ShakeReader: 'Read' UHF RFID using Smartphone

Kaiyan Cui[1,2], Yanwen Wang[3], Yuanqing Zheng[1], Jinsong Han[4]
[1]The Hong Kong Polytechnic University, Hong Kong, China
[2]Xi'an Jiaotong University, Xi'an, Shaanxi, China
[3]Hunan University, Changsha, Hunan, China
[4]Zhejiang University, Hangzhou, Zhejiang, China
kaiyan.cui@connect.polyu.hk, wangyw@hnu.edu.cn, yqzheng@polyu.edu.hk, hanjinsong@zju.edu.cn

***Abstract*—UHF RFID technology becomes increasingly popular in RFID-enabled stores (*e.g.*, UNIQLO), since UHF RFID readers can quickly read a large number of RFID tags from afar. The deployed RFID infrastructure, however, does not directly benefit smartphone users in the stores, mainly because smartphones cannot read UHF RFID tags or fetch relevant information (*e.g.*, updated price, real-time promotion). This paper aims to bridge the gap and allow users to 'read' UHF RFID tags using their smartphones, without any hardware modification to either deployed RFID systems or smartphone hardware. To 'read' an interested tag, a user makes a pre-defined smartphone gesture in front of an interested tag. The smartphone gesture causes changes in 1) RFID measurement data (*e.g.*, phase) captured by RFID infrastructure, and 2) motion sensor data (*e.g.*, accelerometer) captured by the user's smartphone. By matching the two data, our system (named *ShakeReader*) can pair the interested tag with the corresponding smartphone, thereby enabling the smartphone to indirectly 'read' the interested UHF tag. We build a novel reflector polarization model to analyze the impact of smartphone gesture to RFID backscattered signals. Experimental results show that *ShakeReader* can accurately pair interested tags with their corresponding smartphones with an accuracy of >94.6%.**

***Index Terms*—Human-RFID Interaction, Reflector Polarization Model, RFID System**

I. Introduction

Radio Frequency IDentification (RFID) technology has been widely used in retail stores (*e.g.*, UNIQLO [1], Zara [2], *etc.*) for logistics, sales tracking and shopping behavior analysis. Compared with traditional labelling technologies (*e.g.*, QR-code, NFC), Ultra High Frequency (UHF) RFID is more attractive to stores, because it allows quick scanning of a large number of RFID-labelled items, achieving much higher operation efficiency. Leveraging the deployed RFID infrastructure, merchants can also capture customers' interests by analyzing RFID data and optimize marketing strategy to maximize their profits [3]. As such, more and more stores are expected to deploy UHF RFID systems in the future.

Such a deployed RFID infrastructure, however, does not directly benefit customers during shopping. For example, while detailed item information (*e.g.*, coupon, promotion, price comparison, matching tips) could be potentially accessed, flexibly updated, and presented on smartphones, such item-specific information is not available to customers in physical stores. That is mainly because smartphones are limited by the unavailability of any direct communication with UHF RFID tags. This paper aims to enable users to 'read' on-the-fly item-specific information by bridging the gap between the deployed RFID infrastructure and smartphones without making any hardware modification to either RFID system or smartphones.

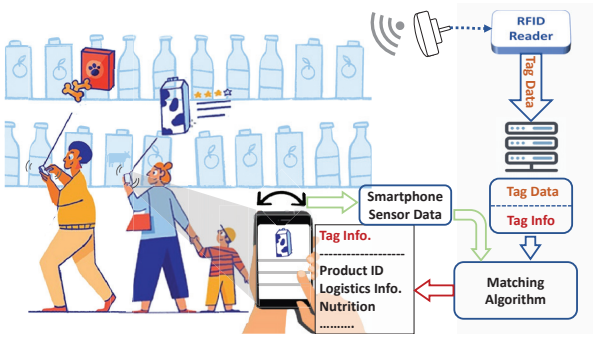


Fig. 1: Application scenario: A lady 'reads' the item-specific information by making a gesture with her smartphone.

In this paper, we develop a system named *ShakeReader*, which allows a user to interact with an RFID-labelled item by simply performing a pre-defined gesture (*e.g.*, shaking a smartphone) nearby the interested tag and automatically delivering item-specific information to the smartphone. Fig. 1 illustrates a usage scenario. Interested in a box of milk, a user makes a pre-defined gesture with her smartphone. Such a gesture causes changes to backscattered signal of the labelled RFID tag attached to the milk box. The changes in backscattered signal can be captured by an RFID reader. Meanwhile, the user's smartphone detects the smartphone gesture using motion sensors. By matching the two data capturing the same smartphone gesture, *ShakeReader* can deliver the interested tag information to the corresponding smartphone user.

We note that our objective is *not* to replace other labelling technologies (*e.g.*, QR-code, NFC), but is to provide a technology that could allow users to read the readily-deployed UHF tags in stores. We believe this technology can complement other labelling technologies in practice.

Although useful in practice and simple in concept, the system entails tremendous technical challenges. First, despite plenty of previous works on RFID and mobile sensing, it is still challenging to use only one tag, which remains static and is not attached on the smartphone, for accurately recognizing the smartphone gesture performed nearby. Second, users in stores may influence the gesture detection accuracy as other human activities may influence backscattered signal of RFID tags. Third, many users may perform similar gestures near multiple tags in the same store. How to correctly pair each tag with its corresponding smartphone is challenging in practice.

In this paper, we address all the above challenges. First, *ShakeReader* builds a reflector polarization model to charac-

terize the backscattered signal of a single tag caused by smartphone gestures. This reflection model simultaneously captures backscattered signal propagation and the polarization caused by smartphone reflection. By leveraging the polarization of reflected signal from smartphones, RFID readers can identify smartphone gestures even with a single tag. Second, we notice that irrelevant user movement indeed influences the backscattered signal measurement and may cause detection errors if not handled properly. To address this problem, *ShakeReader* pre-defines a smartphone gesture (clockwise and counter-clockwise rotation of smartphone in front of an interested tag) to facilitate the detection. Third, to pair the interested tag with its corresponding smartphone, *ShakeReader* leverages the synchronicity of the changes in RFID data and smartphone sensor data simultaneously affected by the same smartphone gesture. The synchronicity allows us to differentiate the smartphone gestures performed by different users in front of their interested tags.

The key contributions can be summarized as follows:

- We present *ShakeReader*, a system that enables a flexible human-RFID interaction using smartphones. *ShakeReader* allows smartphone users to indirectly 'read' UHF RFID tags using their smartphones, without any hardware modification to either the deployed RFID infrastructure or smartphones.
- We characterize and analyze the reflector polarization and its impact on backscattered signal in RFID systems.
- We conduct extensive evaluations on our proposed prototype system using COTS RFID system. The experimental results show that *ShakeReader* achieves >94.6% matching accuracy.

II. Background and Motivation

A. UHF RFID Technology and Existing Works

UHF RFID technology in stores. UHF RFID technology has been increasingly used in retail stores. For example, UNIQLO is currently using UHF RFID tags to label all the items to improve operational efficiency [1]. As UHF RFID supports wireless identification from afar, retailers are freed from manually scanning items one-by-one using handheld QR-code/NFC readers. The UHF RFID technology also helps reduce customers' waiting time in the checkout queue, as RFID-labelled items can be instantly identified with RFID readers at checkout counters. As such, we expect more stores will deploy UHF RFID systems to improve operational efficiency. We note that the objective of *ShakeReader* is not to replace alternative labelling technologies (*e.g.*, QR-code, NFC) but allow users to read the already-deployed UHF RFID tags in stores with their smartphones.

Current smartphones cannot read UHF RFID tags. While NFC tags can be read by NFC-enabled smartphones, most smartphones cannot read the deployed UHF RFID tags in stores. In order to wirelessly energize UHF RFID tags, a UHF reader needs to transmit continuous waves at high transmission power, which may quickly drain the battery of a smartphone. Although retailers can afford a handheld UHF reader and recharge the reader more frequently in stores, customers could be reluctant to purchase extra hardware to read the UHF tags and concerned about the battery life of the smartphone.

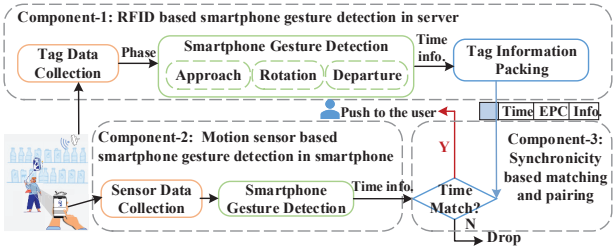


Fig. 2: System architecture of *ShakeReader*.

Existing works. Research works strive to enable smartphones to read UHF RFID tags. For example, TiFi [4] proposes to read tag IDs using RFID readers and broadcast tag IDs as Wi-Fi beacons, so that smartphones equipped with Wi-Fi modules can receive the tag IDs. However, as all tag IDs will be broadcast to smartphones, it is very challenging to correctly identify the interested tag among all the tag IDs.

B. System Architecture and Problem Definition

We assume that all N items are labelled with UHF RFID tags and the tags are covered by RFID readers. In practice, one reader can connect multiple reader antennas deployed in different locations. The readers continuously interrogate the tags and measure backscattered signal of the tags (*e.g.*, phase, signal strength). M clients in the environment specify their interests in tags by making pre-defined smartphone gestures (*i.e.*, clockwise and counter-clockwise rotation of smartphone) near the interested tags.

Fig. 2 illustrates the system overview and dataflow in the system. A client makes a smartphone gesture to specify the intention to fetch information about an interested tag. The server collects tag data from RFID readers and identifies the interested tag among many coexisting tags in the environment. The server also records the starting and finishing timestamps of the smartphone gesture. Along with the coarse-grained timing information, the server also examines the fine-grained patterns in RFID measurement data caused by smartphone gesture. Meanwhile, a mobile application running in client's smartphone records the motion sensor data and identifies the smartphone gesture.

The key objective is to pair an interested tag T_i ($1 \leq i \leq N$) with its corresponding client C_j ($1 \leq j \leq M$) based on RFID and sensor measurements. The smartphone gesture generates two different data streams: 1) backscattered signal data in RFID system, and 2) motion sensor data in smartphone, respectively. The synchronicity of the same event (*i.e.*, smartphone gesture) provides an opportunity to correctly pair the interested tag with its corresponding smartphone.

III. Modelling Reflector Polarization

Referring to Fig. 3, we illustrate the signal propagation and polarization of a rotating smartphone. The RFID system uses a circularly-polarized antenna, which transmits a combination of vertical waves v and horizontal waves h with the phase difference of $\pi/2$. We use ρ_T to denote the tag polarized direction, and ρ_R to denote the long-axis direction of the reflector (*i.e.*, smartphone). α, β, and γ represent different angles between the polarized directions.

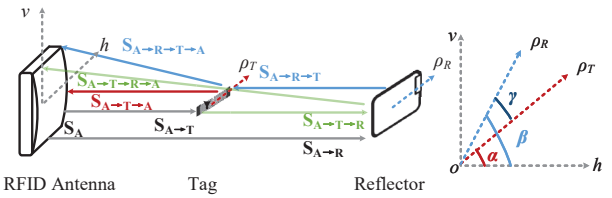


Fig. 3: Reflector polarization model and angle relationship between tag, reflector and RFID antenna.

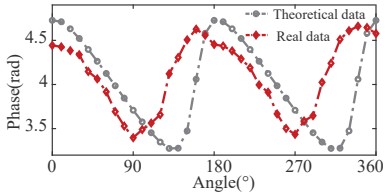


Fig. 4: The comparison between the real phases and the theoretical phases.

Suppose the reader transmits $S_A(t)$:

$$S_A(t) = h \cdot \cos(kt - \phi_A) + v \cdot \sin(kt - \phi_A) \tag{1}$$

where ϕ_A is the constant phase offset induced by the transmitter circuit.

A. Antenna-Tag-Antenna

Due to the tag polarization, the signal emitted by the reader and arrived at the tag $S_{A\to T}(t)$ will be projected to the direction of the tag polarization ρ_T. Thus, we have:

$$\begin{cases} S_{A\to T}(t) = \rho_T \cdot S_A(t - t_{A\to T}) \\ \quad = (\rho_T \cdot h)\cos(kt - \phi_{AT} - \phi_A - \phi_T) \\ \quad + (\rho_T \cdot v)\sin(kt - \phi_{AT} - \phi_A - \phi_T) \\ \quad = \cos(\alpha)\cos(kt - \phi_{AT} - \phi_A - \phi_T) \\ \quad + \sin(\alpha)\sin(kt - \phi_{AT} - \phi_A - \phi_T) \\ \phi_{AT} = 2\pi d_{A\to T}/\lambda \mod 2\pi \end{cases} \tag{2}$$

where $t_{A\to T}$ represents the propagation time from the reader antenna to the tag, ϕ_{AT} represents the phase change corresponding to the signal distance change $d_{A\to T}$, and ϕ_T denotes the phase shift caused by the tag's hardware.

Similarly, the backscattered signal of tag to reader $S_{A\to T\to A}(t)$ projects to both the reader polarized directions h and v. Therefore, we will receive two sub-signals $S^h_{A\to T\to A}(t)$ and $S^v_{A\to T\to A}(t)$ corresponding to the antenna polarized direction h and v, respectively. Thus, we have:

$$\begin{cases} S^h_{A\to T\to A}(t) = \cos(\alpha) S_{A\to T}(t - t_{T\to A}) \\ S^v_{A\to T\to A}(t) = \sin(\alpha) S_{A\to T}(t - t_{T\to A}) \end{cases} \tag{3}$$

The backscattered signal of tag $S_{A\to T\to A}(t)$ is the combination of $S^h_{A\to T\to A}(t)$ and $S^v_{A\to T\to A}(t)$ as follows:

$$\begin{cases} S_{A\to T\to A}(t) = S^h_{A\to T\to A}(t) + S^v_{A\to T\to A}(t - t_{\pi/2}) \\ \quad = \cos(2\alpha)\cos(kt - 2\phi_{AT} - \phi') \\ \quad + \sin(2\alpha)\sin(kt - 2\phi_{AT} - \phi') \\ \phi_{AT} = 2\pi d_{A\to T}/\lambda \mod 2\pi \\ \phi' = \phi_A + \phi_T + \phi'_A \end{cases} \tag{4}$$

where ϕ'_A is the phase offset induced by the receiver circuit of the reader antenna. ϕ' is a constant value related to hardware of tag and reader. As a result, we can see that the backscattered signal of tag $S_{A\to T\to A}$ is influenced by both the distance $d_{A\to T}$ and the angle between the tag and antenna α.

Previous works [5, 6] have studied the influence of the tag's orientation on phase measurements (*i.e.*, antenna-tag-antenna). However, the previous models do not consider the reflector polarization and its impact on backscattered signal.

B. Modelling Reflector Polarization

To further characterize the backscattered signal in our scenario, we consider a scenario with a reflector (*i.e.*, smartphone). The signal emitted by the reader and arriving at the reflector $S_{A\to R}(t)$ is:

$$\begin{cases} S_{A\to R}(t) = \rho_R \cdot S_A(t - t_{A\to R}) \\ \quad = \cos(\beta)\cos(kt - \phi_{AR} - \phi_A - \phi_R) \\ \quad + \sin(\beta)\sin(kt - \phi_{AR} - \phi_A - \phi_R) \\ \phi_{AR} = 2\pi d_{A\to R}/\lambda \mod 2\pi \end{cases} \tag{5}$$

where ϕ_R is the phase offset caused by the reflector.

Then $S_{A\to R}(t)$ will be reflected to the tag and the signal $S_{A\to R\to T}(t)$ can be expressed as:

$$S_{A\to R\to T}(t) = \cos(\gamma) S_{A\to R}(t - t_{R\to T}) \tag{6}$$

$S_{A\to R\to T}(t)$ will arrive at the reader antenna and project on two antenna's polarization direction $S^h_{A\to R\to T\to A}(t)$ and $S^v_{A\to R\to T\to A}(t)$ as follows:

$$\begin{cases} S^h_{A\to R\to T\to A}(t) = \cos(\alpha) S_{A\to R\to T}(t - t_{T\to A}) \\ S^v_{A\to R\to T\to A}(t) = \sin(\alpha) S_{A\to R\to T}(t - t_{T\to A}) \end{cases} \tag{7}$$

Thus, the final arrived signal at the reader $S_{A\to R\to T\to A}(t)$ can be formulated as follows:

$$\begin{cases} S_{A\to R\to T\to A}(t) = S^h_{A\to R\to T\to A}(t) + S^v_{A\to R\to T\to A}(t - t_{\pi/2}) \\ \quad = \cos(\alpha + \beta)\cos(\gamma)\cos(kt - \phi_{ARTA} - \phi'') \\ \quad + \sin(\alpha + \beta)\cos(\gamma)\sin(kt - \phi_{ARTA} - \phi'') \\ \phi_{ARTA} = 2\pi d_{A\to R\to T\to A}/\lambda \mod 2\pi \\ \phi'' = \phi_A + \phi_R + \phi_T + \phi'_A \end{cases} \tag{8}$$

From Eq.(8), we observe that the backscattered signal $S_{A\to R\to T\to A}$ is a function of the distance and the relative angles among reader, tag and reflector.

Similarly, the received signal propagated along another path $S_{A\to T\to R\to A}$ can be modelled. Note that $S_{A\to R\to T\to A}$ and $S_{A\to T\to R\to A}$ are reciprocal with the same propagation distance and the same polarization directions.

Finally, the received signal of antenna $R(t)$ can be modelled:

$$\begin{cases} R(t) = S_{A\to T\to A}(t) + S_{A\to R\to T\to A}(t) + S_{A\to T\to R\to A}(t) \\ \quad = \cos(kt - 2\phi_{AT} - \phi' - \mathbf{2\alpha}) \\ \quad + 2\cos(\boldsymbol{\gamma})cos(kt - \phi_{ARTA} - \phi'' - \boldsymbol{\alpha} - \boldsymbol{\beta}) \\ \phi_{AT} = 2\pi d_{A\to T}/\lambda \mod 2\pi \\ \phi_{ARTA} = 2\pi d_{A\to R\to T\to A}/\lambda \mod 2\pi \\ \phi' = \phi_A + \phi_T + \phi'_A \\ \phi'' = \phi_A + \phi_R + \phi_T + \phi'_A \\ \gamma = |\beta - \alpha| \end{cases} \tag{9}$$

Key observation: *The distance and the polarization directions of tag, reflector, and antenna jointly affect the received backscattered signal.*

We conduct an experiment to validate our proposed reflector polarization model. In the experiment, we ensure that both tag and reader antenna are fixed and only rotate the reflector (*i.e.*,

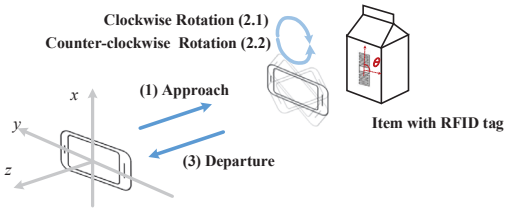


Fig. 5: Illustration of the pre-defined smartphone gesture.

change of β) for one circle. Specifically, we use an iPhone 7 ($67.1mm \times 138.3mm$) as a reflector to rotate 360 degrees counter-clockwise at $5cm$ in front of the tag. The distance between the tag and reader's antenna is $15cm$ and the angle between them is 0 (*i.e.*, $\alpha = 0$). The result is shown in Fig. 4. We observe that the phase changes with the rotation of the reflector and the changes of the measured phases are consistent with the theoretical phases. Note that the overall deviations of the phase values are introduced by the unknown parameters ϕ' and ϕ'' in Eq.(9). The experiment result demonstrates the validity of our reflector polarization model, which can be applied when capturing and differentiating a pre-defined gesture from other movements nearby.

IV. System Design

Our system consists of three key functional components: Component-1) RFID based smartphone gesture detection in server; Component-2) motion sensor based smartphone gesture detection in smartphone; and Component-3) synchronicity based matching and pairing for interested tags and their corresponding smartphones.

A. RFID based Smartphone Gesture Detection

Based on our reflector polarization model, we design a simple yet effective pre-defined smartphone gesture to specify user's interest in a tag, as illustrated in Fig. 5. The user first holds the smartphone horizontally then approaches the interested tag. Next, the user rotates the smartphone clockwise rotation followed by a symmetric counter-clockwise rotation and finally departs from the tag. Note that the pre-defined gesture does not require strict rotation angle from users.

To visualize the changes in RFID data as well as the sensor data caused by the gesture, we ask a volunteer to perform a smartphone gesture and measure both RFID data and motion sensor data in Fig. 6. We observe that the phase measurements remain flat before the smartphone gesture and start to fluctuate during the interaction. The phase changes caused by the interaction are divided into three periods: approach, rotation and departure. On the other hand, when approaching and leaving, acceleration readings in Y-axis are very small, since Y-axis is mostly perpendicular to gravity. As a user rotates the phone, the acceleration readings clearly exhibit two increasing-and-decreasing patterns. In the following, we first focus on the RFID data and analyze the phase changes.

1) ***Approach and Departure Patterns****:* As shown in Fig. 6, when the phone is far away from the tag, the phase values remain stable. As the distance does not change during this period, the phase readings remain almost constant subject to small noise. Once the phone starts to approach or depart from the tag, the reflected signal from the smartphone will affect

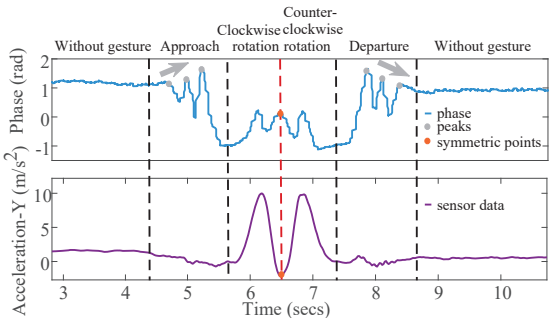


Fig. 6: Phase measurements (upper panel) and sensor data (lower panel) during the interaction.

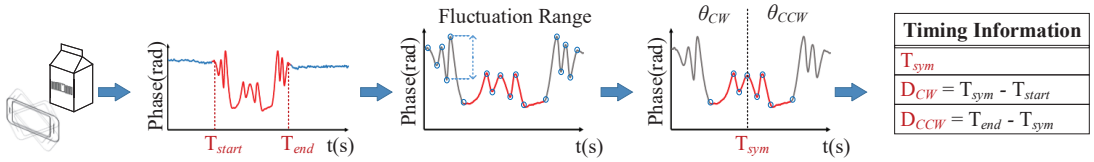
the phase measurements. As a result, the phase measurements of the interested tag will fluctuate with the distance change between the tag and the phone.

More importantly, as the phone approaches, the backscattered signal exhibits the specific approach pattern and its fluctuation range (*i.e.*, the difference between the local maximum and the local minimum of phase readings) is becoming larger because the reflected signal strength from the smartphone increases. In contrast, the fluctuation range will decrease when the phone departs.

Based on this observation, we measure the standard deviation of phase readings to detect the start and the end of a gesture. In particular, we apply a moving window to scan the phase measurements and continuously calculate the standard deviation of the phase measurement in the window. The standard deviation will remain small without gestures. When the standard deviations of three consecutive windows exceed a threshold, we consider that one gesture starts to affect the tag. If the standard deviations of three consecutive windows are below the threshold and the phase readings return to the original phase readings measured before the gesture, we consider the gesture to be finished. We record the starting point timestamp $T_{start}^{\texttt{RFID}}$ and finishing point timestamp $T_{end}^{\texttt{RFID}}$ as shown in Fig. 7(a).

However, we note that dynamics in the environment are likely to cause various changes in the tag phase readings. In order to accurately detect approach and departure patterns, we first find the local maximums and local minimums of phase readings, then measure the *differences between two adjacent local maximum and local minimum* defined as *fluctuation range*. If there are two or more consecutive fluctuations and the fluctuation range exhibits an increasing trend (as illustrated in Fig.7(b)), we consider that the phone is approaching. In contrast, the continuous decreases in the fluctuation range indicate that the smartphone is departing from the tag. In practice, some movements may cause similar phase changing patterns as in approach and departure events. In the following, we design a unique smartphone gesture to facilitate the detection and improve the detection robustness.

2) ***Rotation Pattern****:* To improve the detection robustness against the dynamics and background noise in the environment, we define a smartphone gesture (clockwise and counter-clockwise rotation of smartphone). As analyzed in Section III, smartphone polarization can affect received backscattered signal. In Fig. 6, we have an interesting observation.



(a) Cut the phase readings associated with the gesture

(b) Detect the approach pattern and departure pattern

(c) Find the symmetric point using DTW algorithm DTW(θ_{CW} ,*flip*(θ_{CCW}))

(d) Extract three key timing information on the tag signal

Fig. 7: Timing information extraction on tag signal.

Observation: *Phase changes caused by the defined smartphone gesture are generally symmetric.*

We observe that the phase reading shows an 'M' or 'W' shape because the smartphone gesture is symmetrical. As a result, RFID readers can leverage such prior knowledge and detect a pre-defined smartphone gesture. Note that such a symmetric pattern in our pre-defined gesture can be used to disambiguate human activities (*i.e.*, human movement), which do not generate symmetric patterns.

Although the rotation angles of the clockwise and counter-clockwise are generally symmetrical, the rotation time and speed can be slightly different, resulting in misaligned phase waveforms. To accurately detect the symmetric point and use that as the timing information, we adopt the Dynamic Time Warping (DTW) algorithm to match the slightly misaligned phase waveforms measured in clockwise and counter-clockwise rotations. We first select the local maximums and local minimums on phase readings of rotation as a candidate set of symmetric points $\{SP_1, SP_2, \cdots, SP_k, \cdots, SP_K\}$. Next, we divide the tag signals into two parts: clockwise signal $\theta_{CW}(k)$ before the symmetric point SP_k and counter-clockwise signal $\theta_{CCW}(k)$ after the symmetric point as shown in Fig.7(c). Then, we use DTW algorithm to calculate the distance between the $\theta_{CW}(k)$ and the flipped counter-clockwise signal, $flip(\theta_{CCW}(k))$:

$$Distance(k) = DTW(\theta_{CW}(k), flip(\theta_{CCW}(k))), k \in [1, K] \quad (10)$$

The minimum distance indicates the highest similarity of $\theta_{CW}(k)$ and $flip(\theta_{CCW}(k))$. We notice that the time difference between clockwise and counter-clockwise rotation of smartphone performed by users are generally less than 1 second. Therefore, the DTW algorithm in our experiment tolerates clockwise and counter-clockwise rotation waveforms with a maximum misalignment of 1 second. As a result, we can find the true symmetric point and filter out noise in the environment (*e.g.*, user movement, random signal fluctuation).

3) ***Timing Information Extraction on Tag Signal****:* Based on the observations, we can extract three key timing information on the backscattered signal of RFID tag T_i $(1 \leq i \leq N)$ as shown in Fig. 7(d):

- Absolute timestamp of symmetric point $T_{sym}^{\texttt{RFID}}(T_i)$.
- Clockwise rotation duration $D_{CW}^{\texttt{RFID}}(T_i)$: the difference between symmetric point timestamp and starting point timestamp $T_{start}^{\texttt{RFID}}(T_i)$, *i.e.*, $D_{CW}^{\texttt{RFID}}(T_i) = T_{sym}^{\texttt{RFID}}(T_i) - T_{start}^{\texttt{RFID}}(T_i)$.
- Counter-clockwise rotation duration $D_{CCW}^{\texttt{RFID}}(T_i)$: the difference between symmetric point timestamp and finishing point timestamp $T_{end}^{\texttt{RFID}}(T_i)$, *i.e.*, $D_{CCW}^{\texttt{RFID}}(T_i) = T_{end}^{\texttt{RFID}}(T_i) - T_{sym}^{\texttt{RFID}}(T_i)$.

B. Motion Sensor based Smartphone Gesture Detection

After detecting the gesture from the RFID data, we need to perform gesture detection on user's smartphone and pair the smartphone to the corresponding tag.

1) ***Smartphone Gesture Detection****:* In the above discussion, we only focus on the acceleration readings in the Y-axis for concise presentation. In practice, X-axis and Z-axis acceleration readings can complement and enhance the gesture detection as shown in Fig. 8.

Since the phone is held horizontally in the initial state, we observe that the acceleration readings in Y-axis and Z-axis are close to zero, and the acceleration readings in X-axis are close to the gravitational acceleration $9.8m^2/s$. Therefore, we can determine the initial state of our defined gesture by measuring the initial patter of acceleration readings. Next, we need to detect the approach pattern and departure pattern. We find when the phone starts moving toward the tag along the Z-axis, the Z-axis acceleration readings will increase from 0. To detect the starting time and finishing time of smartphone gesture, we calculate the standard deviations of Z-axis readings in each moving window. If the standard deviations exceed a threshold for three consecutive windows, we consider that the smartphone is approaching the tag and departing when the standard deviations drop below the threshold for three consecutive windows. When a user finishes this interaction gesture, the acceleration readings in all three axes will return to the initial state. Meanwhile, we record the starting point timestamp $T_{start}^{\texttt{Phone}}$ and finishing point timestamp $T_{end}^{\texttt{Phone}}$.

Then, we identify smartphone rotation by measuring the acceleration readings in Y-axis. In the initial state, the acceleration readings in Y-axis are expected to be small and stable. In contrast, once the phone starts rotation, its readings change from 0 to $9.8m^2/s$. As the user rotates clockwise and then counter-clockwise, the acceleration readings in Y-axis exhibit two peaks. Hence, we search for local maximum values and local minimum values and extract the key timing information. Our observation is that the smartphone gesture is symmetric, and the symmetric point is the local minimum (corresponding to the horizontal pose after clockwise rotation) between two local maximums (corresponding to the two vertical poses during the clock-wise and counter clock-wise rotations, respectively). As a result, we can identify the symmetric point P_{sym}: the local minimum between two peaks and its Y-axis acceleration reading near zero. In this way, we can obtain the timestamp of symmetric point $T_{sym}^{\texttt{Phone}}$.

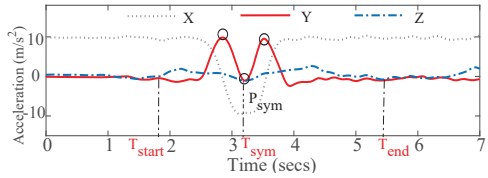


Fig. 8: The changes of acceleration readings in the x, y, and z axes during the interaction.

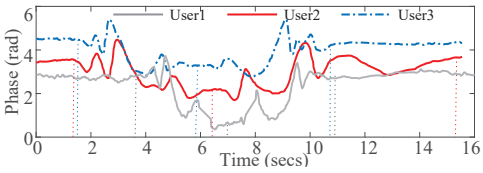


Fig. 9: Phase changes caused by different users.

*2) **Timing Information Extraction on Sensor Data**:* Based on the above observation, Component-2 detects the pre-defined smartphone gesture and extracts the timing information for each client smartphone C_j $(1 \leq j \leq M)$ as follows.

- Absolute timestamp of symmetric point $T_{sym}^{\texttt{Phone}}(C_j)$.
- Clockwise rotation duration $D_{CW}^{\texttt{Phone}}(C_j)$: the difference between symmetric point timestamp and starting point timestamp, *i.e.*, $D_{CW}^{\texttt{Phone}}(C_j) = T_{sym}^{\texttt{Phone}}(C_j) - T_{start}^{\texttt{Phone}}(C_j)$.
- Counter-clockwise rotation duration $D_{CCW}^{\texttt{Phone}}$: the difference between symmetric point timestamp and finishing point timestamp $T_{end}^{\texttt{Phone}}(C_j)$, *i.e.*, $D_{CCW}^{\texttt{Phone}}(C_j) = T_{end}^{\texttt{Phone}}(C_j) - T_{sym}^{\texttt{Phone}}(C_j)$.

C. Synchronicity based Matching and Pairing

As the backscattered signal and the sensor data are simultaneously affected by the same gesture, we leverage the synchronicity of the signals to pair the interacted tag and the corresponding smartphone. Instead of mapping all the data points in two data streams, we only match backscattered signal and the sensor data using the extracted key time information to reduce computation time and network traffic.

We design a sequence matching algorithm based on the following three key observations: (1) The rotation gesture is generally performed within a certain period P (*e.g.*, 5s); (2) Different users may generate different key timing information; and (3) The key timing information of backscattered signal and sensor data caused by the same gesture are synchronized. Based on these observations, we match tag T_i $(1 \leq i \leq N)$ with client C_j $(1 \leq j \leq M)$ (denoted as $T_i \mapsto C_j$), if all following conditions are satisfied:

- **C1**: $D_{CW}^{\texttt{RFID}}(T_i) + D_{CCW}^{\texttt{RFID}}(T_i) \leq P$,
- **C2**: $D_{CW}^{\texttt{Phone}}(C_j) + D_{CCW}^{\texttt{Phone}}(C_j) \leq P$
- **C3**: $T_{sym}^{\texttt{RFID}}(T_i) = T_{sym}^{\texttt{Phone}}(C_j)$
- **C4**: $D_{CW}^{\texttt{RFID}}(T_i) = D_{CW}^{\texttt{Phone}}(C_j)$
- **C5**: $D_{CCW}^{\texttt{RFID}}(T_i) = D_{CCW}^{\texttt{Phone}}(C_j)$

However, such strict timing requirements may not be satisfied in practice. For example, due to the ALOHA protocol of RFID system as well as the different sampling rates of the backscattered signal and the sensor data, the RFID signal and sensor readings may not be exactly matched. To address this practical issue, we relax the conditions (**C3** - **C5**) by tolerating a small mismatch δ in the time domain. For example, we relax **C3** as follows:

- **Relaxed C3**: $|T_{sym}^{\texttt{RFID}}(T_i) - T_{sym}^{\texttt{Phone}}(C_j)| \leq \delta$

We note that a smaller δ indicates a tighter timing requirement, which can reduce the possibility of incorrectly matching two streams generated by different gestures but meanwhile increase the chance of missing two streams originated by the same gesture. We empirically tune δ and set δ to $400ms$.

Why do we extract three key timing information for matching? Fig. 9 plots the phase readings when three volunteers perform smartphone gestures in front of their interested tags concurrently. We notice that the timestamps of three symmetric points can be very close in time, making it hard to differentiate. Fortunately, as users tend to perform gestures differently (*e.g.*, different speed, different duration), the clockwise and the counter-clockwise duration can be different in practice. For example, the gesture duration of user 1 is shorter than that of user 2. Therefore, we extract three key timing information to differentiate users and improve robustness.

As the network traffic involved in transmitting the timing information as well as tag ID is small, the server can encapsulate the timing information of RFID data and its tag ID and broadcast the packet to all clients. Receiving a broadcast packet, clients test the above matching conditions if the client's smartphone has detected a smartphone gesture recently. If no smartphone gesture has been detected, a client can simply drop the broadcast packet. If all the above conditions are satisfied, the client can read the tag ID from the broadcast packet, and fetch more information about the tag from the server using the tag ID as an index. The computation overhead involved in testing the above conditions is very low and can be afforded by smartphones.

V. Implementation and Evaluation

We implement a prototype of *ShakeReader* using the COTS RFID system and conduct extensive experiments to evaluate its performance in this section.

Hardware: As shown in Fig. 10, our prototype system consists of an Impinj R420 RFID reader, which is connected to a circularly-polarized directional antenna. We adopt Network Time Protocol (NTP) to synchronize the reader's time [7] with smartphones. Three different types of RFID tags (*i.e.*, Impinj E53, Alien ALN-9640, and Impinj H47) are tested in our experiments. A PC with Intel Core i7-10510U 2.30GHz CPU and 16GB RAM is used as the server to control the reader and process the received RFID data. We test three popular smartphones including an iPhone 7 with aluminum back cover, a HUAWEI P20 Pro with glass back cover, and an iPhone 7 with a common soft rubber case.

Data collection: Our server adopts the LLRP (Low-Level Reader Protocol) to communicate with the RFID reader and the software is implemented using C#. We use MATLAB Mobile Apps [8] to collect sensor data and the data processing algorithm is implemented using MATLAB.

Experiment setting: We conduct experiments in an office environment with a size of $4m \times 10m$ and a bookshelf scenario in another office to evaluate the performance of *ShakeReader*. By default, the reader uses its maximum transmit power at $32.5dBm$ and works on $920.625MHz$. In our experiment, the

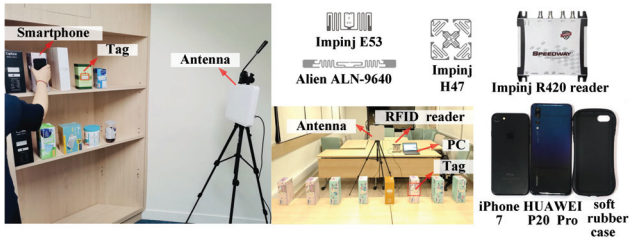


Fig. 10: Experimental environment and devices.

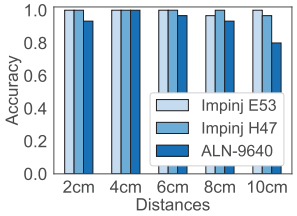


Fig. 11: Impact of smartphone-to-tag distance.

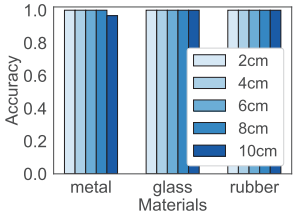


Fig. 12: Impact of different reflective materials.

read rate is about 260 tags/s. On the client side, we adopt the sampling rate of $100Hz$ to collect data from the smartphone's accelerometer.

Metrics: For each component, we mainly focus on detection accuracy. We adopt three metrics, *i.e.*, Accuracy, False Accept Rate (FAR) and False Reject Rate (FRR) to evaluate the overall performance of the system. Accuracy is defined as the rate that one tag is correctly matched to its corresponding client. FAR is the rate that *ShakeReader* incorrectly accepts the uninterested tag information and FRR is the rate that *ShakeReader* incorrectly rejects the interacted tag information.

A. RFID based Smartphone Gesture Detection

Component-1 detects smartphone gestures based on the phase measurements of RFID tags. In the following, we consider various factors that may affect the detection accuracy.

Impact of smartphone-to-tag distance. To evaluate the effective interaction range of *ShakeReader*, we vary the smartphone-to-tag distance from $2cm$ to $10cm$. A volunteer is asked to perform the smartphone gesture 30 times at each interaction distance.

Fig.11 shows the detection accuracy at different distances. The smartphone gestures can be detected with an average accuracy over 95%. In the figure, we see that within interaction distance of $10cm$, the gesture detection accuracy for Impinj E53 and Impinj H47 tags keeps stable and exceeds 95% at all tag-smartphone distances. The interaction with ALN-9640 tag exhibits a lower detection accuracy of around 90% and decreases to 80% at the distance of $10cm$. This is because the ALN-9640 tag is not fully covered by the smartphone, resulting in an asymmetric pattern during smartphone rotation. Therefore, we choose the Impinj E53 as our default RFID tag in the next experiments.

We note that a longer distance between the tag and the smartphone results in weaker reflected signals. As such, the smartphone may not cause sufficient impact on the backscattered signal, which degrades the detection accuracy. Therefore, in order to 'read' a tag, a user needs to make a smartphone gesture within $10cm$. More importantly, the result implies that a smartphone gesture will not cause ambiguity in identifying the interacted tags as long as the interacted ones are separated from their near tags by $10cm$. As such, we do not intend to increase smartphone-to-tag distance in the current implementation. Possible approaches to increase the distance is to increase the transmission power of readers, and decrease the distance between antenna and smartphone, thereby increasing reflected signals from smartphones.

Impact of smartphone materials. Different smartphones may have different back cover materials. The reflected signal is impacted by the reflection coefficient of the material. A higher reflection coefficient of the reflector can reflect more radio waves. To test the impact of smartphone materials, we conduct an experiment using three smartphones with different materials: an iPhone 7 with metal back cover, a HUAWEI P20 Pro with a glass back cover and an iPhone 7 with a soft rubber case. A volunteer performs the pre-defined smartphone gesture at $10cm$ interaction distance. Each smartphone is used to interact with three different tags 30 times.

Fig.12 shows gesture detection accuracy when using smartphones with different materials to interact with the tag. We observe that almost all the gestures performed using smartphones with different back cover materials can be detected. We note that along with the external back cover, the internal circuit board also reflects continuous waves to the tags. As such, smartphones with glass and rubber back cover can also be used to interact with tags.

Impact of tag-to-reader distance. In the above experiments, we fixed the distance between the tag and the reader's antenna at $1m$. To evaluate the impact of distance between the tag and the reader's antenna, we vary the tag-reader distance ranging from $1m$ to $2.5m$. A volunteer is asked to perform the smartphone gesture 100 times in front of the tag while the tag-reader distance is varied. In the experiment, we only use the Impinj E53 tag and the interaction distance between the tag and the smartphone is within $10cm$.

Fig.13 illustrates the gesture detection accuracy at different tag-reader distances. When the tag-reader distance is $1m$, the RFID system can reliably measure the changes in backscattered signal and our algorithm can correctly detect almost all gestures. As the tag-reader distance increase to $1.5m$ and $2m$, the backscattered signal becomes weak, resulting in miss detection of some gestures. In practice, one COTS reader can be connected with multiple antennas. To achieve high detection accuracy, we can deploy multiple antennas to reduce tag-to-reader distance.

Impact of tag-to-tag distance. When a user is interacting with the tag of interest, the adjacent tags may be affected as well, leading to detection ambiguity. To evaluate the impact of tag-to-tag distance, we fix the tag-reader distance to $1m$ and the interaction distance within $10cm$ while varying the tag-to-tag distances from $10cm$ to $30cm$ in Fig.14. A volunteer performs a gesture in front of one tag, while we move away the other tag from the interacted tag. We observe that when the tag-to-tag distance reaches $15cm$, the adjacent tag incurs minute influence on the detection. As such, when the tag-to-tag distance exceeds $15cm$, our system can detect almost all gestures. In practice, considering the size of the items, the spacing distance between two adjacent items is much smaller

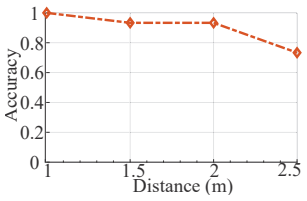


Fig. 13: Impact of tag-to-reader distances.

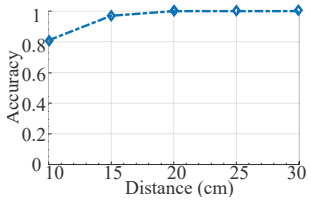


Fig. 14: Impact of tag-to-tag distances.

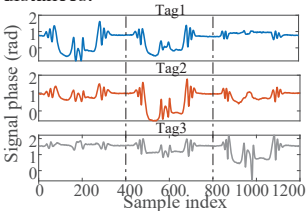


Fig. 15: The impact of adjacent tags on phase measurements.

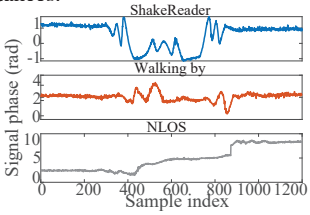


Fig. 16: The impact of human movement.

than tag-to-tag distance.

To visualize the effects of the neighboring tags, we place three tags in a straight line separated by around $10cm$ and ask a volunteer to interact with the three tags sequentially. Fig.15 plots the phase measurements of all the three tags. We can observe that the signal phase of the adjacent tags indeed exhibits a similar fluctuation pattern. However, the signal phase of the interacted tag fluctuates more drastically than those of neighboring tags. For instance, when Tag1 is interacted, we see that the backscattered signal of Tag1 is influenced most dramatically. Based on this observation, we can resolve the ambiguity by examining the fluctuation magnitude in the phase measurements. When the tag-to-tag distance is less than $10cm$, due to the coupling effect of the tags and the ambiguity caused by the neighboring tag, the detection accuracy would decrease. To mitigate the impact of very close nearby tags, a user can manually move the interested tag away from nearby tags before performing smartphone gestures.

Impact of tag orientation. In real applications, an interested tag can be attached to an item in various orientations. To investigate the impact of tag orientation relative to the smartphone, we vary the tag's orientation θ from $0°$ to $180°$ as shown in Fig.5. We perform the pre-defined gesture 30 times at each tag's orientation and measure recognition accuracy. In the experiment, the smartphone rotates in the XY plane, while the tag's initial orientation attached to the item is varied as illustrated in Fig.5. According to our experiments, the tag orientation does not affect the gesture recognition accuracy. That is because we leverage the symmetry of our pre-defined gesture to pair the interested tag with its corresponding smartphone, which is irrelevant to the tag's initial orientation. We note that if the smartphone rotates in the XZ plane, since the reflection from the smartphone to the tag is weak due to small reflection surface, it becomes hard to notice substantial phase changes during smartphone gesture. In this case, we need to manually adjust the RFID tag to ensure that the tag plane is parallel to the smartphone.

Impact of human movement. Human movement near a tag may cause the change in backscattered signal. We consider the human movement near a tag as well as the blockage of the line-of-sight path between a tag and reader's antenna by a user. In the first scenario, we ask a volunteer to walk near a tag and stay in front of the tag for a while. In the second scenario, we ask a volunteer to stand between the tag and the reader to block the line-of-sight path. Fig.16 plots the phase measurements in the two scenarios. Compared with the pre-defined gesture of *ShakeReader*, the phase measurements in the two scenarios exhibit different patterns. Even if Component-1 accidentally triggers a false alarm and incorrectly broadcasts a potential smartphone gesture to clients, the clients can filter out the packets using Component-3 (*i.e.*, synchronicity based matching and pairing).

B. Overall System Performance

System accuracy. We conduct the experiments in an office environment as shown in Fig.10. We ask three volunteers (2 males and 1 female) to randomly interact with nine tags attached to paper boxes separated by $15cm$ concurrently. Each volunteer interacts with one of the tags within $10cm$ interaction range. We record the ground truth of interactions and test whether our system can accurately match the interacted tags to their corresponding smartphones. We note that volunteers do not interact with the same tag simultaneously, but they can interact with different tags at the same time.

In this dynamic environment with multiple people, we collect 810 RFID tag records and 270 smartphone gesture records in total. As shown in Fig.17, *ShakeReader* achieves the matching accuracy of $> 94.6\%$. Even in the case of multi-user interaction, the FAR and FRR of each user are less than 6.1% and 3.3% respectively. The results indicate that *ShakeReader* can accurately match the interacted tags to their corresponding smartphones. In our applications, we care more about FRR than FAR, because false rejects mean that the user performed the pre-defined gesture but did not receive any item information. In contrast, false accepts indicate that it is possible for a user to receive broadcast information of an uninterested tag. When two users interact with two different tags at the same time and their phase and accelerometer waveforms exhibit similar patterns, *ShakeReader* may not be able to differentiate the two gestures and associate the tags to their corresponding tags. To address this problem, we can examine tag location and phone location to further improve matching accuracy in future work.

System performance in the shelf scenario. To simulate real application scenarios, we divide 10 items attached with RFID tags into two columns and put them on the shelf to conduct the experiment as shown in Fig.10. The shape of selected items is various and the distance of the tag on the items is around $15cm$. A volunteer randomly chooses an item and performs the pre-defined gesture in front of the interested item. In this process, we read phase samples when performing 100 smartphone gestures in total. As shown in Fig.17, the accuracy of *ShakeReader* reaches 96.9% and FRR is 2%. However, the dense placement of items makes it easier for users to receive the information of adjacent tags and the FAR is 3.2%. Yet, we note that the interested tags can still be 'read' with a very high accuracy.

System latency. We measure the execution time of each component as shown in Fig.18. The average values are around

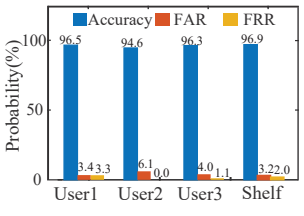


Fig. 17: Overall performance.

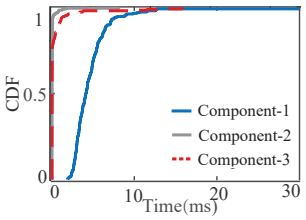
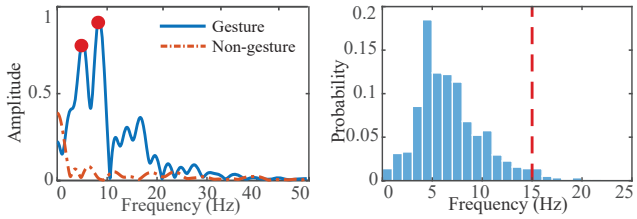


Fig. 18: Execution time.



(a) After FFT (b) Frequency Distribution

Fig. 19: Gesture frequency component Analysis.

$4.83ms$, $0.13ms$, and $0.48ms$ for Component-1, Component-2 and Component-3, respectively. We find that the DTW algorithm in Component-1 is most time-consuming. To reduce the time complexity, instead of scanning all sampling points of tag signals, we select the segments between the local maximums and local minimums to execute the DTW algorithm to find the symmetric point. In addition, our system matches interacted tags and corresponding users using timing information rather than raw data, which further reduces computational complexity. Overall, the average processing time of *ShakeReader* is $7.6ms$ for each smartphone gesture matching, which is acceptable for most interaction applications.

System capacity. A low read rate of reader will result in a low resolution of measured timing information extracted from RFID data, which may affect the matching accuracy. To determine the maximum capacity of *ShakeReader*, we first analyze the frequency component of the pre-defined interaction gestures with different users. We use the Fast Fourier Transform (FFT) to measure the frequency domain information of RFID data when users perform gestures as shown in Fig.19a. We can see that the main frequency components corresponding to the gestures are concentrated below $20Hz$. Thus, we plot the top-2 frequency distribution from 370 RFID tag records of four users in Fig.19b. We can see that 96.8% of gesture frequencies is less than $15Hz$. Therefore, according to the Nyquist theorem, the read rate of the RFID reader needs to be higher than 30 readings/s for a single tag. One potential method to improve the reading rate is to utilize the SELECT command to focus on the potential interacted tags.

VI. Related Work

Most commercial smartphones available on the market cannot directly read UHF RFID tags. To read UHF RFID tags, one may extend smartphone by adding external UHF modules [9], which incurs extra cost and power consumption to smartphones. Recent research aims to allow smartphone users to read UHF RFID tags based on Cross-Frequency Communication. TiFi [4] first reads RFID tags using RFID readers and broadcasts the tag IDs as Wi-Fi beacons. However, the signal strength based association is subject to background noise and interference. In addition, it is very challenging to correctly identify the interested tag among all the tag IDs. Our work uses a pre-defined smartphone gesture and leverages the synchronicity of RFID and sensor data to accurately match an interacted tag to the corresponding smartphone.

Human-object interaction based on passive RFID has attracted much attention in recent years. COTS RFID systems have been used to achieve high accuracy in tracking RFID-labelled objects [10–18] and enable innovative RFID sensing applications [19–24]. RF-IDraw [25] tracks the trajectory of an RFID tag by measuring the angle of arrival using customized antenna arrays. Tagyro [6] attaches RFID tags to an object and measures the object orientation by leveraging the polarity of tag antenna. PolarDraw [26] infers the orientation and position of RFID-labelled items based on tag polarization. TACT [27] builds a contact-free reflection model for activity recognition which does not need to attach tags to target users. RFIPad [28] enables in-air handwriting using an array of RFID tags. RF-finger [29] tracks finger writings and recognizes multi-touch gestures using tag arrays deployed in the environment. Spin-Antenna [30] enhances object tracking accuracy by combing tag arrays and spinning polarized antenna, which can effectively suppress ambient signal interference and noise.

Unlike these works, *ShakeReader* does not need to attach tags to smartphones. Instead, *ShakeReader* detects the symmetric smartphone rotation by leveraging the polarization of the reflected signal and the prior knowledge of pre-defined smartphone gesture.

VII. Conclusion

In this paper, we aim to enable smartphone users to interact with UHF RFID tags using their smartphones without making any hardware extension to either deployed RFID infrastructure or smartphones. To this end, we define a smartphone gesture which can be simultaneously detected by both RFID systems and smartphones. We overcome many technical challenges involved in smartphone gesture detection especially using RFID systems. In particular, we characterize the polarization of reflected signals from smartphone and detect smartphone rotations. We leverage the synchronicity of RFID data and sensor data caused by the same smartphone gesture to match the interacted tag with the corresponding smartphone. Experimental results show that *ShakeReader* can achieve up to 94.6% matching accuracy.

Acknowledgement

This work is supported in part by the National Nature Science Foundation of China under grant 61702437, 61872285 and Hong Kong GRF under grant PolyU 152165/19E. This work is also supported in part by the major project of the National Social Science Foundation under Grant 20ZDA062, Research Institute of Cyberspace Governance in Zhejiang University, Leading Innovative and Entrepreneur Team Introduction Program of Zhejiang (Grant No. 2018R01005), Zhejiang Key R&D Plan (Grant No. 2019C03133). Yuanqing Zheng and Jinsong Han are the corresponding authors.

REFERENCES

[1] HUAYUAN. (2020) UNIQLO Global Stores Applied RFID Tags. [Online]. Available: https://www.huayuansh.com/uniqlo-global-stores-applied-rfid-tags/

[2] N. Kumar. (2018) Zara: Fast Fashion and RFID. [Online]. Available: https://nirmalyakumar.com/2018/06/02/zara-fast-fashion-and-rfid/

[3] L. Shangguan, Z. Zhou, X. Zheng, L. Yang, Y. Liu, and J. Han, "ShopMiner: Mining Customer Shopping Behavior in Physical Clothing Stores with COTS RFID Devices," in *Proceedings of ACM SenSys*, 2015.

[4] Z. An, Q. Lin, and L. Yang, "Near-field identification of uhf rfids with wifi!" in *Proceedings of ACM MobiCom*, 2018.

[5] C. Jiang, Y. He, S. Yang, J. Guo, and Y. Liu, "3D-OmniTrack: 3D tracking with COTS RFID systems," in *Proceedings of ACM IPSN*, 2019.

[6] T. Wei and X. Zhang, "Gyro in the Air: Tracking 3D Orientation of Batteryless Internet-of-things," in *Proceedings of ACM MobiCom*, 2016.

[7] M. Lenehan. (2019) Synchronize and Set the Clock on Speedway RAIN RFID Readers. [Online]. Available: https://support.impinj.com/hc/en-us/articles/202756558-Synchronize-and-Set-the-Clock-on-Speedway-RAIN-RFID-Readers/

[8] Matlab. (2020) MATLAB Mobile APP. [Online]. Available: https://www.mathworks.com/products/matlab-mobile.html

[9] Amazon. (2019) RFID ME: Mini ME UHF RFID Reader for Android Powered Devices. [Online]. Available: https://www.amazon.com/RFID-ME-Android-Powered-Devices/dp/B007KXC1NO

[10] L. M. Ni, Y. Liu, Y. C. Lau, and A. P. Patil, "LANDMARC: Indoor Location Sensing Using Active RFID," in *Proceedings of IEEE PerCom*, 2003.

[11] L. Yang, Y. Chen, X.-Y. Li, C. Xiao, M. Li, and Y. Liu, "Tagoram: Real-time Tracking of Mobile RFID Tags to High Precision Using COTS Devices," in *Proceedings of ACM MobiCom*, 2014.

[12] J. Wang and D. Katabi, "Dude, Where's My Card?: RFID Positioning That Works with Multipath and Non-line of Sight," in *Proceedings of ACM SIGCOMM*, 2013.

[13] H. Xu, D. Wang, R. Zhao, and Q. Zhang, "AdaRF: Adaptive RFID-based Indoor Localization Using Deep Learning Enhanced Holography," *Proceedings of ACM Interact. Mob. Wearable Ubiquitous Technol.*, vol. 3, no. 3, pp. 1–22, 2019.

[14] Z. Wang, M. Xu, N. Ye, R. Wang, and H. Huang, "RF-Focus: Computer Vision-assisted Region-of-interest RFID Tag Recognition and Localization in Multipath-prevalent Environments," *Proceedings of ACM Interact. Mob. Wearable Ubiquitous Technol.*, vol. 3, no. 1, pp. 1–30, 2019.

[15] X. Shi, M. Wang, G. Wang, B. Huang, H. Cai, J. Xie, and C. Qian, "TagAttention: Mobile Object Tracing without Object Appearance Information by Vision-RFID Fusion," in *Proceedings of IEEE ICNP*, 2019.

[16] Z. Liu, X. Liu, and K. Li, "Deeper Exercise Monitoring for Smart Gym using Fused RFID and CV Data," in *Proceedings of IEEE INFOCOM*, 2020.

[17] C. Duan, W. Shi, F. Dang, and X. Ding, "Enabling RFID-Based Tracking for Multi-Objects with Visual Aids: A Calibration-Free Solution," in *Proceedings of IEEE INFOCOM*, 2020.

[18] G. Wang, C. Qian, K. Cui, X. Shi, H. Ding, W. Xi, J. Zhao, and J. Han, "A Universal Method to Combat Multipaths for RFID Sensing," in *Proceedings of IEEE INFOCOM*, 2020.

[19] H. Ding, L. Shangguan, Z. Yang, J. Han, Z. Zhou, P. Yang, W. Xi, and J. Zhao, "FEMO: A Platform for Free-Weight Exercise Monitoring with RFIDs," in *Proceedings of ACM SenSys*, 2015.

[20] Y. Wang and Y. Zheng, "TagBreathe: Monitor Breathing with Commodity RFID Systems," in *Proceedings of IEEE ICDCS*, 2017.

[21] L. Yang, Q. Lin, X. Li, T. Liu, and Y. Liu, "See Through Walls with COTS RFID System!" in *Proceedings of ACM MobiCom*, 2015.

[22] J. Guo, T. Wang, Y. He, M. Jin, C. Jiang, and Y. Liu, "Twinleak: RFID-based Liquid Leakage Detection in Industrial Environments," in *Proceedings of IEEE INFOCOM*, 2019.

[23] P. Yang, Y. Feng, J. Xiong, Z. Chen, and X.-Y. Li, "RF-Ear: Contactless Multi-device Vibration Sensing and Identification Using COTS RFID," in *Proceedings of IEEE INFOCOM*, 2020.

[24] Z. Chen, P. Yang, J. Xiong, Y. Feng, and X.-Y. Li, "TagRay: Contactless Sensing and Tracking of Mobile Objects using COTS RFID Devices," in *Proceedings of IEEE INFOCOM*, 2020.

[25] J. Wang, D. Vasisht, and D. Katabi, "RF-IDraw: Virtual Touch Screen in the Air Using RF Signals," in *Proceedings of ACM SIGCOMM*, 2014.

[26] L. Shangguan and K. Jamieson, "Leveraging Electromagnetic Polarization in a Two-Antenna Whiteboard in the Air," in *Proceedings of ACM CoNEXT*, 2016.

[27] Y. Wang and Y. Zheng, "Modeling RFID Signal Reflection for Contact-free Activity Recognition," *Proceedings of ACM Interact. Mob. Wearable Ubiquitous Technol.*, vol. 2, no. 4, pp. 193:1–193:22, 2018.

[28] H. Ding, C. Qian, J. Han, G. Wang, W. Xi, K. Zhao, and J. Zhao, "RFIPad: Enabling Cost-Efficient and Device-Free In-air Handwriting Using Passive Tags," in *Proceedings of IEEE ICDCS*, 2017.

[29] C. Wang, J. Liu, Y. Chen, H. Liu, L. Xie, W. Wang, B. He, and S. Lu, "Multi-Touch in the Air: Device-Free Finger Tracking and Gesture Recognition via COTS RFID," in *Proceedings of IEEE INFOCOM*, 2018.

[30] C. Wang, L. Xie, K. Zhang, W. Wang, Y. Bu, and S. Lu, "Spin-Antenna: 3D Motion Tracking for Tag Array Labeled Objects via Spinning Antenna," in *Proceedings of IEEE INFOCOM*, 2019.