Leveraging Electromagnetic Polarization in a Two-Antenna Motion Tracking System

March 15, 2016

Longfei Shangguan[1], Kyle Jamieson[1,2]

[1]Princeton University, [2]University College London

Abstract

Wireless sensing, tracking, and drawing technologies are enabling exciting new possibilities for human-mobile interaction. They primarily rely on measurements of backscattered phase, amplitude, and Doppler signal distortions, and often require many measurements of these quantities—in time, or from multiple antennas. In this paper we present the design and implementation of PolarDraw, the first system for pen motion tracking that sends differentially-polarized wireless signals to glean more information from the environment. Leveraging information received from each polarization angle, our novel algorithms infer orientation and position of an RFID-tagged pen using just two antennas, writing in the air or on a whiteboard. An experimental comparison in a cluttered indoor office environment compares two-antenna PolarDraw with recent state-of-the-art object tracking systems that use double the number of antennas, demonstrating comparable centimeter-level tracking accuracy and character recognition rates (88–94%), thus making a case for the use of polarization in many other tracking systems.

1. Introduction

We are rapidly moving toward a pervasively-sensed wireless world where most of our interactions with machines will be accomplished via gestures [3, 25] and writing in the air [32], and machines will be able to recognize our activities and pinpoint our location [1, 2, 13, 38]. On the industrial side, recent efforts include both short-range [9] and medium-range [8] indoor radar technologies, as well as more mature electronic whiteboard technologies based on ultrasound/infrared [20] and laser curtains [19].

While the basic technology is compelling, the real-world utility of passive wireless sensing systems depends on several factors, all of which must be addressed:

1. Infrastructure requirements— Systems that use angle-of-arrival information generally require a significant number of physically-separated antennas [3, 41], reducing deployability. Other systems require the simultaneous use of multiple widely-separated carrier frequencies [33], requiring simultaneous or added communication on these frequencies.

2. Pointing equipment— Some systems achieve millimeter accuracy, but require expensive ($100) styluses in order to interact with ultrasound and infrared receivers [20].

3. Location and orientation estimation— In many cases, estimating the orientation of an object or person helps improve accuracy or activity recognition.

Electromagnetic wave *polarization* is a fundamental property of a wireless transmission, and refers to the orientation of the electrical field in the plane transverse to the wave's propagation, as shown in Figure 1. In the context of mobile devices, polarization has received the most attention in recent years as increasing smartphone usage in an orientation facing the user (roughly level with the ground) has resulted in a loss of signal strength from a vertically-polarized transmissions, both indoors and outdoors [27]. Outdoors, the horizontally- and vertically-polarized paths from base station to mobile (of length *ca.* 2 mi.) are very different [16] due to differing reflectors. Indoors, for smaller Wi-Fi cell sizes, the same is true for longer non-line-of-sight paths, but shorter paths tend to share similar reflectors (and a more dominant line-of-sight path) when viewed from orthogonal polarizations [28].

Our key observation in this paper is that this similarity between horizontally- and vertically-polarized paths at modest ranges presents a unique opportunity for wireless motion tracking systems to leverage a new source of information: polarization. We use a simple RFID reader as a test case, but replace the reader's standard circularly polarized antennas with linearly polarized antennas as shown in Figure 1, resulting in the linearly polarized transmission shown.

In this paper, we present the design and implementation of PolarDraw, the first motion tracking system that can accurately reproduce handwritten letters in the air or on a whiteboard with just *two* nearby antennas. With two antennas, PolarDraw sets a new standard for minimal supporting infrastructure. PolarDraw estimates both the position and orientation of an RFID-tagged whiteboard pen, adding just pennies to the cost of each item to be used as a stylus. Thus PolarDraw meets each of the three preceding objectives for a practical and highly-deployable motion tracking system.

In brief, PolarDraw works as follows. Our system uses RFID antennas to measure the phase and amplitude of an RFID tag at a rate of *ca.* 100 Hz. Variations in the angle between tag and reader antenna result in fluctuations of power received from the

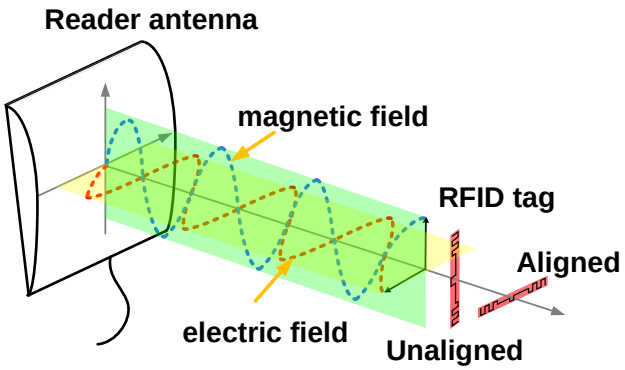


Figure 1— Linearly-polarized electromagnetic waves propagate with a fixed electric field angle, exciting antennas or tags in proportion to alignment with this angle.

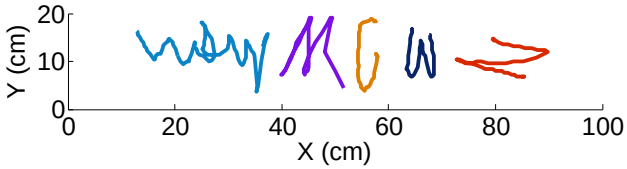


Figure 2— Recovered trajectory: *WoW*, *M*, *C*, *W*, *Z*.

tag, but as our feasibility study next (§2) shows, symmetry properties result in equal power changes when the tag rotates left or right. PolarDraw uses a second, linearly-polarized antenna to view the rotation from a 30° shift in perspective, thus overcoming this *rotational ambiguity* problem (§3.2). Rotation is just part of what determines the pen's true trajectory, however. PolarDraw also estimates the displacement of the pen on the whiteboard using phase difference readings across both time and space (§3.3). Since they do not attempt to discriminate polarization, current trajectory tracing systems have not yet explored how best to fuse the above phase readings (which estimate pen displacement) with power readings (which estimate pen rotation), and doing so carefully is key to getting highly accurate results. We describe our novel algorithm, which incorporates a simple Viterbi-based probabilistic search (§3.4), to complete the design of PolarDraw.

Contributions. PolarDraw is the first system we are aware of that uses differentially-polarized radio transmissions to estimate an object's orientation. The techniques introduced here introduce a new source of position information, and thus have the potential to be applied to many other indoor radar and indoor localization systems to enhance their accuracy. PolarDraw is also the first radio-based system we are aware of that simultaneously models changes in the pen's displacement and orientation. As our experimental evaluation shows, PolarDraw benefits from the pen orientation estimation.

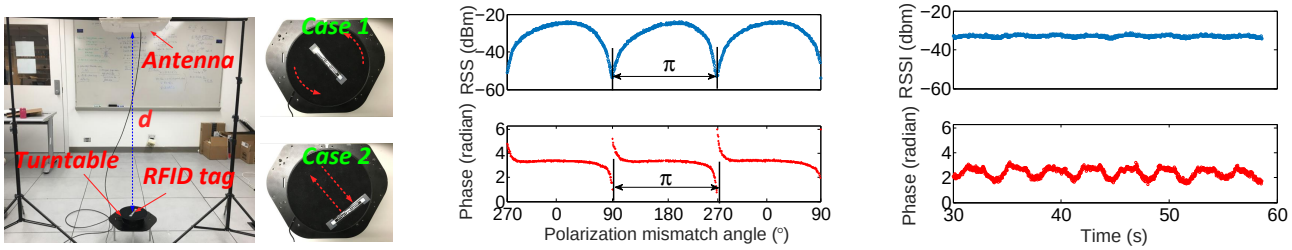
Roadmap. The rest of this paper is organized as follows: §2 presents initial microbenchmark-style measurements in a cluttered office environment to establish the basic experimental possibility of measuring orientation through polarized transmissions. §3 presents the design of PolarDraw, after which we describe our implementation (§4). Our experimental evaluation (§5) tests PolarDraw's performance against the two leading motion tracking systems in the research literature mentioned above, RF-IDraw and Tagoram, with all three systems running in the same experimental environment. Our results in this cluttered, real-world office environment demonstrate motion tracking accuracy that is competitive with the above two systems (median 10 cm for two-antenna PolarDraw, 8 cm for four-antenna RF-IDraw and Tagoram), but with just two antennas, making PolarDraw significantly more deployable. Further microbenchmark experiments justify parameter choices and test the experimental limits of our system. In the remainder of the paper, we discuss related (§6) and future (§7) work before concluding (§8).

2. Feasibility: Measuring polarization

To determine whether we can measure polarization in a real indoor environment where multipath is prevalent, we have conducted an emprical feasibility study, using the hardware setup shown in Figure 3(a). We situate an ImpinJ RFID reader connected to a linearly polarized antenna 2.5 m above an RFID tag placed on a turntable below, as shown in the picture.[1] The RFID reader interrogates the tag at a rate of approximately 100 Hz. In the first experiment, the tag rotates at a constant angular velocity on the turntable. In the second experiment, we manually translate the tag back and forth, keeping its orientation fixed.

Figure 3(b) shows the received signal strength (RSS) and phase measured at the RFID reader during the tag's rotation. As expected, RSS changes periodically during the tag's rotation. It peaks at -24 dBm when the tag and the reader antennas are

[1] Our hardware setup is more completely described below in §4.1.



(a)— Feasibility study hardware setup. **(b)—** Rotating a tag. (Case 1) **(c)—** Moving a tag. (Case 2)

Figure 3— An initial feasibility study experimental setup and experimental results.

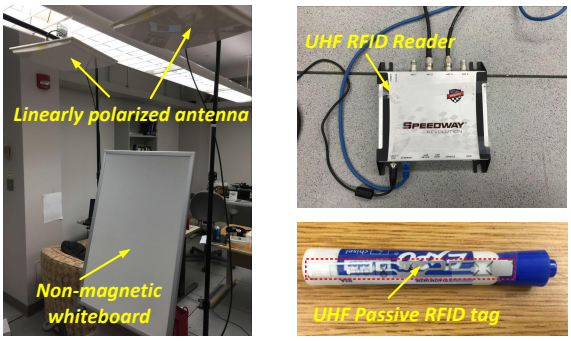


Figure 4— PolarDraw's high-level design. Two linearly-polarized antennas are mounted above a whiteboard. Users draw with standard whiteboard pens, RFID-tagged.

aligned, and drops gradually as the polarization mismatch angle between the two increases. Finally, the tag fails to acquire any power and there is no RSS reading when the orientation of the tag is perpendicular to the polarization angle of the reader antenna. On the other hand, since the tag-to-antenna distance does not change during the tag's rotation, the phase reading keeps around a constant value. Interestingly, we can see that the phase reading jumps when the polarization mismatch angle is around 90° and 270°. This is because the tag fails to acquire energy from the line-of-sight path due to the polarization mismatch. Nonetheless, it acquires energy along non-line-of-sight signal propagation paths, where the signal bounces off nearby objects, changing its polarization angles. PolarDraw is designed to cope with and overcome the spurious data that these differently-polarized reflection paths convey to the reader antenna.

Figure 3(c) shows RSS and phase during the tag's movement over a distance of 8 cm. As the result indicates, the RSS value stays more or less constant, which is expected since RSS is insensitive to small changes in distance over the tag-to-antenna link. In contrast, we can see the phase reading increases when the tag moves in one direction, remains stable when the tag stands still, and decreases when the tag moves back.

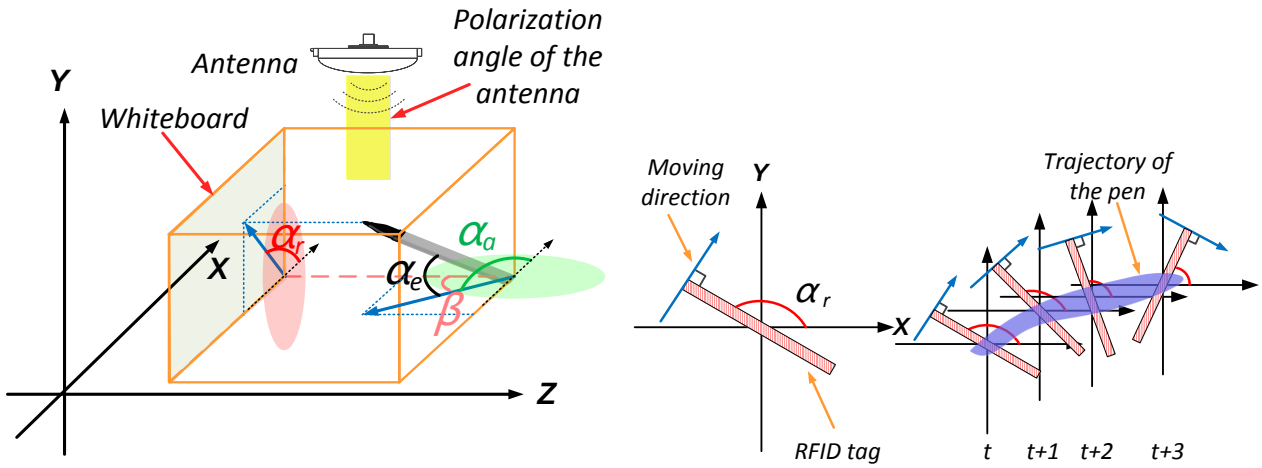
We conclude from the above empirical results:

1. At moderate distances in our line-of-sight indoor environment, RSS is sensitive to the polarization mismatch, yet it is insensitive to small changes in the length of the tag-to-antenna link.
2. With the exception of a "corner" case when the tag antenna and reader antenna are orthogonal to each other, phase measurements are insensitive to polarization mismatch, yet are extremely sensitive to small changes in the length of the tag-to-antenna link.

Combining these two findings, we can conclude that in our experimental setup, separate measurements of RSS and pha-se can respectively estimate rotational and translational mov-ement of the pen, with a reasonable degree of reliability. In the next section, we describe a design that extracts a high degree of reliability from these two signals.

3. Design

In this section we detail PolarDraw's design, starting with a simple model of pen rotation and translation (§3.1). Leveraging this model, we next present a method for estimating the direction (§3.2) and amount (§3.3) of the pen's movement. We conclude



(a)— Pen angle projection. **(b)—** Tracking pen position due to rotation and translation.

Figure 5— Modeling pen movement during writing.

the section with the entire pen trajectory tracking algorithm (§3.4).

Design overview. PolarDraw comprises a UHF RFID reader connected to two linearly-polarized antennas mounted above a whiteboard as shown in Figure 4, or simply tracking a "virtual whiteboard" in the air. Users draw below with standard whiteboard pens, to which we attach low-cost RFID tags. The RFID reader works on the 902–928 MHz ISM band, interrogating the RFID tag at a rate of 100 Hz (10 milliseconds between each reading).

3.1 Modeling pen movement

A *pen trajectory* is composed of a set of discrete trajectory *fragments*, each of which can be characterized by a movement direction and distance on the (virtual or real) whiteboard. When a user is writing on a whiteboard, wrist movements tend to cause azimuthal rotations clockwise when the pen moves to the right, and counterclockwise when the pen moves to the left. The pen is of course free to rotate in three-space, and so we define angular measurements of the pen's elevation from the X–Z plane and azimuthal rotation measured from the X-axis and projected onto the X–Z plane:

α_e:	Pen *elevation angle*, from the X–Z plane,
α_r:	Pen *rotation angle*, as projected on the X–Y plane,
α_a:	Pen *azimuthal angle*, as projected on the X–Z plane.

Table 1— Pen angle as illustrated in Figure 5(a).

as shown in Figure 5(a). Consequently, if the rotation angle of the pen α_r is determined, then the direction of the movement of the pen, which is in perpendicular to α_r, can be estimated.

Pen movement direction. PolarDraw relies on the phenomenon of polarization angle mismatch to infer changes in the rotation angle of the pen. As noted above, the tag reflects the most energy when its physical orientation is aligned with the polarization angle of the interrogating antenna. In contrast, the tag reflects negligible energy when its rotation angle α_r is 90° from the polarization angle of the reader antenna above.

As Figure 5(a) shows, once the polarization mismatch angle $\alpha_a - 90°$ is determined (detailed in Section 3.2), we can deduce α_a and estimate pen rotation angle α_r as follows:

$$\alpha_r = \pi - \arctan\left(-\frac{\sin\alpha_e}{\cos\alpha_e \cdot \cos\alpha_a}\right) \tag{1}$$

$\alpha_e \in (-90°, 90°)$ indicates the pen elevation angle when wri-ting on the whiteboard. We set α_e to be a constant determined by experiments described in 5.4.1. There we also show that recognition accuracy is insensitive to PolarDraw's choice of α_e.

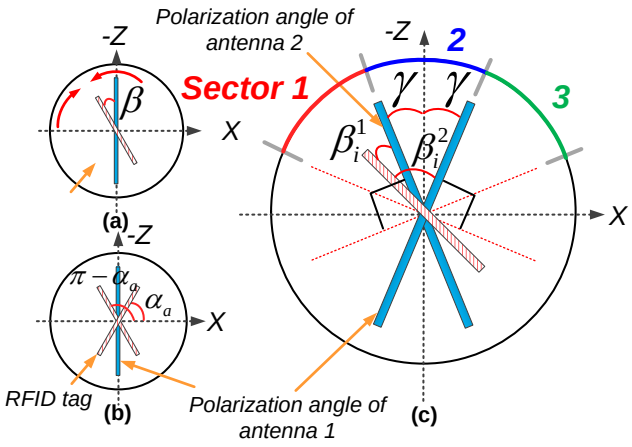


Figure 6— Polarization angle mismatch. The thick blue line indicates the polarization angle of the reader antenna. (a): rotation direction ambiguity. (b): azimuthal angle ambiguity. (c): breaking the rotation ambiguity by spatially separated antennas.

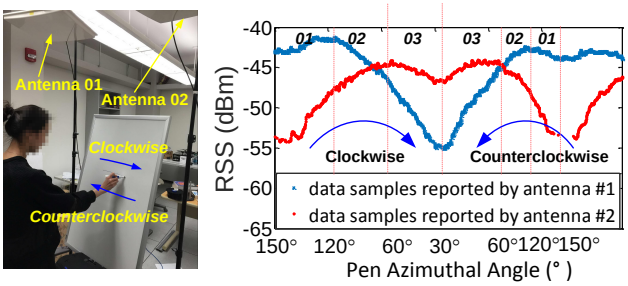


Figure 7— The RSS trend during human writings. $\gamma = 30°$.

PolarDraw continuously estimates α_r and the moving distance along the arc formed by change in α_r at each time point, and inherently recoveries the trajectory of the pen on the whiteboard, as shown in Figure 5(b). One thing need to be noticed that when $\alpha_a = 90°$, $\alpha_r = 0°$. Hence we do not take $\alpha_a = 90°$ into Eq. 1 in this case.

Pen moving distance. The RF phase reading depicts the traveling distance of the backscatter signal. PolarDraw uses RF phase readings to estimate the tag moving distance due to the following two reasons. First, as our empirical feasibility study shows, the polarization mismatch has little impact on the phase readings. Second, the COTS RFID reader provides fine-grained resolution of phase readings.

3.2 Estimating the pen's moving direction

As noted above, once the azimuthal angle α_a is estimated, we can infer the pen rotation angle α_r by Equation 1 and deduce the pen direction of movement. PolarDraw exploits both the RSS trend and the phase trend to estimate and track pen direction of movement. The basic workflow of the algorithm is as follows: at each timestamp, PolarDraw first examines the RSS trend. If there is no significant change in RSS, *i.e.*, RSS change remains below some threshold δ[2], PolarDraw then exploits the phase trend to determine the pen's coarse-grained direction of movement (detailed in §3.2.2). Otherwise, PolarDraw tracks the pen azimuthal angle α_a bas-ed on its continuous azimuthal angle tracking algorithm (detailed in §3.2.1), and translates α_a to the pen rotation angle α_r using Equation 1. With α_r, PolarDraw further infers the pen moving direction at each timestamp.

3.2.1 Azimuthal angle estimation

In designing PolarDraw's α_a estimation algorithm, we face the following two challenges. Firstly, as Figure 6(a) shows, both a clockwise and a counterclockwise rotation could result in an identical polarization mismatch angle β, hence the same end RSS. Consequently, we are unable to differentiate the rotation direction by simply observing the end RSS value alone. We term this challenge the *rotation direction ambiguity*. Secondly, as Figure 6(b) shows, within the range $[0, 180°]$, there are always two azimuthal angles that result in the same RSS value. Therefore, we are also unable to estimate the azimuthal angle based on RSS alone. We term this challenge the *azimuthal angle ambiguity*.

Breaking ambiguities. PolarDraw addresses the above challenges by leveraging the RSS trend acquired from its two linearly polarized antennas. As Figure 6(c) shows, we assume the angles between the Z-axis and the polarization angle of these two antennas are the same, denoting them as γ. Let β_i^j be the polarization mismatch angle between the tag and antenna j at time i. Then, the lines along the polarization angles of these two antennas together with perpendicular lines drawn across each

[2]We test various threshold, and empirically set δ =2 dBm, which optimizes the overall performance

Table 2— RSS trend. ("–" means don't care.)

Area	Direction	RSS trend		RSS Changing rate
Sector 1	$\Rightarrow$	$s_i^1 \uparrow$	$s_i^2 \uparrow$	$\lvert\Delta s_i^1\rvert < \lvert\Delta s_i^2\rvert$
	$\Leftarrow$	$s_i^1 \downarrow$	$s_i^2 \downarrow$	$\lvert\Delta s_i^1\rvert < \lvert\Delta s_i^2\rvert$
Sector 2	$\Rightarrow$	$s_i^1 \downarrow$	$s_i^2 \uparrow$	–
	$\Leftarrow$	$s_i^1 \uparrow$	$s_i^2 \downarrow$	–
Sector 3	$\Rightarrow$	$s_i^1 \downarrow$	$s_i^2 \downarrow$	$\lvert\Delta s_i^1\rvert > \lvert\Delta s_i^2\rvert$
	$\Leftarrow$	$s_i^1 \uparrow$	$s_i^2 \uparrow$	$\lvert\Delta s_i^1\rvert > \lvert\Delta s_i^2\rvert$

naturally separate the rotation plane into three *sectors*, as shown in Figure 6(c). The pen does not rotate azimuthally too much when a user writes on the whiteboard, and so we assume the azimuthal angle α_a is within the range of the union of these three sectors. Therefore, if we can determine in which sector the current α_a is, then we can break the azimuthal angle ambiguity.

Figure 7 shows RSS trends reported by different antennas when a user is writing on a whiteboard. Let s_i^j be the RSS reading reported by antenna j at time i, $\Delta s_i^j = s_{i+1}^j - s_i^j$. As indicated in this figure, when the pen rotates in different direction within different sectors, the polarization mismatch angle β_i^1 and β_i^2 changes in a different way, which leads to a different direction of change in RSS as well as a different rate of change of RSS: Table 2 summarizes the result. From this table we find that by jointly considering RSS trends and changing rates, we can successfully determine the moving direction of the pen as well as the range of the current azimuthal angle α_a that the pen points to, thereby breaking both the rotation direction ambiguity and the azimuthal angle ambiguity.

Continuous azimuthal angle tracking. Let α_a^i be the azimuthal angle of the pen at time i. When the user begins writing, PolarDraw estimates in which sector the pen points and in which direction the pen orients. Based on these, PolarDraw assigns an initial azimuthal angle as follows:

$$\alpha_a^0 = \begin{cases} \pi - \gamma, & \text{if clockwise and in Sector 1,} \\ \frac{\pi}{2} + \gamma, & \text{if clockwise and in Sector 2,} \\ \frac{\pi}{2} - \gamma, & \text{if clockwise and in Sector 3,} \\ \frac{\pi}{2} + \gamma, & \text{if counterclockwise and in Sector 1,} \\ \frac{\pi}{2} - \gamma, & \text{if counterclockwise and in Sector 2,} \\ \gamma, & \text{if counterclockwise and in Sector 3.} \end{cases} \tag{2}$$

PolarDraw then tracks α_a^i at each time i as follows:

$$\alpha_a^i = \begin{cases} \alpha_a^{i-1} - \Delta\beta, & \text{if clockwise rotation,} \\ \alpha_a^{i-1} + \Delta\beta, & \text{if counterclockwise rotation.} \end{cases} \tag{3}$$

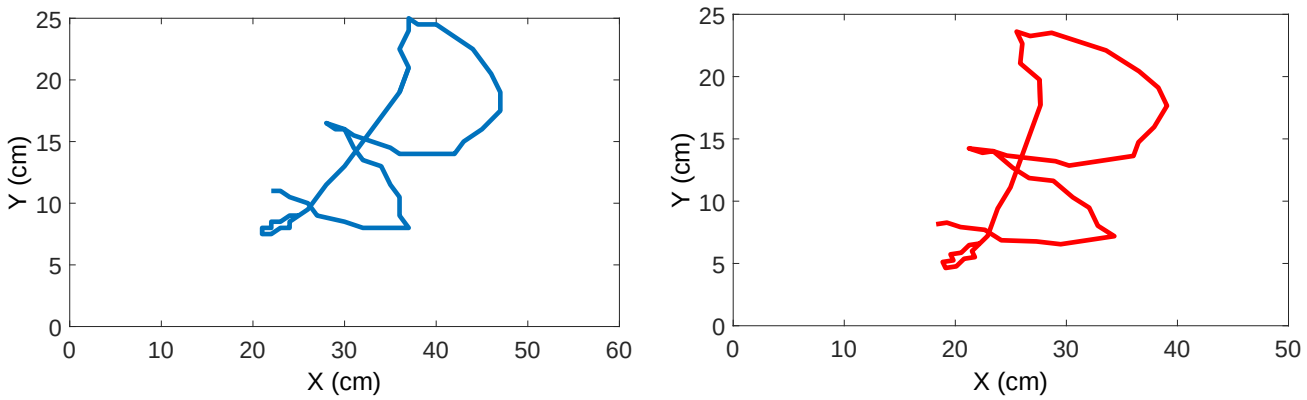
where $\Delta\beta$ is a variable, indicating the average changing rate of the azimuthal angle when human writes on the whiteboard. The assignment of $\Delta\beta$ is as follows:

$$\Delta\beta = \begin{cases} \frac{\pi}{30}, & \text{if } \lvert\Delta s_i^1\rvert > \delta \text{ and } \lvert\Delta s_i^2\rvert > \delta, \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

We test various thresholds and empirically set δ to 1.5 dBm, which optimizes the overall detection accuracy.

Initial azimuthal angle correction. In the tracking process, the initial azimuthal angle we assigned will likely deviate from its true value by some amount $\tilde{\alpha}_a$. Consequently, later azimuthal angle estimates suffer from this error $\tilde{\alpha}_a$ as well, resulting in an inaccurate tracking result.

PolarDraw corrects $\tilde{\alpha}_a$ based on detecting when the azimuthal angle of the pen crosses over the boundary of two sectors. Suppose at time i PolarDraw detects that the pen crosses over the boundary of two neighboring sectors based on the principle shown in Table 2. The azimuthal angle of the pen at time i should be approximately equal to the azimuthal boundary angle of these two sectors, denoted as $\hat{\alpha}_a^i$. Hence the difference between $\hat{\alpha}_a^i$ and α_a^i (the estimated azimuthal angle) indicates the initial



(a)— Before correction. **(b)—** After correction.

Figure 8— The recovered pen trajectory before and after azimuthal angle correction.

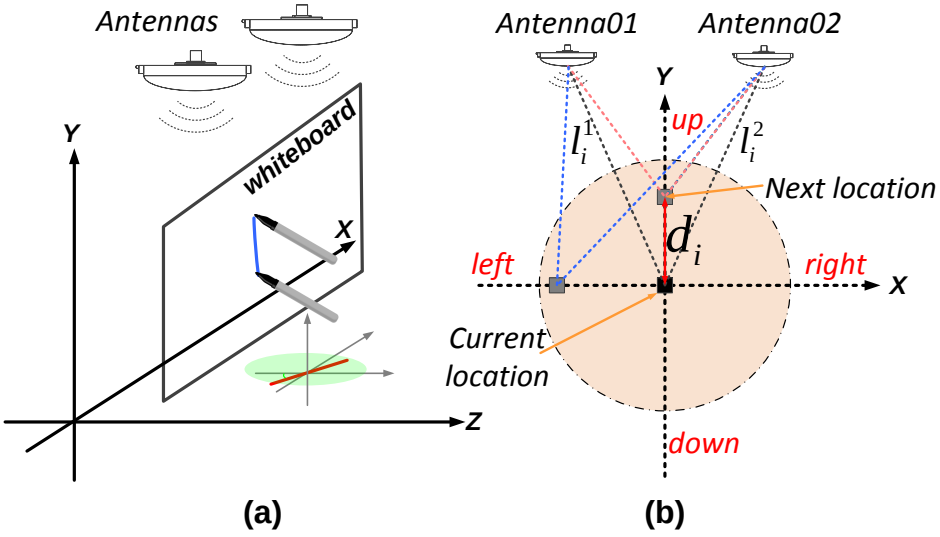


Figure 9— Writing without rotation. (a): the user writes on the whiteboard with a constant rotation angle; (b): geometry relationship of moving directions;

azimuthal angle error $\tilde{\alpha}_a$. PolarDraw corrects the estimated azimuthal angles by subtracting $\tilde{\alpha}_a$ from all α_a^i, thus achieving a more accurate result. Figure 8 shows a snapshot of our system's output before and after azimuthal angle error correction.

3.2.2 Estimating moving direction via phase trend

Writing styles vary and some people may write on the whiteboard without pen rotation, as shown in Figure 9(a). Consequently, the azimuthal angle α_a fails to reflect the true moving direction of the pen. To solve this problem, PolarDraw further exploits phase changes to estimate the pen's direction of movement. Specifically, we denote the distance between reader antenna j and the tag at time i as l_i^j. Let θ_i^j be the phase reading reported by the antenna j at time i. As Figure 9(b) shows, when the pen moves up, both l_i^1 and l_i^2 decrease, resulting in $\theta_i^1 \downarrow$ and $\theta_i^2 \downarrow$. Conversely, when the pen moves down, both l_i^1 and l_i^2 increase, leading to $\theta_i^1 \uparrow$ and $\theta_i^2 \uparrow$. Table 3 shows phase trends when the pen moves to different directions. Hence we leverage phase trends to determine a coarse-grained moving direction of the pen.

3.3 Estimating the pen's moving distance

PolarDraw again exploits phase trends to estimate the moving distance of the pen. Our approach is based on an assumption that the moving distance during consecutive tag readings is within the half wavelength (≈16 cm). This assumption holds in

Table 3— Phase changing trends during movement.

Moving direction	**Phase trends**	
Up	$\theta_i^1 \downarrow$	$\theta_i^2 \downarrow$
Down	$\theta_i^1 \uparrow$	$\theta_i^2 \uparrow$
Left	$\theta_i^1 \downarrow$	$\theta_i^2 \uparrow$
Right	$\theta_i^1 \uparrow$	$\theta_i^2 \downarrow$

practice since the sampling rate of the COST RFID reader is around 100 Hz, yielding a maximum differentiable speed of 16 m/s, significantly larger than human writing speed.

Moving distance lower bound. Let d_i be the moving distance of the pen at time period i, and l_i^j be the distance between the pen and reader antenna j at time i. $\Delta l_i^j = l_{i+1}^j - l_i^j$ and can be computed as follows:

$$\Delta l_i^j = \begin{cases} (\theta_{i+1}^j - \theta_i^j) \cdot \lambda/(4\pi), & \text{if} \quad |\theta_{i+1}^j - \theta_i^j| < \pi \\ (\theta_{i+1}^j - \theta_i^j - 2\pi) \cdot \lambda/(4\pi), & \text{if} \quad \theta_{i+1}^j - \theta_i^j \geq \pi \\ (\theta_{i+1}^j - \theta_i^j + 2\pi) \cdot \lambda/(4\pi), & \text{if} \quad \theta_{i+1}^j - \theta_i^j \leq -\pi \end{cases} \tag{5}$$

According to the triangle inequity we have $d_i \geq \max\{|\Delta l_i^1|, |\Delta l_i^2|\}$.

Moving distance upper bound. We denote the maximum moving speed of the pen as v_{max}. Then the upper bound of the displacement is $v_{max}\Delta t$, where Δt is the time period between two consecutive tag readings. Based on above analysis, we know the pen moving distance d_i should be within an annulus: $\max\{|\Delta l_i^1|, |\Delta l_i^2|\} \leq d_i \leq v_{max}\Delta t$ (as shown in Figure 10(a)), which we term the *feasible region*. Within the feasible region, the ligature between the current location and each block forms the displacement *candidate set*. Since the pen's next location should be aligned with the pen's current moving direction, we can eliminate a large portion of infeasible locations in the feasible region, as shown in Figure 10(b). However, a large moving distance uncertainty still remains as many blocks are aligned with the pen moving direction.

Minimizing the moving distance uncertainty. PolarDraw further exploits the inter-antenna phase difference $\theta_i^2 - \theta_i^1$ to minimize the pen moving distance uncertainty. The relationship between phase readings and the tag-to-antenna distance can be formulated as follows:

$$\begin{cases} \theta_i^1 + 2k_1\pi = 4\pi l_i^1/\lambda \\ \theta_i^2 + 2k_2\pi = 4\pi l_i^2/\lambda \end{cases} \tag{6}$$

Subtracting the above equations and dropping subscripts:

$$\Delta l_i^{2,1} = \frac{\lambda}{4\pi}(\Delta\theta_i^{2,1} + 2k\pi) \tag{7}$$

where k is an unknown integer, $\Delta\theta_i^{2,1} = \theta_i^2 - \theta_i^1$ and $\Delta l_i^{2,1} = l_i^2 - l_i^1$. Let (x_1, y_1) and (x_2, y_2) be the location of two antennas, $\Delta l_i^{2,1}$ be the distance difference of these two antenna-to-tag links, we can construct a hyperbola with the two foci at the location (x_1, y_1) and (x_2, y_2) as our location estimation. Due to the phase ambiguity, we acquire multiple hyperbolas. Therefore, the next location of the pen may locate on all these hyperbolas. We minimize the displacement uncertainty by intersecting the hyperbolas with the location candidates already acquired, as shown in Figure 10(c).

3.4 Tracking pen trajectory

The design so far focuses on how to track the moving direction and the moving distance of the pen, separately. Now we put them together and show how to leverage the moving direction and the moving distance to infer the pen's trajectory fragment at each time interval.

We approach pen trajectory tracking problem as a discrete-time state estimation problem, where the state of the system at time t, X_t, is the location of the pen at that time. The measurement at time t, Y_t, is the phase and RSS readings reported by the two antennas. PolarDraw employs a Hidden Markov Model (HMM) to find the most likely sequence of pen trajectory segments.

HMM Prior. The HMM consists of a set of interconnected states, each of which emits an observable output. Each state is characterized by two probabilities: the *transition probability* over states and the *emission probability* over the output symbol.

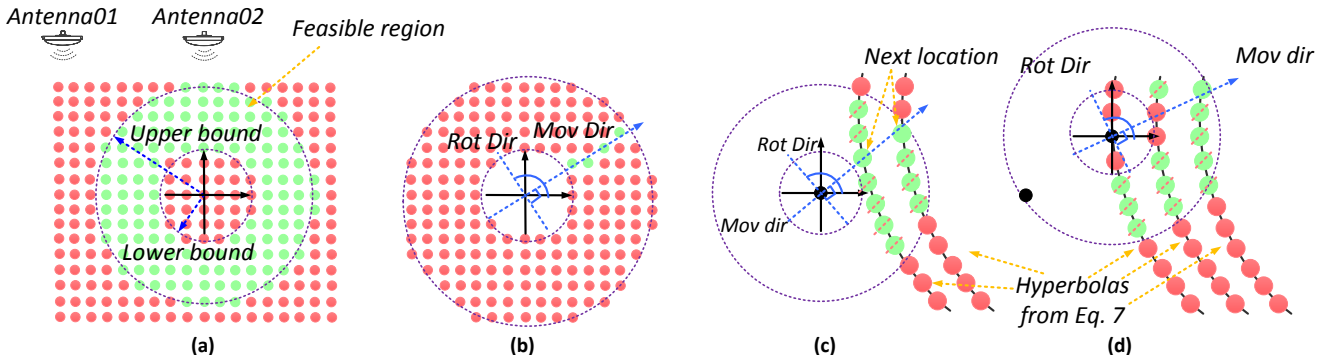


Figure 10— Moving distance inference; (a): under the lower and upper distance bound. (b). under the moving direction constraint; (c): under the hyperbola constraint. (d): tracking the second trajectory segment of the pen.

The moving trajectory of an object described by HMM generates a sequence of output symbols according to the highest emission probability of the current state, transitioning to the next state based on the highest transition probability.

Modeling the state transition probability. Without loss of generality, we divide the whiteboard into equal-length blocks, denoted $B_{i,j}$. Then the state space of HMM consists of all blocks on the whiteboard. We consider the transitions between consecutive states with equal probability [3]. The state transition probability $Pr(X_{t+1} = B_{i,j} | X_t = B_{k,l})$ is defined as:

$$\begin{cases} \frac{1}{M}, & \text{if } \max\{|\Delta l_t^1|, |\Delta l_t^2|\} \leq d(B_{i,j}, B_{k,l}) \leq v_{max}\Delta t, \\ 0, & \text{otherwise.} \end{cases} \tag{8}$$

where M is the the number of blocks within the feasible region. $d(B_{i,j}, B_{k,l})$ is the distance between $B_{i,j}$ and $B_{k,l}$.

Modeling emission probability. At time t, the observation Y_t is represented as a quad: $[\theta_t^1, \theta_t^2, s_t^1, s_t^2]$. The emission probability for a given state represents the likelihood of seeing the pen located at the block conditioned on the observation, *i.e.*, RSS and phase readings. In PolarDraw, we model the emission probability by jointly considering both the moving direction constraint and the hyperbola constraint. Let $B_{k,l}$ and α_r^{t-1} be the location and the rotation angle of the pen at time $t-1$, respectively. Consequently, we use a linear function $f(B_{k,l}, \tan(\alpha_r^{t-1} + \frac{\pi}{2}))$, which passes through $B_{k,l}$ and has a slope [4] of $\tan(\alpha_r^{t-1} + \frac{\pi}{2})$, to describe the approximate trajectory of the pen. The pen location at the next timestamp t should be on this trajectory. On the other hand, the pen's next location should be on the hyperbolas we acquired based on the inter-phase difference of two antennas $\Delta\theta_t^{1,2}$. Thus, by jointly considering the moving direction and the hyperbola constraint, we define the emission probability $Pr(X_t = B_{i,j} | Y_t = \theta_t^1, \theta_t^2, s_t^1, s_t^2)$ as follows:

$$\begin{aligned} Pr(X_t = B_{i,j} | Y_t) = & \frac{4\pi - |\Delta\theta_t^{1,2} - \theta_{B_{i,j}}^{1,2}|}{4\pi} \\ & \cdot \frac{d_{max} - d(B_{i,j}, f(B_{k,l}, tan(\alpha_r^{t-1} + \frac{\pi}{2})))}{d_{max}} \end{aligned} \tag{9}$$

where $\Delta\theta_{B_{i,j}}^{1,2}$ is the theoretical inter-phase difference of two antennas on block $B_{i,j}$. d_{max} is the maximum moving distance of the pen within the feasible region. In this equation, the first factor describes the likelihood that the block $B_{i,j}$ is on the hyperbolas computed from $\Delta\theta_i^{1,2}$; the second factor indicates the likelihood that the block $B_{i,j}$ is on the pen trajectory.

Trajectory Rotation. PolarDraw leverages Viterbi decoding to find the most likely pen trajectory. As mentioned in §3.2, the initial azimuthal angle may have an error $\tilde{\alpha_a}$. Hence after the trajectory tracking, PolarDraw corrects the azimuthal angle error to acquire a more accurate pen trajectory. Let $\mathbb{P} = \{P_{x_1,y_1}, P_{x_2,y_2}, ..., P_{x_T,y_T}\}$ be pen's trajectory recovered by the Viterbi algorithm, where P_{x_t,y_t} is the location of the pen on time t. PolarDraw eliminates the impact of $\tilde{\alpha_a}$ on pen's trajectory as follows:

[3] We leave more sophisticated motion modeling, such as Kalman filter and particle filter for future work.
[4] $\tan(\alpha_r^{t-1} - \frac{\pi}{2})$ when $\frac{\pi}{2} \leq \alpha_r^{t-1} \leq \pi$.

$$\hat{\mathbb{P}} = \mathbb{P} \cdot \begin{bmatrix} \cos\tilde{\alpha}_r & -\sin\tilde{\alpha}_r \\ \sin\tilde{\alpha}_r & \cos\tilde{\alpha}_r \end{bmatrix} \tag{10}$$

where $\hat{\mathbb{P}}$ is the new trajectory of the pen; $\tilde{\alpha}_r$ is the rotational angle error induced by the azimuthal angle error.

4. Implementation

This section details the implementation of PolarDraw.

4.1 Frontend software and Backend Software

RF sensing module. The front-end hardware of PolarDraw consists of an ImpinJ Speedway R420 RFID reader [12]; two LAIRD PA912 (LP) linearly polarized antennas [15], and an Avery Dennison AD-227m5 UHF passive RFID tag [6]. The passive tag is attached on a standard whiteboard pen.

Software processing module. The two main tasks of PolarDraw are to control the RFID reader to interrogate RFID tag and process the tag readings for pen trajectory tracking. The tag interrogation module is implemented in Java, and runs on a Lenovo ThinkCentre PC equipped with an i5-4590 CPU and 4 GB RAM. It collects tag readings (*e.g.*, RSS, phase, timestamp etc.) through the Low Level Reader Protocol (LLRP) [18] and stores them in Comma separated value (CSV) files. The pen trajectory tracking module is implemented in Matlab. It fetches data from the stored CSV files and processes them for pen trajectory tracking.

4.2 Implementation issues

We face two issues when implementing PolarDraw.

Modulation scheme selection. A typical EPC GEN 2 reader supports a set of pre-configured Gen2 modes of tag interrogation. These modes differ in the modulation scheme, hence each mode has a different reading rate and a different level of sensitivity to RF interference. In PolarDraw, a higher reading rate is beneficial to the fine-grained pen trajectory tracking. However, the low SNR would lead to severe phase noises. So it is crucial to balance the reading rate and the capability of resistance to noise. PolarDraw round-robins all the selectable modulation schemes in an order from the highest reading rate to the lowest. The first scheme with the standard phase variances $Var(\theta) \leq \delta$ is selected for tag interrogation. We conduct extensive experiments to test various thresholds, and set $\delta = 0.1$, which optimizes the system performance.

Data smoothing. Data smoothing aims to improve the stability of RSS and phase samples by mitigating the phase and RSS noises of the raw data. The noise sources could be RFID hardware, multipath signal propagation *etc.*. To mitigate these nosies. PolarDraw first segments the RSS and phase series into windows. The window size is set to 50 ms. Within each window, PolarDraw further averages the RSS and phase readings. These averaged data form the new RSS and phase series. Finally, PolarDraw performs moving average smoothing on these new RSS and phase series.

5. Evaluation

In this section, we begin with experimental methodology (§5.1), after which we conduct end-to-end experiments in a laboratory environment to evaluate PolarDraw's performance (§5.2). Then we compare PolarDraw with two state-of-the-art motion tracking systems, Tagoram and RF-IDraw. Finally, we present micro-benchmark experiments to provide insights into PolarDraw, particularly to understand which fa-ctors impact PolarDraw's performance (§5.4).

5.1 Experimental methodology

To determine the groundtruth, we photograph the user's writing on the whiteboard and leverage edge detection algorithm-s to extract writing trajectories. We use LipiTk [10] for hand writing recognition.

Metric. We use the following metrics:

- *Recognition Accuracy*: the fraction of successful character recognitions over the total number of characters.

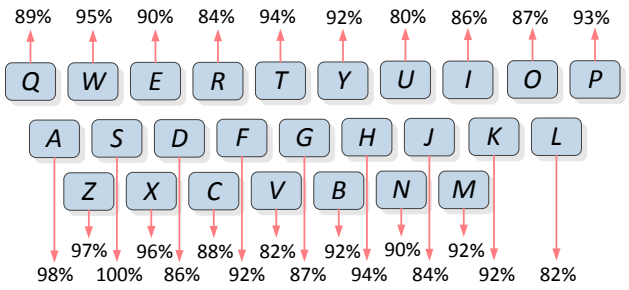


Figure 11— PolarDraw's letter recognition accuracy.

- *Similarity*: the procrustes distance between the recovered trajectory and the groundtruth trajectory. The procrustes distance determines a linear transformation (translation, rotation, and scaling) of the points in trajectory A to best conform them to the points in Trajectory B. The goodness-of-fit criterion is the sum of squared errors.
- *Confusion matrix*: each row shows the actual character and each column shows the recovered character.

Compared schemes. We compare PolarDraw with two oth-er tracking algorithms: RF-IDraw [37] and Tagoram [42]. RF-IDraw adopts eight spatially separated antennas for RFI-D tag localization and tracking. Most COTS RFID reader support four antennas a piece, so we compare four-antenna version of RF-IDraw with PolarDraw, for equal hardware resources. Tagoram adopts four antennas to locate and track an RFID tag based on the phase readings. Similarly, we implement Tagoram using four omnidirectional antennas.

5.2 End-to-end performance

We first take field studies to evaluate PolarDraw. The algorithm parameters are set according to the result in §5.4.

5.2.1 Recognition accuracy over different character

We first look at the recognition accuracy over 26 characters. In these experiments, we invite a volunteer to write all 26 characters 100 times. The result is shown in Figure 11. PolarDraw achieves 93.6% recognition accuracy on average. Specifically, 15 out of 26 characters are correctly recognized with a probability higher than 90%. 21 out of 26 characters are correctly recognized with a probability higher than 85%. The remaining 5 characters have a much lower recognition accuracy. However, their recognition accuracies are still above 80%. We believe by applying natural language processing techniques, we can further increase the recognition accuracy.

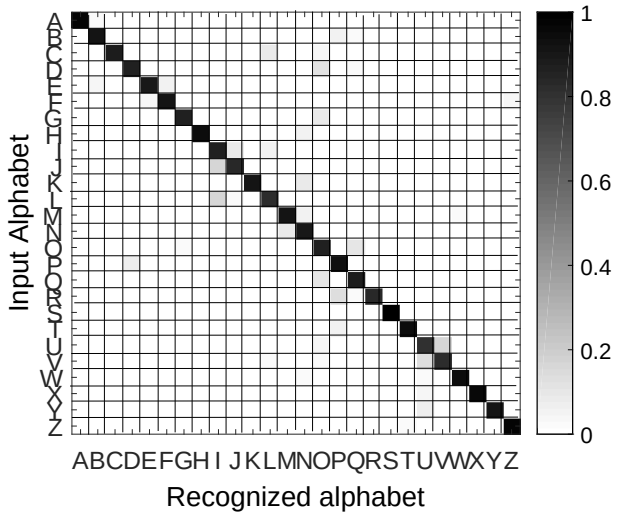


Figure 12— PolarDraw's letter confusion matrix.

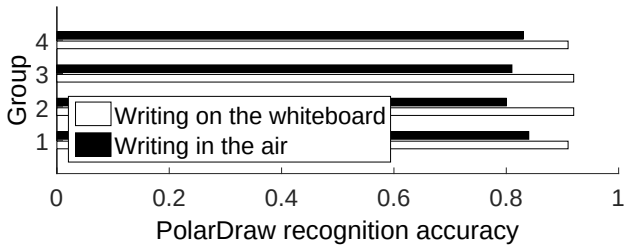


Figure 13— Writing in air vs. on the whiteboard.

5.2.2 Confusion matrix over different character

To better understand why PolarDraw performs poorly in som-e characters, we further show the confusion matrix in Figure 12. The darker areas represents higher rate of recognitions. We observe from this figure that the number of misclassified characters vary for different characters. A large portion of false negatives or positives usually correspond to those characters that share the similar writing style. For example, The character L and V are prone to be incorrectly recognized as I and U. Besides, the recognition accuracy also varies with the complexity of character writing. For example, those characters that can be written in a single stroke usually achieve a higher recognition accuracy.

5.2.3 Performance of writing in the air

We further remove the whiteboard and let the user write in the air. We conduct four groups of experiments here. In each group, we randomly choose 10 letters and let the user write each letter 10 times in air. For comparison, this user is also required to repeat these experiments but write on the whiteboard. Figure 13 shows the resulting recognition accuracy. As the result shows, PolarDraw performs consistently when the user is writing on the whiteboard, achieving a recognition accuracy around 91% on average. In contrast, PolarDraw's performance declines about 8% when the user writes in the air. The reason is that without the whiteboard, human writings will not be confined to 2D space, which leads to errors in the pen moving distance inference. However, PolarDraw still achieves over 80% recognition accuracy in this case.

5.3 Comparison vs. RF-IDraw and Tagoram

We then compare PolarDraw with other two state-of-the-arts: Tagoaram and RF-IDraw. The setup is in Figure 14.

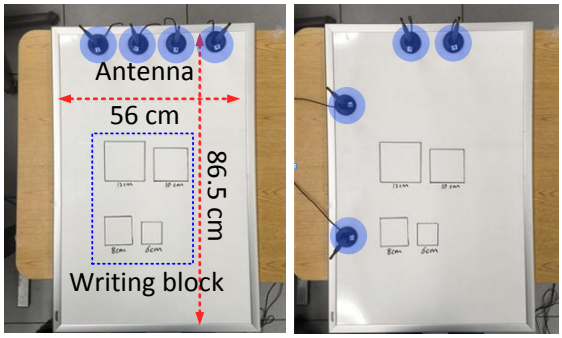


Figure 14— Hardware setups of Tagoram and RF-IDraw.

5.3.1 Recognition accuracy over writing words

We first compare these three algorithms' ability on written words tracking. We divide the testing words into four groups according to the word length. Within in each group, we randomly select 10 words from the Oxford English Dictionary (O.E.D) [22]. Figure 15 shows the word recognition accuracy of PolarDraw, Tagoram and RF-IDraw. When the word contains two characters, PolarDraw achieves a similar performance with both RF-IDraw and Tagoram, with a recognition accuracy over 91%. As the number of characters in the word increases, the performance of all these three algorithms drops slightly. The performance of

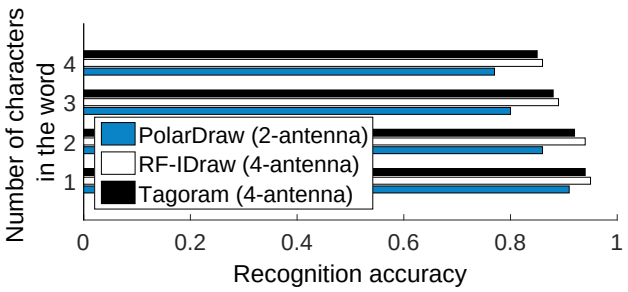


Figure 15— Recognition accuracy for various words.

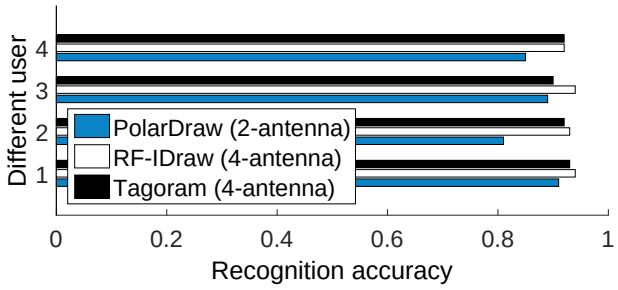


Figure 16— Recognition accuracy over different users. User 2 writes with a "stiff" style, rotating the pen slightly.

PolarDraw degrades much significantly than other two algorithms. However, with just two antennas, the minimum recognition accuracy of PolarDraw is still above 75%. With nature language processing techniques, we believe the recognition accuracy can be improved further. Besides, we find that the recognition accuracy of RF-IDraw here is lower than that in [37]. This is due to the less number of antennas used in this experiments (four antennas here and eight antennas in [37]).

5.3.2 Recognition accuracy over different users

In this experiment we examine the recognition accuracy over different users. Figure 16 shows the resulting recognition accuracy. As the result shows, both PinIt and Tagoram achieve consistent performance for all these four users. While PolarDraw's performance varies due to the different user writing style. This is as expected since PolarDraw uses two antennas and relies on the pen rotation to determine the pen moving direction, while some user may not rotate when writing.

5.3.3 Trajectory similarity

We further examine the trajectory similarity of handwritten letters recovered by these three algorithms. Here we randomly choose 5 characters and invite one volunteer to write each character 10 times. The font size is set to 20cm. Figure 17 shows the cumulative distribution function (CDF) of the Procrustes distance between the recovered trajectories and the groundtruth. Tagoram and RF-IDraw achieves similar performance, with a 90^{th} percentile of 11.3 cm and 10.2 cm, respectively. PolarDraw's performance decreases slightly. However, the 90^{th} percentile is constrained to 13.8 cm, which is not too large than other two algorithms.

We further show a snapshot of the pen trajectory recovered by these three algorithms in Figure 18. Compared with the groundtruth, we can see that all the recovered trajectories are stretched or rotated due to the localization and tracking errors. Comparing the trajectories recovered by these three algorithms, we find that these trajectories are distinct from each other, especially at the beginning and the ending part of the pen trajectory. Nevertheless, all of them preserve the basis profile of the handwritten letters.

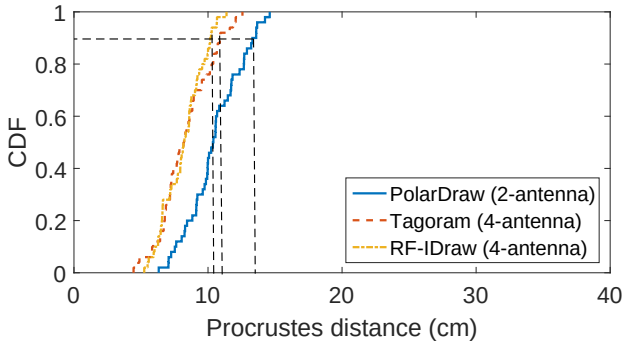


Figure 17— The CDF of the procrustes distance between the groundtruth and the trajectory recovered by algorithms.

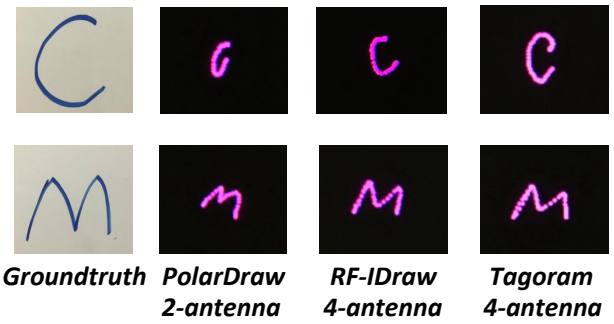


Figure 18— Pen trajectory recovered by three algorithms.

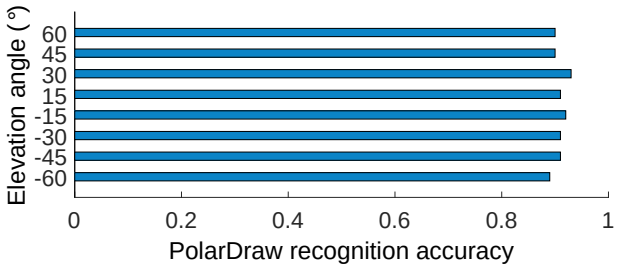


Figure 19— PolarDraw's recognition accuracy over α_e.

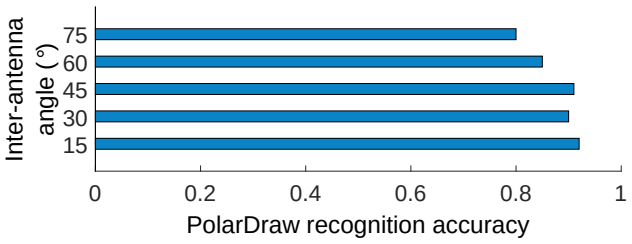


Figure 20— PolarDraw's recognition accuracy over γ.

5.4 Microbenchmarks

We then conduct microbenchmarks to give insight into PolarDraw's performance. In particular, we seek to understand which factors will most affect tracking performance as we turn various operating parameters, and understand how polarization angle mismatch helps to improve the pen trajectory tracking accuracy. The hardware setup of our microben-chmarks is shown in Figure 4 on page 4. In the following experiments, we randomly choose 10 letters from the English alphabet and invite a volunteer to write each letter 10 times under different parameter settings.

5.4.1 Choosing elevation angle

We first examine how the elevation angle (§3.1) α_e affects the trajectory tracking performance. Figure 19 shows the resulting recognition accuracy over different elevation angles. As the result shows, PolarDraw achieves similar performance under different α_e settings. Suggested by this result, we set $\alpha_e = 30^\circ$ as our default settings in all experiments.

5.4.2 Choosing inter-antenna angle

We then examine how the inter-antenna angle (§3.2) γ affects the trajectory tracking performance. Figure 20 shows the resulting recognition accuracy under different γ settings. PolarDraw achieves similar recognition accuracy when γ is relatively small (*e.g.*, $\gamma = 15^\circ, 30^\circ, 45^\circ$). This is because the pen is prone to cross over the boarder of neighbouring sectors (as shown in Figure 6 on page 6) when the user writes on the whiteboard. As expected, when γ increases, the probability that the pen crosses over the border of neighboring sectors decreases, resulting in a lower recognition accuracy. Suggested by the experiment result, we set $\gamma = 15^\circ$ as our default settings in all experiments.

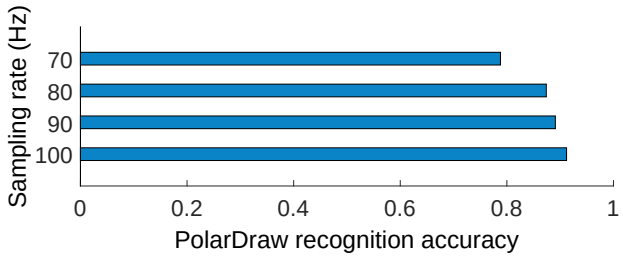


Figure 21— Recognition accuracy over sampling rates.

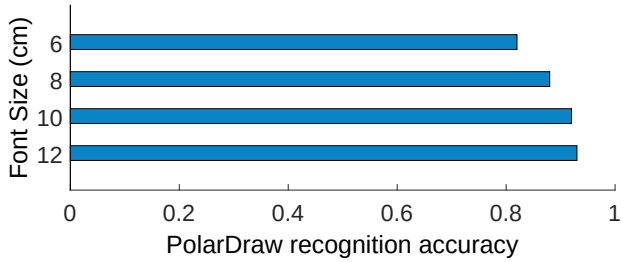


Figure 22— Recognition accuracy over different font size.

5.4.3 Impact of the sampling rate

We then evaluate the tracking performance of PolarDraw under different sampling rate settings. Figure 21 shows the resulting accuracy. PolarDraw achieves 92% recognition accuracy when the sampling rate is 100 Hz. The recognition accuracy decreases slightly as we reduce the sampling rate. This is as expected since the lower sampling rate lead to a much coarse-grained pen trajectory which is difficult to be recognized. Nevertheless, recognition accuracy still exceeds 82% at the lowest sampling rate.

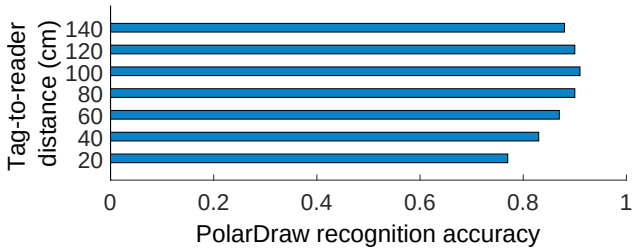


Figure 23— PolarDraw's recognition accuracy over different tag-to-reader distance settings.

5.4.4 Impact of the writing character size

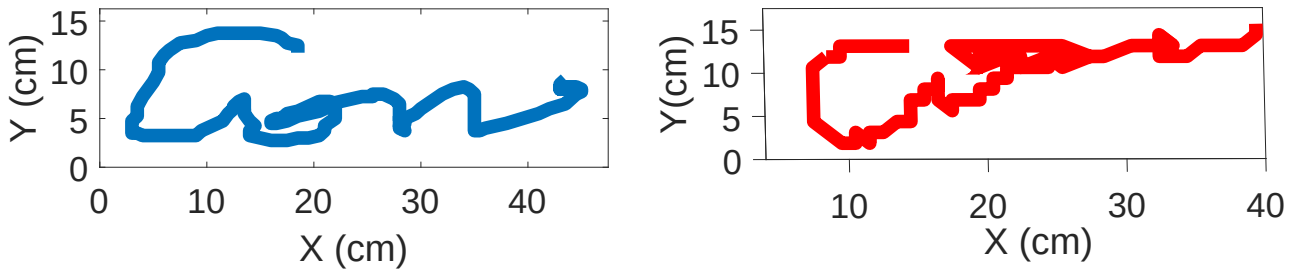
We examine the trajectory tracking performance under different font size of human writings. In these experiments, we manually split the writing area into different blocks, ranging from 12 cm to 6 cm, hence covering the normal size of human writings on the whiteboard. Figure 22 shows the resulting recognition accuracy. The recognition accuracy maintains in a high level (around 92%) when the font size is relatively large (*e.g.*, 12cm and 10cm). It then drops slightly to 88% when the font size is 8 cm, and further to 82% when the font size is 6 cm. This is because the pen rotation is less significant when the user writes in small font size. Such a minor change in the polarization mismatch angle leads to a small and even undetectable RSS change. Hence, PolarDraw fails to correctly estimate the pen moving direction.

5.4.5 Impact of the tag-to-reader distance

We further examine the impact of the tag-to-reader distance on the trajectory tracking performance. In these experiments, we vary the tag-to-reader distance from 20 cm to 140 cm, with a step of 20 cm. Figure 23 shows the recognition accuracy at each distance. The recognition accuracy is relatively low (around 78%) when the tag is close to the reader antenna (*e.g.*, 20 cm spacing). This is because both the polarization angle mismatch and the tag movement contribute to the RSS change. Hence PolarDraw fails to correctly estimate the pen moving direction. As we expand the spacing between the tag and the reader, the RSS tends to be less sensitive to the pen movement. Hence, PolarDraw could accurately estimate the pen moving direction, which results in an increased recognition accuracy, as shown in the experimental result. The recognition accuracy drops as we further expand the tag-to-reader distance from 100 cm to 120 cm and further to 140 cm. A possible explanation may be that the backscatter signal bounces off nearby objects and changes the polarization angle, which leads to an unusual RSS trend during human writing. We plan to investigate that use of more directional antennas to increase range in futhre work.

5.4.6 Gain of using polarization

To examine how polarization benefits the pen trajectory trac-king, we implement a version of PolarDraw that tracks pen trajectory based solely on the phase readings. Figure 24 shows a snapshot of the pen trajectory of the word *Cow*, as reconstructed by PolarDraw and this new version of PolarDraw. PolarDraw could successfully reconstruct the profile of the word by tracking both the moving direction and the moving distance of the pen. In contrast, without polarization, PolarDraw is unable to acquire the pen moving direction hence fails to recover any meaningful trajectory.



(a)— With polarization.

(b)— Without polarization.

Figure 24— Pen trajectory recovered by PolarDraw with and without polarization information.

6. Related work

Our system draws on prior works in multiple areas, chiefly gesture recognition, Wi-Fi sensing and RFID positioning.

6.1 Gesture recognition

Human gesture recognition plays a key role in mobile computing and human-computer interaction systems. A rich bod-y of works, however, have focused on computer-vision based methods [11], which are sensitive to light conditions. With the prosperity of smart phone market and widely accessible WiFi networks, researchers begin to address the gesture recognition problem using smartphone and Wi-Fi signals.

Based on inertial sensors: E-Gesture [24] employs gyroscope and accelerometer to characterize eight kinds of predefined hand gestures and designs a HMM-based classification model to better fit multiple mobility situations. Simiarly, RisQ [23] recognizes smoking gestures by using the accelerometer and gyroscope readings from the smart wristband. However, both these two systems only recognize simple and pre-defined hand gestures like throw, draw and smoking *etc.*, and they fail to characterize the fine-grained finger motion and hand writings. PhonePoint Pen [4] enables in-air human writing by tracking the hand-hold smart phone's motion trail with the accelerometer. MoLe [35] designs a inference algorithm to detect and identify the human typing input through the smart watch accelerometer readings. However, since the moving distance error propagates significantly after twice integration of noisy accelerometer readings, these systems fail to precisely recover the written words.

Based on acoustic/Wi-Fi signals: WiDraw [32] enables in-air writing by exploiting the change in the AoA spectrum caused by human arm motions. However, WiDraw requires many antennas' support and sizable human writings. AAMouse [44] designs a acoustic-ranging based method to track the smart phone's trajectory with an median error of around 1.4 cm. However, it may works in problem due to the acoustic noises. mTrack [40] leverages the highly-directional 60 GHz millimeter wave to pinpoint and track the target with 90-percentile error below 8 mm. However, it requires costly infrastructures and only works in near-field. RF-IDraw [37] designs a multi-resolution positioning scheme for virtual input recognition by leveraging the power of spatially deploye-d antenna arrays. However, it requires a significant number of physically-separated antennas and rigid calibration of antenna locations, reducing deployability. In contrast, PolarDraw requires minimal supporting infrastructures yet achiev-es high tracking accuracy.

6.2 Wi-Fi sensing

Researchers model the impact of human body on Wi-Fi signals to infer the human location and human activities.

Localization and Tracking: There are a surge of works focusing on human/device localization and tracking based on the analysis of WiFi signal patterns. However, most of these proposals achieve an order of several meters localization accuracy [7, 21, 26, 43], hence they are unable to characterize the fine-grained human writing. Although some Time-of-flight(ToF) or Angle-of-arrival (AoA)-based localization schemes could achieve decimeter-level or even centimeter-level localization accuracy [2, 13, 14, 29, 30, 41, 45], they either rely on dedicated hardware for signal processing (like USRP) or require densely deployed infrastructures, which may infeasible in many mobile application cases.

Human activity recognition: There are also extensive wor-ks in decoding human activities based on Wi-Fi signals. E-eyes [39] detects and classifies different kinds of coarse-grained human activities (washing, sleeping, cooking *etc.*) at home through the use of commercial Wi-Fi access pints. WiHear [34] recognizes human speakings by analyzing the CSI pattern of Wi-Fi signals reflected by human lips. WiKey [5] decodes user input on keyboards by matching the Wi-Fi signal patterns to the fingerprint. However, all these works are fingerprinting-based and only work for predefined coarse-grained activities. Hence they cannot be directly applied to fine-grained motion tracking and human writing recognition since the writing style varies from people to people.

6.3 RFID positioning

Initial attempts in RFID positioning are using RSS to characterize the distance between the reader and the tag. Later on researchers exploit fine-grained phase information of the backscatter signal for more accurate tag localization. BackPos [17] designs a hyperbolic-based positioning system with multiple RFID reader (≥ 3) antennas. However, the localization accuracy

of BackPos is around 13 cm, which is still unsuitable for fine-grained hand writing tracking. Tagoram [42] introduces a holography-based tag tracking algorithm, which achieves an accuracy of around 1 cm based on four reader antennas. In contrast, PolarDraw requires only two antennas hence sets a new standard for minimal supporting infrastructure. PinIt [36] localize RFID tags in Non-line-of-sight (NLOS) environment by exploiting the multipath profile of each tag. STPP [31] adopts the mobile RFID reader to acquire the spatial order of tags without localizing them. However, the localization error of these two systems retain in a relatively high level (around 12 cm for PinIt, and 8 cm for STPP), hence they may fail in hand motion tracking.

7. Limitations and future works

We discuss limitations and opportunities for improvement.

Confined to separated words. The tag attached on the pen responses to the reader all the time, hence PolarDraw is unable to distinguish the words from a sentence. However, we believe with light-weighted inertial sensors on the tag (*e.g.*, WISP), it is possible to detect whether the pen is touching the whiteboard or not by examining the accelerometer readings. We leave this problem for our future work.

Limited performance for stiff users. As shown in Figure 16 on page 14 in our experimental results, PolarDraw achi-eves best performance when the user writes naturally, with a small rotation during writing, and high but sub-optimal when the user writes in an unnatural way, not rotating the pen at all when writing on the whiteboard. But we have found early evidence that the accelerometer readings on the RFID tags offer valuable hints on the moving direction of the pen, which we leave for future work.

Phase jumping correction. As shown in Figure 3(b) on page 4, the phase reading jumps when the polarization angle of the tag and the reader antenna are totally mismatched. Although this boarder case seldom occurs when the user is writing on the whiteboard where the pen rotation is usually confined to a small angle, it may occurs frequently when the user writes in the air. We leave this as our future work and plan to detect and drop these irregular data and use the remaining for pen trajectory tracking.

8. Conclusion

In this paper we present the design and implementation of PolarDraw, the first motion tracking system that can accurately reproduce and recognize handwritten letters in the air or on a whiteboard with just two nearby antennas. PolarDraw tracks the tag moving direction and the distance by leveraging the polarization mismatch-induced power loss and the tag displacement-induced phase changing, based on which it carefully crafts the tag motion trail. The experimental comparison with other state-of-the-art multi-antenna tracking systems demonstrates that PolarDraw could achieve competitive tracking performance yet requires significantly lower hardware support.

Acknowledgement. The research leading to these results has received funding from the European Research Council under the European Union's Seventh Framework Programme (FP/2007-2013) / ERC Grant Agreement no. 279976.

References

[1] F. Adib, Z. Kabelac, D. Katabi. Multi-person localization via rf body reflections. *NSDI*, 2015.

[2] F. Adib, Z. Kabelac, D. Katabi, R. Miller. 3D Tracking via body radio reflections. *NSDI*, 2014.

[3] F. Adib, D. Katabi. See through walls with Wi-Fi! *SIGCOMM*, 2013.

[4] S. Agrawal, I. Constandache, S. Gaonkar, R. Roy Choudhury, K. Caves, F. DeRuyter. Using mobile phones to write in air. *MobiSys*, 2011.

[5] K. Ali, A. X. Liu, W. Wang, M. Shahzad. Keystroke recognition using WiFi signals. *MobiCom*, 2015.

[6] Avery-Dennison UHF RFID inlay *AD-227m5*.

[7] P. Bahl, V. N. Padmanabhan. RADAR: An in-building RF-based user location and tracking system. *Infocom*, 2000.

[8] Emerald.

[9] Google Project Soli .

[10] LipiTk: online hand writing recognition toolkit.

[11] L. He, Z. Zhang. Real-time whiteboard capture and processing using a video camera for remote collaboration. *IEEE Trans. on Multimedia*, **9**(1), 198–206, 2007.

[12] ImpinJ Speedway R420 reader.

[13] K. Joshi, D. Bharadia, M. Kotaru, S. Katti. WiDeo: Fine-grained device-free motion tracing using RF backscatter. *NSDI*, 2015.

[14] S. Kumar, S. Gil, D. Katabi, D. Rus. Accurate indoor localization with zero start-up cost. *MobiCom*, 2014.

[15] Laird PA9-12 (LP) outdoor RFID antenna.

[16] W. Lee, Y. Yeh. Polarization diversity system for mobile radio. *IEEE Trans. on Commms.*, (5), 912–923, 1972.

[17] T. Liu, L. Yang, Q. Lin, Y. Guo, Y. Liu. Anchor-free backscatter positioning for RFID tags with high accuracy. *Infocom*, 2014.

[18] LLRP Toolkit.

[19] MimioProjector Touch Projector.

[20] MimioTeach Interactive Whiteboard.

[21] A. Musa, J. Eriksson. Tracking unmodified smartphones using Wi-Fi monitors. *SenSys*, 2012.

[22] oxford english dictionary.

[23] A. Parate, M.-C. Chiu, C. Chadowitz, D. Ganesan, E. Kalogerakis. Risq: Recognizing smoking gestures with inertial sensors on a wristband. *MobiSys*, 2014.

[24] T. Park, J. Lee, I. Hwang, C. Yoo, L. Nachman, J. Song. E-gesture: a collaborative architecture for energy-efficient gesture recognition with hand-worn sensor and mobile devices. *SenSys*, 2011.

[25] Q. Pu, S. Gupta, S. Gollakota, S. Patel. Whole-home gesture recognition using wireless signals. *MobiCom*, 2013.

[26] A. Rai, K. K. Chintalapudi, V. N. Padmanabhan, R. Sen. Zee: Zero-effort crowdsourcing for indoor localization. *MobiCom*, 2012.

[27] T. Rappaport. *Wireless Communications: Principles and Practice*. Prentice Hall, second edn., 2002.

[28] T. Rappaport, D. Hawbaker. Wide-band microwave propagation parameters using circular and linear polarized antennas for indoor wireless channels. *IEEE Trans. on Comms.*, **40**(2), 240–245, 1992.

[29] S. Sen, R. R. Choudhury, B. Radunovic, T. Minka. Precise indoor localization using PHY layer information. *MobiSys*, 2011.

[30] S. Sen, B. Radunovic, R. R. Choudhury, T. Minka. You are facing the Mona Lisa: Spot localization using PHY layer information. *MobiSys*, 2012.

[31] L. Shangguan, Z. Yang, A. X. Liu, Z. Zhou, Y. Liu. Relative localization of RFID tags using spatial-temporal phase

profiling. *NSDI*, 2015.

[32] L. Sun, S. Sen, D. Koutsonikolas, K.-H. Kim. WiDraw: Enabling hands-free drawing in the air on commodity WiFi devices. *MobiCom*, 2015.

[33] D. Vasisht, S. Kumar, D. Katabi. Decimeter-level localization with a single WiFi access point. *NSDI*, 2016.

[34] G. Wang, Y. Zou, Z. Zhou, K. Wu, L. M. Ni. We can hear you with Wi-Fi! *MobiCom*, 2014.

[35] H. Wang, T. Lai, R. Choudhury. MoLe: Motion leaks through smartwatch sensors. *MobiCom*, 2015.

[36] J. Wang, D. Katabi. Dude, where's my card? RFID positioning that works with multipath and non-line of sight. *SIGCOMM*, 2013.

[37] J. Wang, D. Vasisht, D. Katabi. RF-IDraw: Virtual touch screen in the air using RF signals. *SIGCOMM*, 2014.

[38] W. Wang, A. Liu, M. Shahzad, K. Ling, S. Lu. Understanding and modeling of WiFi signal based human activity recognition. *MobiCom*, 2015.

[39] Y. Wang, J. Liu, Y. Chen, M. Gruteser, J. Yang, H. Liu. E-eyes: device-free location-oriented activity identification using fine-grained WiFi signatures. *MobiCom*, 2014.

[40] T. Wei, X. Zhang. mTrack: High-precision passive tracking using millimeter wave radios. *MobiCom*, 2015.

[41] J. Xiong, K. Jamieson. ArrayTrack: A fine-grained indoor location system. *NSDI*, 2013.

[42] L. Yang, Y. Chen, X.-Y. Li, C. Xiao, M. Li, Y. Liu. Tagoram: Real-time tracking of mobile RFID tags to high precision using COTS devices. *MobiCom*, 2014.

[43] M. Youssef, A. Agrawala. The Horus WLAN location determination system. *MobiSys*, 2005.

[44] S. Yun, Y.-C. Chen, L. Qiu. Turning a mobile device into a mouse in the air. *MobiSys*, 2015.

[45] Y. Zhu, Y. Zhu, B. Zhao, H. Zheng. Reusing 60GHz radios for mobile radar imaging. *MobiCom*, 2015.