]:

$$H(f, t) = H_s(f, t) + H_d(f, t) = H_s(f, t) + A(f, t)e^{-j2\pi \frac{d(t)}{\lambda}} \quad (1)$$

where $H_s(f, t)$ is the static phasor component, $A(f, t)$, $e^{-j2\pi \frac{d(t)}{\lambda}}$ and $d(t)$ are the attenuation, the phase shift and the path length of the dynamic phasor component $H_d(f, t)$, respectively. The CSI waveform is the magnitude of the complex CSI data $H(f, t)$ over time.

As the finger moves, the reflected path changes its length and the dynamic phasor component rotates accordingly. If the reflected path changes less than a wavelength in length, the superimposed CSI changes along a circular arc. The rotation direction of the dynamic component is tightly coupled with the motion direction of the finger, causing the reflected path increases or decreases in length.

An interesting finding in previous studies [37, 51] shows that when an object moves across a series of Fresnel Zones, the received signal shows a continuous sinusoidal-like waveform. If the reflection path length change caused by a moving object is shorter than one wavelength, the received signal is just a fragment of the sinusoidal-like waveform. This finding implies CSI waveform patterns may be different for similar finger motions, it depends on how many Fresnel Zones a finger moves across at the time of drawing, as well as the initial position in the Fresnel Zone.

3.2 Pattern Inconsistency Problem in Sub-Wavelength Level Gesture Recognition

The common problem in the existing gesture recognition when recognizing finger drawings is pattern inconsistency. Previous researches [35, 55] show that the gesture orientation and location influence CSI waveform shapes.

The problem is even worse in recognizing centimeter-level finger-draws, as drawing size and initial drawing position also matters. Slight change of finger drawings may result in totally different CSI waveform. To reveal the issue, we conduct an experiment by placing a pair of WiFi transmitter and receiver at fixed locations with a spacing of 70 cm. We ask a volunteer to draw a letter ‘d’ with a finger in the air in three cases. Case a) and Case b) are different in letter size, while Case b) and Case c) are different in their initial drawing positions. The results are shown in Fig. 3. It is clear to see that the recorded patterns of CSI waveforms in the three cases are quite different. The waveform of Case a) in Fig. 3(a) has much more observable fluctuations than that of Case b) in Fig. 3(b), as a longer stroke in Case a) moves across more Fresnel Zone boundaries. To understand why initial drawing position matters in Cases b) & c), we study the following simulation.

We fix the other variables and change only the initial position to observe CSI waveforms. Suppose the reflected path length decreases first for one-third wavelength, then increases for two-thirds wavelength. According to the Fresnel Zone model, we expect the dynamic phasor component rotates one-third of a circle clockwise and then two-thirds of a circle counter-clockwise, respectively. The initial phase of the dynamic phasor component is determined by the initial position where an object starts to move. If we repeat the same movement two times each with a slightly different initial position, the CSI signals are different as they contain the same static phasor component but different time-varying dynamic phasor component, as shown in Fig. 4.

Centimeter-level finger motions usually induce sub-wavelength scale reflected path changes, thus the CSI waveforms are highly depend on the drawing position and size. Learning-based approaches essentially rely on intensive training to achieve good results. Previous learning-based gesture recognition approaches are sensitive to the drawing position and drawing sizes. It’s difficult to collect the CSI waveforms of different people with all gesture-drawing positions and gesture-drawing sizes for training, thus the gesture recognition performance cannot be guaranteed.

4 SENSING MOTION WITH CHANNEL QUOTIENT

Instead of adopting learning-based approaches, in this paper, we use a mathematical method to reconstruct centimeter-level finger-draw trajectories, thus are free from the difficulties of location dependency and drawing size dependency in traditional pattern-based machine learning methods. In achieving this goal, we construct a new transformed signal by taking the CSI quotient between two antennas of the same WiFi receiver. The advantage of using the CSI quotient is two-fold: 1) it cancels out most noise in raw CSI signals, thus it is very sensitive to tiny finger movements, 2) it provides fully available phase and amplitude information of a complex value and still preserves the correlation between motion displacement and CSI signals.

4.1 Noise Cancellation

MIMO technology has been widely used in wireless communication. Multiple antennas increase signal diversity, and hence provide us an opportunity to cancel out signal noise more efficiently. For example, the phase difference over different antennas has been explored in Angle-of-Arrival (AoA) estimation [15] and motion speed estimation [17]. The benefit of using phase difference is that it contains no offset noise. Since different antennas on a receiver share the same clock, the phase offsets such as CFO and SFO are identical [33, 48]. The subtraction of the phases on two antennas of a receiver removes these phase offsets completely.

CSI quotient signal cancels noise in a similar way. The channel quotient information on a receiver is calculated by dividing the CSI value of the two antennas on that receiver. The result is a new complex value called CSI quotient (also named as channel quotient). Based on the division operation of complex values, the phase of CSI quotient is the phase difference between two antennas with all the offset subtracted and the amplitude of CSI quotient is the amplitude ratio between two antennas. We find that the high amplitude impulse and bursty noise which are hard to eliminate in raw CSI amplitude have been canceled out in this division operation. This

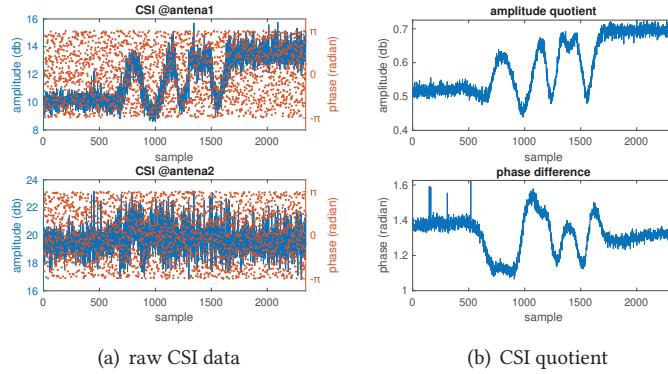


Fig. 5. both the impulse noise in the raw CSI amplitude and random offset in the raw CSI phase are greatly suppressed in the channel quotient signal

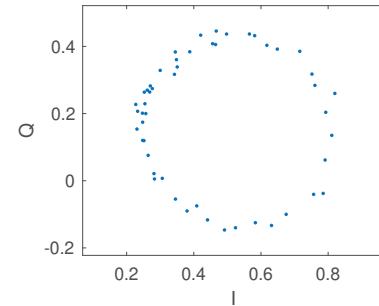


Fig. 6. the samples of channel quotient form a perfect circle on the complex plane as the reflected path changes for exactly one wavelength due to an object moves

cancellation works because the impulse noise in CSI amplitude is scaling noise, which amplifies the power at each antenna on the same receiver at the same level. In other words, although the power scaling varies over time, it is consistent across the antennas of the same receiver.

Figure 5 shows raw CSI and channel quotient. In Fig. 5(a), the samples of CSI phase are randomly distributed in $[-\pi, \pi]$ at all the antennas, and we observe severe impulse noise in the amplitude. After the division of two CSI signals, we have most noise removed both in amplitude and phase of the channel quotient.

Unlike phase difference that has been explored previously, amplitude quotient between two antennas has not been studied before in WiFi sensing. The main reason is probably due to lack of clear physical meanings. However, we observe that CSI quotient can form a perfect circle in the complex plane when the reflected path of a moving object changes for exactly one wavelength, as shown in Fig. 6. This implies that correlation exists between the displacement of an object and the CSI quotient dynamics, which implies the possibility of using the orthogonal information of a complex CSI quotient value, i.e., both phase difference and amplitude quotient, for motion sensing of a moving target. With CSI quotient as a new transformed base signal, it not only achieves high SNR by canceling the noise in raw CSI amplitude and random offsets in the raw CSI phase but also preserves orthogonality between phase difference and amplitude quotient, making it possible to capture the tiny motion of a finger.

In the next section, we introduce our CSI-quotient model and explore its properties for sub-wavelength level finger tracking.

4.2 CSI-quotient Model

The CSI-quotient model builds upon the connection between the motion displacement of a target and the signal variations of CSI quotient like what an ideal CSI signal does. Suppose we have a pair of transceivers at two fixed locations and a moving target such as a finger. There are static signal components of the combination of signals from the LoS and the reflected environmental static objects, and a dynamic signal component reflected from the finger, as introduced in Sec. 3.1. Then the CSI quotient has the form of:

$$H_q(f, t) = \frac{A_{noise}(f, t) e^{-j\theta_{offset}(f, t)} (H_{s1}(f, t) + H_{d1}(f, t))}{A_{noise}(f, t) e^{-j\theta_{offset}(f, t)} (H_{s2}(f, t) + H_{d2}(f, t))} = \frac{H_{s1}(f, t) + H_{d1}(f, t)}{H_{s2}(f, t) + H_{d2}(f, t)} \quad (2)$$

where A_{noise} is the impulse noise in amplitude, θ_{offset} is the random phase offset, $H_{s(1,2)}(f, t)$ are the static phasor components and $H_{d(1,2)}(f, t)$ are the dynamic phasor components for the two antennas of the same receiver, respectively.

We make two assumptions to simplify the CSI quotient model. Firstly, we assume the static components and the magnitude of a channel dynamic component remain unchanged at the two antennas of a receiver as a target moves for some short distance. This assumption also holds in CSI-Speed model [41] and Fresnel Zone model [51]. Secondly, we assume the angle-of-arrival of the signal reflected from moving target changes very little as the target moves for some short distance.

With these assumptions, we rewrite Equ. 2 into a new form as follows:

$$H_q(f, t) = \frac{H_{s1}(f, t) + A_1(f, t)e^{-j2\pi\frac{d_1(t)}{\lambda}}}{H_{s2}(f, t) + A_2(f, t)e^{-j2\pi\frac{d_2(t)}{\lambda}}} = \frac{H_{s1}(f) + A_1(f)e^{-j2\pi\frac{d_{diff}}{\lambda}}e^{-j2\pi\frac{d_1(t)}{\lambda}}}{H_{s2}(f) + A_2(f)e^{-j2\pi\frac{d_2(t)}{\lambda}}} = \frac{az + b}{cz + d} \quad (3)$$

where $A_{(1,2)}(f, t)$, $e^{-j2\pi\frac{d_{(1,2)}(t)}{\lambda}}$ and $d_{(1,2)}(t)$ are the attenuation, the phase shift and the path length of dynamic component $H_{d(1,2)}(f, t)$ for the two antennas of a receiver, respectively, and $z = e^{-j2\pi\frac{d_2(t)}{\lambda}}$ is a unit complex variable, its phase representing the change of reflected path, the coefficients a, b, c, d are complex numbers.

We observe from the new form that there is only one independent variable z left. It means the value of channel quotient solely depends on how the total length of the reflected path changes for any given radio frequency. Mathematically, the latter part of Equ. 3 is the Möbius transformation known in the field of geometry and complex analysis [5].

Möbius transformations are fundamental complex maps. They are invertible meromorphic functions, and so are conformal everywhere. The transformations preserve angles, map every circle to a circle on the complex plane [5], provided $|c| \neq |d|$. When applied in sub-wavelength level motion sensing, it has a geometrical property that they map arcs of circles in CSI to arcs of circles in CSI quotient. That is to say, we can sense subtle finger movements by observing the rotations of arcs in CSI quotient, instead of observing it in raw CSI.

We now investigate the correlation between the rotation direction of the arc in CSI and the one in CSI quotient, as it indicates the motion direction of a finger. To study how the rotation direction of arcs in CSI quotient is related to the moving direction of an object which causes the reflected path length increases or decreases, we rewrite Equ. 3 into a more concise form as follows.

$$g(z) = \frac{az + b}{cz + d} = \left(\frac{bc - ad}{c}\right)\frac{1}{cz + d} + \frac{a}{c} = \frac{\rho e^{i\theta}}{z + \alpha} + \beta \quad (4)$$

for appropriate $\alpha, \beta \in \mathbb{C}$ and $\rho, \theta \in \mathbb{R}$. z is the unit vector on the complex plane with its phase represents the signal phase of the reflected path d , subject to $z = e^{-j2\pi\frac{d(t)}{\lambda}}$.

Assume an object moves with its reflected path length changing, then z rotates accordingly. Geometrically, a Möbius transformation $g(z)$ can be obtained as the composition of translation $z \mapsto z + \alpha$, inversion $z \mapsto 1/z$, dilation $z \mapsto \rho z$, rotation $z \mapsto ze^{i\theta}$ and another translation $z \mapsto z + \beta$. This decomposition makes many properties of Möbius transformation obvious.

As we can see, translation, rotation, and dilation transformations are quite straightforward, as shown in Fig. 7. They do not change the rotation direction and rotation phase of variable z . The only transformation we need to be aware of is inversion transformation (Fig. 7(d)).

Inversion transformation has the form of $1/z$. It has a property that $1/\bar{z}$ and z are collinear with the origin and that the product of their distances from the origin is equal to 1, where \bar{z} is the conjugate complex of z . This transformation is conformal, thus it maps arc of z to arc of $1/z$. As an example shown in Fig. 7(d), $|P_1||P_2| = 1$, and $\overline{P_3} = P_2$, so $1/P_1 = P_3$. The rotation direction of the arc depends on whether the circle of z contains the

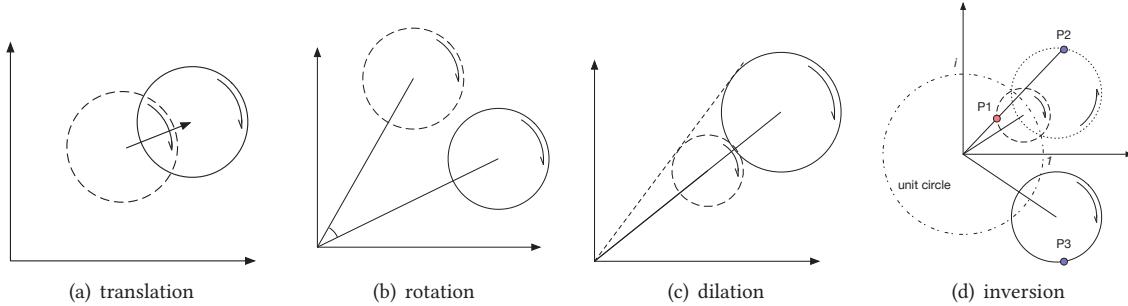


Fig. 7. a Möbius transformation can be decomposed to four transformations. All the four transformations preserve the rotation direction of the arc in our finger tracking case.

origin point. For finger tracking, the signal reflected from a finger is much weaker than the environmental static component. We always have $|d| > |c|$ in Eqn. 4 which means the circle of z does not contain the origin point. It is easy to verify that the rotation direction of the arc in z remains unchanged before and after inversion.

We summarize the properties of our CSI-quotient model as follows.

- P1 The trace of CSI quotient in the complex plane is a perfect circle when the reflected path of a moving object changes for exactly one wavelength. As the reflected path changes for multiple wavelengths, the circle of CSI quotient rotates the same number of rounds accordingly.
- P2 When the reflected path changes less than a wavelength in length, the CSI quotient forms a circular arc whose radian measure roughly matches the length change of the reflected path.
- P3 The rotation direction of the arc in CSI quotient is tightly related to the moving direction of the target in the Fresnel Zone.

4.3 Model Verification

In this subsection, we conduct experiments to empirically verify the above three properties of CSI-quotient model. We use a transmitter (Tx) and a receiver (Rx) to record CSI, and a metal plate acts as a perfect signal reflector as shown in Fig. 8(a). We drive the metal plate using a high precision THK programmable linear motion slider at a constant speed. The slider is arranged perpendicular to the LoS between transceivers. The frequency of WiFi signal is set to 5.24 GHz in the experiment.

Verification of P1 We separate the transmitter and the receivers apart for 1.2 m and the metal plate moves outward for 1 m in distance. We count the number of circles in CSI quotient. The initial position of the plate is at 1.2 m perpendicular to the LoS. We expect to see 33.09 periods based on the calculation. And we get 33 circles in the CSI quotient signals, which is very close to our claculation.

Verification of P2 (Figure 9 Case 1-4). We separate the transmitter and the receivers apart for 1.8 m and the start positions of the metal plate are set to 1 m in Case 1&2 and 1.6 m in Case 3&4, respectively. The moving distances for the plate are 5 mm in Case 1&3 and 20 mm in Case 2&4, respectively. The motion direction of the plate in all the four cases are towards outside. Figure 8(b) shows the settings of the four cases.

The results are shown in Fig. 9. In all the cases, the length changes of the reflected path are less than a wavelength, so the channel quotients are only arcs. The motions for Cases 1 and 2 start from the same location, their initial phases are identical. The same as Cases 3 and 4. The moving distances in Cases 1&3 are shorter than that in Cases 2&4, so the phase change of the arcs are smaller in Fig. 9(a) and Fig. 9(c), compared with the arcs in Fig. 9(b) and 9(d).

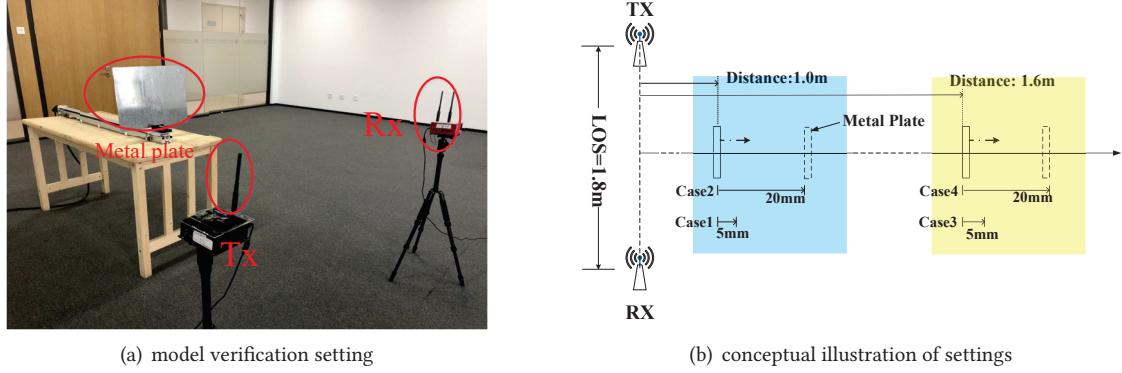


Fig. 8. model verification using a linear motion slider and a metal plate

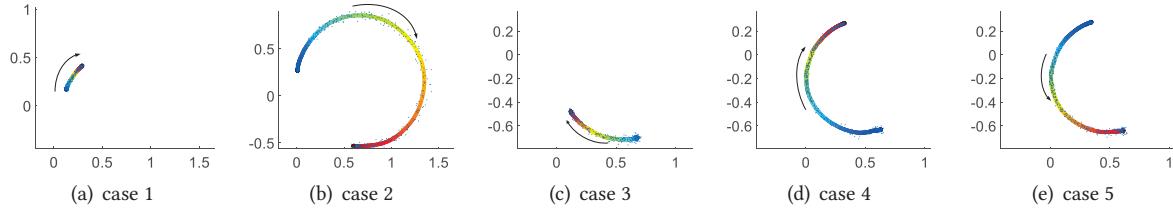


Fig. 9. the exhibition of CSI quotient arcs in the complex plane as object moves in five cases, the dots are the samples of CSI quotient, and the line is the denoised CSI quotient with the color indicates the rotation direction of arcs.

Verification of P3 (Figure 9 Case 5). The setting is similar to Case 4 except the moving direction is backward. As shown in Fig.9, the directions of how arcs of CSI quotient rotate are marked with arrows and colors. The sequence order of the samples is painted from dark blue to light red. We can observe that, as the object moves outward, the phase of arc in channel quotient rotates clockwise, as in Fig. 9(d); while as the object moves with its reflected path length decreases, the phase of arc in channel quotient rotates counter-clockwise, as shown in Fig. 9(e).

In summary, the channel quotient among antennas on the same receiver efficiently removes most impulse noise in CSI amplitude and random phase offsets in CSI phase. It provides much higher SNR than raw CSI signals on a single antenna. The channel quotient provides fully orthogonal information on the complex plane and thus is an ideal base signal for location independent sub-wavelength level sensing. The phase change of an arc in the channel quotient can be mapped to the motion of the object in the sub-wavelength scale. It makes sub-wavelength level finger tracking possible.

5 UNDERSTANDING CSI DYNAMICS OF FINGER DRAWINGS

We use an example to illustrate how the motion of finger-draw dynamically changes CSI signal, as well as channel quotient. The key observation is that the stroke is segmented into parts due to the dynamic reflection path changes. Each piece of stroke corresponds to a piece of arc in CSI quotient signal. Then we explain how to extract displacement information from CSI quotient signal piece by piece and splice them together. The reconstruction of finger drawing trajectory in 2D plane requires information from two pairs of transceivers. We

find the optimal strategy is to place the devices orthogonally based on the understanding of Fresnel Zones theory. The reconstruction involves an iterative process to calculate each location point of the finger. With the above understandings, we summarize the steps of reconstructing any finger drawing trajectories from CSI quotient signals.

5.1 Mapping Finger Stroke to CSI Quotient Signal by Segmentation

Assume the power of the reflected path of a finger is consistent as the finger moves for some short distance, then the CSI signal is determined by the phase of the dynamic component, according to the background introduced in Sec. 3.1. That is, the dynamic phasor component of CSI rotates with respect to the changes in the reflected path of the finger. As the length of the reflected path decreases, the dynamic component rotates counter-clockwise, and vice versa.

Consider we write a letter ‘d’ with a finger near a pair of WiFi transceivers, as shown in Fig. 1. To understand how the reflected path changes in length as we draw the letter, we study the drawing in Fresnel Zones. As illustrated in Fig. 10(a), the transceiver devices are located at two foci of the Fresnel Zone ellipses, each elliptical boundary of Fresnel Zones indicates the reflected path is equal in length, as long as the target is on the same ellipse. With the auxiliary lines of Fresnel Zone boundaries, it is easy to observe the reflected path length changes as we write. When the finger moves towards inner ellipse of Fresnel Zone boundary, the length of the reflected path of the finger decreases, and vice versa.

Drawing of letter ‘d’ is composed of a circle stroke and a straight line stroke. In the specific setting illustrated in Fig. 10(a), when we draw the circle stroke of letter ‘d’, the length of the reflected path decreases first, then increases. The same observation can be found when we draw the line stroke. If we consider the monotonous changing section of the drawing based on the increase and decrease of the reflected path, then we can segment the continuous drawing of letter ‘d’ naturally into four stroke pieces. In this view, each piece of the stroke causes the dynamic component of CSI to rotate in only one direction, leaving a single trace of an arc in the complex plane, as shown in Fig. 10(b). Between two stroke pieces, the reflected path stops to change its length, and the arc in complex plane stops to rotate. It is also true for CSI quotient, as the arcs in CSI quotient rotates in the same direction with similar phase angles as the arcs in CSI in case of finger tracking, as we have learnt in Sec. 4.2. We can see in Fig. 10(c) that our real data recording of the channel quotient in the complex plane contains four pieces of arcs, each represents a piece of stroke for letter ‘d’.

The segmentation in Fig. 10 solely depends on how the transceiver devices are placed relative to the drawing. If the same drawing happens in the Fresnel Zones of another pair of devices, then the length of the reflected path changes in a different way. Fig. 11 illustrates a drawing in Fresnel Zones of another pair of WiFi transceivers, where there exist only three monotonous segments for the same drawing.

5.2 Mapping CSI Quotient Signal to Finger Motion Displacement by Extracting Phase Changes

After segmentation, we have several pieces of arcs in CSI quotient signal. Next, we extract the displacement in Fresnel Zones, i.e., the change of the reflected path length from it. The basic idea of inferring the change of the reflected path is to extract the phase change in a piece of rotating circular arc. As we have learned in Sec. 3.1, the length changes of the reflected path can be estimated by measuring the phase change of the dynamic phasor component, and the displacement information can be calculated from it. Some skills are needed in order to extract this information from pieces of arcs. Since the finger motion causes sub-wavelength changes in the reflected path, there are no complete circular rotations. As a consequence, we cannot use the same method proposed in QGesture [48] to estimate the static component by averaging the CSI data over a sliding window and then separating out the dynamic phasor component. It is because determining the circular center of a piece of arc accurately becomes challenging. Instead, we track the phase change of the dynamic component by measuring the

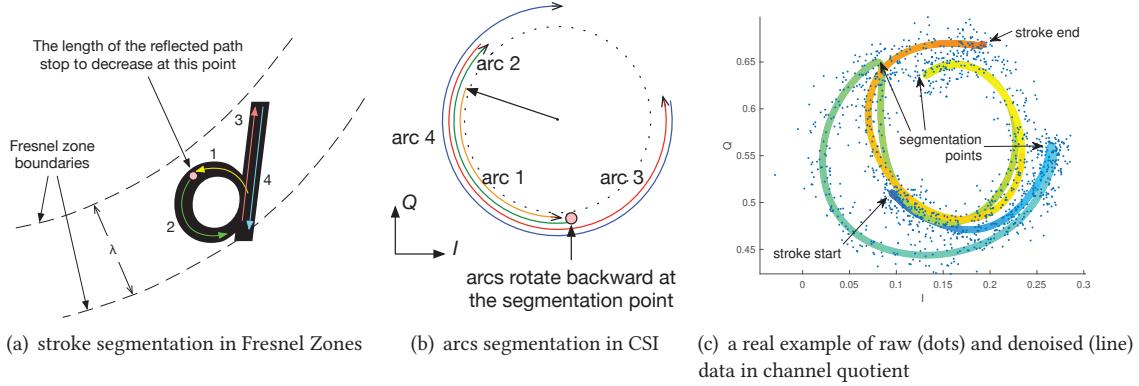


Fig. 10. A finger-draw of ‘d’ is segmented into four pieces based on the displacement in Fresnel Zones. The arcs in CSI quotient rotate accordingly to the displacement in Fresnel Zones, with the phase changes represent the displacement

changes of the slopes for the tangent line at each position on the arc. The two phase changes are identical based on some geometry knowledge, as illustrated in Fig. 12. This method has two advantages, i.e., 1) no need to worry about the static component, 2) computationally efficient, as the tangent calculation only involves subtraction of two adjacent sample points on the arc.

With the tangent slope information for a segment piece of drawing, we can calculate how phase changes over time by subtracting the tangent slope of two consecutive samples. The physical meaning of the phase change is the length difference of the reflected path between two consecutive observation samples as the finger moves. With this information, we can calculate the displacement information in Fresnel Zone for each sample interval, based on the following equation:

$$\Delta d(t) = \frac{\Delta\theta(t)\lambda}{2\pi} \quad (5)$$

where $\Delta d(t)$ is the length change of the reflected path, and $\Delta\theta(t)$ is the phase change of the dynamic component. Then the displacement information of each segment piece of stroke is spliced together to provide the whole information of the finger drawing of a letter.

5.3 Mapping Displacement to the Finger-draw Trajectory

Now, we have the displacement information that describes changes of reflected path length in Fresnel Zone of one transceiver pair. However, the patterns of displacement changes for a drawing obtained from a single pair of transceivers is not enough to recover the trajectory. As a 2D finger-draw trajectory is composed of displacement in two orthogonal directions. We need at least two displacement information from different transceiver pairs. Hence, we add another receiver which is placed orthogonally to the first pair of transceivers to provide more information and use the displacement information reported at the two receivers to fully recover the finger-draw trajectory.

The placement strategy of WiFi device pairs is based on the observation from Fresnel Zones. Theoretically, the trajectory can be recovered by solving a group of ellipse equations for any placement of WiFi devices provided they are not co-linear. However, we find that the optimal placement of devices is placing another receiver orthogonally to the first pair of transceivers, and keeping their length of LoS equal, as shown in Fig. 13. Then at

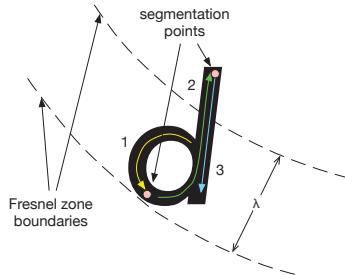


Fig. 11. the segmentation of a finger-draw depends on where the transceivers locate

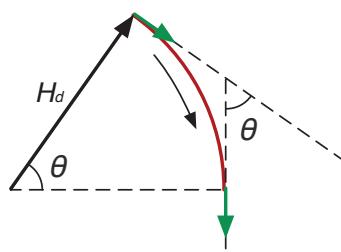


Fig. 12. the phase change of the dynamic phasor component can be estimated by measuring the slope change of its tangent line on the arc

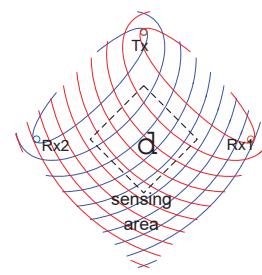


Fig. 13. the crossed Fresnel Zones and the optimal sensing area

the center of the crossed Fresnel Zones, the space intervals between each Fresnel Zone boundary are equal for the two receiver pairs. The benefit of this placement strategy is that it tolerates more noise in the displacement input, and produces a better shape of the reconstructed finger-draws.

The displacement information in Fresnel Zones we obtain describes how the length of the reflected path changes, not the moving finger itself. We reconstruct the finger-draw trajectory iteratively. We calculate the position of a finger based on the information of the previous position and the displacement related to the two pairs of transceivers, as shown in Fig. 14. This can be done by solving a group of ellipse equations. We first use the previous finger position and the displacement of the reflected path to calculate the total length of reflected path. With the positions of two foci and the total reflected path length, we can write the equation of the ellipse. With two pairs of transceivers, we have two ellipses intersected. The new position of the finger locates at the intersection point of two ellipses. Solving a group of ellipse equations may result in multiple location candidates. However, with the constraint of the sensing area, there will be only one possible solution left.

From the concrete example illustrated above, we summarize the steps of reconstructing any finger drawing trajectories from CSI quotient signals. We know that drawing any characters with a finger in the Fresnel Zones of a transceiver pair produces several segments of arcs in the CSI quotient signals. The segmentation is solely based on how the reflected path changes in length. The phase change on an arc in the CSI quotient corresponds to the displacement changes in Fresnel Zones. With these understandings, to recover the trajectory of a letter needs the following steps:

- (1) Segment the CSI quotient signals of each transceiver pair into pieces of arcs based on its rotation direction.
Each arc rotates in only one direction.
- (2) Extract displacement information of finger for every sample interval on each arc, and splice the information from each arc together for the whole stroke.
- (3) With the displacement information from a pair of orthogonally placed transceiver pairs, we reconstruct the trajectory of finger-draw by calculating the finger position based on the information of previous position and the displacement iteratively. It can be done by solving a group of ellipse equations.

6 IMPLEMENTATION

We now discuss how to put FingerDraw into practice, and detail our design and implementation in this section. The hardware configuration includes three WiFi devices, i.e., a WiFi access point (AP) and two computers (as receivers) equipped with wireless cards. Each receiver is equipped with at least two external omni-directional antennas. The placement of the devices is shown in Fig. 13.

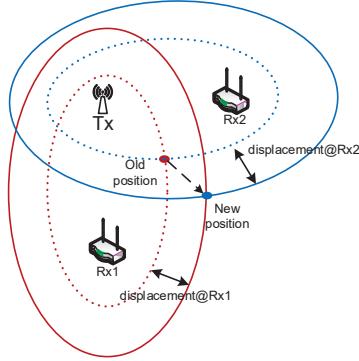


Fig. 14. calculate trajectory iteratively by finding the cross points of two ellipses which are determined by the previous location and the displacement information at the two receivers

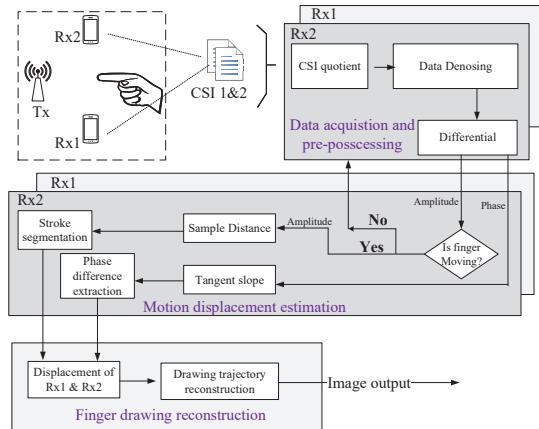


Fig. 15. FingerDraw system overview

FingerDraw consists of three modules: data collection & preprocessing, motion displacement estimation, and drawing trajectory reconstruction. The three modules are sequentially connected and all the steps are essential to the successful reconstruction of finger drawings. The data collection module gathers CSI information from the WiFi receivers and divides them to produce CSI quotient data streams. The motion displacement estimation module extracts displacement information in Fresnel Zones from the CSI quotient streams. And finally, the drawing trajectory reconstruction module rebuilds finger drawing trajectories based on the calculated displacement information. The system information flow is illustrated in Fig. 15. We build a real-time prototype system with MATLAB. It collects CSI streams from TCP/IP sockets and reconstructs finger drawing trajectories as output images. The computation is lightweight to support the real-time processing. It takes 12ms to process the input data of 1 second on an old MacBook Pro (2012 version).

6.1 Data Collection & Preprocessing

6.1.1 Data Collection. We collect CSI data from the two receivers, each receiver has at least two antennas. In the case of Intel 5300 wireless chip, CSI data contains information of 30 sub-carriers. Along with the three antennas that 5300 chip supports, each CSI data is a 3×30 complex matrix. We take element-wise division of two CSI data of the two antennas on the same receiver to get the channel quotient. It produces a 1×30 complex array. With two receivers, we have two CSI quotient streams. We send all these CSI quotient data to a dedicated computer via TCP/IP socket for further processing.

6.1.2 Data Denoising. Data preprocessing involves data denoising and segmentation. Although the division of CSI over two antennas greatly suppresses CSI noise, the CSI quotient data still need to be smoothed in order to reveal the circular shape of CSI quotient clearly for phase change extraction. We use Savitzky-Golay filter to smooth out the CSI quotient data. Savitzky-Golay smoothing, which fits successive subset of data points with low degree polynomial by the method of linear least square and introduces less distortion than band-pass filters in [44]. It is also efficient in smoothing out the complex data. We show an example of denoised CSI quotient data in Fig. 10(c).

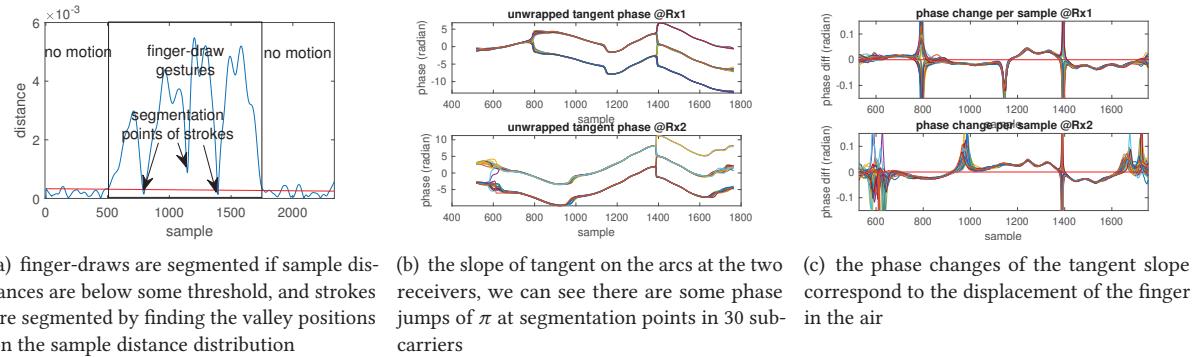


Fig. 16. the displacement are estimated by calculating the slope changes on the arcs of the CSI quotient

6.1.3 Segmentation. The segmentation step is designed to identify a sequential piece of CSI quotient data which corresponds to a single drawing such as a letter. In order to segment finger drawings, we require the subject to keep still for a short while in between the drawings. Since the data samples of CSI-quotient move along a circular arc as a finger moves, and while a finger keeps still, the data samples of channel quotient do not move either. The Euclidean distances between the consecutive samples of CSI quotient data on the complex plane are larger in case of moving than that of in a still case. Fig. 16(a) shows the distribution of the sample distances of a letter ‘d’ drawing. We can distinguish whether the finger moves easily with a threshold value. The threshold can be pretty stable, and it can be determined on the fly by observing how CSI quotient data distributed in an environment with no moving objects.

6.2 Motion Displacement Estimation

6.2.1 Tangent Slope Extraction. Once we cut out a bulk of smoothed CSI quotient data that correspond to a single drawing, we are ready to extract the displacement information in Fresnel Zones due to finger motion. According to Sec. 5.2, the displacement changes in Fresnel Zones can be measured as the phase changes of the tangent lines on the arcs of the CSI quotient data. The tangent line can be approximated by subtracting a sample point with its successor on the arc, thus it is computationally efficient. The result is a sequence of complex vector data, its amplitude represents the Euclidean distances between samples, and its phase represents the slopes of the tangent lines. We extract the phase sequence and unwrap it by changing absolute jumps greater than π to their 2π complement. Figure 16(b) shows the tangent slopes of all the sub-carriers. We can see all the 2π phase jumps are eliminated. However, there are still several noticeable phase jumps of approximately π in the data. The extra π phase jumps need to be handled in the next step before the displacement estimation.

6.2.2 Stroke Segmentation. The most tricky part of our implementation is stroke segmentation. Stroke segmentation refers to the breaking of the CSI quotient data into segments according to the monotonous changes of the reflected path length as we write. This step is crucial to the success of trajectory reconstruction. Although data samples on the arcs of CSI-quotient data correspond to the finger motions, there are plenty of data samples near the segmentation points, which does not correspond to any finger motions. These data samples are also denoised and smoothed as curves in the previous steps, thus extra phase changes are introduced that correspond to no motion. As the dynamic phasor component changes its rotation direction before and after each segmentation point, the extra phase changes are roughly π in the data sequence. It corresponds to additional displacement of

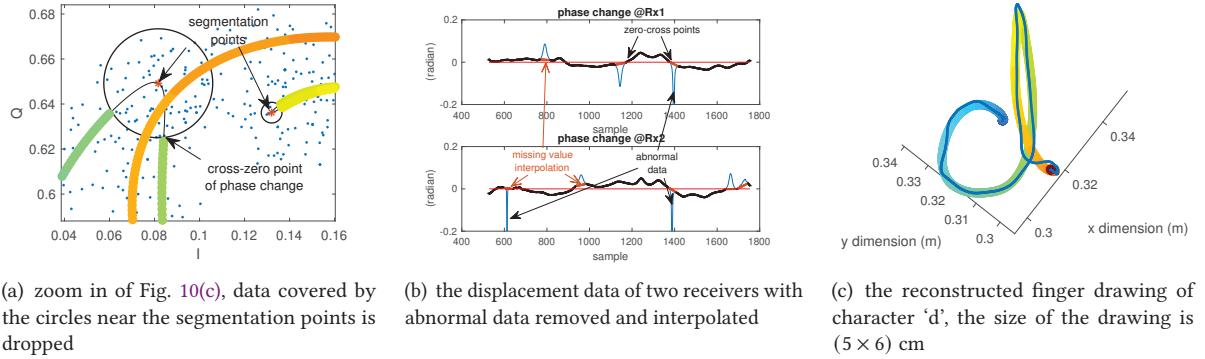


Fig. 17. the stroke segmentation step and the reconstruction of finger drawing from the displacement data

$\lambda/2$ in the reconstructed trajectory, greatly distorting the shape of a drawing. As a result, the data samples near segmentation points need to be discarded before the displacement calculation. It is illustrated in Fig. 17(a).

The segmentation points are identified with the help of sample distance information. As we have learned in Sec. 5.1, the segmentation points located at the spacial positions where a finger changes its moving direction in the air. As illustrated in Fig. 16, we use the fact that the moving speed of the finger changes its direction near these positions. As a result, the sample distances on the complex plane have local minimums at these positions which we can identify by searching all the valley positions in Fig. 16(a).

Once we have all the indexes of the segmentation points identified, we locate the CSI quotient data samples as the centers and draw circles on the complex plane to mark the discard zones. The data samples covered by the circles in these areas are discarded. As shown in Fig. 17(a), the segmentation points are marked as the red asterisk and the continuous CSI quotient curves are broken down into disjointed arcs by the circle of the discard zone. The radii of the circle are decided by finding the positions of the cross-zero data samples in the phase change diagram, as shown in Fig. 17(b). The cross-zero points always exist because the phase changes its sign before and after the segmentation points. After we find the cross-zero points, we locate their positions in the complex plane. The radii are then decided as the distances between these positions and the segmentation points.

6.2.3 Displacement Extraction. The next step is to extract displacement information in Fresnel Zones. We first calculate the phase change information and then convert it to displacement. The phase change information is calculated by subtracting two consecutive phases, as shown in Fig. 16(c). The physical meaning of the phase change is the angle in which a dynamic phasor component rotates between two consecutive CSI quotient samples. Then we discard data near the segmentation points in the phase change sequence and fill the missing data segments by interpolation. It is implemented using a simple linear interpolation method. Finally, we have the phase change data as shown in Fig. 17(b). Based on Equ. 5, we convert phase change to the signal propagation distance change of the reflected path.

6.3 Drawing Trajectory Reconstruction

The displacement information in Fresnel Zones we obtained describes how the length of reflected path changes, not moving finger itself. In order to map the reflected path length change to finger motions, we adopt the similar solution proposed in Widar [26] and Doppler-MUSIC [18]. We assume the initial drawing position is at the center



Fig. 18. illustration of the experimental setting in a meeting room

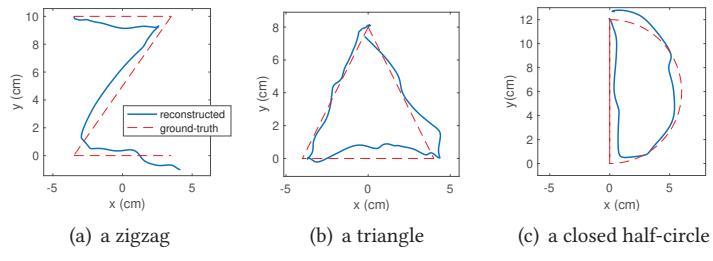


Fig. 19. tracking accuracy evaluation using three templates and the reconstructed trajectories

of the sensing area. Start from this point, we calculate the trajectory position point-by-point using the technique introduced in Sec. 5.3. The reconstructed trajectory is illustrated in Fig. 17(c).

7 EVALUATION

In this section, we evaluate the performance of FingerDraw with commodity WiFi devices. We first describe details about the devices, the environments, and the participants. Then, we show finger-drawing trajectory reconstruction errors using three templates under various conditions. After that, we show the proposed method can recognize finger-drawing digits in natural settings. Then, we compare with the state-of-the-arts to show existing solutions are both position and drawing size dependent in a controlled setting, while our solution achieves consistent performance in these cases. Finally, we show some free-style drawing examples.

7.1 Experimental Settings

7.1.1 Devices: In all the experiments, we use three GigaByte mini PCs as WiFi transceivers. Each receiver is equipped with an Intel 5300 wireless card and three omnidirectional antennas. The receivers are configured to work under the monitor mode, to capture packets from the transmitter simultaneously. We mount one transmitter (marked as Tx) and two receivers (marked as Rx1 and Rx2, respectively) on tripods placed orthogonally. The direction of antennas is placed in parallel to the ground, in order to better capture finger motions as a user sit in front of it. The distance between transmitter and receiver is set to 0.6 m, as shown in Fig. 18. We use open-source Linux CSI tool released by Washington University to collect CSI data from the two receivers [12]. The frequency of WiFi signal is set to 5.24 GHz and the bandwidth is set to 20 MHz. The transmitter sends 500 packets per second at a transmitting power of 15 dBm.

7.1.2 Environments: We test FingerDraw in three environments, i.e., an office ($4\text{ m} \times 5\text{ m}$), a meeting room ($6\text{ m} \times 8\text{ m}$), and an empty hall ($8\text{ m} \times 12\text{ m}$). Both office and meeting room are multi-path rich environments. The office has many types of furniture, including a sofa, two set of desks, a cabinet, and several chairs. The meeting room also has a large conference table, four small long tables, and a lot of chairs. The example settings and the surrounding environments are shown in Fig. 18.

7.1.3 Participants: We recruit 20 participants, i.e., 7 females and 13 males, aging from 21 to 40. They are college students and staffs. Among the participants, four are the authors of this paper (participant No.1-4). The rest of the participants know nothing about our system. All the participants follow the same instructions to conduct experiments. We use cardboard hanging in front of the participants to mark the drawing area. The writing area is about $30\text{ cm} \times 30\text{ cm}$ in size. The participants sit in front of the WiFi devices with their forearms lifted, writing in

the air with their index fingers, as shown in Fig. 18. For each drawing, the participants are asked to keep still first, then move their hands with fingers to draw letters, and in the end stop moving. While drawing, the participants cannot perform other activities. All the finger drawing data are recorded individually and processed off-line for evaluation.

7.2 Tracking Accuracy

We use three templates (a triangle, a letter ‘Z’, and a closed half-circle) to evaluate the accuracy. These templates contain straight-line strokes of different angles and curves, which comprise letters and digits. We print these templates on cardboard and hang them in front of the subjects to act as the ground-truth. We ask the participants to draw the templates with their index finger touching the cardboard and tracking the printed templates on the cardboard. The drawing speed is roughly 5 cm per second if not mentioned.

We set the starting positions of the reconstructed trajectory to coincide with the template and calculate the distance between the template and the reconstructed one as errors. An example is illustrated in Fig. 19. We collect 150 drawings for each template from 6 volunteers in three environments. The overall median error is 1.27 cm with a standard deviation of 1.28 cm. The 50% errors are 1.12 cm, 1.46 cm and 1.29 cm for triangle, letter ‘Z’ and half-circle templates, and the 90% errors are 2.98 cm, 3.38 cm and 3.52 cm for triangle, letter ‘Z’ and half-circle templates, respectively. Fig. 20(a) shows the overall CDF of tracking accuracy of the three templates.

Impact of different drawing sizes: We print the templates on the cardboard in three sizes (4.5 cm × 4.5 cm, 8 cm × 8 cm and 11.5 cm × 11.5 cm for the triangle template, 5 cm × 7 cm, 7 cm × 10 cm and 9 cm × 12 cm for the letter ‘Z’ template, and diameter of 8 cm, 12 cm and 15 cm for the half-circle template, respectively). Some of the templates are smaller than a wavelength (5.8 cm in our case), while others are a litter bigger. We ask a volunteer to draw the templates 50 times each in the office room. The results are illustrated in Fig. 20(b). We can see there is no obvious difference in tracking accuracy among different template sizes, expect that the small half-circle and small triangle perform slightly better.

Impact of different LoS distances: We deploy the system with three different LoS distance between the transmitter and the receiver from 50 cm, 75 cm and 100 cm in the office room and ask a participant to draw the template of letter ‘Z’ 50 times under each LoS setting. The results are shown in Fig. 20(c). We find that the performance under a shorter LoS distance is slightly better than that under a longer LoS setting. The probably reason is that the smaller LoS distance produces stronger reflection signal from hand and finger, thus it is less likely to be influenced by other moving objects in the environment and EM interference.

Impact of different wireless environments: We deploy the system in three different environments (i.e., an office, a meeting room, and an empty hall) and test it over a period of two weeks with all the six participants using the template of ‘Z’ with a size of 7 cm × 10 cm. The results are shown in Fig. 20(d). We find that the environment has noticeable impact on accuracy, but not too much. It is reasonable since our method relies on a stable combination of the static environmental reflection signals at the time of the drawing. It doesn’t matter how these multi-paths superimposed. We also test the influence of activities performed by other people in the room. We observe other people’s activities have little influence provided they stay 3 meters away from the WiFi devices. However, activities such as walking and waving hands near the devices deteriorate our performance greatly.

Impact of lower sampling rates: We use a sampling rate of 500 Hz in most of the experiments, but in practice, the sampling rate can be much lower. To evaluate the influence of sampling rate on accuracy, we test our algorithm using different sampling rates, i.e., 25 Hz, 50 Hz, 125 Hz, 250 Hz and 500 Hz, respectively. A participant draws a template ‘Z’ 25 times for each sampling rate. The results are shown in Fig. 20(e). We find that a higher sampling rate does lead to higher accuracy, but the performance degradation with lower sampling rates is not significant. For example, when the sampling rate is 100 Hz, the accuracy does not change too much compared with the results of higher sampling rates. However, when the sampling rate is lower than 25 Hz, the performance becomes

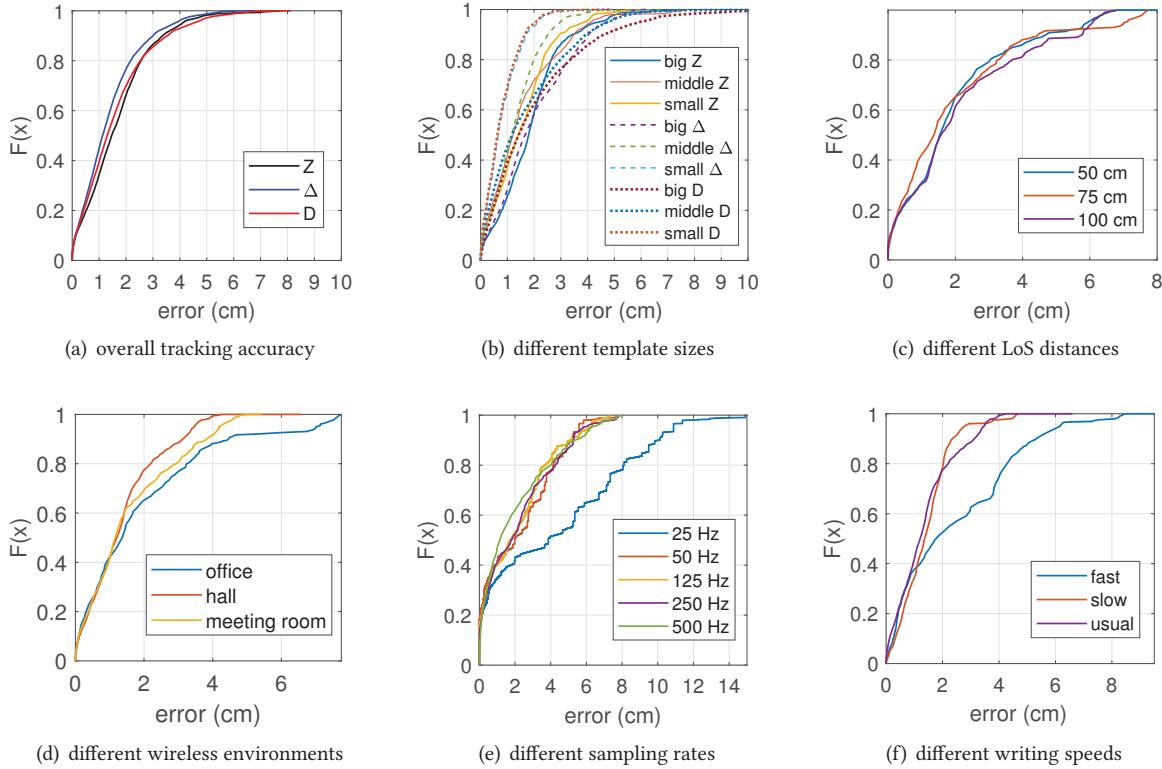


Fig. 20. the tracking accuracy under various settings

unacceptable. Hence the minimum sampling rate we recommend in this setting is 50 Hz. If the participants draw at a higher speed, we need a higher minimum sampling rate accordingly.

Impact of different drawing speeds: We test the drawing accuracy of a template ‘Z’ with three different writing speeds in the office room. The speed is calculated based on the stroke length in the printed template, as well as the time a user spends to finish drawing. We asked a participant to draw the template at three different speeds. The measured motion speed is 9 cm/s, 6 cm/s, and 3.5 cm/s for the fast, usual, and slow speed, respectively. We collect 20 data samples for each speed. The tracking accuracy at three different speeds is shown in Fig. 20(f). We observe that the reconstructed trajectories are recognizable at all three speeds in most cases. But in some tests, the shapes of the reconstructed trajectories at the fast speed is worse than that of at two lower speeds. It comes from the distortions of the misidentified uncut segmentation points.

7.3 Text Recognition

In this section, we evaluate FingerDraw by allowing the users to input digits to a computer by writing in the air naturally, that is, the users write digits without templates. We interface FingerDraw with Microsoft Azure OCR service. As users write in the air, we feed the image of reconstructed trajectories of digits into Azure OCR and

use the recognition functionality to interpret the digits written by users. We collect and count the recognized characters to evaluate the accuracy of text input.

The experimental settings are the same as the one in tracking accuracy test. We hang blank cardboard to mark the writing area. The participants can draw digits in normal or italic type, and they can draw in different sizes and initial positions as well. We do not limit the drawing speed of the participants. They use the speed from 3 cm/s to 6 cm/s based on the off-line analysis.

We collect 20 samples for each digit from 20 participants over a period of several weeks. In total, $20 \times 10 \times 20 = 4000$ data are collected. Among all the data, the tests of participant No.1,4,6,8,12,13,14 are conducted in the office, the tests of participant No.2,7,9,11,18 are conducted in the hall, the rest tests are conducted in the meeting room. An example of the reconstructed digits is illustrated in Fig. 24(a) Table 1 shows the detail of recognition accuracy. We plot the overall recognition rate for each participant in Fig. 22. We can see that most of the users achieve an accuracy of greater than 90%. The first four participants marked with an asterisk indicate they are the authors. We observe no obvious difference in recognition rates among the participants.

In all the digits, the overall recognition rate of digit ‘8’ is lower than the others. It is because the starting and ending strokes of digit ‘8’ need to be closed. Because of the accumulated error of FingerDraw, when the glyph is not closed properly, the reconstructed trajectory is sometimes difficult to recognize. This also happens to digits ‘0’ and ‘6’, as in some rare cases these two digits are confusable for the same reason.

Table 1. The digits recognition accuracy of 20 participants (in percentage)

| User | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | Avg |
|-------|-----|------|------|------|-----|------|-----|------|------|------|------|
| No.1* | 90 | 80 | 100 | 100 | 90 | 100 | 100 | 100 | 80 | 90 | 93 |
| No.2* | 100 | 100 | 100 | 100 | 100 | 90 | 100 | 100 | 85 | 100 | 97.5 |
| No.3* | 100 | 100 | 100 | 100 | 95 | 100 | 80 | 100 | 80 | 100 | 95.5 |
| No.4* | 100 | 100 | 90 | 90 | 90 | 100 | 90 | 100 | 80 | 80 | 92 |
| No.5 | 100 | 100 | 90 | 80 | 95 | 90 | 100 | 90 | 80 | 80 | 90.5 |
| No.6 | 90 | 100 | 90 | 100 | 90 | 90 | 90 | 90 | 85 | 100 | 92.5 |
| No.7 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| No.8 | 90 | 100 | 90 | 90 | 100 | 100 | 90 | 100 | 90 | 100 | 95 |
| No.9 | 90 | 100 | 90 | 80 | 85 | 90 | 90 | 80 | 90 | 90 | 88.5 |
| No.10 | 90 | 100 | 100 | 100 | 90 | 90 | 80 | 100 | 90 | 80 | 92 |
| No.11 | 100 | 100 | 100 | 100 | 95 | 100 | 80 | 90 | 100 | 100 | 96.5 |
| No.12 | 100 | 100 | 100 | 100 | 90 | 90 | 90 | 100 | 90 | 90 | 95 |
| No.13 | 90 | 100 | 80 | 75 | 85 | 80 | 100 | 100 | 80 | 90 | 88 |
| No.14 | 80 | 100 | 100 | 100 | 90 | 95 | 90 | 100 | 100 | 80 | 93.5 |
| No.15 | 90 | 90 | 95 | 100 | 95 | 100 | 90 | 70 | 90 | 90 | 91 |
| No.16 | 90 | 100 | 90 | 85 | 90 | 95 | 90 | 85 | 90 | 95 | 91 |
| No.17 | 90 | 100 | 90 | 85 | 85 | 95 | 90 | 90 | 80 | 100 | 90.5 |
| No.18 | 100 | 100 | 100 | 100 | 90 | 100 | 80 | 100 | 85 | 90 | 94.5 |
| No.19 | 100 | 90 | 90 | 90 | 95 | 90 | 80 | 100 | 90 | 80 | 90.5 |
| No.20 | 90 | 90 | 95 | 90 | 90 | 90 | 90 | 100 | 90 | 95 | 92 |
| Avg | 94 | 97.5 | 94.5 | 93.3 | 92 | 94.3 | 90 | 94.8 | 87.8 | 91.5 | 93.0 |

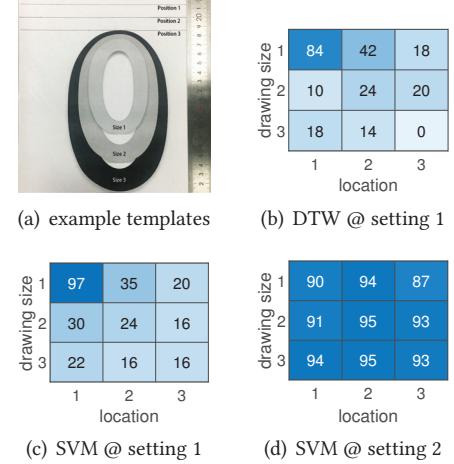


Figure 21. The result of DTW and SVM methods in two tests. Test 1 use the data of setting 1 for training, while test 2 use the data of all the settings for training. The performance is highly depend on how the data distributed in the training set

7.4 Comparison with the State-of-the-Arts

Existing learning-based gesture-recognition solutions can recognize centimeter-level finger gestures [1, 32, 35]. In this section, we compare FingerDraw with state-of-the-arts.

As we have observed in this paper, previous gesture recognition works face a pattern inconsistent problem in sensing centimeter-level finger motions, as both the finger position and drawing size matter. To evaluate

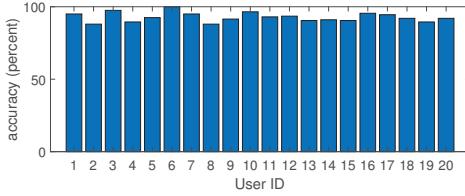


Fig. 22. recognition accuracy per participant of Finger Draw

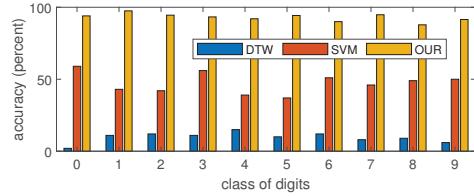


Fig. 23. baseline comparison of natural finger drawing in the air

it, we use the same setting as in Sec. 7.3 and the same set of digits for recognition. We control the position and the drawing size of each digit in the test. We print each digit in three sizes, which are 18.4 cm × 11.4 cm, 15.2 cm × 9.2 cm, and 12 cm × 6.8 cm, respectively, and attach each template on a cardboard at three locations with a spacial interval of 1.4 cm. Fig. 21(a) shows an example of the templates of different sizes and positions. The three locations are marked by a fixed ruler, to ensure all the templates are attached to the three fixed places in the tests. This combination produces $3 \times 3 = 9$ test settings. We ask five participants to draw these digits according to the printed shape of digits. Each digit is drawn 10 times in each setting. Totally we collect $10 \times 3 \times 3 \times 5 \times 10 = 4500$ data. We choose one of the CSI data streams from multiple antennas on a WiFi receiver and take its amplitude to get CSI waveforms.

We use two commonly used methods in previous WiFi-based gesture recognition systems as our baseline. That is the Dynamic Time Warping (DTW) for comparison of waveform shapes, and the Fast Fourier Transform (FFT) for comparison of frequency distributions. All the data follow the same preprocessing steps before comparison. We first cut off both ends of the data to ensure that the data contains only a single digit written by fingers. Then we smooth data with S-G filter. After that, we normalize the data by Z-score.

7.4.1 Waveform Shapes of CSI. WiFinger [32] uses time-domain waveform shapes for recognizing some pre-defined gestures. To evaluate this method in centimeter-level finger motion sensing task, we utilize DTW to align and compare the waveform patterns to the reference profiles in the library. DTW provides a robust metric for measuring the similarity between the test CSI patterns and profiles in the library. The one with the profile in the library that has the highest and also sufficient similarity with the test CSI pattern is then identified as the recognized digit. We build the reference profiles collected from setting No.1 by one participant, that is, the user draws the digits with the smallest drawing size at location 1. Then, we compare the other participant's data collected from all the nine settings with the reference profiles for classification. Before comparison, all the waveforms are down-sampled so that each pattern contains 20 samples.

From Fig. 21(b) we can see, the DTW performance highly depends on the setting that the test data and the reference profiles are collected. The test gets the best match score of 84% when both the test data and the profiles are in the same test setting group and deteriorate quickly in other settings. The more differences in drawing sizes and locations, the fewer similarities of their waveform shapes. As a result, the waveform-shape-based pattern matching method is not a good choice for recognizing centimeter-level finger drawings.

7.4.2 Waveform Frequencies of CSI. WiGest [1] and WiMu [34] are based on the patterns of frequency distributions for gesture recognition. They use the Discrete Wavelet Transform (DWT) and FFT for frequency feature extraction. Both the two methods convert waveforms to its frequency representations, thus provide similar information. We evaluate the FFT method by down-sample the denoised CSI waveforms to 128 samples, then apply 128-point FFT on it. The calculated frequency distribution data is 65-sample vectors. As the spectrum of frequency is not linearly separable in a finite-dimensional space, we use Radial Basis Function (RBF) kernel SVM

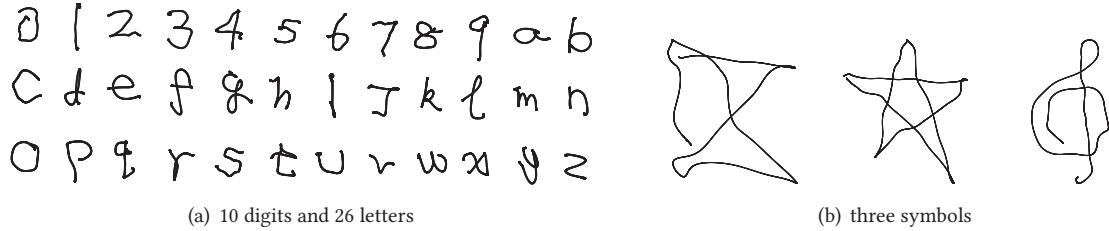


Fig. 24. illustration of some free-style drawings

to classify digits on frequency. Since we collect 450 samples for each digit in all the nine settings and each sample contains 30 sub-carriers, we obtain 13500 SVM vectors for each digit.

We evaluate the SVM method with two test settings. In test setting No.1, we train the SVM model with the data from four participants conducted in the test setting No.1 and predict data of the fifth person collected in all the nine settings. The test result is in Fig. 21(c), we can see the classification result is 97% when the test data is coming from the same experimental setting group, but below 35% when the test data is coming from other settings. In test setting No.2, we train the SVM model with the data from four participants conducted in all the nine test settings and predict data of the fifth person. We can see the performance increase significantly compared with test setting No.1. These results suggest the performance of CSI-waveform-based learning systems highly depends on how the training data distributed. The data from all the possible test settings are needed to train the SVM model. However, consider plenty number of combinations of drawing sizes and locations, traditional CSI-waveform-based learning systems require labor-intensive training, yet their performances are still not guaranteed.

7.4.3 The Comparison Result. We compare FingerDraw with state-of-the-arts in a natural setting. This comparison use dataset from Sec. 7.3. In this dataset, the finger-draw position, size, and font style are not controlled deliberately. The overall recognition results of FingerDraw and the baselines are compared in Fig. 23. The data is reported per recognition class. DTW, SVM, and OUR correspond to the waveform-shape-based gesture recognition, frequency-distribution-based gesture recognition, and our implementation, respectively. In this comparison, DTW use profiles taken from user No.1 and SVM use data of the first 15 participants for training. The rest data is used for testing. The overall accuracy of DTW and SVM is 9.6% and 47%, respectively. We can see the previous learning-based approaches (such as the shape or frequency-distribution-based gesture recognition solutions) are sensitive to finger-drawing positions, sizes, and font styles, they cannot maintain robust recognition performance when finger-drawing gestures vary in size, drawing position and writing style. Compared with the gesture recognition works, FingerDraw achieves an average accuracy of over 93.0% for all the ten digits, which is consistent on change of drawing size and location.

7.5 The Free-Style Drawings

Before we conclude this section, we show some free-style tracking examples of characters and symbols. Fig. 24(a) shows examples of all the digits and letters. Fig. 24(b) shows the example of connected line segments, a five-point star, and a treble clef symbol.

8 LIMITATION AND DISCUSSION

In this section, we present and discuss several limitations of this work.

8.1 Sub-wavelength Motion Accuracy

The phase change of dynamic phasor component in CSI has a precise mapping to the length change of reflected path. It rotates uniformly as the finger moves at a constant speed. However, the mapping of arc rotation in channel quotient is not precisely related to finger motion distance. As we can see in Fig. 9(a) and Fig. 9(c), the same moving distance of an object lead to different phase changes of the arcs of CSI-quotient signals. It is a side effect of channel quotient for a precise sub-wavelength motion measurement. Nevertheless, it has limited influences in finger drawing recognition, as slightly distorted letters can also be recognized. By studying the Möbius transformation, we can alleviate the side effect by choosing the antenna with the biggest CSI magnitude as the denominator in the division. It will produce a more accurate mapping to the motion distance.

8.2 Segmentation of Letters and Multi-stroke Alphabets or Symbols

Our system segments each finger drawing with the assumption that a participant has to keep her/his hand still before and after each drawing. It is required to ensure the success of segmenting each letter for further recognition. Our current implementation does not consider the word input which is composed of several letters. The main reason is that our method is not based on range information, it will have the accumulative tracking errors.

Since our system keeps track of finger movements continuously, we currently only support drawings that consist of a single continuous stroke. Alphabets or symbols that contain multiple strokes will be constructed as single-stroke drawings. It makes letters quite different from what we normally write on paper. It is much better to remove the connections between the strokes to make the letter easier to recognize. We can potentially solve this problem by adding one more WiFi receiver to sense hand motions from Z-axis. We leave the multi-stroke letter segmentation problem as our future work.

9 CONCLUSION

This paper presents FingerDraw, the first sub-wavelength level tracking system for finger drawings with commodity WiFi devices. FingerDraw significantly increases the motion sensing capability into sub-wavelength level granularity. The theory behind the scene is the CSI-quotient model which removes most of the noise in raw CSI signals leveraging on channel quotient between two antennas of a receiver, thus it is very sensitive to minute-level finger movements. In addition, we place two WiFi transceivers orthogonally where we obtain orthogonal CSI signals. We then design our methods to extract micro-displacement information of stroke and segment the strokes based on reflected path length changes. By obtaining the displacement from orthogonal directions, we are able to reconstruct finger drawings in 2D plane.

We believe the technology developed in this paper contributes to not only fine-grained WiFi sensing, but also other wireless sensing scenarios where there is a lack of clock synchronization between transceivers. The sub-wavelength sensing capability is even more important for signals with longer wavelengths, such as LTE.

ACKNOWLEDGMENTS

This research is supported by National Key Research and Development Plan under Grant No. 2016YFB1001200, EU CHIST-ERA RadioSense Project, Australian Research Council (ARC) Discovery Project under Grants No. DP190101888 and DP180103932, and Peking University Information Technology Institute (Tianjin Binhai).

REFERENCES

- [1] Heba Abdelnasser, Moustafa Youssef, and Khaled A Harras. 2015. WiGest: A ubiquitous wifi-based gesture recognition system. In *2015 IEEE Conference on Computer Communications (INFOCOM)*. IEEE, 1472–1480.
- [2] Fadel Adib, Zachary Kabelac, Dina Katahi, and R Miller. 2014. WiTrack: motion tracking via radio reflections off the body. In *Proceedings of the USENIX Symposium on Networked Systems Design and Implementation (NSDI)*.

- [3] Mohammed Al-qaness and Fangmin Li. 2016. WiGeR: WiFi-based gesture recognition system. *ISPRS International Journal of Geo-Information* 5, 6 (2016), 92.
- [4] Kamran Ali, Alex X Liu, Wei Wang, and Muhammad Shahzad. 2015. Keystroke recognition using wifi signals. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*. ACM, 90–102.
- [5] Douglas N. Arnold and Jonathan Rogness. 2008. Möbius transformations revealed. *Notices Amer. Math. Soc.* 55 (2008), 1226–1231.
- [6] Alex Butler, Shahram Izadi, and Steve Hodges. 2008. SideSight: multi-touch interaction around small devices. In *Proceedings of the 21st annual ACM symposium on User interface software and technology*. ACM, 201–204.
- [7] Liwei Chan, Rong-Hao Liang, Ming-Chang Tsai, Kai-Yin Cheng, Chao-Huai Su, Mike Y Chen, Wen-Huang Cheng, and Bing-Yu Chen. 2013. FingerPad: private and subtle interaction using fingertips. In *Proceedings of the 26th annual ACM symposium on User interface software and technology*. ACM, 255–260.
- [8] Amy D Droitcour, Olga Boric-Lubecke, Victor M Lubecke, Jenshan Lin, and Gregory TA Kovacs. 2004. Range correlation and I/Q performance benefits in single-chip silicon Doppler radars for noncontact cardiopulmonary monitoring. *IEEE Transactions on Microwave Theory and Techniques* 52, 3 (2004), 838–848.
- [9] Makiko Funasaka, Yu Ishikawa, Masami Takata, and Kazuki Joe. 2015. Sign language recognition using leap motion controller. In *Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA)*. The Steering Committee of The World Congress in Computer Science, 263.
- [10] Wei Gong and Jiangchuan Liu. 2018. SiFi: Pushing the Limit of Time-Based WiFi Localization Using a Single Commodity Access Point. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 1 (2018), 10.
- [11] Google. [n.d.]. Project Soli. <https://www.youtube.com/watch?v=0QNjZfSsPc0>. Accessed Jan 15, 2020.
- [12] Daniel Halperin, Wenjun Hu, Anmol Sheth, and David Wetherall. 2011. Tool release: gathering 802.11 n traces with channel state information. *ACM SIGCOMM Computer Communication Review* 41, 1 (2011), 53–53.
- [13] Wolf Kienzle and Ken Hinckley. 2014. LightRing: always-available 2D input on any surface. In *Proceedings of the 27th annual ACM symposium on User interface software and technology*. ACM, 157–160.
- [14] David Kim, Otmar Hilliges, Shahram Izadi, Alex D Butler, Jiawen Chen, Iason Oikonomidis, and Patrick Olivier. 2012. Digits: freehand 3D interactions anywhere using a wrist-worn gloveless sensor. In *Proceedings of the 25th annual ACM symposium on User interface software and technology*. ACM, 167–176.
- [15] Manikanta Kotaru, Kiran Joshi, Dinesh Bharadia, and Sachin Katti. 2015. SpotFi: Decimeter Level Localization Using WiFi. In *Proceedings of the 2015 ACM Conference on Special Interest Group on Data Communication (SIGCOMM '15)*. ACM, 269–282.
- [16] Sven Kratz and Michael Rohs. 2009. HoverFlow: expanding the design space of around-device interaction. In *Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services*. ACM, 4.
- [17] Xiang Li, Shengjie Li, Daqing Zhang, Jie Xiong, Yasha Wang, and Hong Mei. 2016. Dynamic-music: accurate device-free indoor localization. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM, 196–207.
- [18] Xiang Li, Daqing Zhang, Qin Lv, Jie Xiong, Shengjie Li, Yue Zhang, and Hong Mei. 2017. IndoTrack: Device-free indoor human tracking with commodity Wi-Fi. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 3 (2017), 72.
- [19] Xiang Li, Daqing Zhang, Jie Xiong, Yue Zhang, Shengjie Li, Yasha Wang, and Hong Mei. 2018. Training-Free Human Vitality Monitoring Using Commodity Wi-Fi Devices. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 3 (2018), 121.
- [20] Jian Liu, Yan Wang, Yingying Chen, Jie Yang, Xu Chen, and Jerry Cheng. 2015. Tracking vital signs during sleep leveraging off-the-shelf wifi. In *Proceedings of the 16th ACM International Symposium on Mobile Ad Hoc Networking and Computing*. ACM, 267–276.
- [21] Xuefeng Liu, Jiannong Cao, Shaojie Tang, and Jiaqi Wen. 2014. Wi-Sleep: Contactless sleep monitoring via WiFi signals. In *Real-Time Systems Symposium (RTSS), 2014 IEEE*. IEEE, 346–355.
- [22] Yongsen Ma, Gang Zhou, Shuangquan Wang, Hongyang Zhao, and Woosub Jung. 2018. SignFi: Sign Language Recognition Using WiFi. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 1 (2018), 23.
- [23] Rajalakshmi Nandakumar, Vikram Iyer, Desney Tan, and Shyamnath Gollakota. 2016. Fingerio: Using active sonar for fine-grained finger tracking. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM, 1515–1525.
- [24] Sameera Palipana, David Rojas, Piyush Agrawal, and Dirk Pesch. 2018. FallDeFi: Ubiquitous Fall Detection using Commodity Wi-Fi Devices. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 4 (2018), 155.
- [25] Qifan Pu, Sidhant Gupta, Shyamnath Gollakota, and Shwetak Patel. 2013. Whole-home gesture recognition using wireless signals. In *Proceedings of the 19th annual international conference on Mobile computing & networking*. ACM, 27–38.
- [26] Kun Qian, Chenshu Wu, Zheng Yang, Yunhao Liu, and Kyle Jamieson. 2017. Widar: Decimeter-level passive tracking via velocity monitoring with commodity Wi-Fi. In *Proceedings of the 18th ACM International Symposium on Mobile Ad Hoc Networking and Computing*. ACM, 6.
- [27] Kun Qian, Chenshu Wu, Zimu Zhou, Yue Zheng, Zheng Yang, and Yunhao Liu. 2017. Inferring motion direction using commodity wi-fi for interactive exergames. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. ACM, 1961–1972.
- [28] Muhammad Shahzad and Shaohu Zhang. 2018. Augmenting User Identification with WiFi Based Gesture Recognition. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 3 (2018), 134.

- [29] Chao Sun, Tianzhu Zhang, and Changsheng Xu. 2015. Latent support vector machine modeling for sign language recognition with Kinect. *ACM Transactions on Intelligent Systems and Technology (TIST)* 6, 2 (2015), 20.
- [30] Ke Sun, Ting Zhao, Wei Wang, and Lei Xie. 2018. VSkin: Sensing Touch Gestures on Surfaces of Mobile Devices Using Acoustic Signals. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*. ACM, 591–605.
- [31] Li Sun, Souvik Sen, Dimitrios Koutsikolas, and Kyu-Han Kim. 2015. Widraw: Enabling hands-free drawing in the air on commodity wifi devices. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*. ACM, 77–89.
- [32] Sheng Tan and Jie Yang. 2016. WiFinger: leveraging commodity WiFi for fine-grained finger gesture recognition. In *Proceedings of the 17th ACM International Symposium on Mobile Ad Hoc Networking and Computing*. ACM, 201–210.
- [33] Deepak Vasisht, Swaran Kumar, and Dina Katabi. 2016. Decimeter-Level Localization with a Single WiFi Access Point.. In *Proceedings of the USENIX Symposium on Networked Systems Design and Implementation (NSDI)*, Vol. 16. 165–178.
- [34] Raghav H Venkatnarayan, Griffin Page, and Muhammad Shahzad. 2018. Multi-User Gesture Recognition Using WiFi. In *Proceedings of the 16th Annual International Conference on Mobile Systems, Applications, and Services*. ACM, 401–413.
- [35] Aditya Virmani and Muhammad Shahzad. 2017. Position and orientation agnostic gesture recognition using wifi. In *Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services*. ACM, 252–264.
- [36] Chuyu Wang, Jian Liu, Yingying Chen, Hongbo Liu, Lei Xie, Wei Wang, Bingbing He, and Sanglu Lu. 2018. Multi-Touch in the Air: Device-Free Finger Tracking and Gesture Recognition via COTS RFID. In *Proceedings of the Conference on Computer Communications (INFOCOM)*. IEEE, 1691–1699.
- [37] Hao Wang, Daqing Zhang, Junyi Ma, Yasha Wang, Yuxiang Wang, Dan Wu, Tao Gu, and Bing Xie. 2016. Human respiration detection with commodity wifi devices: do user location and body orientation matter?. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM, 25–36.
- [38] Hao Wang, Daqing Zhang, Yasha Wang, Junyi Ma, Yuxiang Wang, and Shengjie Li. 2017. RT-Fall: A real-time and contactless fall detection system with commodity WiFi devices. *IEEE Transactions on Mobile Computing* 16, 2 (2017), 511–526.
- [39] Jue Wang, Deepak Vasisht, and Dina Katabi. 2014. RF-IDraw: virtual touch screen in the air using RF signals. In *ACM SIGCOMM Computer Communication Review*, Vol. 44. ACM, 235–246.
- [40] Wei Wang, Alex X Liu, and Muhammad Shahzad. 2016. Gait recognition using wifi signals. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM, 363–373.
- [41] Wei Wang, Alex X Liu, Muhammad Shahzad, Kang Ling, and Sanglu Lu. 2015. Understanding and modeling of wifi signal based human activity recognition. In *Proceedings of the 21st annual international conference on mobile computing and networking*. ACM, 65–76.
- [42] Wei Wang, Alex X Liu, and Ke Sun. 2016. Device-free gesture tracking using acoustic signals. In *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking*. ACM, 82–94.
- [43] Chenshu Wu, Zheng Yang, Zimu Zhou, Xuefeng Liu, Yunhao Liu, and Jiannong Cao. 2015. Non-invasive detection of moving and stationary human with wifi. *IEEE Journal on Selected Areas in Communications* 33, 11 (2015), 2329–2342.
- [44] Dan Wu, Daqing Zhang, Chenren Xu, Yasha Wang, and Hao Wang. 2016. WiDir: walking direction estimation using wireless signals. In *Proceedings of the 2016 ACM international joint conference on pervasive and ubiquitous computing*. ACM, 351–362.
- [45] Yaxiong Xie, Zhenjiang Li, and Mo Li. 2015. Precise Power Delay Profiling with Commodity WiFi. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking (MobiCom '15)*. ACM, 53–64.
- [46] Tong Xin, Bin Guo, Zhu Wang, Pei Wang, Jacqueline Chi Kei Lam, Victor Li, and Zhiwen Yu. 2018. FreeSense: A Robust Approach for Indoor Human Detection Using Wi-Fi Signals. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 3 (2018), 143.
- [47] Zheng Yang, Zimu Zhou, and Yunhao Liu. 2013. From RSSI to CSI: Indoor localization via channel response. *ACM Computing Surveys (CSUR)* 46, 2 (2013), 25.
- [48] Nan Yu, Wei Wang, Alex X Liu, and Lingtao Kong. 2018. QGesture: Quantifying Gesture Distance and Direction with WiFi Signals. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 1 (2018), 51.
- [49] Sangki Yun, Yi-Chao Chen, Huihuang Zheng, Lili Qiu, and Wenguang Mao. 2017. Strata: Fine-grained acoustic-based device-free tracking. In *Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services*. ACM, 15–28.
- [50] Youwei Zeng, Dan Wu, Ruiyang Gao, Tao Gu, and Daqing Zhang. 2018. FullBreathe: Full Human Respiration Detection Exploiting Complementarity of CSI Phase and Amplitude of WiFi Signals. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 3 (2018), 148.
- [51] Daqing Zhang, Hao Wang, and Dan Wu. 2017. Toward centimeter-scale human activity sensing with Wi-Fi signals. *Computer* 50, 1 (2017), 48–57.
- [52] Fusang Zhang, Daqing Zhang, Jie Xiong, Hao Wang, Kai Niu, Beihong Jin, and Yuxiang Wang. 2018. From Fresnel Diffraction Model to Fine-grained Human Respiration Sensing with Commodity Wi-Fi Devices. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 1 (2018), 53.
- [53] Ouyang Zhang and Kannan Srinivasan. 2016. Mudra: User-friendly Fine-grained Gesture Recognition using WiFi Signals. In *Proceedings of the 12th International on Conference on emerging Networking EXperiments and Technologies*. ACM, 83–96.

- [54] Chen Zhao, Ke-Yu Chen, Md Tanvir Islam Aumi, Shwetak Patel, and Matthew S Reynolds. 2014. SideSwipe: detecting in-air gestures around mobile devices using actual GSM signal. In *Proceedings of the 27th annual ACM symposium on User interface software and technology*. ACM, 527–534.
- [55] Yue Zheng, Yi Zhang, Kun Qian, Guidong Zhang, Yunhao Liu, Chenshu Wu, and Zheng Yang. 2019. Zero-Effort Cross-Domain Gesture Recognition with Wi-Fi. In *Proceedings of the 17th Annual International Conference on Mobile Systems, Applications, and Services*. ACM, 313–325.