This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2867578, IEEE Access

IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2018.DOI

Towards System Implementation and Data Analysis for Crowdsensing Based Outdoor RSS Maps

**XIAOCHEN FAN[1], XIANGJIAN HE[1], CHAOCAN XIANG[23], DEEPAK PUTHAL[1]
LIANGYI GONG[4], PRIYADARSI NANDA[1], AND GENGFA FANG[1]**
[1]School of Electrical and Data Engineering, University of Technology Sydney, Ultimo, NSW 2007, Australia
[2]College of Computer Science, Chongqing University, Chongqing, China
[3]Army Logistics University of PLA, Chongqing, China
[4]School of Computer Science and Engineering, Tianjin University of Technology, Tianjin, China

Corresponding author: Chaocan Xiang, email: Xiang.Chaocan@gmail.com.

The preliminary version of this article was published in the 14th IEEE International Symposium on Parallel and Distributed Processing with Applications (ISPA'16), August 23-26, Tianjin, China, 2016.

ABSTRACT

With the explosive usage of smart mobile devices, sustainable access to wireless networks (e.g., WiFi) has become a pervasive demand. Most mobile users expect seamless network connection with low cost. Indeed, this can be achieved by using an accurate received signal strength (RSS) map of wireless access points. While existing methods are either costly or unscalable, the recently emerged mobile crowdsensing (MCS) paradigm is a promising technique for building RSS maps. MCS applications leverage pervasive mobile devices to collaboratively collect data. However, the heterogeneity of devices and the mobility of users could cause inherent noises and blank spots in collected dataset. In this paper, we study (1) how to tame the sensing noises from heterogenous mobile devices, and (2) how to construct accurate and complete RSS maps with random mobility of crowdsensing participants. First, we build a mobile crowdsensing system called iMap to collect RSS measurements with heterogeneous mobile devices. Second, through observing experimental results, we build statistical models of sensing noises and derive different parameters for each kind of mobile device. Third, we present the signal transmission model with measurement error model, and we propose a novel signal recovery scheme to construct accurate and complete RSS maps. The evaluation results show that the proposed method can achieve 90% and 95% recovery rate in geographic coordinate system and polar coordinate system, respectively.

INDEX TERMS RSS map; Crowdsensing; Wireless access points.

I. INTRODUCTION

With the proliferation of smart mobile devices, mobile crowdsensing has become a promising paradigm. Mobile users can exploit their smartphones to cooperatively perform large-scale sensing tasks [1]. Based on mobile crowdsensing, both industry and academia have developed numerous novel applications [2], such as traffic monitoring [3], [4], route planning [5], [6], air quality sensing [7], [8], localization [9], [10] and digital map construction [11]–[13].

Nevertheless, the above crowdsensing applications usually require high network bandwidth for data transmission. In terms of cost and efficiency, WiFi networks enable the compute-intensive applications to provide more reliable computing services for mobile users. For instance, the public WiFi access points (APs) have been pervasively deployed in metropolises, especially in indoor environment (*e.g.*, apartments, shopping centers, airports, *etc.*) [14]. In contrast, the Quality of Service (QoS) of outdoor WiFi network is difficult to quantify [15].

The major concerns of outdoor access points are signal coverage and transmission capacity [16]. These information can be obtained from the Received Signal Strength (RSS). However, it is non-trivial to collect complete RSS data in large areas, and many researchers have put their efforts to

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2867578, IEEE Access

achieve it. Ayon *et al.* [17] proposed SpecSense, a platform for large scale spectrum monitoring. Similarly, Wu *et al.* [18] presented CrowdWiFi, a vehicular crowdsensing system for looking up roadside WiFi networks. In [19], the authors proposed CRAD, a crowdsensing based approach to detect rogue APs. The above works focus more on data collection rather than the accuracy and reliability of raw crowdsensed RSS data.

However, our experimental results show that the accuracy and reliability of RSS data can be seriously influenced by the mobility pattern of users and the heterogeneity of devices. Even on the same observing spot, the RSS measurements from different devices can always have mismatches or misalignments. This is due to the differential capabilities of mobile devices in sensing signals. To address this issue, in [20], an Expectation Maximization based mechanism is proposed to compute the maximum likelihood estimation of sensor noises. Furthermore, Xiang *et al.* [21] proposed an iterative algorithm to reduce the error-rate of crowdsensed RSS data. Moreover, Kim *et al.* presented a mobile crowdsensing framework for large-scale WiFi fingerprinting system [22], using physical-layout and signal-strength measurements.

Nevertheless, assume that the noises in RSS data can be reduced or even eliminated, it is still difficult to construct a complete RSS map. Because the crowdsensed data is usually incomplete and can not cover every spot on the map. Due to the different trajectories of mobile users, there are blank spots without any data in RSS maps. These blank spots are especially challenging in constructing large-scale RSS maps, as the sensing cost is in proportion to the crowdsensing coverage [23]. Wang *et al.* [24] discussed the above issues and designed a general framework for sparse mobile crowdsensing applications. In [25], the authors proposed a crowdsensing based WiFi radio map construction mechanism for mobile users to choose appropriate access points. Wu *et al.* proposed PRESM, a privacy preserving RSS map generation scheme for crowdsensing networks [26]. However, the above works neglect the importance of data quality in building WiFi related digital maps. Therefore, how to construct accurate and complete RSS maps for outdoor APs remains challenging.

Here, two challenges need to be formally addressed.

- First, how to build accurate RSS maps with unpredictable noises in crowdsensed data. In our experiments, the difference of collected RSS data between two smartphones could be up to 40 dBm. Such noises, either from the sensing errors or malicious measurements, should all be fairly tamed for constructing accurate RSS maps.
- Second, how to construct complete RSS maps with the missing data in blank spots. In practical crowdsensing, it is hard to fully cover the target area considering the random mobility of mobile users as well as the overall cost. The RSS maps need to be accurately constructed with incomplete sensing data.

In this paper, we propose iMap, towards system implementation and data analysis for crowdsensing based outdoor RSS maps. Our system enables mobile users to use their sensor-embedded smartphones to collaboratively collect RSS data in the wild. Through systematically analyzing the collected RSS measurements from heterogenous devices, we have found the following facts.

- First, although the noise is inevitable, for each type of smartphones, the crowdsensed data could fit into a statistical model fairly well. Moreover, the variances between two different types of smartphones could roughly fit into a specific linear model. In addition, we recruit a group of volunteers to collect RSS data with three different types of smartphones. The experimental results validate the above claims. Hence, we can leverage the features of RSS data to estimate the data quality.
- Second, even the crowdsensed RSS data is incomplete, we can still form a sufficiently sparse matrix on it. By leveraging compressive sensing methods, we can sample the sparse matrix to adaptively recover the data on unsensed spots.

Based on the above observations, in this work, we apply a model-based mechanism to reduce errors and noises in RSS data. With more reliable data, we can further apply an adaptive sparse sampling algorithm to recover RSS data and build complete RSS maps. The major contributions of this paper are listed as follows.

- To the best of our knowledge, iMap is the first crowdsensing system for constructing accurate and complete RSS maps in the wild. We develop an application for mobile users to collect RSS measurements. Meanwhile, we use a cloud-based central sever for RSS data storage and processing.
- We conduct real-world experiments and analyze the RSS data from diverse aspects. Based on the experimental results, we investigate the error models for heterogenous smartphones.
- We propose a compressive sensing based RSS data sampling and recovery algorithm. The experimental results show that the proposed algorithm can achieve 90% and 95% recovery rates in geographic coordinate system and polar coordinate system, respectively.

The rest of paper is organized as follows. We introduce the design and implementation of iMap system in Section II. Then, in Section III we present and explore the crowdsensing experiments to collect RSS data. In Section IV, we propose the signal propagation model and measurement error model. We further devise a sparse sampling based algorithm to recover the blank spots and show the experimental results of signal recovery. Finally, We conclude this work in Section V.

II. SYSTEM DESIGN

The proposed iMap system is designed for building accurate and reliable RSS maps. The main functions include RSS data collection, data processing and RSS map visualisation. By running iMap application on their smartphones, mobile users can participate in measuring RSS of surrounding wireless access points. In the meantime, iMap will automatically attach

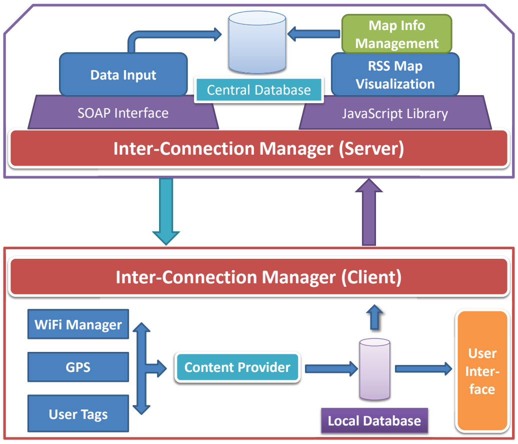


FIGURE 1: The architecture of iMap system

location information to collected RSS data. The crowdsensed data is then uploaded to a cloud-based server that is responsible for calibrating the noises and generating the visualisation data. With the iMap system, we can leverage crowdsensing paradigm to measure the signal strength of access points in large-scale urban areas. Accordingly, we conduct real-world experiments with iMap in an urban square in Wuxi City, China. The details of experimental results will be presented in Section III.

A. DESIGN OVERVIEW

We build the iMap system on two ends, *i.e.*, the user end and the server end. In the user end, we develop an mobile application for users to measure RSS values of surrounding wireless access points. In the server end, we build a cloud-based online server as a data center, The user's data is organised by its location information. We further process the data through the online server to visualize RSS maps.

The architecture of $iMap$ system is shown in Fig. 1, where the user-end mobile application consists of four modules. The content provider is one of the main function modules. It encapsulates the original crowdsensed data, acting as the interface for transmitting data to the local database. To display the collected RSS data of observed at current location, the locally stored data is sent to user interface module periodically. Similarly, the local RSS data will be sent to the connection manager module. The connection manager module will handle the communication and data exchange between smartphones and the central server. We build an online server based on LeanCloud [27] and implement the Simple Object Access Protocol (SOAP) in the data input module. The requests from mobile users of accessing database are processed through the data input module. Furthermore, we emulate a JavaScript based interface to extract RSS data that divided by the geographic information. After that, the visualisation module will iteratively attach RSS data to the map. The map information management module is set up to transfer and store visualisation data into the central database.

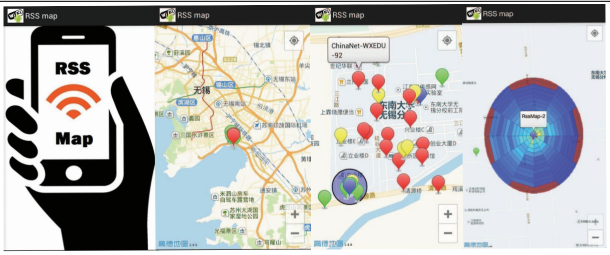


FIGURE 2: The overview of iMap mobile application

In the cloud-based central server, the RSS data is organized by its geographic coordinate. In addition, the online server provides the participating users with latest RSS dataset for visualisation on iMap application. We describe the details of system architectures in the following subsections.

B. SMARTPHONE AS A CLIENT: THE REAL-TIME MEASUREMENT

In the iMap application (as shown in Fig. 2), we build a real-time RSS data processing module. As Android operating system has provided specific classes in signal sensing, we mainly use four important classes (*i.e.*, WifiManager, ScanResult, WifiConfiguration and WifiInfo) in RSS data processing module. The WifiManager class provides a variety of APIs for WiFi management, such as WiFi scanning, establishing network connection and configuration options. We instantiate WifiManager class by simply invoking Context.getSystemService(Context.WIFI_SERVICE). We further call its public method $getScanResults$ to return a table list of access points in the latest scan. From this table list, we can acquire complete information of surrounding access points, including SSID, MAC address, levels (RSS values), capabilities and frequency. Considering the generality, we run the scanning module for 5 times at each sensing spot and take the average value of crowdsensed RSS data.

C. COMMUNICATION TO THE SERVER: GEOGRAPHIC DATA PROCESSING

In fact, it is non-trial to build RSS Maps. During signal sensing process, iMap application uses the LocationManager class to access the system's location services. This allows iMap to obtain real-time updates of each device's geographic location. We leverage the location information provided by either GPS (Global Positioning System) or cellular network to localize each mobile user and tag coordinate information to the RSS data. The iMap application will periodically upload the latest RSS data to the central server through interconnection manager. We use Json as the data transmission format in iMap, and we separate uploaded data in central server by geographic coordinates. More importantly, the coordinate is also the key unit for RSS visualization. As iMap application imports the SDK provided by LeanCloud. mobile users are able to send RSS data from the application to the cloud-based central server. Once a user opens the application, the iMap will send update requests to the central server and download the latest RSS dataset based on the current location.

D. CENTRAL SERVER: RSS MAP VISUALISATION

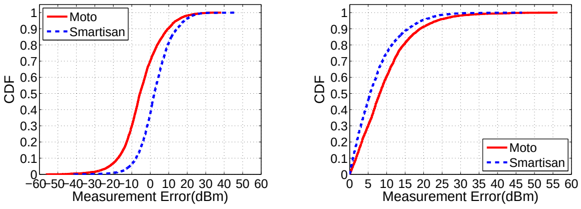
The central server is built on the Lean Cloud and it is responsible for data aggregation and RSS map visualization. Considering efficiency and accuracy, we leverage a commercial map platform called 'Amap' to visualize RSS maps on mobile devices. With Amap's location SDK (software development kit) and API (application programming interface), we apply the $getLongitude$ method and $getLatitude$ method to acquire the geographic location of each mobile user. When users are moving, the $requestLocationData$ method is invoked to capture the real-time longitude and latitude data. In the practical setting, we programme the iMap application to request for updated geographic coordinates when the change of location exceeds 5 meters. Meanwhile, iMap will rescan wireless access points once the geographic coordinates are updated. We visualize WiFi access points on Amap by two steps. First, we use the $marker$ class in the Amap SDK to mark the individual access point on the map. The different colors represent the different levels of RSS (high, medium or low). Second, we apply InfoWindow method to add information windows on the access points. When a user clicks the marker, the information window will pop up and show detailed information about the corresponding access point.

E. INCENTIVE MECHANISM: DATA ACCESS CONTROL

To motivate more users to participate in RSS map crowdsensing, we further design an incentive mechanism with data access control. By the first-time use of iMap application, a mobile user can only access the RSS data within the district he is localized. When a user uploads a new piece data from a different district, the corresponding RSS data of that district will be released to the user. Once the RSS data is unlocked, iMap application will send requests to the central server and download the new data. The above data access control flow is automatically executed in the iMap application. In our future work, we will investigate how to use authentication mechanism to improve user's incentive in crowdsensing.

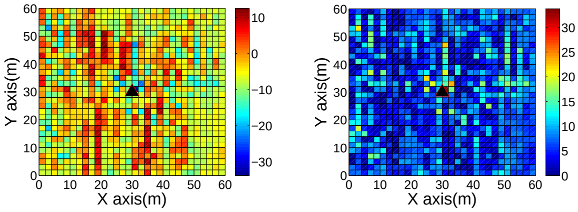
III. EXPERIMENTAL STUDY AND OBSERVATION

By leveraging the iMap system, we first conduct an experimental study with 18 volunteers from the university. The volunteers are divided into three groups, using three different types of smartphones, *e.g.*, Samsung, Motorola and Smartisan. Each participant takes random walk in a 5000 square-meter urban area for two sessions. In the first session, the participants walk randomly in groups for 30 minutes. In the second session, each participant takes random walk individually for another 30 minutes. In both sessions, the iMap application is running on each volunteer's smartphone. At last, each group will upload the crowdsensed data to the cloud server. We make the following observations and analysis on collected data.



(a) Deviation (b) Absolute deviation

FIGURE 3: Measurement deviations of different smartphones



(a) Deviation (b) Abs deviation

FIGURE 4: Spatial distribution for the measurement deviations between Samsung smartphone and Moto smartphone

A. DIVERSITY OF RSS MEASUREMENTS

First, we explore the diversity of RSS measurements with deviation and spatial deviation.

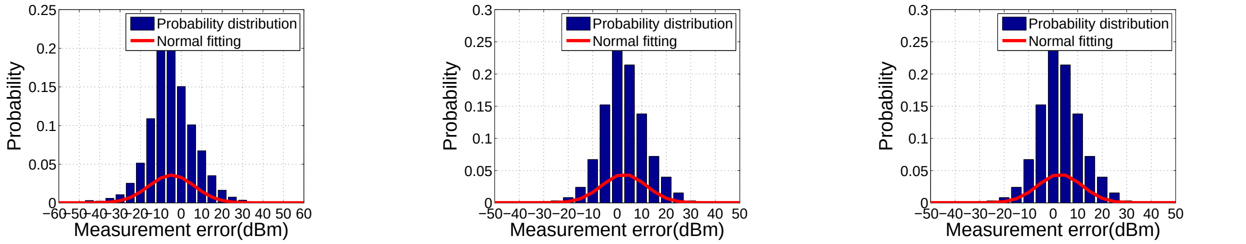
1) Deviations of RSS measurements

Specifically, we compare the deviation of measurements collected by different smartphones in Fig. 3. Here, we use the measurements of Samsung smartphones as the benchmark. In Fig. 3(a), the measurement deviations of both Moto and Smartisan have positive and negative values. This indicates that the noises caused by heterogeneity are fluctuant. As illustrated in Fig. 3(b), about 90% of the absolute deviations between Smartisan and Samsung are less than 15 dBm. Meanwhile, from Smartisan to Samsung, the absolute deviations are within 15 dBm for 80% measurements. If we narrow down the deviation range, still about 60% of absolute deviations are less than 10 dBm for Moto's measurements and 50% of absolute deviations are less than 5 dBm for Smartisan's measurements. Note that, the maximum deviations can be up to 50 dBm for both Moto and Smartisan. The above results show that, the deviations between the measurements of different devices are significant and can cause inaccuracy of RSS map. Hence, noises among heterogeneous devices need to be carefully addressed.

2) Spatial deviations of RSS measurements

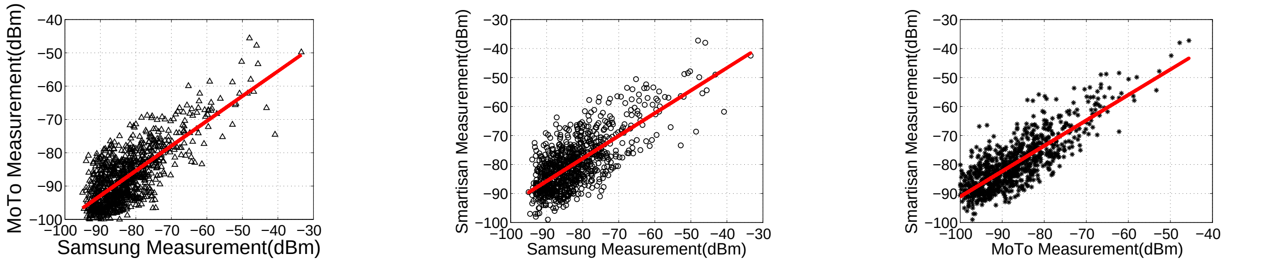
We further explore the spatial distribution of measurement deviations. In this experiment, we still use the measurements of Samsung as the benchmark. Fig. 4 shows the measurement deviations and absolute measurement deviations between Moto and Samsung in spatial distribution. The deviations are

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2867578, IEEE Access



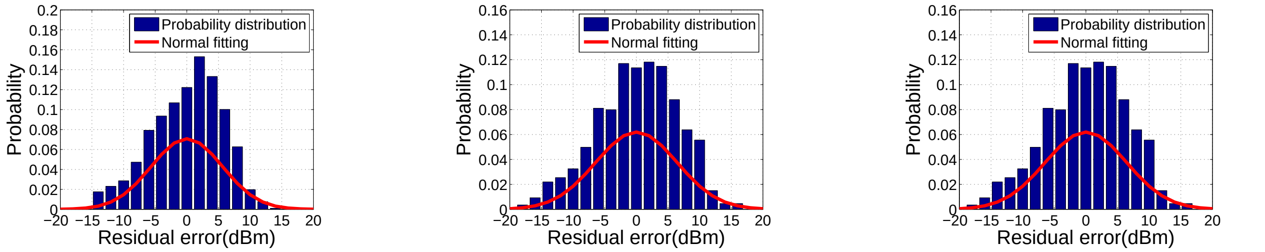
(a) Samsung VS Moto (b) Samsung VS Smartisan (c) Moto VS Smartisan

FIGURE 6: Probability distribution of the measurement deviations of different devices



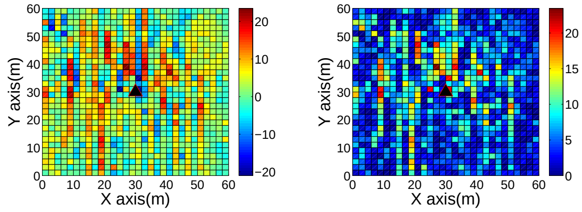
(a) Samsung VS Moto (b) Samsung VS Smartisan (c) Moto VS Smartisan

FIGURE 7: Linear relationship between the measurements of different devices



(a) Samsung VS Moto (b) Samsung VS Smartisan (c) Moto VS Smartisan

FIGURE 8: Probability distribution of the residual errors after linear fitting



(a) Deviation (b) Abs deviation

FIGURE 5: Spatial distribution for the measurement deviations between Samsung smartphone and Smartisan smartphone

randomly distributed, where the large deviations fall into the area between 15 to 40 meter on X axis and 10 to 50 meter on Y axis. Most of absolute deviations are smaller than 10 dBm and only a few exceed 20. Similarly, Fig. 5 shows the spatial distribution of deviations between Smartisan and Samsung. There are more negative deviations and the large deviations fall into the area between 20 to 50 meter on both X and Y axes. In Fig. 5(b), the absolute deviations are sparse on the map, showing that there exist noises in RSS data caused by heterogeneity of devices.

B. EXPLORING MODELS OF MEASUREMENT DEVIATIONS

Next, we explore the error models of measurement deviations from RSS measurements among Samsung, Moto and Smartisan.

Firstly, we explore whether the measurement deviations of different devices satisfy the normal model. As shown in Fig. 6, we compare the probability distribution of measurement errors with normal fittings. By using the Lilliefors test, we find that the statistical significance is only 1%, showing the rejection of normal model assumptions.

Secondly, we explore whether the measurement deviations satisfy the linear model. As shown in Fig. 7, the measurements of different devices follow linear models. Specifically, we calculate the linear fittings as follows. For Samsung vs Moto, the fitting model is $0.74737x \pm 25.6226$ with the standard deviation of 5.63 dBm. For Samsung vs Smartisan, the fitting model is $0.78078x \pm 15.4472$ with the standard deviation of 6.43 dBm. For Moto vs Smartisan, the fitting model is $0.87764x \pm 3.331$ with the standard deviation of 5.37 dBm.

Based on above observations, we further explore whether the residual errors of linear fitting follows the norm model. As illustrated in Fig. 8, we use the Lilliefors method to test this assumption. However, the statistical significance is still 1%. Such that, the residual errors do not follow the normal model.

C. CHALLENGES

Based on the experimental observations, we find that, to achieve accurate RSS map construction with signal propagation model and measurement error model is non-trivial. Adding to the blank spots without any RSS data, there are two challenges need to be formally addressed:

First, modelling the signal propagation and measurement error. The measurement error model is essential for calibrating the noises in RSS data of different devices. However, the model parameters are not known as a prior and the values of parameters usually depend on the type of mobile devices.

Second, recover signal strength data with incomplete measurements. Although compressed sensing can be used for data recovering, how to design measurement matrix and recovery algorithm still remains challenging. The goal is to extract salient information from the k-sparse or compressible signals, without damaging signal by the dimensionality reduction .

IV. SPARSE SIGNAL RECOVERY DESIGN

In this section, we present how to construct accurate and complete RSS maps with sparse sampling and signal recovery. Due to the high cost, it is not practical for participants to collect fully complete RSS data that covers every spot in a large district. Moreover, it is quite daunting to directly construct an accurate and complete RSS map with partial RSS data. Fortunately, compressive sensing methods [28], [29] are capable of recovering sparse signals with limited information. Hence, we leverage compressive sensing techniques to recover the complete RSS maps with partially sampled data. We first build the signal propagation model and measurement error model. Based on that, we propose a compressive sensing algorithm to recover RSS data on unsensed spots.

A. SIGNAL PROPAGATION MODEL AND MEASUREMENT ERROR MODEL

1) Signal propagation model

We adopt the typical signal propagation model from [30], *i.e.*, the Pass-Loss model. The propagation model of a WiFi signal in the wild can be given as

$$P_{ij}^{k} = P_{j}^{0} - 10\gamma_j \log_{10}(\tfrac{d}{d_0}), \tag{1}$$

where P_j^0 denotes the transmit power of j_{th} AP, d_0 and γ_j denote the reference distance and the path-loss exponent, respectively.

2) Measurement error model

According to the experimental results, we observe that the RSS measurements from heterogenous devices are linear with each other. For a specific user i, the k-th measurement on AP j is denoted as M_{ij}^{k} and the fixed error model is given by:

$$C_{ij}^{k} = \pi_i \cdot M_{ij}^{k} + \eta_i, \tag{2}$$

where π_i and η_i are two unknown parameter that depend on the type of the smartphones.

B. PRELIMINARIES IN COMPRESSIVE SENSING

Compressive sensing is an innovative signal sampling paradigm compared with Shannon/Nyquist sampling theorem [31]. It is related to several topics in signal processing [32], including sparse sampling, under-determined linear-systems and heavy hitters. Compressive sensing theory asserts that a relatively small number linear combination of a compressible or sparse signal can contain most of its salient information [29].

1) Compressibility of Signals

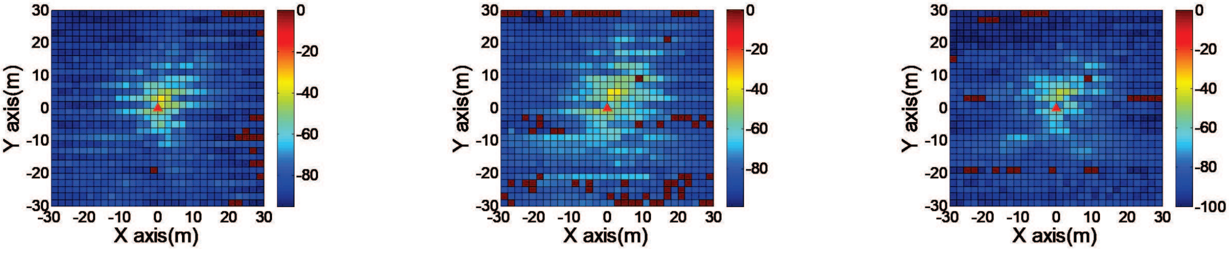
Consider that a signal x is an one-dimensional signal and it can be represented by a $N \times 1$ vector in $\mathbb{R}^N$ with elements $x[n]$, $n = 1, 2, ..., N$. Assuming that the basis is orthogonal and $\Psi = [\psi_1|\psi_2...|\psi_N]$ is a orthogonal $N \times N$ basis with the vectors $\{\psi_i\}$ as columns, then a signal x can be expressed as:

$$x = \psi s, \tag{3}$$

where **s** is the $N \times 1$ column vector of weighting coefficients $s_i = \langle x, \psi_i \rangle = \psi_i^T x$, and $\cdot^T$ denotes transposition. In fact, **x** and **s** are the same signals with different domains. While **x** is a spatial-temporal domain signal, **s** is under the ψ domain. If $\mathbf{x} \in \mathbb{R}^N$ is a K-sparse signal, it is a linear combination of only K basis vectors. Such that, only k nonzero components exist in the s_i coefficients. Thus, the information can be extracted from **s** by $\mathbf{y} = \Phi \mathbf{s}$, where Φ is an $M \times N$ measurement matrix, $\mathbf{y} \in \mathbb{R}^M$ is measurement vector and $M \ll N$.

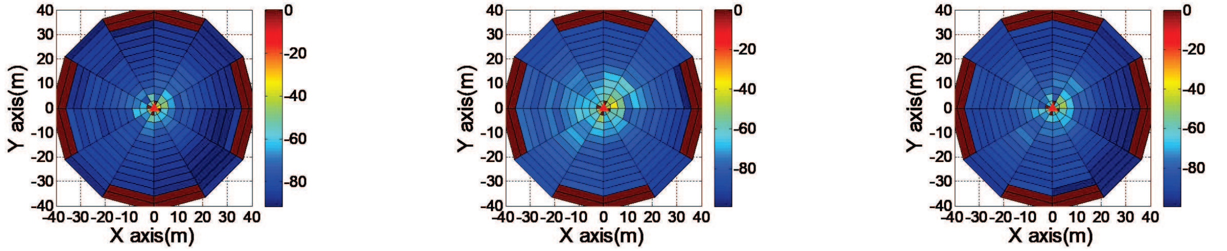
2) The Problem of Recovering Signals

Different from traditional data acquisition method, compressive sensing [33] [34] directly acquires a compressed signal representation without requiring N samples. Considering a general linear measuring process that computes $M < N$ inner products between **x** and vectors $\{\phi_j\}_{j=1}^{M}$ in $y_j = \langle x, \phi_j \rangle$, we arrange the measurements y_j in an $M \times 1$ vector **y** and set measurement vectors ϕ_j^T as rows in an $M \times N$ matrix Φ.



(a) Samsung (b) Smartisan (c) Moto

FIGURE 9: Geographic RSS map constructed with sensing data from different types of smartphones



(a) Samsung (b) Smartisan (c) Moto

FIGURE 10: Polar RSS map constructed with sensing data from different types of smartphones

By substituting Ψ from Equation (3), y can be written as:

$$y = \Phi x = \Phi \Psi s = \Theta s, \tag{4}$$

where $\Theta = \Phi\Psi$ is an $M \times N$ matrix. The measurement matrix Φ must allow the reconstruction of the length-N signal x from $M < N$. The problem of recovering a signal consists of two parts. First, design a stable Φ such that the salient information can be extracted from any K-sparse signal without being damaged by the dimensionality reduction from $x \in \mathbb{R}^N$ to $y \in \mathbb{R}^N$. Second, design an efficient signal recovery algorithm to reconstruct s from y. To address the first part of the problem, the matrix Θ must satisfy restricted isometry property (RIP) [34]:

$$1 - \varepsilon \leq \frac{||\Theta v||_2}{||v||_2} \leq 1 + \varepsilon. \tag{5}$$

That is, the matrix Φ must preserve the lengths of particular K-sparse vectors and satisfy Equation (5) for an arbitrary $3K$-sparse vector v. However, existing studies [34] [33] show that the RIP condition can be simply achieved with high probability by selecting Φ as a random matrix, for example, Gaussian Random Matrices.

To address the second part of the problem, the signal s could be recovered via ℓ_1 optimization as

$$\hat{s} = \arg\min_s ||s||_1, \text{s.t.}, y = \Phi s. \tag{6}$$

Specifically, the signal s can be successfully recovered if Φ satisfies the condition of RIP and $M \geq cK \log(N/K)$, with c as a small constant [29] [35].

In our case, when the measurement vector y contains noise, then the signal s can still be recovered via

$$\hat{s} = \arg\min_s ||s||_1, \text{s.t.}, ||\Phi s - y||_2^2 \leq \varepsilon, \tag{7}$$

where $\hat{s}$ is the recovered signals of s, $s = \psi^{-1}x$ and ε is the bound of the noise.

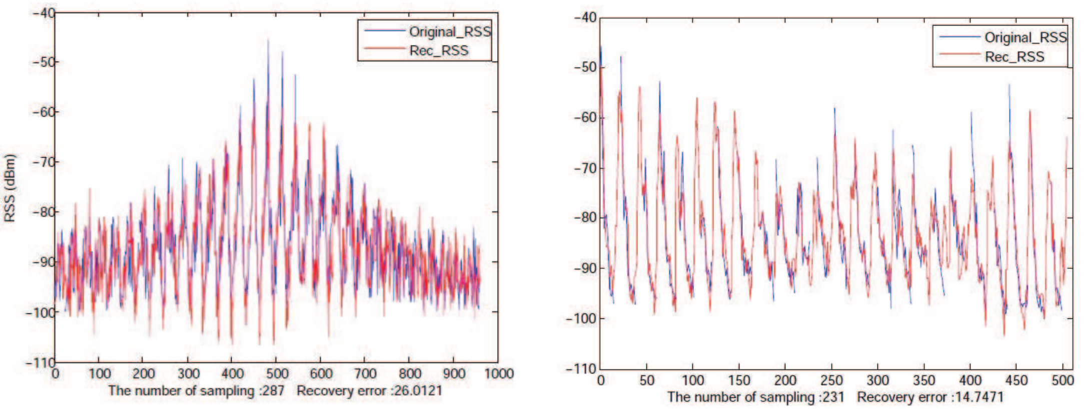
C. RSS MAP CONSTRUCTION WITH PARTIAL RSS DATA

Here, we construct the RSS map with partial RSS data collected from different types of smartphones, *i.e.*, Samsung, Smartisan and Moto. In Fig. 9, we construct RSS maps in geographic coordinate system. The red spots on the maps are unsensed spots, *i.e.*, spots without available data. We find that, the RSS data from Samsung performs the best in the map construction with the least unsensed spots. Meanwhile, the RSS maps constructed with RSS data from Smartisan are with the largest number of unsensed spots. In terms of coverage, Samsung and Moto achieves similar performance on RSS coverage in the maps. However, the Smartisan collects more sparse RSS data, showing the sensing errors cased by heterogeneity of mobile devices.

We further construct RSS maps in polar coordinates system in Fig. 10. In polar RSS maps, the unsensed spots are represented by red quadrilaterals. Obviously, the areas with blank spots are located on the edge of RSS maps. Similarly, the Smartisan's measurements on signal coverage are more sparse and inaccurate comparing with Samsung's and Moto's. Next, we devise an adaptive algorithm for RSS data sampling and recovery.

Algorithm 1 SVT Based RSS Data Sampling and Recovery Algorithm

```
Input:
    Initialize time interval $t$=0; Initialize fraction of the number of uniform samples $\omega \in [0,1]$
Output:
    Recovered measurement matrix $\widehat{T}(t+n)$;
Take uniform sampling to obtain sample set $|\Omega_t| = \eta \times N \times N$ to initial measurement matrix $M(t)$ with $N \times N$ entries.
for $t = [1:n]$ do
    Apply matrix completion and obtain partially recovered matrix $\widehat{T}(t)$:
        Take an extra uniform sample $\Omega_{t+1}$, where $|\Omega_{t+1}| = 0.5N \log N$;
        Add $\Omega_t$ to $\Omega_{t+1}$, i.e., $\Omega_{t+1} = \Omega_t \cup \Omega_{t+1}$;
        Compute new measurement matrix $M(t+1)$;
        Apply matrix completion and obtain new partially recovered matrix $\widehat{T}(t+1)$;
        if $\hat{T}(t+1) \triangleq \hat{T}(t)$
            Sampling Stopping Condition is met, stop sampling;
            Return $M(t+1)$ as the final matrix, break;
            else
            for $(i,j) \notin \Omega_{t+1}$
            Calculate $I_{(x,y)}$;
            Select the largest $\theta N \log N$ entries into $\Omega_{t+1}$ into $\Omega_{t+1}$;
        end if
    t=t+1;
end for
return $\hat{T}(t+i)$;
```



(a) Sparse sampling and recovery in geographic coordinate system

(b) Sparse sampling and recovery in polar coordinate system

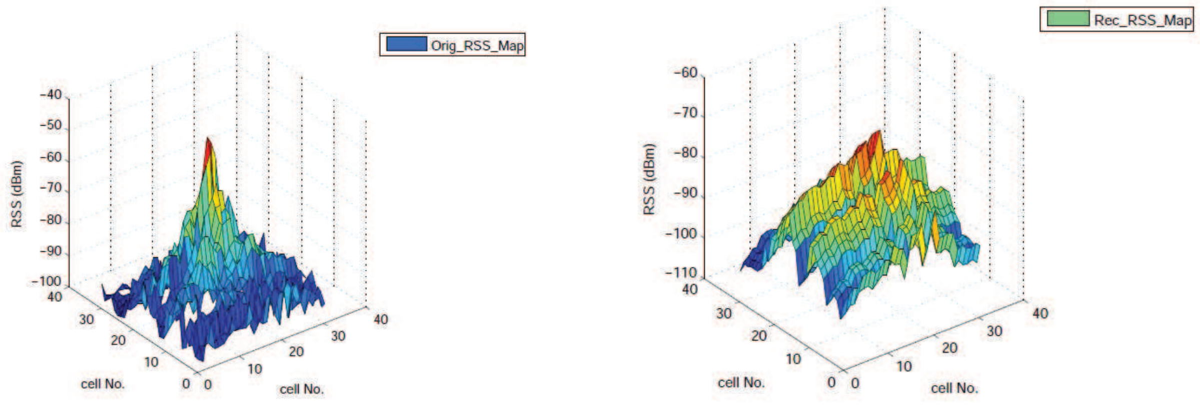
FIGURE 11: One-dimensional sparse sampling and signal recovery

D. ADAPTIVE ALGORITHM FOR RSS DATA SAMPLING AND RECOVERY

Consider that a WiFi AP covers an area consisting of $N \times N$ blocks, and we define a sensing matrix $T_{N \times N}$, where the entry T^i_{xy} represents the received signal strength measured by user i at block (x,y). We propose to adaptively recover the RSS data matrix with a small number of measurements at the initial stage. Then, we add more measurements to the partially recovered data matrix. Here, we adopt Singular Value Thresholding (SVT) from [36] to reconstruct the RSS matrix with sequential and adaptive sampling. Based on [37], we use a information-based metric I to quantify and evaluate the informativeness of an entry in the sensing matrix as follows.

$$I_{(x,y)} = \frac{\left|\widehat{T}_{xy}(t+1) - \widehat{T}_{xy}(t)\right|}{\frac{1}{2}\left|\widehat{T}_{xy}(t+1) + \widehat{T}_{xy}(t)\right|}, \tag{8}$$

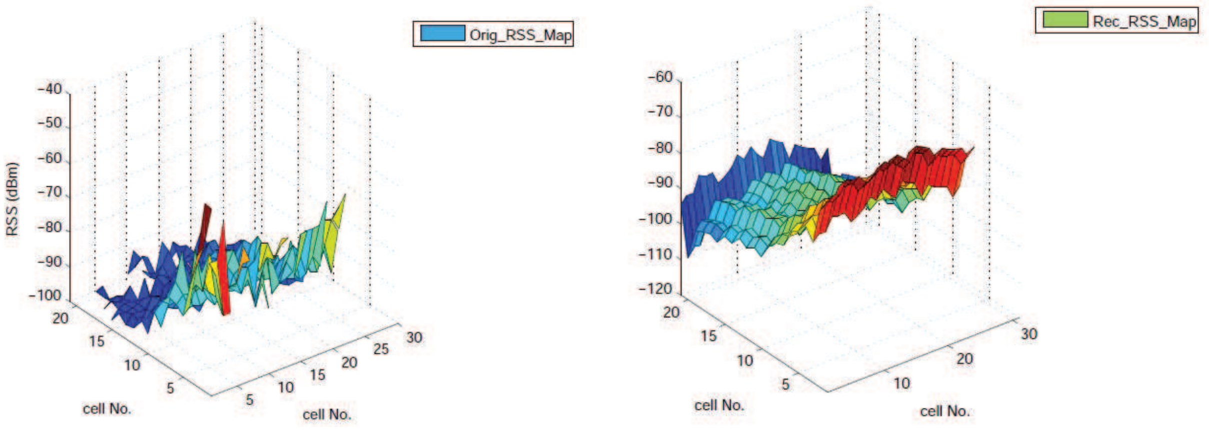
where (x,y) is an entry of the $N \times N$ matrix, $\widehat{T}_{xy}(t)$ and $\widehat{T}_{xy}(t+1)$ denote the recovered matrices at time t and time $t+1$, respectively. The value of $I_{(x,y)}$ shows the informativeness of entry (x,y). If the $I_{(x,y)}$ is large, then the entry (x,y) should be sampled in the next step. The definition of Sampling Stopping Condition is as follows.



(a) Original signals

(b) Signal Recovery after sparse sampling

FIGURE 12: Two-dimensional sparse sampling and signal recovery in geographic coordinate system



(a) Original signals

(b) Signal Recovery after sparse sampling

FIGURE 13: Two-dimensional sparse sampling and signal recovery in polar coordinate system

Definition 1. ***Sampling Stopping Condition*** *Given two matrices $\hat{T}_{N\times N}(t)$ and $\hat{T}_{N\times N}(t+1)$, if they satisfy:*

$$\frac{\sqrt{\sum\left(\hat{T}_{xy}(t)-\hat{T}_{xy}(t+1)\right)^2}}{\sqrt{\sum\left(\frac{1}{2}((\hat{T}_{xy}(t)+(\hat{T}_{xy}(t+1))\right)^2}} \leq \varepsilon, \qquad (9)$$

where ε is a static constant, we denote that $\hat{T}_{N\times N}(t) \triangleq \hat{T}_{N\times N}(t+1)$, That is to say, the original matrix has already been correctly recovered at step t , and the recovery sampling stops at step $t+1$.

We present the sampling and recovering algorithm as in Algorithm 1. First, the algorithm takes uniform sampling to generate a $|\Omega_t| = \eta \times N \times N$, where the η is the fraction number of the total samples. Note that a larger η leads to a higher sampling cost, while a smaller η reduces the reflected real information in data matrix. Next, the Singular Value Thresholding based matrix completion is applied to obtain a partially recovered matrix $\hat{T}(t)$. After that, the adaptive samples are taken for generating a new recovered matrix $\hat{T}(t+1)$. The algorithm compares the two recovered matrices to check whether the sampling stopping condition is satisfied, *i.e.*, $\hat{T}(t+1) \triangleq \hat{T}(t)$. If the condition is satisfied, then the sampling step stops and the algorithm outputs $\hat{T}(t+1)$ as the final recovered data matrix. Otherwise, the algorithm will compute I for each entry that is not included in the latest sample set. The largest $\theta N \log N$ entries will be added into Ω_{t+1}, where $\theta = \frac{\sum_{xy}\phi_{ij}}{N\times N}$ and ϕ is calculated as follows.

$$\phi_{ij} = \begin{cases} 1 & \frac{\left|\hat{T}_{xy}(t+1)-\hat{T}_{xy}(t)\right|}{\frac{1}{2}\left|\hat{T}_{xy}(t+1)+\hat{T}_{xy}(t)\right|} > \mu \\ 0 & otherwise \end{cases}, \qquad (10)$$

where μ is a small constant.

E. RSS DATA SAMPLING AND RECOVERY RESULTS

We apply the SVT based RSS data sampling and recovery algorithm on the collected dataset, in both geographic coordinate system and polar coordinate system. We set η and ε to be 12.5% and 0.05, respectively [37]. We take the average values of RSS data collected by three types of smartphones. The total number of measurements is more than 18000 and the experiment area is divided into 900 sensing blocks.

Fig. 11 shows the one-dimensional sparse sampling and signal recovery results. We plot the original RSS data with blue curves and the recovered RSS data with red curves. Under the geographic coordinate system and polar coordinate system, the number of sparsely sampled data entries are 287 and 231, respectively. The average recovery error in geographic coordinate system is relatively higher (26.0121 dBm) than that of polar coordinate system (14.7471 dBm), showing that the constructed RSS map under polar map is more accurate.

We further attach the geographic and polar coordinates to RSS data, making it become two-dimensional. In recovering the two-dimensional RSS data, the algorithm is set to take only 240 samples. Fig. 12(a) shows that the original RSS map in geographic coordinate system is incomplete and inaccurate. There are blank spots in original RSS Map and the diffusion of signal is not normal. In Fig. 12(b), the recovered RSS data has more smooth diffusion and completely covers the 900 sensing blocks, with the recovery rate of 90%. In polar coordinate system, as revealed by Fig. 13(a), the original RSS data is more sparse on the map. Nevertheless, the proposed algorithm still recovers almost all RSS data and achieves a high recovery rate of 95% in RSS map. The above RSS data sampling and recovery results demonstrate the validation of SVT based algorithm in recovering crowdsensed RSS data. With the recovered RSS data, we can build more accurate and complete RSS maps for outdoor wireless access points.

V. CONCLUSION

In this work, we have investigated the possibility of building accurate and complete RSS maps with raw data collected by heterogeneous mobile devices. We have developed an innovative iMap system for mobile users to crowdsense signals of outdoor wireless access points. We have further tested the system with different types of smartphones and observed the collected RSS measurements with model-based analysis. To construct accurate and complete RSS maps, we have devised an compressive sensing based algorithm to recover RSS data with adaptive sampling. The experimental results show that the proposed method can achieve accurate and complete recovery with partial RSS data. The recovery rates are 90% and 95% in geographic coordinate system and polar coordinate system, respectively.

ACKNOWLEDGMENT

This research is supported by China Scholarship Council (CSC), NSF China under Grants No.61872447, 61502520, 61672038, 61602067 and Chongqing Research Program of Basic Research and Frontier Technology: No.CSTC2018JCYJA1879, No.CSTC2016JCYJA0053.

REFERENCES

[1] W. Gong, B. Zhang, and C. Li, “Task assignment in mobile crowdsensing: Present and future directions,” IEEE Network, 2018.

[2] H. Li, K. Ota, M. Dong, and M. Guo, “Mobile crowdsensing in software defined opportunistic networks,” IEEE Communications Magazine, vol. 55, no. 6, pp. 140–145, 2017.

[3] X. Wang, J. Zhang, X. Tian, X. Gan, Y. Guan, and X. Wang, “Crowdsensing-based consensus incident report for road traffic acquisition,” IEEE Transactions on Intelligent Transportation Systems, 2017.

[4] L. Shao, C. Wang, L. Liu, and C. Jiang, “Rts: road topology-based scheme for traffic condition estimation via vehicular crowdsensing,” Concurrency and Computation: Practice and Experience, vol. 29, no. 3, 2017.

[5] C. Chen, D. Zhang, X. Ma, B. Guo, L. Wang, Y. Wang, and E. Sha, “Crowddeliver: planning city-wide package delivery paths leveraging the crowd of taxis,” IEEE Transactions on Intelligent Transportation Systems, vol. 18, no. 6, pp. 1478–1496, 2017.

[6] C. Chen, S. Jiao, S. Zhang, W. Liu, L. Feng, and Y. Wang, “Tripimputor: Real-time imputing taxi trip purpose leveraging multi-sourced urban data,” IEEE Transactions on Intelligent Transportation Systems, 2018.

[7] Z. Pan, H. Yu, C. Miao, and C. Leung, “Crowdsensing air quality with camera-enabled mobile devices.” in AAAI, 2017, pp. 4728–4733.

[8] S. Yang, F. Wu, S. Tang, X. Gao, B. Yang, and G. Chen, “On designing data quality-aware truth estimation and surplus sharing method for mobile crowdsensing,” IEEE Journal on Selected Areas in Communications, vol. 35, no. 4, pp. 832–847, 2017.

[9] J. Wang, N. Tan, J. Luo, and S. J. Pan, “Woloc: Wifi-only outdoor localization using crowdsensed hotspot labels,” in Proc. IEEE INFOCOM, 2017.

[10] X. Tian, W. Zhang, J. Wang, W. Li, S. Li, X. Wu, and Y. Yang, “Online pricing crowdsensed fingerprints for accurate indoor localization,” in Vehicular Technology Conference (VTC-Fall), 2017 IEEE 86th. IEEE, 2017, pp. 1–5.

[11] Z. Peng, S. Gao, B. Xiao, S. Guo, and Y. Yang, “Crowdgis: Updating digital maps via mobile crowdsensing,” IEEE Transactions on Automation Science and Engineering, vol. 15, no. 1, pp. 369–380, 2018.

[12] C. Cao, Z. Liu, M. Li, W. Wang, and Z. Qin, “Walkway discovery from large scale crowdsensing,” in Proceedings of the 17th ACM/IEEE International Conference on Information Processing in Sensor Networks. IEEE Press, 2018, pp. 13–24.

[13] X. Fan, P. Yang, C. Xiang, and L. Shi, “imap: A crowdsensing based system for outdoor radio signal strength map,” in Trustcom/BigDataSE/ISPA, 2016 IEEE. IEEE, 2016, pp. 1442–1447.

[14] L. Zhang, L. Zhao, Z. Wang, and J. Liu, “Wifi networks in metropolises: From access point and user perspectives,” IEEE Communications Magazine, vol. 55, no. 5, pp. 42–48, 2017.

[15] M. S. Afaqui, E. Garcia-Villegas, and E. Lopez-Aguilera, “Ieee 802.11 ax: Challenges and requirements for future high efficiency wifi,” IEEE wireless communications, vol. 24, no. 3, pp. 130–137, 2017.

[16] J. Ling, S. Kanugovi, S. Vasudevan, and A. K. Pramod, “Enhanced capacity and coverage by wi-fi lte integration,” IEEE Communications Magazine, vol. 53, no. 3, pp. 165–171, 2015.

[17] A. Chakraborty, M. S. Rahman, H. Gupta, and S. R. Das, “Specsense: Crowdsensing for efficient querying of spectrum occupancy,” in IEEE INFOCOM, 2017.

[18] D. Wu, Q. Liu, Y. Li, J. A. McCann, A. C. Regan, and N. Venkatasubramanian, “Adaptive lookup of open wifi using crowdsensing,” IEEE/ACM Transactions on Networking, vol. 24, no. 6, pp. 3634–3647, 2016.

[19] T. Zhou, Z. Cai, B. Xiao, Y. Chen, and M. Xu, “Detecting rogue ap with the crowd wisdom,” in Distributed Computing Systems (ICDCS), 2017 IEEE 37th International Conference on. IEEE, 2017, pp. 2327–2332.

[20] C. Xiang, P. Yang, C. Tian, C. Li, Q. Li, and X. Li, “Accurate quantification of sensor noise in participatory sensing network,” Adhoc & Sensor Wireless Networks, vol. 30, 2016.

[21] C. Xiang, P. Yang, C. Tian, L. Zhang, H. Lin, F. Xiao, M. Zhang, and Y. Liu, “Carm: crowd-sensing accurate outdoor rss maps with error-prone smartphone measurements,” IEEE Transactions on Mobile Computing, vol. 15, no. 11, pp. 2669–2681, 2016.

[22] Y. Kim, Y. Chon, and H. Cha, “Mobile crowdsensing framework for a large-scale wi-fi fingerprinting system,” IEEE Pervasive Computing, vol. 15, no. 3, pp. 58–67, 2016.

[23] Z. Zheng, F. Wu, X. Gao, H. Zhu, S. Tang, and G. Chen, "A budget feasible incentive mechanism for weighted coverage maximization in mobile crowdsensing," IEEE Transactions on Mobile Computing, vol. 16, no. 9, pp. 2392–2407, 2017.

[24] L. Wang, D. Zhang, Y. Wang, C. Chen, X. Han, and A. M'hamed, "Sparse mobile crowdsensing: challenges and opportunities," IEEE Communications Magazine, vol. 54, no. 7, pp. 161–167, 2016.

[25] T. Amano, S. Kajita, H. Yamaguchi, T. Higashino, and M. Takai, "A crowdsourcing and simulation based approach for fast and accurate wifi radio map construction in urban environment," in IFIP Networking Conference (IFIP Networking) and Workshops, 2017. IEEE, 2017, pp. 1–9.

[26] X. Wu, P. Yang, S. Tang, X. Zheng, and Y. Xiong, "Privacy preserving rss map generation for a crowdsensing network," IEEE Wireless Communications, vol. 22, no. 4, pp. 42–48, 2015.

[27] "Lean cloud," https://leancloud.cn/.

[28] Z. Han, H. Li, and W. Yin, Compressive sensing for wireless networks. Cambridge University Press, 2013.

[29] R. G. Baraniuk, "Compressive sensing [lecture notes]," IEEE signal processing magazine, vol. 24, no. 4, pp. 118–121, 2007.

[30] A. Goldsmith, Wireless communications. Cambridge university press, 2005.

[31] M. Unser, "Sampling-50 years after shannon," Proceedings of the IEEE, vol. 88, no. 4, pp. 569–587, 2000.

[32] A. Massa, P. Rocca, and G. Oliveri, "Compressive sensing in electromagnetics-a review," IEEE Antennas and Propagation Magazine, vol. 57, no. 1, pp. 224–238, 2015.

[33] D. L. Donoho, "Compressed sensing," IEEE Transactions on information theory, vol. 52, no. 4, pp. 1289–1306, 2006.

[34] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," IEEE Transactions on information theory, vol. 52, no. 2, pp. 489–509, 2006.

[35] R. Baraniuk, M. Davenport, R. DeVore, and M. Wakin, "A simple proof of the restricted isometry property for random matrices," Constructive Approximation, vol. 28, no. 3, pp. 253–263, 2008.

[36] J.-F. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," SIAM Journal on Optimization, vol. 20, no. 4, pp. 1956–1982, 2010.

[37] K. Xie, L. Wang, X. Wang, G. Xie, G. Zhang, D. Xie, and J. Wen, "Sequential and adaptive sampling for matrix completion in network monitoring systems," in Computer Communications (INFOCOM), 2015 IEEE Conference on. IEEE, 2015, pp. 2443–2451.