TagBooth: Deep Shopping Data Acquisition Powered by RFID Tags

Tianci Liu*, Lei Yang*, Xiang-Yang Li*†, Huaiyi Huang*, Yunhao Liu*

* School of Software and TNLIST, Tsinghua University, China,
† Department of Computer Science, Illinois Institute of Technology
{tian, young}@tagsys.org, {xiangyang.li, huanghy}@gmail.com, yunhao@greenorbs.com

Abstract—To stay competitive, plenty of data mining techniques have been introduced to help stores better understand consumers' behaviors. However, these studies are generally confined within the customer transaction data. Actually, another kind of 'deep shopping data', *e.g. which and why goods receiving much attention are not purchased*, offers much more valuable information to boost the product design. Unfortunately, these data are totally ignored in legacy systems. This paper introduces an innovative system, called *TagBooth*, to detect commodities' motion and further discover customers' behaviors, using COTS RFID devices. We first exploit the motion of tagged commodities by leveraging physical-layer information, like phase and RSS, and then design a comprehensive solution to recognize customers' actions. The system has been tested extensively in the lab environment and used for half a year in real retail store. As a result, TagBooth generally performs well to acquire deep shopping data with high accuracy.

Keywords—*RFID, TagBooth, Deep Shopping Data, Motion Detection, Action Recognition*

I. INTRODUCTION

In the era of big data, numerous detailed and accurate shopping data bring endless benefits to retailers and products suppliers. Based on reliable shopping data, retailers can conduct more accurate market research, wiser sales strategy, and provide more specific product recommendation, while suppliers and manufacturers can formulate more visionary product planning [1]–[3]. Unfortunately, unlike online stores, collecting shopping data is an arduous job for physical stores. Sales data is quite easy to get, as a key part of market research. Deep shopping data, other than just final deal lists, for example which piece of commodity customer touches and moves, is beyond the capability of most physical stores to acquire efficiently and effectively.

General automated state-of-art is based on video monitor and analysis [4], [5]. These methods deploy plentiful cameras in stores and collect shopping data by recognizing customer activity from video through video analyses and image matching technology. However, there are at least following four limitations for video based solutions. First, they demand good light and unobstructed line of sight to keep good accuracy. Second, image feature databases of customer behavior are required for recognizing customer actions, which will take a lot of labor to gather before system deployment. Third, most of these analysis are done off-line and need lots of computing resources. Even so, the accuracy of recognizing actions and target object is hardly satisfactory. And last, installing cameras may violate the customer privacy in some cases, not to say the high cost for installing enough cameras to cover a market. Some work makes use of smartphone tracking to acquire movements of shoppers, whereas they are not intelligent enough to know what actions shoppers perform. There exists other solutions such as [6] to track shoppers and hot spots in stores. However, these solutions either demand to remold stores or special design of commodities arrangement. Besides, they cannot obtain fine-grained deep shopping data, such as when a customer picks an item. Hence, a low cost, efficient and effective method to acquire fine-grained deep shopping data is in great demand by physical stores.

In this work we propose to acquire the deep shopping data by taking advantage of the Radio Frequency IDentification(RFID) tags already attached to merchandizes. The RFID market is worth $7.88 billion in 2013, and grows to $9.2 billion in 2014 [7]. A critical factor making RFID technology so fashionable is the low price of passive RFID tags. Employing RFID technology to acquire deep shopping data for physical stores is a good choice, especially some markets have already equipped goods with RFID tags, such as Walmart [8]. We first conduct an in-depth study of RF features of signals between reader and tags in a set up experimental environment similar to physical stores. Then *TagBooth* is proposed to acquire valuable deep shopping data based on RFID tags. TagBooth divides the deep shopping data acquisition into two main steps: 1) *motion detection*, detect whether commodities are in motion because of customer actions. 2) *action recognition*, identify what kind of behaviors the customer may have just performed and the targets of those actions. We employ RSS to detect which commodities are targets of actions and utilize phase to distinguish subtle customer actions. At last, we implement TagBooth with COTS readers, which is non-intrusive to customer and store structure. No radio fingerprint and location measurement work have to be done during the deployment of our TagBooth system.

In summary, our main contributions are as follows.

We propose a RFID based method to collect fine-grained deep shopping data automatically. We explore RF features of backscatter signals with people moving around and fetching tagged goods. We use these features to model and recognize customer actions. In other words, we use the subtle variations and fluctuations in RFID backscatter signals (mainly, RSS values and phase values of received signals) to infer activities of customers in stores.

TagBooth is a working system based on COTS RFID devices. Some localization systems may also be modified to

collect deep shopping data by finding the location of customers and objects with high accuracy, such as ArrayTrack [9] and PinIt [10]. However, the localization accuracies (around 30 centimeters) achieved by these systems are not enough to distinguish different customer actions, not to say that these systems need expensive devices, such as antenna array, or lots of deployment work, such as reference tags deployment. In contrary, TagBooth is of low cost, non-intrusive and easy to deploy.

To validate our design, we build a proof-of-concept prototype and conduct extensive evaluations of TagBooth. Our evaluations of TagBooth show that the accuracy of detecting target commodities is 89.16% and False Positive Rate(FPR) is 6.76%. The precision of action identification is 88.89%.

The remainder of the paper is structured as follows. We introduce deep shopping data and review RFID in Section II. Then empirical studies of RF features under three basic shopping scenes are shown in Section III. The system architecture of TagBooth is in Section IV and details are in Section V and VI. The implementation and evaluation is given in Section VII. Section VIII states limitations of TagBooth. We overview related works of this paper in Section IX. Finally, Section X concludes the paper.

II. PROBLEM STATEMENT AND OVERVIEW

A. Deep Shopping Data

Shopping data has been shown great importance for stores and manufacturer all the time [1], [2], [11], [12]. This paper concentrates on collecting shopping data for physical stores, while that of online stores is a piece of cake. We summarize shopping data as follows:

(1) What customers buy. Statistic on those commodities people pay for are most direct sales performance of stores, such as who pays, what paid for, by cards or cash, when, etc.

(2) What customers want to buy. When customers shop in stores, they may pick goods up and then put them back. Although these unlucky items would not be on the final shopping lists, they have some merits to impress consumers. Relevant data is of value and worth to study. For instance, stores can recommend the user congeneric products with different prices or specialities.

(3) What customers pay attention to. Sometimes items attract attention of customers to stop by, arising interests of customers at the first glance. Data about these consumer behaviors are useful to explore advantages and disadvantages of commodities, being similar to counts of clicking and visiting for cyber shopping.

We call the second and third kinds of data as **Deep Shopping dAta**, i.e. DSA. Although they are not direct reflection of sales performance, DSA are of great value to collect and study as well [2], [3].

Recalling the shopping experiences in a physical store like a clothing store most of consumers would like to move the cloth back and forth to take a glance at the style, and then pick up the one of interest for detailed information. Actually, these basic behaviors reflect the consumers intention, which contributes to composition of complex DSA. Hence,
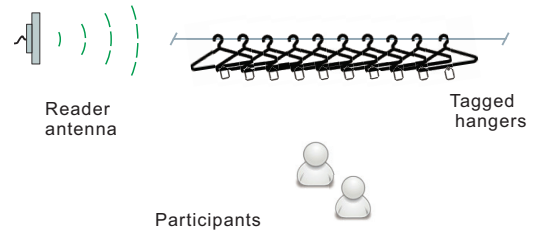

Fig. 1. Clothes store study experiment scene

we acquire the DSA through recognizing two main types of consumer's behaviors (also called events):

(1) Picking events: If a customer wants to buy one piece of clothing, he/she may take it off from the clothes stand and may have a try in fitting room.

(2) Toggling events: Most customers show their interests by toggling clothes on hangers. This is like "clicking and visiting" kind of online shopping data.

An interesting phenomenon has to be mentioned here: when someone picks or toggles one piece of clothes , the target is in motion. Its neighboring objects would sway as well. To describe these two kinds of objects in events, we give a definition:

Definition 1: **Master objects and Slave objects:** The targets in picking events or toggling events are Master objects. The adjacent objects, who are in motion due to the movement of master objects, are Slave objects.

If we desire to collect true DSA, not only events must be recorded accurately, but also the right master objects. However, it is an arduous task to collect those data in most physical stores, because it is difficult to detect those events automatically and effectively. As the rapid development of RFID technology, almost every piece of commodity is attached with a tag, offering an opportunity to acquire the DSA in physical stores by RFID technology.

B. RF Features in RFID

Ultra-low cost of UHF tags ($5 \sim 10$ cents each) become the preferred choice of many industry applications [7]. Following the common practices, we concentrate on UHF RFID system in this paper. The UHF RFID system utilizes the backscatter radio link for communications. Two RF features, *Received Signal Strength(RSS)* and *phase*, are available for Commercial Off-The-Shelf (COTS) readers, like Impinj Speedway Revolution Reader [13].

RSS: The RSS in the unit of dbm is an output parameter reported by the reader, indicating the level of received power. It has log relation to the distance d between the reader and tag, as follows:

$$RSS = RSS_0 - 10\gamma log(d) \qquad (1)$$

where γ is the path loss exponent and depends on the propagation characteristics of the received signal. the γ is related to multipath indoor, and changes as environment changes.

Phase: In backscatter link, the signal traverse a total distance of $2d$ back and forth. Besides the RF phase rotation

(a) RSS SD during walking

(b) Phase SD during walking

(c) Time series phase of Tag#5. Walking starts at 22s and stop at 146s in Near-1. That of Near-2 is 22s and 117s respectively.
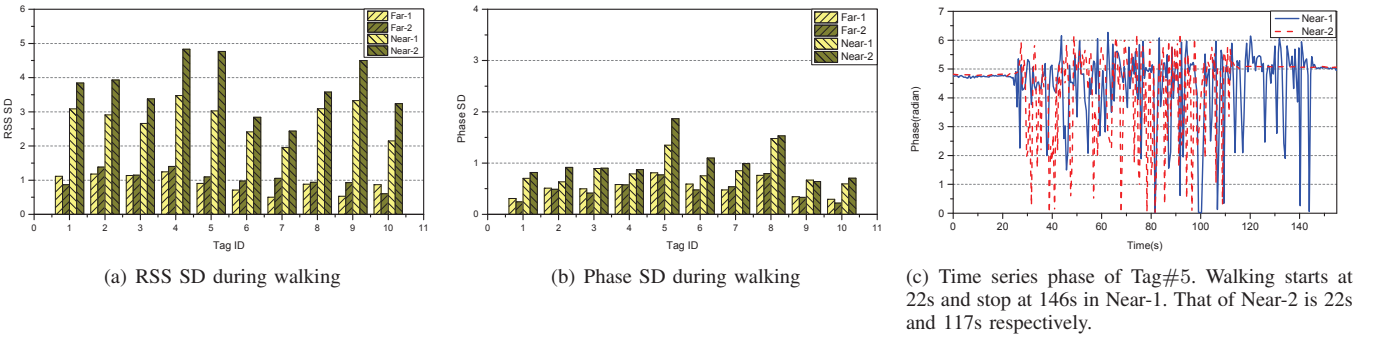
Fig. 2. RF features in Interference Scenario: All tags remain static.

over distance, the reader and tag's characteristics will all introduce some additional phase rotation, denoted as $\theta_{Antenna}$ and θ_{Tag} respectively. The total phase rotation output by the reader can be expressed as:

$$\theta = (2\pi \frac{2d}{\lambda} + \theta_{\text{Antenna}} + \theta_{\text{Tag}}) \bmod (2\pi) \qquad (2)$$

The phase is a periodic function with period 2π radians which repeats every $\frac{\lambda}{2}$ in the distance of backscatter communication.

III. EMPIRICAL STUDIES AND CHALLENGES

Both RSS and RF phase are highly related to distance. A naive solution is to monitor the tag's motion through perceiving the changes of RSS of RF phase to detect customer action events. We conduct a serials of empirical experiments to study whether this naive solution is feasible way for our goal. In the experiments, we employ the ImpinJ reader and 10 tags from Alien, modeled "2x2" tags. The experimental scene is shown in Figure 1. There are total 10 hangers to clothes stand, each of which is attached on a RFID tag and hangs a piece of clothing. The reader is deployed 1.5 meters away from the shelf. Three scenarios are studied here. They are labeled as 'static environment', 'human walking around' and ' human performing action'. Both RSS and RF phase are collected in these three scenarios. We respectively concern these three scenarios and exhibit the results next.

A. Static Scenario

In the first case, we keep the environment as quiet as possible where the hangers remains stationary. It takes 20 minutes to collect the RSS and RF phase. The average Standard Deviation(SD) of collected RSS and phase over these 10 tags are $0.21 dbm$ and $0.22 radians$ respectively, showing that both RSS and phase are stable in a static environment without interference. The Reading Reception Rate(RRR) for each tag is also evaluated as follows. We command the reader to repeatedly identify the 200 tags for 20 minutes. As a result, the lowest RRR is 14.47. It is absolutely high enough to sample feature changes due to human activities if there exists.

B. Human Interfering Scenario

The human activities is the main interference affecting the signal propagation at indoor environment, especially in stores. We consider four cases to understand how the human activities take impact on our two metrics, RSS and RF phase.

(1) Far-1 case: One person walks in the room, at least 1.5 meters away from the reader and hangers.

(2) Far-2 case: 4 people walk in the room, at least 1.5 meters away from the reader and hangers.

(3) Near-1 case: One person wanders near the hanger stand, at most 1.5 meters away.

(4) Near-2 case: People are at most 1.5 meter away from the hanger. They do not touch these clothes hangers and just wander aside. Only 2 participants take part in this case due to space limitations.

We collect data at three stages: before walking, during walking, after walking. Each stage lasts for more than 1 minute. Figure 2 (a) is the Standard Deviation(SD) of RSS data during walking for each tag, while Figure 2 (b) is the SD of phase data during walking. From the figure, we know that the distance of interfering human matters very much. The number of human become very important when the interference distance is small($\leq 1.5m$ in this experiment). However, outlier tags exist in Figure 2 (b): Tag#5 and Tag#8 in Near case. Why this happens while all tags are static? We plot the time series phase of Tag#5 out in Figure 2 (c). The x axis covers the whole time of walking, a small duration before and after walking. We can see that when the phase is near 2π, it drops down sharply to near 0, especially in Near-2 case. This can be explained by Equation 2. Actually the value bounces on edge of 0 or 2π, but is reported after mod operation. That is to say, *periodicity* makes phase of Tag#5 special. We call these tags *Cycle Hop tag*.

C. Action Scenario

We focus on how the consumer's actions, picking and toggling, take impact on RF features.

(1) Picking case: one participant picks one target hanger out. The clothes hanger is in hand for about 30 seconds, during which the participant rotates and watches the hanger over and over again. In the end, he hangs the target back at where it was.

(2) Toggling case: one participant visits and sways one hanger on the stand for about 30 seconds. Because clothes hangers are arranged closely, one hanger moving would lead to motions of neighboring hangers.

The target tag is Tag#4 and we show the SD of RSS and phase during actions in Figure 3 (a) and (b) respectively. We

(a) RSS SD during action

(b) Phase SD during action

(c) Time series phase of Tag#4. Participant starts picking at 8s and hangs hanger back at 39s. Toggling starts at 10s and stops at 37s.
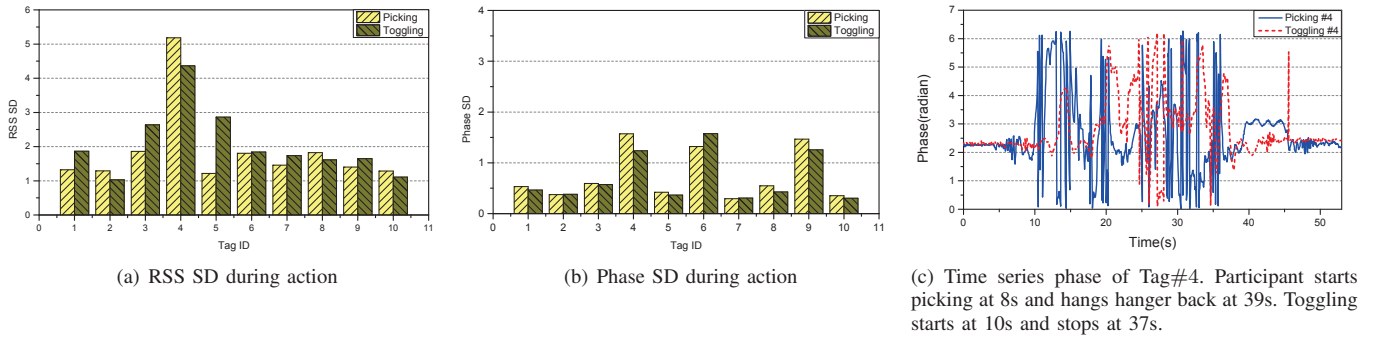
Fig. 3. RF features in Action Scenario: Tag #4 is the target.

observe that the RSS SD of target tag is greater than that of others in action scenario, but close to that in Near case in Figure 2 (a). In terms of phase, there still exists Cycle Hop tags due to periodicity: Tag#6 and Tag#9 here. The SD of target tag is close to that of Cycle Hop tags, because phase value all fluctuates in $[0, 2\pi)$ due to periodicity. That is clear by comparing Figure 2 (c) and Figure 3 (c).

D. Summary

It is a fact that the both RSS and Phase has different exhibition when consumer takes different actions. However, purely utilizing their changes cannot directly infer the consumer's actions because of three main challenges:

(1) Interference from people nearby Multipath makes RSS and phase sensitive to environment. People walking around, block off some paths and generate new paths. RSS and phase value may drift frequently, especially when there are more than one person nearby. It is necessary to eliminate or weaken these interferences to pick tags who are really in motion out.

(2) Hard to differentiate Picking and Toggling As discussed in Problem Formulation subsection, there are two kinds of actions related to DSA: Picking and Toggling. However, RF features of the target tag appear to act similarly under these two actions.

(3) Slave tags and Cycle Hop tags As explained in Problem Formulation, there are master tags and slave tags when customer pick or toggle commodities. RF features of slave tags would fluctuate as well as master tags, such as Tag#3 and Tag#5 in Figure 3. Besides, we have to handle phase periodicity since Cycle Hop tags exist. These two kind of tags hamper judgement of master tags.

These are three practical challenges we have to address.

IV. SYSTEM ARCHITECTURE

We propose TagBooth, a practical RFID based system, to acquire DSA in physical stores. Passive tag is attached on each commodity, with a unique EPC. As an edge device, the reader repeatedly scans the reading zone and reports their readings to backend server where the TagBooth runs. A high-level work flow through this architecture is described as follows: *(1)* The reader reports four tuples including tag's EPC, RSS, $phase$, and time t, which is the input elements for TagBooth. *(2)* The component of *Motion Detection* illustrated in Figure 4
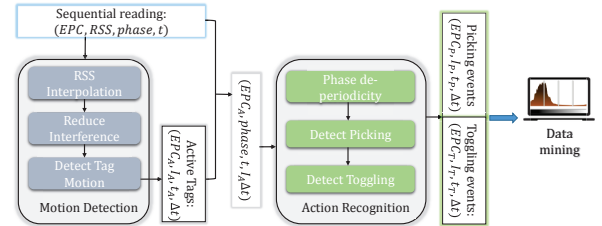

Fig. 4. System diagram of TagBooth

is to detect whether the tag is in motion. If the answer is yes, motion intensity I_A (defined later) would be calculated during $t_A, \Delta t_A$. Hence, the result of motion detection are tuples: $(EPC_A, I_A, t_A, \Delta t_A)$. Tags with ID of EPC_As are taken for active tags. *(3)* Original $(phase, t)$s of active tags are taken out and feed into the second component, called *Action Recognition*. Phase would be preprocessed and utilized to distinguish active tags of Picking Action out. Hence, Picking events are recognized. TagBooth cluster the rest of active tags into two sets. The one with higher average motion intensity is set of master tags, and taken as the targets of Toggling actions.

We employ RSS to address the issue of interference. A sliding window algorithm is adapted to obtain start time and duration of motion. We remove the periodicity of phase and take advantage of phase to recognize two actions in the component of Action Recognition, by solving remaining two challenges. Details of each component are elaborated in next two sections.

V. MOTION DETECTION

Motion detection is fed by a time series of RSS values and outputs tag's motion intensity, start time and time interval.

A. RSS Interpolation

The time interval between two reported data for one tag is not equivalent, which hinders our further process. Hence, interpolation is needed for raw data first, so is a fitted model.

Hypothesis 1: During a very short time interval, the γ could be regarded as a constant.

As the time interval between two collected records is rather small than that between human actions, the γ could be regarded as a constant during the interval. In a similar way, we can
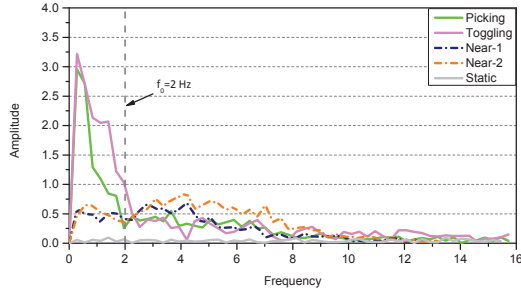
Fig. 5. Frequency domain characteristics of $Tag\#4$ RSS from study experiments. We take $4s$ from each data set and do interpolation. Then FFT is performed.

Fig. 6. Diagram of motion detection flow

assume that customers take the tagged commodity away in a constant velocity between two records. Then the distance between antenna and tag changes linearly by time, represented by $d = v_0 t$. We have:

$$RSS = RSS_0 - 10\gamma_0 log(v_0) \times log(t) \qquad (3)$$

RSS_0 and $\gamma_0 log(v_0)$ are two constants that could be calculated from two neighbouring data points. Equation 3 is the fitted model we use for interpolation of RSS values. Theoretically the higher the sampling frequency is, the better the signal processing result is. However, we just need signal data to analyse contained factors of human actions, which have rather low frequency. Here, 32 data points in 1 second after interpolation is enough. We use $32Hz$ as the sampling frequency in implementation and evaluation of TagBooth.

B. Reducing Interference

Our method to reduce interference on RSS is based on a hypothesis:

Hypothesis 2: The frequency of RSS changes due to picking or toggling remains below a fixed value f_0.

That is easy to understand. A customer, who is really interested in the commodity, would not perform actions as quickly as possible. To consider the process of picking up or moving a commodity, the count of movements per second has rules to follow, though subtle differences exist for individuals. This hypothesis inspires us to utilize the frequency of RSS changes to screen out changes due to customer actions by a reasonable f_0.

We take five original RSS data sets of $Tag\#4$ from study experiments: static scenario, Near-1 case and Near-2 case in interference scenario, Picking and Toggling case in action scenario. We take $4s$ data out from each data set and do interpolation, for instance $4s$ RSS data during walking in Near-case. Then, we do Fast Fourier Transform(FFT) to see what is the difference in frequency domain. The results of FFT is shown in Figure 5. Since the sampling frequency is $32Hz$, the max frequency is $16Hz$ after FFT. From the figure, we can see that the RSS energy of action data sets concentrate on frequencies below $2Hz$. In relative terms, the distribution of energy from interference data sets is more uniform in $[0, 8]$. Besides, the amplitude is more obvious during $[2, 6]$ for Near-2 case. The reason is that multipath effects change very fast when there are more than one persons walking nearby. Propagation
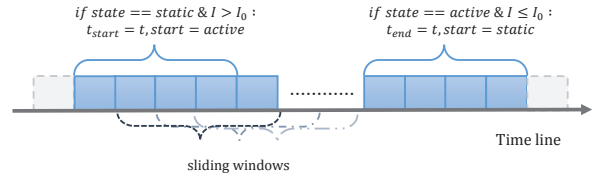
paths are broken and reestablished frequently. While one customer is performing action on the tagged commodity, the dominant reason of RSS variance is the changes of Line-Of-Sight(LOS) path length and path loss, i.e. d and λ in Equation 1. The frequency of these two factors is limited to that of customer actions. From Figure 5, we observe that $f_0 = 2Hz$ is a good value to blank off most impact of interference. We believe that $2Hz$ is reasonable since a customer changes $n \in [0, 2]$ gestures in one second basically.

C. Motion Intensity and Detection Flow

Definition 2: **Motion Intensity:** The integration RSS energy of those components whose frequency less than f_0 after FFT.

$$I_{motion} = \int_0^{f_0} E(f)df \qquad (4)$$

where $E(f)$ is the RSS energy component with frequency of f after FFT.

Motion Intensity embodies the variation of RSS after abandoning components not related to customers' actions. Therefore, it reflects that whether customers' actions are intense or not, which is a valuable information of DSA. Therefore, we employ I_{motion} in Equation 4 to represent the intensity of tag motion. In practice, we use discrete FFT. Hence, we just need to sum energy of those components to get I_{motion}. Although we reduce much of interference from people nearby by abandoning high frequency parts($> f_0$), there still exists interference and noise energy in I_{motion}. We need a threshold to select real moving tags out, named I_0. For any tag, whose $I_{motion} > I_0$, we consider it to be an "active" one, even if it may be a slave tag. To filter slave tag out is the job of next section.

The whole flow of detecting "active" tags is shown in Figure 6. The initial state for every tag is static: $state = static$. We keep a sliding window of RSS data with size of 4 seconds. Every time a new second RSS data is collected, we do interpolation work, abandon the earliest in sliding window and add the new one. Next we perform FFT and check whether $I_{motion} > I_0$. If yes, we record the time of most recently one second data as start time and set $state = active$. After that, the time when $I_{motion} \leq I_0$ is regarded as end time. Duration is calculated by these two values. So far, we get the whole output of Motion Detection module along with motion intensity I_{motion}.

VI. ACTIONS RECOGNITION

Our goal is to collect DSA for stores. One record includes commodity information, customer action, time. Through previous section, we have tags in motion (both master tags and
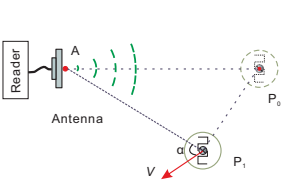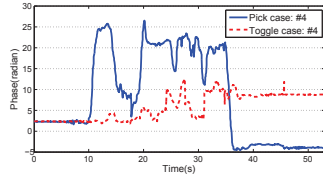
Fig. 7.   Scene of Picking Action

Fig. 8.   Phase of Tag#4 after de-periodicity

slave tags), their motion intensity, the time and duration. Now we have to identify which tag is the target of what kind of action.

We find that slave tags always remain on shelves, as well as targets of toggling events. Hence, we first recognize picking action, whose tags are surely master tags. Then, we separate master tags and slave tags. The rest master tags are targets of toggling actions. First step of this module is to deal with phase periodicity and eliminate Cycle Hop tags.

A. Phase De-periodicity

The periodicity troubles us in Figure 3. We need to deal with phase periodicity and eliminate these Phase Edge tags. We call the process phase de-periodicity. The method depends on a hypothesis:

Hypothesis 3: The absolute difference of two adjacent reading phase value is smaller than π.

Figure 7 shows the scene of picking a commodity out. A is the location of reader antenna; P_0 is the original position of commodity; P_1 is the current position. Customer is taking the target away at the speed of $\overrightarrow{v}$. The angle between $\overrightarrow{AP_1}$ and the direction of $\overrightarrow{v}$ is α. Then the speed of antenna-tag distance decreasing is $v_d = v\cos(\alpha)$. Combining this with Equation 2, we have $\Delta\theta = \frac{4\pi}{\lambda}vt\cos(\alpha)$. To satisfy Hypothesis 3, the speed of fetching target v should satisfy $v < \frac{\lambda}{4t\cos(\alpha)}$. t is the time interval between two readings. Typically, the wavelength λ is $\sim 33cm$ for UHF radio. In study experiments, the lowest RR is 14.47 per second while the tag population is 256. Therefore, the time interval $t > 0.07s$. The demand on speed of fetching commodity comes to:

$$v < \frac{4.7}{\cos(\alpha)}m/s \tag{5}$$

which is apparently true for shopping customers. As of Toggling action, the tag sways going left and right: $\alpha = 0$. The Inequation 5 holds true as well. As a consequence, Hypothesis 3 is valid for readings from tags in physical stores.

We preprocess phase values for every tag according to Hypothesis 3. For each reading, once the phase and previous one does not meet the hypothesis, we add or sub 2π to it. The processed values from Figure 3 (c) are shown in Figure 8.

B. Recognizing Picking Action

After de-periodicity, the difference between Picking case and Toggling case is obvious in Figure 8. We can see that phase in Picking case suffers great continuing ups and continuing downs during the action, while phase in the Toggling case

does not. We call one continuing up or continuing down a step. We use the average absolute phase step length to reveal the difference:

$$\theta_{\text{step}} = \frac{\sum_{i=2}^{N}|\theta_i - \theta_{i-1}|}{1 + \sum_{i=3}^{N}\delta_i} \tag{6}$$

where $\forall i \in [3, N]$

$$\delta_i = \begin{cases} 0 & \text{if } (\theta_i - \theta_{i-1})(\theta_{i-1} - \theta_{i-2}) \geq 0 \\ 1 & \text{otherwise} \end{cases}$$

N is the count of phase readings for the target tag during start time and end time detected last module. A threshold θ_{thres} is used to separate Picking cases and Toggling cases. If $\theta_{\text{step}} > \theta_{\text{thres}}$, we take the tag for object of picking action. Otherwise, the tag is object of toggling action. We discuss impacts of θ_{thres} on action recognition in evaluation experiments.

C. Recognizing Toggling Action

The rest tags are targets of toggling actions and all slave tags after we pick off-shelf tags out. Therefore, once we distinguish master tags and slave tags, we finish the work of detecting toggling actions and their objects.

Data clustering works appropriately to the problem [14]. All active tags own intensity and start time. As described in last section, motion tags are detected every 1 second. Hence, the start time has granularity of 1 second, which is enough for slave tags to response once the master tag is being picked or toggled. We first align motion tags by their start time. Then hierarchy clustering is performed on tags with the same start time. The similarity measurement is the inverse of 1-norm distance based on motion intensity. Two sets of tags are clustered and we claim one with higher mean intensity to be master tags.

VII.   IMPLEMENTATION AND RESULTS

In this section, we present the implementation of TagBooth and evaluate the prototype in two experiment scenes.

A. Implementation

1. Hardware: We implement a prototype of TagBooth using Impinj R420 with one directional antenna. "2 × 2" tags from Alien company are used in our experiments. The reader works at the frequency of $920.5MHz$ by default, which is the legal UHF band in China.

2. Software: The software part running at personal computer is implemented using Java language. It connects to the RFID reader with LLRP protocol. This protocol was ratified by EPCglobal in April 2007. Results of TagBooth would be recorded into database as DSA.

3. Parameter choice: There are two pivotal threshold parameters in TagBooth: $I_0, \theta_{\text{thres}}$. I_0 is a threshold to determine whether one tag is in motion. Choice of I_0 and effect on accuracy are evaluated in both two evaluation scenes. And we employ θ_{thres} to decide which tags are master tags in Picking events. We test how to choose an appropriate θ_{thres} in the second evaluation.
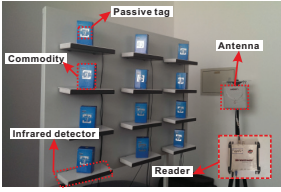
Fig. 9. Motion detection evaluation scene

Fig. 10. Action recognition detection evaluation scene

B. Evaluation Methodology

Two experiment scenes are designed towards measuring the performance of TagBooth. Both are performed in a corridor outside the office room whose width is $350cm$, as shown in Figure 9 and 10. Every commodity in both scenes is attached with a unique "2 × 2" Alien tag. Only one antenna is needed and there are no requirements for deployment, except that the reading zone of reader antenna should cover all tested tags. The first scene (Figure 9) is designed quite simple so that we could use infrared devices to record ground truth automatically. We have 12 rectangle boards, each holding an item. Only picking events are tested and the accuracy of motion detection would be evaluated. The second scene (Figure 10) approaches the real scene in a cloth store. We hang 20 pieces of clothes to a clothes hanger stand. Each stand is attached with a RFID tag. We invite persons in our lab to do picking or toggling actions randomly. Both motion detection and action recognition would be tested.

Ground Truth Recording: 1. In the first experiment, the shelf is simple so we could use infrared devices to record the ground truth. An infrared detector in each board knows whether and when the item leaves. However, we should maintain the one-one relationship of item and board, otherwise we do not know which item is picked. Meanwhile, it is worth noting that the infrared methods does not work in complicated scene, such as multiple items on a board, the second scene, items out-of-order and blocking of LOS between detector and item. 2. We record ground truth by hand in the second experiment. Once the experiment begins, we start a camera to record videos of the experiment. At the end, we analyse the video manually to record shopping data, including the target ,action type, start time and end time.

Metrics: We use two important metrics to measure the accuracy of motion detection and action recognition: *True Positive Rate(TPR)* and *False Positive Rate(FPR)*. TPR is the total number of detected true positives divided by the number of real positives in ground truth. FPR is termed as the rate of number of false positives to the total number of negative events in ground truth. Another important indicator is *Time Error* in s of detecting events. Because in many application scenes, merchants desire to interact with customers timely while they pick or toggle a piece of commodity, such as playing a video showing specialities of the commodity. Interaction effect would be greatly reduced if the delay is too much.

C. Motion Detection Evaluation

Ground Truth: The experiment lasts for 20 minutes. TagBooth gets data stream from RFID reader and executes algorithm to detect motion. During the 20 minutes, 9 persons of our lab take commodities off the shelf, then put them back from whenever they want. 83 picking events are detected by infrared detectors as ground truth. We draw these events in Figure 11 (a) with red lines. Start and end x value of each red line represent the starting time and end time, while the y value stands for tag ID $\lceil y \rceil$ [1]. The ground truth timeline shows that multiple events happen simultaneously.

Evaluation Results: The higher threshold I_0 is, less events would be detected by TagBooth. Figure 11 (b) shows the TPR and FPR of motion detection when I_0 changes. The FPR goes down from 26.9% to 0 as I_0 decreases. Besides, when I_0 is below 18, the precision stays more than 90%. However, the precision drops rapidly when we set $I_0 > 20$, while the improvement of FPR is limited to 6.33%. Therefore, we think 20 is a good choice for I_0 in this experiment.

When we set $I_0 = 20$ as the motion intensity threshold, TagBooth detects 79 events, as drawn in Figure 11 (a) in blue lines. y value of blue lines signifies tag ID of the event to be $\lfloor y \rfloor$. 74 of 79 events have time overlap with ground truth events and we call them True Positive events while the rest are False Positive events. Therefore, the TPR is 89.16% and FPR is 6.33% when I_0 is 20. Meanwhile, we compare the start time and duration time detected to ground truth. Figure 11 (c) plots the Cumulated Distribution Function(CDF) of time error in second. 90% start time error of events is less than $2.6s$ and the mean error is $2.11s$. The reason for start time error is that participants wield hands in LOS path between tag and antenna for a small duration before they pick the commodity out. As of duration time, the mean error is $2.6s$, which is completely tolerable for DSA acquisition. Besides, the visualized detected results in Figure 11 (a) show a good performance of time detection.

D. Action Recognition Evaluation

Ground Truth: The experiment lasts for 20 minutes and 59 action events happened. 34 of them are picking events and others are toggling events. However, target tags of these 59 events are not all tags in motion during the 20 minutes. The ground truth events are shown in Figure 12 (a). As well as the first experiment, multiple events from different persons happen simultaneously.

Evaluation Results: We first exploit the effect of I_0 on TPR and FPR in this evaluation scene, as shown in Figure 12 (a). Numerator of TPR is count of tags in motion detected who are really target tags in ground truth, slave tags excluded. Numerator of FPR is count of tags in motion detected who are not target tags in ground truth. As I_0 increases, less and less tags are judged as events targets, leading lower TPR. However, the FPR does not perform like in Figure 11 (b). This is because slave tags exist in the this evaluation scene. There exist situations like: one participants toggles one piece of clothes with a big range and another toggles with a small range. Motion intensity of slave tags in the former may be larger than master tags in the latter. Therefore, one I_0 may make the slave tags stay and keep the master tags of second event outside. Hence, FPR may become higher as the I_0 is larger. Nevertheless, we do not have to worry about FPR because of

---

[1] $\lceil y \rceil$ means rounding y up to an integer while $\lfloor y \rfloor$ is rounding y down to an integer.
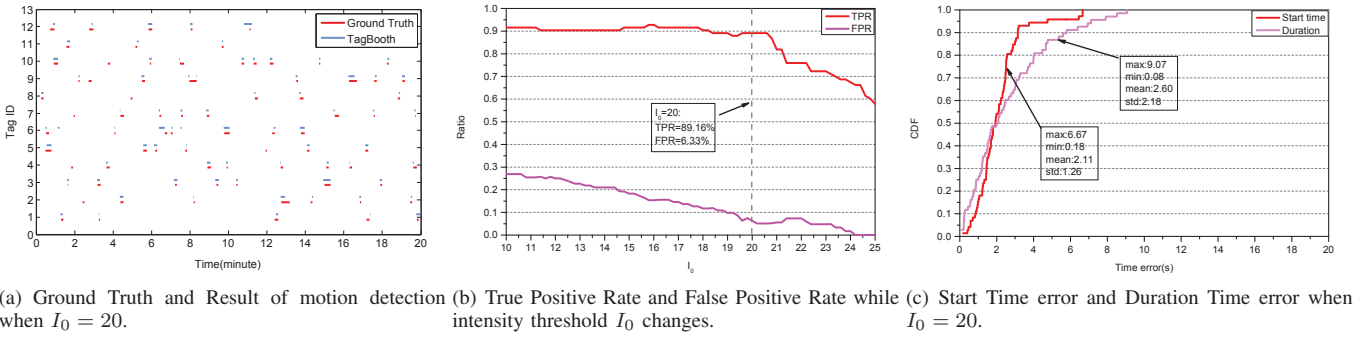
(a) Ground Truth and Result of motion detection when $I_0 = 20$.

(b) True Positive Rate and False Positive Rate while intensity threshold I_0 changes.

(c) Start Time error and Duration Time error when $I_0 = 20$.

Fig. 11.   Results of motion detection evaluation

(a) TPR and FPR of action detection when I_0 changes.

(b) TPR and FPR of picking and toggling respectively: $I_0 = 8$.

(c) Start Time error and Duration Time error: $I_0 = 8, \theta_{thres} = 2.5$.
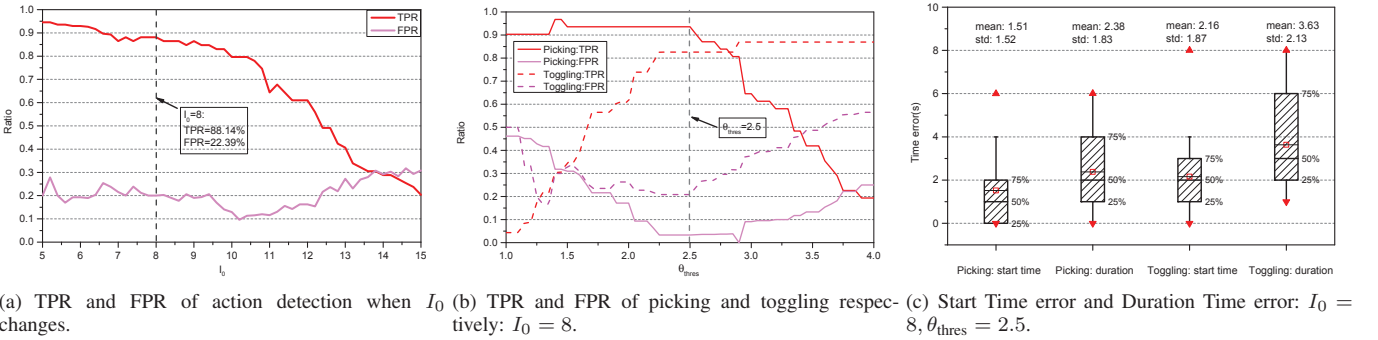
Fig. 12.   Results of action recognition evaluation

slave tags here. They will be abandoned after clustering in recognizing toggling action. That is to say, we can choose a reasonable I_0 here to keep good TPR and leave FPR alone. When we set $I_0 = 8$, TagBooth finds 67 times of tags in motion. 54 of them are master tags of events in ground truth.

Next, we focus on the accuracy of recognizing picking events and toggling events. 67 times of motion detected. We first use θ_{thres} to take picking action out. The rest are divided into small sets, aligned by motion start time. After clustering, toggle targets are recognized out in each set. As θ_{thres} is higher, more events are considered as toggling events, leading to higher TPR of toggling and lower TPR of picking. Figure 12 (b) plots TPR and FPR of picking and toggling as $\theta_{thres} \in [1, 4]$. As we can see, 2.5 is a good choice to maintain good accuracy of recognizing picking and toggling, 93.55% and 82.61% respectively. After clustering and excluding slave tags, the total rate of action events identified correctly is 48 out of 54, i.e. 88.89%.

When we fix I_0 to be 8 and θ_{thres} to be 2.5, we get the time error of event start time and duration in Figure 12 (c). The granularity of ground truth time recorded by hands is 1 second, so is the step of sliding window in TagBooth. Therefore, the granularity of time error here is 1 second. The mean start time error of detecting picking action is $1.51s$, which is real time enough for applications to show product information once a customer picks a commodity. That of toggling action is a little higher: $2.16s$. Duration is important and shows customers' will. The mean error of duration time is $2.38s$ and $3.63s$ respectively. We think the result is accurate enough to reflect customers' intention.

VIII. LIMITATIONS

Evaluation shows that TagBooth has a good performance on acquiring deep shopping data, i.e. two kinds of action events. However, there are two limitations of TagBooth by far: *1. Identification of customer individual:* If each record of DSA contains the identification of customer, the data would be more valuable for applications, such as personal product recommendation. So far, TagBooth is not able to identify any person who is shopping in a physical store. However, if the customer has a RFID VIP(Very Important Person) card of the store, the story is different. As elaborated in PinIt [10], the multipath profiles of two near tags are similar. We explode multiple channels to collect multi-channel profiles of target commodity tag and VIP cards. The most similar VIP card to one tag of action events is considered as the identification of customer. *2. Discriminate staff from customers:* If store staff tidy commodities on shelves, TagBooth would record these events as DSA by mistake. However, if the staff takes a RFID job card during tidying work, we could identify that the actions performer is a staff as stated above, thus not taking these events as DSA.

IX. RELATED WORK

Shopping data acquisition: Lewis [15] deploys portable device on shopping carts to collect data. Some work for physical stores focus on other kinds of data, such as shopping time [16] and shopping paths of customers [12]. Besides, video analysis are possible solutions. Niu *et al.* [4] presents a framework for detecting and recognizing human activities on simple statistics compiled on the tracked trajectories. Ribeiro *et al.* [5] use a Bayesian classifier to recognize human activities from

video sequences. **RFID systems in stores:** [17] describes development fo RFID-based personal shopping assistant .system for retail stores [18] presents a case study of an RFID project at Galeria Kaufhof. [19] develops a customized commodity recommendation algorithm and a shopping route determination and guiding algorithm. **RF localization method:** Some of localization methods model the relations between RSS and distance, and explore localization by RSS [20]–[22]. Others use phase as indicator for distance and propagation direction [10], [23], [24]. PinIt [10] makes use of multipath-effect based fingerprints with the help of reference tags to acquire high accuracy in complex environment. However, it requires special devices and a lot of deployment work. **Other related RFID issues:** [25] designs a query protocol for serverless mobile RFID systems. ASAP [26] explores how to arbitrate collisions in large scale RFID systems. PMTI [27] identifies missing tags and OTrack [28] sequences a steam of RFID tags. Frogeye [29] is a more lightweight work, focusing on monitoring whether tag is moved instead of locating it. Twins [30] tracks device-free objects using passive tags.

X. CONCLUSIONS

This work presents a novel solution to acquire Deep Shopping dAta(DSA) for physical stores. Our observations from extensive preliminary experiments motivate the design of TagBooth, a practical RFID system to achieve the goal. We have built the proof-of-concept system and conducted extensive evaluations to test the performance of TagBooth. Our evaluations show a very good performance. Future work of TagBooth includes breaking limitations stated before and detecting complex customer actions.

ACKNOWLEDGEMENT

REFERENCES

[1] D. R. Bell and J. M. Lattin, "Shopping behavior and consumer preference for store price format: Why large basket shoppers prefer edlp," *Marketing Science*, vol. 17, no. 1, pp. 66–88, 1998.

[2] G. L. Lohse, S. Bellman, and E. J. Johnson, "Consumer buying behavior on the internet: Findings from panel data," *Journal of interactive Marketing*, vol. 14, no. 1, pp. 15–29, 2000.

[3] A. G. Close and M. Kukar-Kinney, "Beyond buying: Motivations behind consumers' online shopping cart use," *Journal of Business Research*, vol. 63, no. 9, pp. 986–992, 2010.

[4] W. Niu, J. Long, D. Han, and Y.-F. Wang, "Human activity detection and recognition for video surveillance," in *ICME*, vol. 1. IEEE, 2004, pp. 719–722.

[5] P. C. Ribeiro and J. Santos-Victor, "Human activity recognition from video: modeling, feature selection and classification architecture," in *Proceedings of International Workshop on Human Activity Recognition and Modelling*. Citeseer, 2005, pp. 61–78.

[6] *Shopper Tracking*, http://www.vizualize.net/solutions/shopper-tracking.html.

[7] R. Das and P. Harrop, "Rfid forecasts, players and opportunities 2014-2024," 2011.

[8] E. P. Kelly and G. S. Erickson, "Rfid tags: commercial applications v. privacy rights," *Industrial Management & Data Systems*, vol. 105, no. 6, pp. 703–713, 2005.

[9] J. Xiong and K. Jamieson, "Arraytrack: A fine-grained indoor location system." in *NSDI*, 2013, pp. 71–84.

[10] J. Wang and D. Katabi, "Dude, where's my card?: Rfid positioning that works with multipath and non-line of sight," in *Proc. of SIGCOMM*. ACM, 2013, pp. 51–62.

[11] G. Häubl and V. Trifts, "Consumer decision making in online shopping environments: The effects of interactive decision aids," *Marketing science*, vol. 19, no. 1, pp. 4–21, 2000.

[12] T. Fujino, M. Kitazawa, T. Yamada, M. Takahashi, G. Yamamoto, A. Yoshikawa, and T. Terano, "Analyzingin-store shopping paths from indirect observation with rfidtags communication data," *Journal on Innovation and Sustainability. RISUS ISSN 2179-3565*, vol. 5, no. 1, pp. 88–96, 2014.

[13] *Low Level Data Support*. http://www.impinj.com/: Impinj Inc.

[14] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," *CSUR*, vol. 31, no. 3, pp. 264–323, 1999.

[15] C. E. Lewis and T. P. O'Hagan, "Shopping cart mounted portable data collection device with tethered dataform reader," Oct. 13 1998, uS Patent 5,821,513.

[16] C.-W. You, C.-C. Wei, Y.-L. Chen, H.-H. Chu, and M.-S. Chen, "Using mobile phones to monitor shopping time at physical stores," *IEEE Pervasive Computing*, vol. 10, no. 2, pp. 37–43, 2011.

[17] E. Ngai, K. Moon, J. N. Liu, K. Tsang, R. Law, F. Suk, and I. Wong, "Extending crm in the retail industry: an rfid-based personal shopping assistant system," *Communications of the Association for Information Systems*, vol. 23, no. 1, p. 16, 2008.

[18] F. Thiesse, J. Al-Kassab, and E. Fleisch, "Understanding the value of integrated rfid systems: a case study from apparel retail," *European Journal of Information Systems*, vol. 18, no. 6, pp. 592–614, 2009.

[19] J.-L. Hou and T.-G. Chen, "An rfid-based shopping service system for retailers," *Advanced Engineering Informatics*, vol. 25, no. 1, pp. 103–115, 2011.

[20] L. M. Ni, Y. Liu, Y. C. Lau, and A. P. Patil, "Landmarc: indoor location sensing using active rfid," *Wireless networks*, vol. 10, no. 6, pp. 701–710, 2004.

[21] A. Bekkali, H. Sanson, and M. Matsumoto, "Rfid indoor positioning based on probabilistic rfid map and kalman filtering," in *Proc. of IEEE WiMob*, 2007.

[22] C. Wang, H. Wu, and N.-F. Tzeng, "Rfid-based 3-d positioning schemes," in *Proc. ofNFOCOM*. IEEE, 2007.

[23] T. Liu, L. Yang, Q. Lin, Y. Guo, and Y. Liu, "Anchor-free backscatter positioning for rfid tags with high accurancy," in *Proc. of INFOCOM*. IEEE, 2014.

[24] 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 the 20th annual international conference on Mobile computing and networking*. ACM, 2014, pp. 237–248.

[25] J. Lim, S. Kim, H. Oh, and D. Kim, "A designated query protocol for serverless mobile rfid systems with reader and tag privacy," *Tsinghua Science and Technology*, vol. 17, no. 5, pp. 521–536, 2012.

[26] C. Qian, Y. Liu, R. H. Ngan, and L. M. Ni, "ASAP: scalable collision arbitration for large RFID systems," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 7, pp. 1277–1288, 2013.

[27] Y. Zheng and M. Li, "P-mti: Physical-layer missing tag identification via compressive sensing," in *IEEE INFOCOM*, 2013.

[28] L. Shangguan, Z. Li, Z. Yang, M. Li, and Y. Liu, "Otrack: Order tracking for luggage in mobile rfid systems," in *IEEE INFOCOM*, 2013, pp. 3066–3074.

[29] L. Yang, Y. Qi, J. Fang, X. Ding, T. Liu, and M. Li, "Frogeye: Perception of the slightest tag motion," in *Proc. of INFOCOM*. IEEE, 2014.

[30] J. Han, C. Qian, D. Ma, X. Wang, and J. Zhao, "Twins: Device-free object tracking using passive tags," in *Proc. of INFOCOM*. IEEE, 2014.