Towards Large-Scale RFID Positioning: A Low-cost, High-precision Solution Based on Compressive Sensing

†Liqiong Chang, †Xinyi li, †Ju Wang, ‡Haining Meng, †Xiaojiang Chen*
†Dingyi Fang, †Zhanyong Tang, §Zheng Wang
†Northwest University, ‡Chinese Academy of Sciences, §Lancaster University

Abstract—RFID-based positioning is emerging as a promising solution for inventory management in places like warehouses and libraries. However, existing solutions either are too sensitive to the environmental noise, or require deploying a large number of reference tags which incur expensive deployment cost and increase the chance of data collisions. This paper presents CSRP, a novel RFID based positioning system, that is highly accurate and robust to environmental noise, but relies on much less reference tags compared with the state-of-the-art. CSRP achieves this by employing an noise-resilient RFID fingerprint scheme and a compressive sensing based algorithm that can recover the target tag's position using a small number of signal measurements. It provides a set of new analysis, algorithms and heuristics to guide the deployment of reference tags and to optimize the computational overhead. We evaluate CSRP in a deployment site with 270 commercial RFID tags. Experimental results show that CSRP can correctly identify 84.7% of the test items, achieving an accuracy that is comparable to the state-of-the-art, using an order of magnitude less reference tags.

I. INTRODUCTION

Radio Frequency IDentificaiton (RFID) is rapidly emerging as a viable means for tracking and localizing objects [1]–[6]. By simply attaching tags to objects, one can then identify the ID of an object of interest and locate its position – by using an contactless RFID reader to gather data from object tags based on a pre-defined radio frequency and protocol. RFID-based positioning is widely seen as a promising solution for inventory management in places like warehouses and libraries [7].

Prior works in RFID-based positioning systems exploit the received signal strength (RSS) information of surrounding object tags to estimate the target's location [1]–[4]. However, such an approach is unreliable as the RSS reading is inherently sensitive to the environmental noise. As an alternative, the Angel of Arrival (AOA) approach [8]–[10] employs multiple antennas to cancel the environmental noise. An AOA scheme calculates the target object's location by measuring the differences of the radio frequency (RF) signal phases between the target tags and antennas. The working mechanism of AOA assumes that the RF signal travels in a direct, line-of-sight (LOS) path from the source to the receiver. This assumption means that the positioning precision can be greatly affected by any physical object that obstructs the signal transmission path, a problem known as non-line-of-sight (NLOS) propagation.

PinIt is the current state-of-the-art RFID-based positioning system [6]. It was proposed to address the NLOS issue. PinIt works by first deploying reference tags at known locations, and then using a reference tag that has the most similar signal characteristics to the target tag to estimate the target's position. To achieve a high accuracy, PinIt requires placing a large number of reference tags, sometimes having more reference tags than the deployed ones. This large number of reference tags not only incurs high deployment cost, but also increases the chance of data collision between tags and leads to deleterious performance in a large-scale deployment. To make RFID-based positioning practical at scale, we must find ways to greatly cut down the number of reference tags required.

This paper introduces CSRP, a novel RFID positioning system that works with NLOS. One of the advantages of CSRP is that it requires significantly less reference tags than the state-of-the-art but without scarifying the precision. Our key insight is that the number of tags to be positioned at a given moment in a large-scale deployment is highly likely to be smaller than the total number of tags. If we consider the problem of positioning as finding a location vector where an element with a value of '1' marks a target tag of a specific location and '0' otherwise, then most of the elements of the vector will be zero, i.e., the location vector is sparse. The sparsity of the location vector suggests that through optimization, it is possible to recover the sparse location vector by sampling a small number of reference tags – using a recently established signal processing technique called compressive sensing (CS) [11]. Essentially, CS allows one to reconstruct a sparse signal (i.e., the location vector in our case) from far fewer samples (i.e., reference tags in our case) than required by the Shannon-Nyquist sampling theorem under certain conditions [12]. If this can be achieved, we can then greatly reduce the number of reference tags required; and consequently, we can save cost and time, and decrease the chance of data collision in a large-scale RFID deployment.

Transforming this high-level idea into a practical system is non-trivial, because we have to rely on a small number of reference tags. It is undesirable to use the signal measurement of a deployed target tag as the fingerprint for positioning, like prior works did [1], [4]. This is because an instantaneous measurement for a tag can significantly differ from the previous measurement of the same tag due to dynamic environmental

noise. We cannot use the signal multipath profiles of nearby reference tags as fingerprints to estimate the location of a deployed target tag like PinIt does either. This is because our small number of reference tags will have to be sparsely distributed in the deployment site and as a result, many object tags will be far from a reference tag.

Our solution is to calculate, for a deployed ojbect tag, its spatial relation with respect to each reference tag in terms of their multipath profiles. We then use this relation to build a fingertip matrix (or database) for all object tags. The key of our approach is to use the relatively stable spatial relation between the ojbect tag and static, fixed reference tags to reduce or even cancel the dynamic environmental noise. Since the spatial relation is automatically calculated, the fingertip matrix will be updated if any of the object tags has been replaced or removed. This ability allows CSRP to adapt to the changing deployment environment without human involvement.

To locate a target tag, we first use an RFID reader to measure the multipath profile of that tag. Next, we calculate the spatial relation between the multipath profiles of the target tag and each static reference tag. We organize the calculated spatial relations as an one-dimensional vector of real values, for which we called a ***real-time measurement***. We then use a sparse recovery algorithm under the `CS` framework to estimate the target tag's position. Specifically, the algorithm tries to find a one-dimensional vector, whose multiplication production with the fingerprint matrix is as close as possible to the real-time measurement vector. The found vector is essentially the sparse location vector that we are looking for. The element with a value of '1' in the location vector indicates the estimated position or location id for the target tag.

However, performing the recovery algorithm on a large fingerprint matrix could be expensive. To reduce the computational overhead, we decouple the problem into two sub-problems. We first divide the deployment region into subareas (e.g., different book shelves in a library), so that each subarea contains a subset of the object tags – on which a smaller fingerprint matrix can be constructed. To locate a tag, we first use an offline learned classifier to predict which subarea the target tag belongs to, and then apply the sparse recovery algorithm to the smaller fingerprint sub-matrix to quickly find the location vector. This divided-and-conquer strategy allows our algorithm to scale to a large RFID deployment.

A natural question is: "how many reference tags are needed"? Deploying too many reference tags will waste money and time, but having too few of them will result in poor performance. In this paper, we prove that organizing the reference tags as a mesh of equilateral triangles is the most cost-effective way to achieve high precision, and develop a heuristic to determine the minimum number of reference tags and how to organize them in a specific region.

We have developed a prototype of CSRP and evaluated it in a library room with rich multipath and `NLOS`. Our experimental results show that CSRP is able to correctly locate 84.7% of the test items with an average location accuracy of less than ten centimetres across all test items. This performance is comparable to PinIt, but CSRP uses an order of magnitude less reference tags and does not rely on specialize hardware.

This paper makes the following contributions:

- We present a novel RFID-based positioning system. Our system offers high positioning precision but requires an order of magnitude less reference tags than the state-of-the-art. As a result, it saves cost and time in a large-scale deployment.
- Our work is the first to employ compressive sensing for RFID-based positioning.
- We show how to translate our novel idea to a practical system. We present new analysis and algorithms to calculate the optimal number of reference tags as well as the best way for organizing reference tags.
- Our approach is immediately deployable on off-the-shelf RFID hardware.

II. OUR APPROACH

A. Overview

Fig. 1 illustrates a high-level overview of CSRP, which consists of four offline and online stages described as follows.

Offline Deployment and Data Collection: The first step is to deploy a number of reference tags in the deployment site. We then collect the signal phase readings for all object tags and reference tags. This process only needs to perform once. If an object tag is replaced, we recollect the phase reading of the new tag. If an object tag is removed, we take its reading out from our database. We show that placing the reference tags at the intersections of a mesh of equilateral triangles gives the best cost-effective performance. See Section II-D.

Offline Fingerprint Matrix Construction: After collecting the data for each tag, we fingerprint each object tag by calculating the spatial relation between the object tag's multipath profile and each reference tag's. This process gives us a fingerprint matrix for all object tags. See Section II-C.

Online Measurement Vector Acquisition: To locate an item with an known tag ID, we first use an RFID reader to track the `RF` signal associated with the tag ID. We take the real-time signal phase measurements of the target tag and each reference tag. The measurements are then used to compute the real-time measurement of the spatial relation between the target and reference tags. This results in a real-time measurement vector.

Online Target Tag Positioning: The real-time measurement is fed into a machine learning classifier to predict which subarea the target tag belongs to. We then use a sparse recovery algorithm to take in the real-time measurement and the fingerprint matrix of the predict subarea to estimate the location of the target object. See Section II-E.

B. Problem Formulation

Fig. 2 depicts an example deployment setup of our approach. In this example, there are in total N objects where each object is attached with an RFID tag. We also deploy M $(M \ll N)$ reference tags. Suppose that there are K target
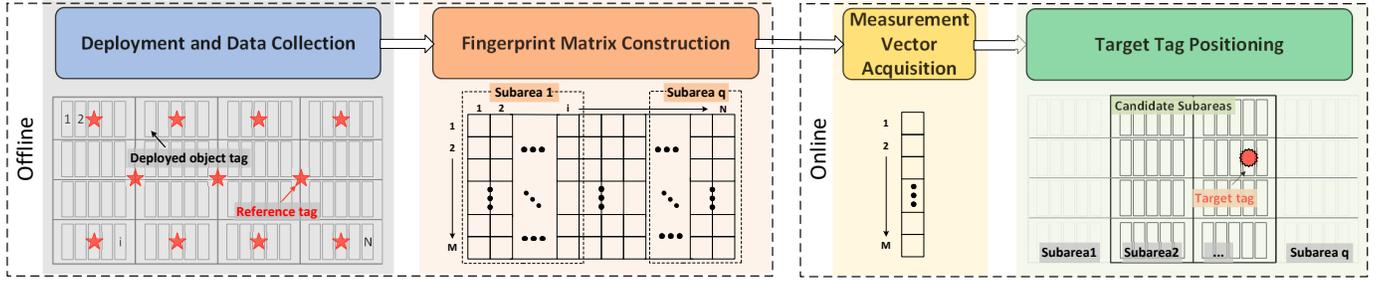
Fig. 1: Overview of CSRP. At the offline deployment phase, the object tags and a few number of reference tags are deployed in the entire region, and CSRP automatically calculates the fingerprint matrix. During online localization, CSRP obtains a measurement vector, predicting which subarea the target tag belongs to, and then using the measurement together with the fingerprint matrix to derive a location vector that indicates the target's location.
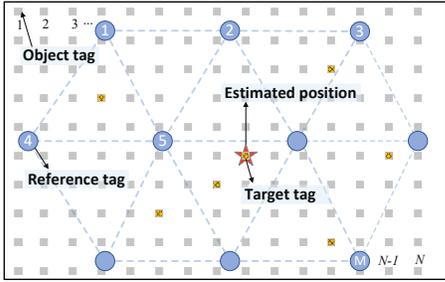

Fig. 2: An example CSRP deployment setup with N object tags and M reference tags. Unlike [6], the number of reference tags is signifantly less than the total number of object tags, i.e., $M \ll N$.

tags that we want to locate. These target tags are randomly distributed among N objects. We use a location vector Φ to represent the positions of K tags as:

$$\boldsymbol{\phi} = [\phi(1), \cdots, \phi(j), \cdots \phi(N)]^T, \quad (1)$$

where $\phi(j) \in \{0, 1\}$, and when the tag is in the position j, $\phi(j) = 1$, otherwise, $\phi(j) = 0$. We want to highlight that the number of target tags is equal to the number of non-zero elements in Φ, i.e., $K = \sum_{j=1}^{N} \phi(j)$. If the number of target tags, K, we want to locate at a given moment is significantly smaller than the total number of object tags, N, then the location vector, Φ, will be a sparse vector. Based on the sparse property of Φ and according to the CS theory in sparse recovery [13], when the number of target tags equals 1 (i.e., we want to locate an object at a time), the target tag positioning can be formulated as:

$$\mathbf{y}_{M \times 1} = \mathrm{X}_{M \times N} \cdot \boldsymbol{\phi}_{N \times 1} + \mathcal{N}, \quad (2)$$

where $\mathbf{y}_{M \times 1}$ is the real-time measurement vector, and $\mathrm{X}_{M \times N}$ is the fingerprint matrix, $\mathcal{N}$ is the measurement noise. According to prior works [11]–[14], when the fingerprint matrix satisfies the restricted isometry property (RIP), the location vector can be accurately recovered by solving Equation (2).

Our approach can also be used to solve a multiple-target positioing problem (i.e. when K is greater than 1). Because the readings of different tags can be separated using their IDs (which are given to the RFID reader), we can estimate the

position of each tag one by one. Specifically, each tag would have its own real-time measurement vector, $\mathbf{y}$, and a generated location vector, $\boldsymbol{\phi}$. Thus, Equation (2) can be re-defined as:

$$Y_{M \times K} = \mathrm{X}_{M \times N} \cdot \Phi_{N \times K} + \mathcal{N},$$
$$\mathrm{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_j, \cdots, \mathbf{x}_N], \mathbf{x}_j \in \mathbb{R}^{M \times 1},$$
$$Y = [\mathbf{y}_1, \cdots, \mathbf{y}_j, \cdots \mathbf{y}_K], \mathbf{y}_j \in \mathbb{R}^{N \times 1},$$
$$\Phi = [\boldsymbol{\phi}_1, \cdots, \boldsymbol{\phi}_j, \cdots \boldsymbol{\phi}_K], \boldsymbol{\phi}_j \in \mathbb{R}^{N \times 1}. \quad (3)$$

where $\mathbf{x}_j$ is the fingerprint of j-th target tag, $\mathbf{y}_j$ and $\boldsymbol{\phi}_j$ are the measurement vector and location vector, respectively.

As shown in Equation (3), only measurements of M reference tags are needed to locate one target tag. In contrast, PinIt would have to collect the phase readings from at least N reference tags (i.e., the number of reference tags is equivalent to the total number of object tags) [6]. Therefore, CSRP reduces the number of measurements by $\frac{N-M}{N}$ ($M \ll N$) over PinIt. This reduction means that we can save the deployment cost and reduce the data conflict in large-scale deployment.

C. Fingerprint matrix construction

Like the common practice in RFID positioning [1], [5], [6], CSRP also use the signal fingerprint information of tags to locate a specific object. There are two types of signal measurments can be directly obtained from an object tags using a RFID reader, RSS and the phase. We choose to use the phase to construct the fingerprint because it is more robust to the multipath effect and NLOS when compared to RSS [6].

In this paper, we use the multipath profile [6] as a measurement feature for each tag. Formally, a multipath profile is a vector $P(\theta)$ which records the power of tag's received signal in the beam with the direction $\theta \in [0°, 180°]$. Given a reader with Q uniformly spaced antennas as shown in Fig. 3(a), the received signal will be projected by each antenna to obtain a narrow beam in the $\theta \in [0°, 180°]$ direction, and the power received in the θ direction can be obtained by (see [6]):

$$P(\theta) = \left| \sum_{q=0}^{Q-1} w(q, \theta) \cdot s_q \right|^2, \quad (4)$$
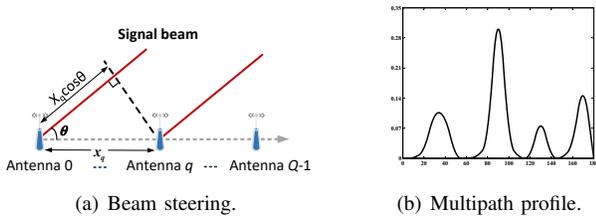$$w(q, \theta) = e^{-j \frac{2\pi}{\lambda} \cdot x_q \cos \theta}, \quad (5)$$

(a) Beam steering. (b) Multipath profile.

Fig. 3: Collecting the phase readings for feature extraction.

where s_q is the tag's signal observed from the qth antenna, $w(q,\theta)$ is the weight to s_q when steering the beam to θ, λ is the wave length and x_q is the position of the kth antenna.

Traditional fingerprint methods suffer from low accuracies due to subtle environmental changes. In this work, we design a novel fingerprint to avoid the issue of environmental noise. Specifically, we compute the spatial relation of each object tag with the reference tags. Because the spatial relationship keeps unchanged as long as the deployment pattern is fixed, thus we use the spatial similarities of each deployed tag with the reference tags as a new fingerprint. We use the dynamic time warping (DTW) distance [15], a widely used similarity metric for time series, to compute the spatial similarity, and the fingerprint for the object tag j can be represented as a column vector:

$$\mathbf{x}_j = [x_{1j}, \cdots, x_{ij}, \cdots, x_{Mj}]^T,$$
$$x_{ij} = \|P_j(\theta) - P_{r_i}(\theta)\|_{DTW}, \tag{6}$$

where $P_j(\theta)$ and $P_{r_i}(\theta)$ are the multipath profiles of the j-th object tag and the ith reference tag, $\|\cdot\|_{DTW}$ quantifies the similarity of multipath profiles between them [6]. By collecting the multipath profiles of N object tags, the fingerprint matrix can be constructed.

Online measurement vector acquisition. To locate a specific tag, we collect the phase readings of the target tag and the reference tags within a region. This is done using Equation (6). Here, the real-time measurement vector $\mathbf{y}_j$ for target tag j is defined as:

$$\mathbf{y}_j = [y_j(1), \cdots, y_j(i), \cdots, y_j(M)]^T,$$
$$y_j(i) = \|P_j(\theta) - P_{r_i}(\theta)\|_{DTW}, \tag{7}$$

where $y_j(i)$ is the DTW distance between multipath profiles of the target tag and the ith reference tag.

There are a number of reasons that we set the fingerprint matrix X as the form described in Equation (6). First, since the number of reference tags is much smaller than the object tags, by using the multipath profile distance between each location and M reference tags as the fingerprint, the row number for the fingerprint matrix M is much smaller than the number of possible positions N. Then the location vector ϕ_j can be recovered by only collecting M measurements for the vector $\mathbf{y}_j$. Second, the measurement vector $\mathbf{y}_j$ is the inner product of the location vector ϕ_j and the row vector $\mathbf{x}_i$ (i^{th} row of fingerprint matrix), thus $\mathbf{x}_i$ should include all the multipath profile distances between the reference tag

i and N different locations. Third, based on the rigorous proof in Section II-E4, the constructed fingerprint matrix in (6) obeys the RIP, which is the sufficient condition to conduct CS recovery. The detailed process about solving the location vector in (3) will be presented in Section II-E.

D. Offline Reference Tag Deployment

In this subsection, we will introduce the principles about the reference tags' deployment. We present two theorems that provide guidance on how to deploy reference tags to achieve a good performance in a cost-effective manners. We want to answer two specific questions: "how many reference tags are needed?" and "how should they be deployed?".

1) How many reference tags are needed?: In a practical deployment, the number of target tags K to look for is known. According to the CS theory, the number of measurements needed to accurately recover an K-sparse location vector, ϕ, should at least $O(K\log(N/K))$. Therefore, the number of reference tags needed to recover the K-sparse location matrix, Φ, should satisfy the term, $M > O(K\log(N/K))$. Because $O(K\log(N/K)) < M \ll N$, the number of reference tags would be much smaller than the number of object tags. Compared to the state-of-the-art work PinIt, which needs to deploy at least N number of reference tags, CSRP significantly reduces the deployment cost. Besides, the package collision among tags would decrease, as shown in Section IV-D.

2) How to deploy reference tags?: The goal of CSRP is to maximize the positioning accuracy under a given budget of reference tags. This requires us to maximize the distinction between different elements of the constructed fingerprint matrix. Specifically, this require us to satisfy two criterions. First, the distance between any two neighboring reference tags should be maximized. For the fingerprint matrix, neighboring elements in one column represent the difference of one object tag to two neighboring reference tags. That is to say the neighboring rows in the fingerprint matrix should have the maximum dissimilarity. Secondly, the distance between each tag and its nearest reference tag should be minimized. For the fingerprint matrix, there exists a minimum for each column and the positions of these minimum in different columns are different. As a result, the neighboring columns in the fingerprint matrix can reach the maximum dissimilarity.

By making sure our fingerprint matrix meets the two criterions described above, we can make the neighboring elements along both the rows and columns as different as possible. Hereafter, we have two theorems to prove that when the locations of reference tags form a mesh of equilateral triangles, the above two requirements can be satisfied. The detailed proofs are similar to the one presented in [16] which is used to choose the measurement points.

Theorem 1: The distances between each pair of reference tags can be maximized when they are at the intersections of a mesh of equilateral triangles.

Proof 1: Given the number of reference tags and the deployment area, the theorem can be proved through its opposite proposition. We just need to prove that when the distance of
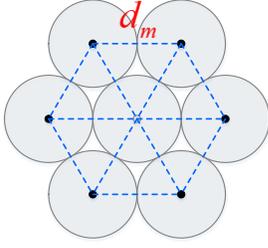
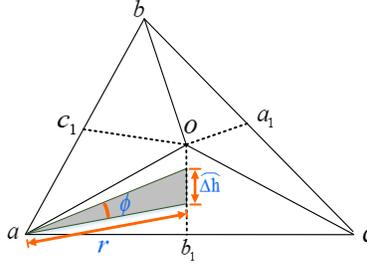Fig. 4: Deployment of theorem 1.
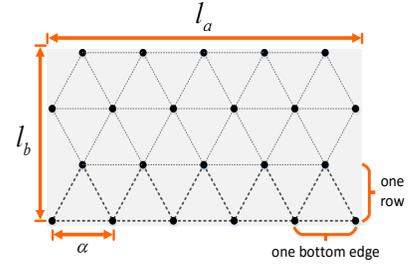

Fig. 5: Deployment of theorem 2.

Fig. 6: Deployment of triangles.

neighboring reference tags is fixed, the deployment area can be minimized when the reference tags are deployed at the intersections of a mesh of equilateral triangles. Note that when the reference tags are evenly deployed, the area can be divided into acute or right neighboring triangles[1].

Suppose that the distance between neighboring reference tags is d_m, and we draw circles centering on each reference tags with radius $d_m/2$ and there is no overlap between them, as shown in Fig. 4. According to the Thue's Theorem, the ratio between the total area of these triangles S_n and the area of the deployment region S is restricted to $\pi/\sqrt{12}$ as:

$$S_m/S = 2M\pi(0.5d_m)^2/S \le \pi/\sqrt{12}, \qquad (8)$$
$$S \ge \sqrt{3}Md_m^2, \qquad (9)$$

and when the edge length of the triangles are equivalent, S achieves its minimum value with $\sqrt{3}Md_m^2$. This prove that the deployment of reference tags with equilateral triangles maximize the distance between reference tags.

Theorem 2: When the deployment of reference tags are a mesh of equilateral triangles, the distance between each tag and its nearest reference tag can be minimized.

Proof 2: Since the area can be divided into triangles, each tag can be treated as uniformly distributed in one of the triangles. Suppose the tag p is located in $\triangle abc$, as shown in Fig. 5, oa_1, ob_1 and oc_1 are three midperpendiculars of edge bc, ac and ab, respectively. Thus the nearest reference tag to p is a, b or c if $p \in \square ab_1oc_1$, $\square ba_1oc_1$ or $\square ca_1ob_1$, respectively.

To proof the above theorem, we first compute the expected distance between p and it's nearest reference tag, let E_{pa} denotes the expected distance between p and a in $\triangle aob_1$, $S_{\widehat{\Delta h}}$ and $S_{a\widehat{\Delta h}}$ denote the area of radian $\widehat{\Delta h}$ and the sector $a\widehat{\Delta h}$:

$$
\begin{aligned}
E_{pa} &= \frac{1}{|ob_1|} \int_0^{|ob_1|} \frac{S_{a\widehat{\Delta h}}}{S_{\widehat{\Delta h}}} d\widehat{\Delta h} \\
&= \frac{1}{|ob_1|} \int_0^{|ob_1|} \frac{\int_0^\phi \int_0^r x^2 d\Delta r d\Delta\phi}{r^2\phi/2} d\widehat{\Delta h} \\
&= \frac{|oa|}{3} + \frac{|oa|\sin^2\angle aob_1}{3\cos\angle aob_1} \ln\left(\cot\frac{\angle aob_1}{2}\right).
\end{aligned} \qquad (10)
$$

[1]The evenly deployed reference tags are bound to form a mesh of parallelograms, and any parallelogram can be divided into two acute or right triangles.

In the same way, the expected distance between p and b in $\triangle boc_1$, p and c in $\triangle coa_1$ can be obtained. Since $S_{\triangle aob_1} = S_{\triangle aoc_1}$, $S_{\triangle boc_1} = S_{\triangle boa_1}$, $S_{\triangle coa_1} = S_{\triangle cob_1}$ and $|oa| = |ob| = |oc|$, the expected distance between p and its nearest reference tag in $\triangle abc$ can be obtained by:

$$E_p = \frac{2(E_{pa} \cdot S_{\triangle aob_1} + E_{pb} \cdot S_{\triangle boc_1} + E_{pc} \cdot S_{\triangle coa_1})}{S_{\triangle abc}}, \quad (11)$$

as $S_{\triangle aob_1} = |oa|^2 \sin 2\angle aob_1/4$, $S_{\triangle boc_1} = |oc|^2 \sin 2\angle boc_1/4$, $S_{\triangle coa_1} = |oc|^2 \sin 2\angle coa_1/4$, (11) becomes:

$$
\begin{aligned}
E_p &= \frac{1}{3}\sqrt{\frac{2S_{\triangle abc}}{\sin 2\angle aob_1 + \sin 2\angle boc_1 + \sin 2\angle coa_1}} \\
&+ \frac{2}{3}\sqrt{\frac{2S_{\triangle abc}}{(\sin 2\angle aob_1 + \sin 2\angle boc_1 + \sin 2\angle coa_1)^3}} \cdot \\
&\left(\sin^3\angle aob_1 \ln(\cot\frac{\angle aob_1}{2}) + \sin^3\angle boc_1 \ln(\cot\frac{\angle boc_1}{2})\right. \\
&+ \left.\sin^3\angle coa_1 \ln(\cot\frac{\angle coa_1}{2})\right).
\end{aligned} \qquad (12)
$$

Since $S_{\triangle abc}$ is fixed, E_p can be minimized if and only if $\angle aob_1 = \angle boc_1 = \angle coa_1 = \pi/3$. Finally, the expected distance is minimized.

Furthermore, we introduce how to choose the edge length of the triangles to accurately recover the location vector. As shown in Fig. 6, suppose a rectangular region with area size $l_a \times l_b$ (length l_a and width l_b). Suppose the region is divided into γ rows of equilateral triangles, and the maximum number of bottom edge for one row is β. Let α represents the edge length of one triangle. The segmentation goal is to make the total area of triangles coverage the region as much as possible, meanwhile satisfies the following three conditions: (a) the number of triangles vertexes is equal to the number of reference tags and $M > O(K \log(N/K))$; (b) the total length of bottom edges for β triangles should smaller than the region length; and (c) the total length of triangle heights for γ rows should smaller than the region width.

The edge length α for each triangle can be obtained by solving a linear programming problem described as follows:

$$
\max \quad \beta \times 2\gamma \times \frac{\sqrt{3}}{4}\alpha^2,
\begin{cases}
(\beta+1) \times (\gamma+1) \le M, \\
\beta \times \alpha \le l_a, \\
\gamma \times \frac{\sqrt{3}}{2}\alpha \le w_b.
\end{cases} \qquad (13)
$$

E. Target Tag Positioning

We first present the detailed positioning process. Then we prove that the fingerprint matrix satisfies the restricted isometry property (RIP) with a high probability.

1) Locating the target tags: To estimate the position of the target tags in a large-scale case, a recovery algorithm with low complexity and high accuracy is needed. If we directly apply the sparse recovery algorithm on the fingerprint matrix, it will cause high computation cost. This problem will be exacerbated when the number of deployed object tags increases. Besides, we prefer to provide a contextual positioning result such that "firstly in which shelf, further in the second rack, and lastly the exact position" rather than a general answer like "the fifth or fifteenth position.". Considering these reasons, we propose a hierarchical algorithm which consists of two steps.

In the first step, we divide the fingerprint matrix into sub-matrixes corresponding to different subareas (shelves). We employ the marginal fisher nalysis (MFA) [17] to the fingerprint matrix to train a classifier. The target tag is first positioned to two candidate shelves in this step. We have also explored several alternative classification techniques. This is discussed in Section IV-G.

In the second step, we apply the sparse recovery algorithm on the candidate sub-matrixes to recover the location vectors for multiple target tags. Each target tag's real-time measurement vector is used to recover the location vector. To get a high accuracy, we utilize the popular orthogonal matching pursuit (OMP) algorithm which is robust to errors and noise [18].

2) Step 1: Predicting the target tag's subarea: MFA is a widely used supervised classification algorithm. Specifically, the samples are projected into a subspace, where the samples in the same class are made as compacted as possible, and the different classes are made as far as possible. To illustrate the intraclass compactness, MFA constructs an intrinsic graph where each sample is connected to its k_1 nearest neighbors in the same class. And a penalty graph is constructed to describe the adjacency relationship between interclass marginal samples [19]. Through minimizing the ratio of the intrinsic graph and penalty graph, each target tag can be first pointed into a specific subarea.

Specifically, we term each fingerprint (column vector) as a sample $\mathbf{x}$, and E_{ij}^{IN} represents the weight of sample $\mathbf{x}_i$ and $\mathbf{x}_j$, which in the same class. $E_{ij}^{IN} = 1$, if $\mathbf{x}_j$ is a k_1-nearest neighbor of $\mathbf{x}_i$ or vice versa; otherwise, $E_{ij}^{IN} = 0$. In a same way, the weight between different class samples $\mathbf{x}_i$ and $\mathbf{x}_j$ is $E_{ij}^P = 1$, if $\mathbf{x}_j$ is a k_2-nearest neighbor of $\mathbf{x}_i$ or vice versa; otherwise, $E_{ij}^P = 0$. And if the number of subareas is L, the MFA subspace minimizes:

$$
\begin{aligned}
Y(W) &= \frac{\sum_{i=1}^{L}\sum_{j=1}^{L}(W^T\mathbf{x}_i - W^T\mathbf{x}_j)^T(W^T\mathbf{x}_i - W^T\mathbf{x}_j)E_{ij}^{IN}}{\sum_{i=1}^{L}\sum_{j=1}^{L}(W^T\mathbf{x}_i - W^T\mathbf{x}_j)^T(W^T\mathbf{x}_i - W^T\mathbf{x}_j)E_{ij}^P} \\
&= \frac{tr\left(W^T X\left(D^{IN} - E^{IN}\right)W^T X\right)}{tr\left(W^T X\left(D^P - E^P\right)W^T X\right)}
\end{aligned}
\tag{14}
$$

where D^{IN} and D^P are diagonal matrices with the ith entry $D_{ii}^{\mathrm{IN}} = \sum_{j=1}^{N} E_{ij}^{IN}$ and $D_{ii}^{\mathrm{P}} = \sum_{j=1}^{N} E_{ij}^P$.

$$
W = \arg\min_W Y(W),
\tag{15}
$$

and $Y(W)$ is the direction of linear project. To ensure the following positioning accuracy, each time we choose two candidate subareas for further processing in the experiment.

3) Step 2: Positioning using the sparse recovery algorithm: In this step, the measurement vector of each target tag together with the candidate sub-matrixes are used to recover the location vector. Due to the data lost and noise during the signal propagating, the measurement vector for one tag will mismatch the fingerprint. The large-scale deployment will exacerbate this problem. We thus leverage the robust and simple recovery algorithm OMP to accurately positioning the target tags:

$$
\begin{aligned}
&\min \|\hat{\Phi}\|_2, \\
&s.t. \ \|X^\dagger(Y - X\hat{\Phi})\|_2 < \delta.
\end{aligned}
\tag{16}
$$

It has been proved that if the fingerprint matrix X satisfies the RIP and the dimension of measurement vector $\mathbf{y}_k$ for the k-th target tag obeys $M = O(K\log(N))$, the location vector ϕ_k can be accurately recovered by the OMP algorithm [18].

Complexity analysis. Now we discuss the computation complexity of our recovery algorithm. In fact, the complexities of the MFA algorithm and OMP algorithm are $O(L^2)$ and $O(KMN)$, respectively. By using our hierarchical algorithm, the complexity is $O(L^2 + \frac{KMN}{L})$. Note that the number of subareas is usually smaller than the number of reference tags, that is $L < M$ and $M \leq N$, thus the complexity of our algorithm is about $O(\frac{KMN}{L})$. When compared with the case of solely applying OMP algorithm which has a complexity of $O(KMN)$, our algorithm reduces the complexity by L times.

4) Meeting the RIP constraint: We now prsent the theoretical analysis, showing that the fingerprint matrix X is highly likely to satisfy the RIP constraint.

Theorem 3: When the number of reference tags satisfies $M = O(K\log N)$ and each row $\mathbf{x}^i$ ($i \in [1, M]$) in the fingerprint matrix is independent identically distributed (i.i.d.), the probability for X to satisfy:

$$
(1-\delta) \leq \|X\Phi\|_2^2/\|\Phi\|_2^2 \leq (1+\delta).
\tag{17}
$$

for all N-dimensional K-sparse matrix Φ tends to 1, i.e., X obeys RIP, where $\delta \in (0, 1)$.

III. Experimental Setup

A. Hardware

Our system is built upon omnidirectional antennas and commercial UHF RFIDs, and we use a commercial off-the-shelf contactless RFID reader to track tags.

RFID Reader: We use an ImpinJ Speedway RFID reader (modeled R420) [20] during test. This reader is compatible with the EPC Gen2 standard [21], operating in a frequency range of 920.5–924.5 MHz. We did not modify any hardware component or firmware of the reader.
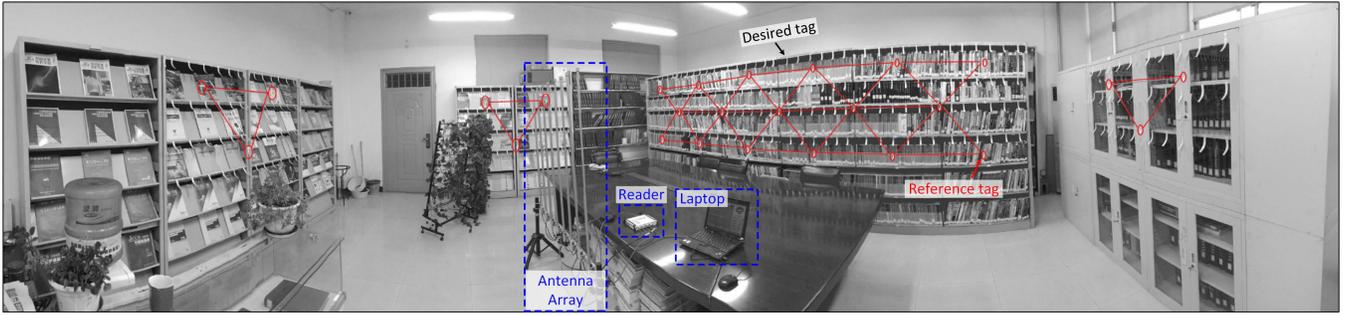
Fig. 7: Experiment deployment layout. The antenna array is deployed at the middle region of the library room. We choose 12 racks to deploy object tags and the reference tags. There are in total 270 object tags and 26 reference tags deployed.

Antenna Array: We deployed eight Q900F-900 omnidirectional antennas [22] to form a linear antenna array with a 16 cm half-wavelength space between adjacent antennas. The antenna provides a communicate range of around 12 meters even when the RFID devices backscatter a week signal.

Tags: We use the Alien Squiggle General Purpose UHF RFIDs as object and reference tags. These are passive backscatter RFIDs working in the UHF band, but other passive RFIDs can also be used. Furthermore, reference tags are placed at the selected reference positions (found using the heuristic described in Section II-D), which are expected to remain unchange throughout their lifetime.

B. Evaluation Scenarios

Evaluation Environment: We evaluate our approach in an indoor library room with rich multipath and `NLOS`. The reference tags and object tags are deployed on 12 racks (46 book shelves), as shown in Fig. 7. The racks are made of mental and wood. The reference tags are set at the vertex of the equivalent triangles to cover the whole test region. Note that we did not put any tags outside of the shelves, so there are no tags except for the reference and object tags. Each wall of the indoor room is attached with shelves and each shelf is full of books, which result in a complex multipath environment.

Offline Data Collection: To construct the fingerprint matrix, we first extract the phase by using the RFID reader to read data of each tag to measure its multipath profile. Next, we calculate the `DTW` values between the multipath profiles of each object tag and reference tags.

Competitive Approach: We compare our approach against PinIt, a state-of-the-art RFID-based positioning system [6]. We faithfully repeated PinIt's setup in our deployment site, by evenly distrributing reference tags between objects. In total, we have placed 285 reference tags to be used by PinIt. To locate an object tag, PinIt selects a reference tag (whose location is known ahead of time) which has a multipath profile that is most similar to the target tag's, and then uses the selected reference tag to estimate the target tag's location.

Evaluation Methods: We have conducted extensive experiments to evaluate the performance of CSRP, and compare

it with PinIt. We also evaluate the deployment cost and data collisions of both approaches, and analyze the working mechanism of our approach. To provide a fair comparison, we have used all deployed objects as test items in our experiments.

IV. EXPERIMENTAL RESULTS

In this section, we first present the overall accuracy of CSRP, showing that it achieves a similar accuracy as the state-of-the-art. Next, we evaluate the effectiveness of our sparse recovery algorithm, and how does the number of reference tags affects positioning accuracy, data collisions, and deployment cost. We then show that CSRP works effectively in `LOS` and `NLOS` environments, before discussing the alternative modeling techniques for subarea predictions.

A. Overall Performance

From Fig. 8, both CSRP and PinIt achieve a good positioning performance. CSRP and PinIt are able to correctly identify the location id for 84.7% and 86.4% of the test items respectively. For 95% of the object tags, the average location accuracy of CSRP is within 7cm. This is slightly better than the average accuracy of 8.5cm given by PinIt. This improved accuracy is largely due to the fact the CSRP uses significantly less reference tags (26 vs 285) and the spatial relation to cancel the environmental noise, which in combination reduces the negative impact of data collisions. Overall, CSRP delivers a similar positioning performance when compared with PinIt, but uses an order of magnitude less reference tags.

B. Comparison of Positioning Recovery Methods

In this experiment, we compare our sparse recovery algorithm against two alternatives: a standard `CS` approach and the `DTW`-based matching method used by PinIt. In this experiment, we use the 26 reference tags deployed under our scheme for positioning.

The Cumulative Distribution Function (CDF) chart in Fig. 9 describes how positioning accuracy for each method changes as the accepted location error changes. If we consider a location error of within 10cm as an accepted criterion, then CSRP gives a average accuracy of 87.9%. Under this assessment criteria, the traditional `CS` method and PinIt achieves an accuracy of 74.9% and 17%, respectively. Clearly, CSRP
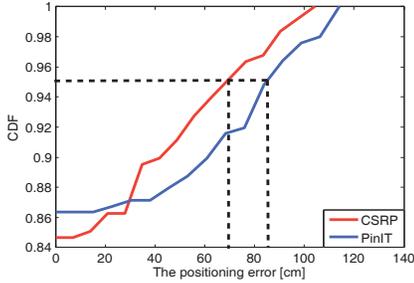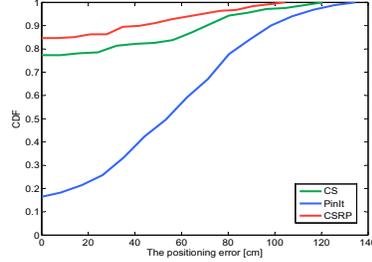
Fig. 8: Positioning accuracy comparison.

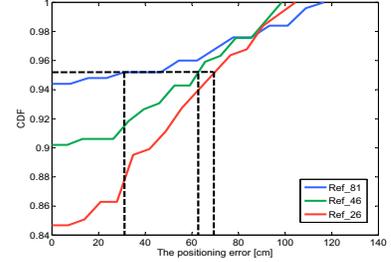Fig. 9: Performance of different positioning methods.

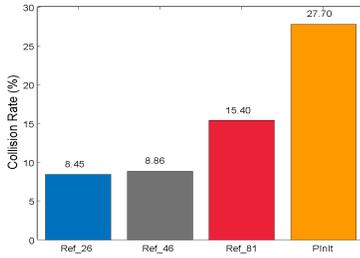Fig. 10: Impact of number of reference tags.
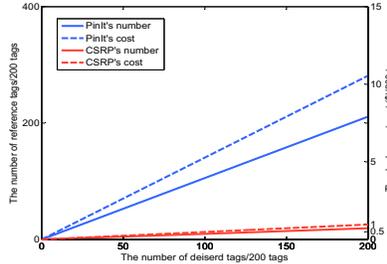

Fig. 11: Tag collision comparison.
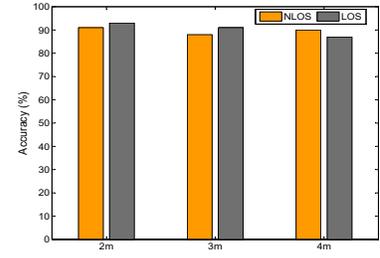

Fig. 12: Deployment cost comparison.

Fig. 13: Impact of different environments.

outperforms both approaches. If we relax the assessment criteria to allow the positioning error to be less than 80cm, the sparse recovery algorithm still remains the best method among the three approaches.

We can also see from the diagram that CS-based methods clearly outperforms PinIt. This is large because PinIt needs a large number of reference tags to work efficiently. For the CS method, owning to its high robustness against the environment noise, it achieves a reasonably good performance. Our approach enhances the standard CS scheme by first predicting which of the subareas the target tag belongs to. This prediction helps us in removing most of the outliers from the candidate locations, leading to better performance over the CS-based approach. Note that there is a huge accuracy decline for PinIt with respect to the accuracy of 13 cm reported in [6].

C. Impact of The Number of Reference Tags

In this experiment, we deploy 26, 46 and 81 reference tags in the our evaluation scene and test how does the number of reference tags affect the positioning accuracy of CSRP. Fig. 10 shows the corresponding positioning errors under different reference tag configurations. When using 26 reference tags, 84.7% of the target tags are positioned in the correct positions. While for the cases of 46 and 81 reference tags, the number of correctly positioned target tags are 90.2% and 94.3%, respectively. For 95% of the testing tags, the positioning error for the configurations of 26, 46 and 81 reference tags are within a distance of 7 cm, 6.3 cm and 3.2 cm, respectively. Using more reference tags indeed can improve the positioning accuracy, but the number of reference tags used by our approach is still considerably smaller than PinIt.

D. Data Collisions

In order to quantify the data collision between tags (both the reference tags and the object tags) under four different deployment cases, we calculate the collected data rate ω within a time period to characterize the collision probability. Specifically, ω can be written as: $\omega = \frac{S_{all} - S_{rec}}{S_{all}}$, where S_{all} is the sum of the number of deployment tags and S_{rec} is the total number of tags received by the reader, S_{all} and S_{rec} both contain the data of object tags and reference tags. Fig.11 shows that the collision probability increases with more tags, i.e., fewer number of reference tags leads to slighter tag collision when the number of object tags remains the same. In conclusion, CSRP can reduce the tag collision by deploying fewer and sparsely distributed reference tags.

E. Deployment Cost

To compare the deployment cost, we calculate the number of reference tags for CSRP and PinIt when the positioning area increases. According to the default setups, when the number of object tags increases to 200, only 48 extra reference tags are needed to implement CSRP. While for PinIt, 528 reference tags are needed. Fig.12 shows how does the deployment cost increase as the number of reference tags increase. In this example, we assume each tag costs $ 0.05. When the total number of object tags are 40,000, the number of reference tags and deployment cost for CSRP and PinIt are 3,800, $190 and 4,2400, $2,120, respectively. As a result, CSRP reduces 91.0% deployment cost compared with PinIt. This massive reduction also means that there will be less human involvement in installing and maintaining the reference tags

TABLE I: Classification accuracy per model

Classifier	Accuracy (%)	Classifier	Accuracy (%)
Naive Bayes	85.2	SVM	91.2
MLP	88.1	KNN	85.4
Decision Tree	89.2	ANN	89.6
MFA	100		

F. Performance of LOS and NLOS environments

To test the performance of our system in different environment, we place 11 reference tags and 150 object tags on four movable shelves. To mimic a LOS setting, we put the book shelves in an empty hall. For NLOS environment, we block the direct paths by setting the shelves in an office environment. As shown in Fig.13, we can infer that: i) CSRP achieves the positioning accuracy about 90% in both LOS and NLOS when the tags are spaced in a range of 2 m to 8 m. ii) No matter in LOS or NLOS scene, the accuracy of positioning has no regular changes when the distance between the tags and the receiver is increasing. Thus, we conclude that the accuracy of CSRP dose not affect obviously by distance.

G. Alternative Classification Methods for Subarea Prediction

Table I gives the sub-area prediction accuracy of various alternative classification techniques and our RF model. The alternative models that were built using the same features and training data. Due to the high-quality features, all classifiers deliver reasonably good prediction accuracy. We choose MFA because it gives the best prediction accuracy and is proven to be effective in modeling time series data [17].

V. RELATED WORK

In this section, we summarize the most related researches from the RFID positioning technologies.

RSS based methods. RSS reflects the received signal strength in the power level. Early systems [1], [4] locate a tag with the help of some anchor tags with know locations. Specifically, by comparing the RSS measurement of a target tag with a dense deployed reference tags, the location of target tag is estimated. To achieve the anchor-free localization purpose, work [23] tries to model the signal propagation in a complex environment. Work [3] combines the RSS and the tag's reading rate to design a tag order identification system. However, RSS is sensitive to multipath or non-line-of-sight paths environment [24], and it is highly related to the antenna gain and tag orientation. As a result, these systems have a limited localization accuracy [5].

Phase based methods. Phase is an indication of the distance that a wireless signal goes through. Phase based methods can be generally divided into two categories, AoA (Angle of Arrival) and SAR (synthetic aperture). AoA based work [8]–[10] locate the target tag by measuring the phase differences of different antennas. The major limitation for these methods is the poor performance in multipath environment. Work [25] track the trajectory of target tag using grating lobes, while the working distance is only tens of centimeters. To reduce the

hardware cost, work [26]–[29] utilizes SAR to simulate multiple antennas. Minsen *et al.* [28] employs a moving antenna to simulate the SAR and location the target by constructing a hologram. Similarly, PinIt [6] exploits the moving antenna and locates the target tag by comparing the multipath profile with a dense deployed reference tags. Benefiting from the SAR method, PinIt is able to locate the target tag with a sub-meter level accuracy even in the non-line-of-sight path and rich multipath environment. However, PinIt requires placing a large number of reference tags, which not only incurs high deployment cost, but also increases the chance of data collision between tags and leads to deleterious performance in a large-scale deployment.

Other methods. Besides the above methods, there is a number of related RFID localization work [5], [7], [30]–[32]. Liu *et al.* [30] introduces an hyperbolic localization method to associate the phase measurement and the target tag's location. Li *et al.* [31] leverages a multi-frequency based ranging method to locate the target tag. Tagoram [5] achieves a centimeter level tracking accuracy through the designed hologram which taking the thermal noise into consideration. Shangguan *et al.* [32] designs a tag ordering system by means of the spatial-temporal dynamics in the phase profiles. Further, a autonomous wheeled robot reader [7] is implemented achieves an exact spatial order in very close spacings. Although great progress has been made by these systems, it still remains a great challenge to realize a large-scale deployment.

Comparing to the existing works, CSRP is the first work designed for large-scale deployment scene. In accordance with the state-of-the-art work PinIt, CSRP employs the SAR method to capture the multipath profile. While, different from PinIt, CSRP achieves a comparable high accuracy with only a small number of reference tags deployed. On the other hand, the measurements needed to locate multiple tags in a large-scale deployment are greatly reduced benefiting from the CS theory.

VI. CONCLUSIONS

This paper presents CSRP, a novel, fine-grained RFID positioning system based on compressive sensing. CSRP works on commercial RFID hardware, aiming to greatly reduce the number of reference tags used in RFID-based positioning systems. CSRP utilizes the stable spatial relationship between the item tag and reference tags to construct a noise-resilient fingerprint matrix. With a compressive sensing based recovery algorithm, the fingerprint matrix is used in combination with the real-time signal measurement to localize the target item. CSRP offers analysis and methods for the optimal deployment of reference tags and optimization of computational overhead. We evaluated CSRP by applying it to a real-world deployment of 255 commercial RFID tags. Experimental results show that CSRP can successfully locate 84.7% of the test items, reaching a performance-level that is comparable to the one provided by the state-of-the-art, but with only 9% of the reference tags used by the state-of-the-art. The massive reduction in reference tags suggests that significant saving in deployment cost and time can be achieved in large-scale RFID deployment.

References

[1] 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.

[2] J. Hightower, G. Borriello, and R. Want, "Spoton: An indoor 3d location sensing technology based on rf signal strength," *Uw Cse*, 2000.

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

[4] Y. Zhao, Y. Liu, and L. M. Ni, "Vire: Active rfid-based localization using virtual reference elimination," in *ICPP*, p. 56, 2007.

[5] L. Yang, Y. Chen, X.-Y. Li, C. Xiao, M. Li, and Y. Liu, "Tagoram: real-time tracking of mobile rfid tags to high precision using cots devices," in *ACM MobiCom*, pp. 237–248, 2014.

[6] J. Wang and D. Katabi, "Dude, where's my card?: Rfid positioning that works with multipath and non-line of sight," in *ACM SIGCOMM Computer Communication Review*, vol. 43, pp. 51–62, 2013.

[7] L. Shangguan and K. Jamieson, "The design and implementation of a mobile rfid tag sorting robot," in *ACM MobiSys*, 2016.

[8] C. Hekimian-Williams, B. Grant, X. Liu, Z. Zhang, and P. Kumar, "Accurate localization of rfid tags using phase difference," in *IEEE RFID*, pp. 89 – 96, 2010.

[9] P. V. Nikitin, R. Martinez, S. Ramamurthy, H. Leland, G. Spiess, and K. V. S. Rao, "Phase based spatial identification of uhf rfid tags," in *IEEE RFID*, pp. 102–109, 2010.

[10] S. Azzouzi, M. Cremer, U. Dettmar, R. Kronberger, and T. Knie, "New measurement results for the localization of uhf rfid transponders using an angle of arrival (aoa) approach," in *IEEE RFID*, pp. 91–97, 2011.

[11] E. J. Candès and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 21–30, 2008.

[12] E. J. Candès, J. K. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Communications on Pure & Applied Mathematics*, vol. 59, no. 8, pp. 1207–1223, 2006.

[13] J. Wang, D. Fang, Z. Yang, H. Jiang, X. Chen, T. Xing, and L. Cai, "E-HIPA: An Energy-Efficient framework for High-Precision Multi-Target adaptive Device-Free localization," *IEEE Trans. on Mobile Computing*, vol. 12, no. 5, pp. 1–12, 2016.

[14] D. Boufounos, D. Liu, and P. T. Boufounos, "A lecture on compressive sensing," *IEEE Signal Processing Magazine*, pp. 118–121, 2007.

[15] S. Salvador and P. Chan, "Toward accurate dynamic time warping in linear time and space," *Intelligent Data Analysis*, vol. 11, no. 5, pp. 561–580, 2007.

[16] K. Sheng, Z. Gu, X. Mao, X. Tian, W. Wu, X. Gan, and X. Wang, "The collocation of measurement points in large open indoor environment," in *IEEE INFOCOM*, pp. 2488–2496, 2015.

[17] S. Yan, D. Xu, B. Zhang, and H.-J. Zhang, "Graph embedding: A general framework for dimensionality reduction," in *IEEE CVPR*, vol. 2, pp. 830–837, 2005.

[18] D. L. Donoho, Y. Tsaig, I. Drori, and J.-L. Starck, "Sparse solution of underdetermined systems of linear equations by stagewise orthogonal matching pursuit," *IEEE Trans. Information Theory*, vol. 58, no. 2, pp. 1094–1121, 2012.

[19] S. Si, D. Tao, and B. Geng, "Bregman divergence-based regularization for transfer subspace learning," *IEEE Trans. on Knowledge and Data Engineering*, vol. 22, no. 7, pp. 929–942, 2010.

[20] "Impinj, Inc." www.impinj.com/products/readers/speedway-revolution/.

[21] "Epc gen2, epcglobal." www.gs1.org/epcglobal.

[22] "Q900f-900 rfid antenna." www.hrtantenna.com/en/products.

[23] L. Yang, Y. Qi, J. Fang, X. Ding, T. Liu, and M. Li, "Frogeye: Perception of the slightest tag motion," in *IEEE INFOCOM*, pp. 2670–2678, 2014.

[24] J. D. Griffin and G. D. Durgin, "Complete link budgets for backscatter-radio and rfid systems," *IEEE Antennas and Propagation Magazine*, vol. 51, no. 2, pp. 11–25, 2009.

[25] J. Wang, D. Vasisht, and D. Katabi, "Rf-idraw: virtual touch screen in the air using rf signals," in *S3 Workshop on ACM MobiCom*, pp. 235–246, 2014.

[26] A. Parr, R. Miesen, and M. Vossiek, "Inverse sar approach for localization of moving rfid tags," in *IEEE RFID*, pp. 104–109, 2013.

[27] J. Wang, F. Adib, R. Knepper, D. Katabi, and D. Rus, "Rf-compass: Robot object manipulation using rfids," *Association for Computing Machinery*, pp. 3–14, 2013.

[28] R. Miesen, F. Kirsch, and M. Vossiek, "Holographic localization of passive uhf rfid transponders," in *IEEE RFID*, pp. 32–37, 2011.

[29] A. Parr, R. Miesen, F. Kirsch, and M. Vossiek, "A novel method for uhf rfid tag tracking based on acceleration data," in *IEEE RFID*, pp. 110–115, 2012.

[30] T. Liu, L. Yang, Q. Lin, Y. Guo, and Y. Liu, "Anchor-free backscatter positioning for rfid tags with high accuracy," in *IEEE INFOCOM*, pp. 379 – 387, 2014.

[31] X. Li, Y. Zhang, and M. G. Amin, "Multifrequency-based range estimation of rfid tags," in *IEEE RFID*, pp. 147–154, 2009.

[32] L. Shangguan, Z. Yang, A. X. Liu, Z. Zhou, and Y. Liu, "Relative localization of rfid tags using spatial-temporal phase profiling," in *USENIX NSDI*, pp. 251–263, 2015.