

# WILL: Wireless Indoor Localization Without Site Survey

Chenshu Wu<sup>1</sup>, Zheng Yang<sup>1,3</sup>, Yunhao Liu<sup>1,3</sup>, Wei Xi<sup>2,3</sup>

<sup>1</sup>MOE Key Lab for Information System Security, School of Software,  
Tsinghua National Lab for Information Science and Technology, Tsinghua University

<sup>2</sup>Xi'an Jiao Tong University

<sup>3</sup>Department of Computer Science and Engineering, HKUST  
{wu, yang, yunhao, xiwei@greenorbs.com}

**Abstract**—Indoor localization is of great importance for a range of pervasive applications, attracting many research efforts in the past two decades. Most radio-based solutions require a process of site survey, in which radio signatures are collected and stored for further comparison and matching. Site survey involves intensive costs on manpower and time. In this work, we study unexploited RF signal characteristics and leverage user motions to construct radio floor plan that is previously obtained by site survey. On this basis, we design WILL, an indoor localization approach based on off-the-shelf WiFi infrastructure and mobile phones. WILL is deployed in a real building covering over 1600m<sup>2</sup>, and its deployment is easy and rapid since site survey is no longer needed. The experiment results show that WILL achieves competitive performance comparing with traditional approaches.

## I. INTRODUCTION

Pervasive and mobile systems for context-aware computing are growing at a phenomenal rate. In most of today's applications such as pervasive medicare, smart space, wireless sensor surveillance, mobile peer-to-peer computing, [1], [2], [3] etc., location is one of the most essential contexts. In the literature of pervasive computing, wireless indoor localization has been extensively studied and many solutions are proposed to provide room-level localization services, such as locating a person or a printer in an office building.

A majority of previous localization approaches employ Received Signal Strength (RSS) as a metric for location determination. RSS fingerprints can be easily obtained for most off-the-shelf equipments, such as WiFi- or ZigBee-compatible devices. In these methods, localization is divided into two phases: training and serving. In the first phase, traditional methods involve a site survey process, in which engineers record the RSS fingerprints (e.g., WiFi signal strengths from multiple Access Points, APs) at every position of an interesting area and accordingly build a fingerprint database. Next in the serving phase, when a user sends a location query with its current RSS fingerprint, localization algorithms retrieve the fingerprint database and return the matched fingerprints as well as corresponding locations.

Although site survey is time-consuming, labor-intensive, and easily affected by environmental dynamics, it is inevitable for those RSS fingerprint matching based approaches based on RSS fingerprint matching, since the fingerprint database is constructed based on on-site fingerprint collection.

To avoid site survey, researchers turn to characterizing wireless signal propagation. They aim to build accurate signal attenuation models and use RSS as an indication of signal propagating distance. Unfortunately, attenuation models perform poorly due to unpredictable signal propagation in complex and dynamic indoor environments, lacking technical potentials for practical uses.

The advance of wireless and embedded technology has fostered the flourish of smartphone market. Nowadays, mobile phones possess powerful computation and communication capability, and are equipped with different kinds of built-in sensors for various functions. Accompanying with users round-the-clock, mobile phones can be viewed as an increasingly important information interface between users and environments. These advances lay solid foundations of breakthrough technology for indoor localization.

On this basis, we reassess existing localization schemes and explore the possibility of using previously unavailable information for wireless indoor localization. Considering user movements, originally separated RSS fingerprints are connected under certain semantics. Similarly, studying the penetrating-wall effect of wireless signals is a good starting point for characterizing different rooms or functional areas. These observations motivate us to design rapidly deployed localization approaches without the laborious site survey process.

In this study, we propose WILL, a wireless indoor logical localization approach. By exploiting user motions from mobile phones, we successfully remove the site survey process of traditional approaches, while achieving competitive localization accuracy. The rationale behind WILL is that human motions can be applied to connect previously independent radio signatures under certain semantics. WILL requires no prior knowledge of AP locations, and users are not required for explicit participation to label measured data with corresponding locations, even in the training phase. In all, such features introduce new prospective techniques for indoor localization.

To validate this design, we deploy a prototype system and conduct extensive experiments in a middle-size academic building in Tsinghua University. Experiment results show that RSS-based indoor localization can achieve room-level location accuracy even without site survey. The average room localization accuracy, namely, accuracy of locating fingerprints

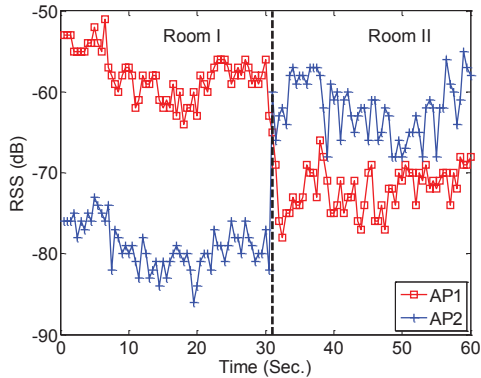


Figure 1: Abrupt signal changes through a wall. AP1 is deployed in Room I and AP2 in an adjacent Room II. Both data are measured at fixed locations.

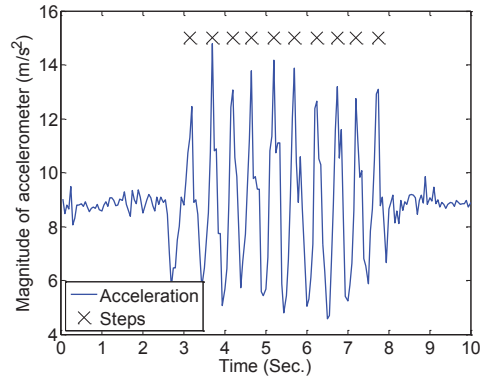


Figure 2: Acceleration signatures of 10 steps (each step marked with a cross).

to the rooms they are actually collected from, is over 80%, which is competitive to existing solutions.

The rest of the paper is organized as follows. We investigate the state-of-the-art on indoor localization technology in Section II. Section III presents our design overview. The generation of virtual rooms is studied in Section IV. In Section V, the techniques of floor plan mapping, a key step of constructing the relation between virtual rooms and ground-truth floor plan without site survey, are discussed in detail. Section VI summarizes the entire working process of WILL when it receives a location query. The prototype implementation and experiments are discussed in Section VII. We conclude the work in Section VIII.

## II. RELATED WORK

Location information is essential for a wide range of pervasive and mobile applications, such as wireless sensor networks, mobile social networks, location-based services, smart space, [1], [2], [3] etc. Especially, in the literature of indoor localization, a well-known research direction in pervasive and mobile computing, many techniques have been proposed in the past two decades.

**Infrastructure-based techniques.** Earliest approaches rely on installing specific infrastructure for indoor localization, such as LANDMARC [4] based on RFID systems, Active Badge [5] based on infrared beacons, Cricket [6] based on ultrasound devices, VOR [7] using VOR base station, and PinPoint [8] relying on special hardware they called PP2. Infrastructure-based techniques are cost-consuming, labor-intensive and improper to scale as ubiquitous indoor localization schemes.

**Fingerprinting-based techniques.** Adopting the fingerprint matching method, some localization approaches bypass pre-installed hardware. The main idea is to fingerprint the surrounding signatures at every location in the areas of interests and build a fingerprint database. The location is then estimated by mapping the measured fingerprints against that database. Researchers have striven to exploit different signatures of the existing devices or reduce the mapping effort. Most of these techniques utilize the RF signals such as [9], [10], [11], [12], [13], [14], [15]. SurroundSense [16] performs logical location estimation based on ambience features including sound, light, color, WiFi, etc. They all need site survey over areas of interests. The considerable manual cost

and efforts, as well as the inflexibility to environment dynamics are the main drawbacks of site-survey-based methods.

**Model-based techniques.** An RF propagation model, e.g., the log-distance path loss (LDPL) model, is used to estimate locations according to the measured RSS values. These techniques reduce the measurement efforts at the cost of decreasing localization accuracy due to the irregular RF propagation in indoor environment. In addition, most of these techniques require the placement of additional infrastructure, modifications to off-the-shelf APs, or knowledge of AP locations and power settings. The systems described in [17], [18] use the LDPL model and [19] uses a more sophisticated ray-tracing model, while [20] uses a Bayesian hierarchical model. Moreover, model based techniques are vulnerable to environment dynamics.

Different from previous work relying on infrastructure and propagation model, WILL adopts the fingerprinting technique but avoids site survey. WILL users are not involved in any work of data collection.

## III. OVERVIEW

### A. Unexploited Potential for Localization

WiFi technology has shown its great potentials for ubiquitous localization as it is available in a large amount of buildings through personal electronic devices like mobile phones and laptops.

By investigating the temporal and spatial characteristics of indoor RF propagation of WiFi signals, we discover some easily overlooked but dramatically useful characteristics. A key observation is that signals may encounter a considerable abrupt change while passing through a wall (as shown in Figure 1). As a result, RSS of a same AP can vary significantly in two rooms. We have been observing this wall-penetrating effect of radio signals when we use wireless routers in everyday life. Such characteristic, however, has not been fully utilized for positioning. As shown in Figure 1, the variation of AP's signal strength can be used to distinguish different rooms.

On the other hand, smartphones integrate various kinds of sensors such as accelerometer, magnetometer, gyroscope, etc., offering new opportunities to capture environment signatures and to detect user behaviors. WILL exploits accelerometers to obtain user movements and utilizes moving traces to assist localization.

Tri-axial accelerometers provide the apparent evidence of human walking patterns [21]. As illustrated in Figure 2, the acceleration variation for walking users is clearly different from those static. Amplitude of about  $2\text{m/s}^2$  is caused by foot lifting and around  $3\text{m/s}^2$  by foot down. This signature is integrated in WILL to detect user motions and collect user traces.

WILL provides human localization service through locating mobile phones. Even though mobile phones can integrate sensors like compasses, cameras, microphones, gyroscopes, WILL uses only accelerometers since no human participation is involved for such sensors. Moreover, different from many previous work using accelerometers for step counting or displacement estimation, WILL utilizes accelerometer sensors to explore reachability between different areas.

### B. System Architecture

In this subsection, we present the overall vision of WILL, as shown in Figure 3. The working process of WILL consists of two phases: training and service. We describe high level architecture and present the details later.

During the training phase (database construction), users in a building work with routine business while their mobile phones automatically measure WiFi signal strengths and record built-in sensor readings. The raw data are collected in the fingerprint collection module on the mobile phone side. All raw fingerprints (not tagged with a known location) are pre-processed in fingerprint processing module and divided into two types: space-continuous and space-discontinuous according to users' motion states when the fingerprints are measured. Both types of fingerprints are classified into different *virtual rooms*, which are virtual containers of fingerprints with high similarity. Space-continuous data are further used for building *logical floor plan*. A logical floor plan shows a view of relative location relationship, especially the connectivity and reachability, between the virtual rooms. Logical floor plan is constructed by leveraging user trace information from the space-continuous data which connects previously independent fingerprints. Afterwards, the logical floor plan is mapped to a given ground truth one by using a novel method. By doing so, we associate the isolated fingerprints with physical rooms. Floor plan database stores these associated relationships.

In the service phase, when a user sends a location query with his currently measured data using mobile phone, WILL server will response the user with the estimated location. The query may contain a variety of information, including WiFi measurements and sensory data. The localization engine consults the fingerprint database to localize the virtual room and then obtain the corresponding physical room from floor plan database. Then location estimation and, if possible, the floor plan that the user currently locates at are sent back to the user. The querying data can be simultaneously used as collected fingerprints to update the databases.

## IV. VIRTUAL ROOM GENERATION

In this section, we define virtual room and describe how to extract unique features of fingerprints from raw data. Afterwards, we classify fingerprints into a number of virtual rooms.

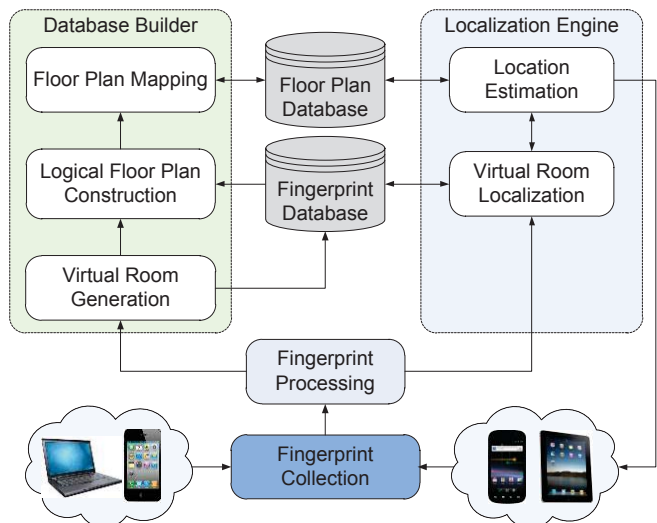


Figure 3: WILL architecture.

### A. Fingerprint Collection

WILL users do not need to deliberately collect data even in the database building phase. They just work in offices, consume in shopping malls, or have a rest at coffee shops, walking or sitting. The information of WiFi signals and sensor readings is collected automatically by their cell phones.

A regular record can be represented as  $D_i = \langle F, A \rangle$ , where  $F$  and  $A$  indicate the WiFi signal fingerprint and accelerometer value, respectively. Assuming totally  $n$  APs in the building, the WiFi signals fingerprint  $F$  can be represented as

$$F = [f_1, f_2, \dots, f_n] \quad (1)$$

where  $f_i$  denotes the RSS value of the  $i^{\text{th}}$  AP.

The motion state of users, walking or staying, is determined by accelerometer readings. Records of walking users are integrated as an entire user trace  $U = \langle \mathcal{F}, \mathcal{A} \rangle$ , where  $\mathcal{F}$  is a set of RSS fingerprints and  $\mathcal{A}$  a set of acceleration values of a user trace. Such records are called *space-continuous* as they are measured during user's movements. For ease of presentation, we refer to *continuous* data as space-continuous data in this paper hereafter.

Note that the raw data collected by the mobile phones are noisy due to the measurement errors as well as the signal fluctuations. Some pre-processing schemes are employed in the fingerprint collection module to smooth raw data.

### B. Fingerprint Processing

Due to signal instability, it is inadequate to utilize absolute RSS values directly for location estimation. In this work, we propose the *RSS stacking difference* as the fingerprint feature, which means the cumulative difference between one AP and all other APs. RSS stacking difference embodies the RSS gap relations of the RSS fingerprint at a specific time and location and acquires a relatively stable feature of radio signals.

Formally, given two fingerprints  $F = [f_1, f_2, \dots, f_n]$  and  $F' = [f'_1, f'_2, \dots, f'_n]$ , the dissimilarity (Euclidean distance) between them using feature of RSS stacking difference can be calculated by the following formulae:

$$\phi(F, F') = \sqrt{\sum_{i=1}^n (\omega(f_i) - \omega(f'_i))^2} \quad (2)$$

$$\omega(f_i) = \sum_{j=1}^n I(f_i - f_j > 0) (f_i - f_j) \quad (3)$$

where  $I$  is an indicative function.

### C. Virtual Rooms

Fingerprints are partitioned into different virtual rooms according to the values of RSS stacking difference. A *virtual room* is a virtual container which consists of the fingerprints with high similarity. Formally, if  $\phi(F_1, F_2) < \zeta$ ,  $F_1$  and  $F_2$  are treated to be in the same virtual room, where  $\zeta$  is a dissimilarity threshold of the room.

Virtual rooms are generated by applying data mining approaches on fingerprints. We adopt several clustering techniques, including KMeans, FarthestFirst, EM, and FilteredCluster, which are implemented in WEKA, a popular classification and clustering tool. Among different techniques, KMeans demonstrates its high accuracy and efficiency for this application. Detailed results of performance comparison are shown in Figure 9. After virtual room generation, each fingerprint is tagged with a virtual room label. In addition, each room  $R$  is marked with a representative fingerprint  $F[R]$  for fast location estimation (See Section VI). This representative fingerprint, along with the dissimilarity threshold  $\zeta$  is dynamically determined and updated according to the fingerprint database. In addition, both parameters are room specific, namely, each room has a distinct value for each parameter.

## V. FLOOR PLAN CONSTRUCTION

Without site survey, the key challenge of localization is how to figure out the locations where fingerprints are measured. In this section, we provide a matching based technique that finds a perfect match between logical floor plan of generated virtual rooms and the ground truth one. The mapping of the logical and ground truth floor plan tells the correspondence of fingerprints and their measured locations.

### A. Logical Floor Plan

Previous work mostly focuses on a single location or a single room. The relationship of different rooms is not sufficiently excavated. In WILL, traces of user's motion indicate the reachability among virtual rooms, which are used to construct the logical floor plan for virtual rooms.

A *logical floor plan* is a diagram showing the view of the reachability among virtual rooms. It is formalized as an undirected graph  $P = (V, E)$ , a.k.a. the *logical graph*, where each vertex  $v \in V$  denotes a virtual room and an edge  $(u, v) \in E$  indicates that the virtual room  $u$  and  $v$  are reachable. Two rooms A and B are referred to be connected in logical floor plan if and only if a user can walk from A to B seamlessly without passing any other room. We observed that user movements inside a building, from one room to another or through the corridors, might indicate the connectivity between rooms. For ease of understanding, we take an example in the ground truth floor plan (as shown in Figure 4). If a user walks from room A to room B through a corridor segment C, then it

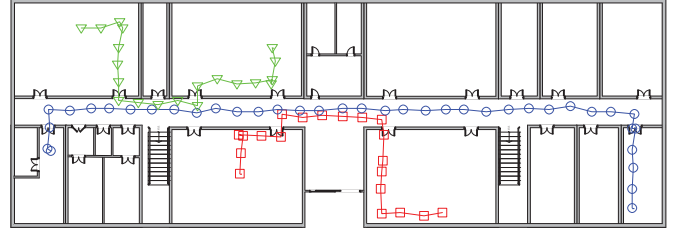


Figure 4: Examples of user traces from users walking inside the building

can be derived that C is reachable from both A and B but A is not directly connected to B on only this condition. We assume that reachability is bidirectional, namely, if room A is reachable from room B, then B is also considered reachable from A.

A series of fingerprints can be collected during their movements. As fingerprints are labeled with virtual rooms, an entire trace may traverse different virtual rooms. In addition, the sequence of the virtual rooms being traversed can be obtained because the trace is timestamped and ordered.

Concretely, we consider a single user trace  $U = \langle \mathcal{F}, \mathcal{A} \rangle$  where  $\mathcal{F} = [F_1, F_2, \dots, F_m]$  and  $\mathcal{A} = [A_1, A_2, \dots, A_m]$  indicate the a sequence of  $m$  fingerprints and acceleration readings collected during the user's movement. Each  $F_i$  belongs to a virtual room  $R_i$ . Hence  $\mathcal{F}$  corresponds to a series of virtual rooms  $\mathcal{R} = [R_1, R_2, \dots, R_m]$ . Accordingly, the reachability between virtual rooms can be obtained by following rule: if  $R_i \neq R_{i+1}$ , which means the user walks into virtual room  $R_{i+1}$  from  $R_i$ , then  $R_i$  and  $R_{i+1}$  is marked to be reachable to each other. In other words, an edge  $(R_i, R_{i+1})$  is added to the logical floor plan  $P$  if  $(R_i, R_{i+1}) \notin E$ . Fusing a large amount of user traces together, the logical floor plan  $P$  is constructed.

### B. Floor Plan Mapping

Logical floor plan needs to be mapped to the ground truth floor plan, which is available to the estate manager of a building who is also supposed to be the provider of location services in this building. For convenience, the ground truth floor plan is also referred to *physical floor plan* hereafter. In the following, we provide a graph mapping method and improve the mapping result later by introducing assistive techniques like global reference point.

The physical floor plan, i.e., the *physical graph*, is modeled with an undirected graph  $P' = (V', E')$  where each vertex  $v \in V'$  indicates a room (or a functional area) and each edge  $(u, v) \in E'$  means the reachability of two rooms  $u$  and  $v$ . Under this scheme, the corridors are connected to most rooms while the adjacent rooms are not connected if no door exists between them. Note that the corridors can be divided into several segments, mainly according to the sizes. The modeled physical floor plan of our experiments is shown in Figure 6, where the corridor is segmented into four parts. Given the logical floor plan  $P = (V, E)$  and the ground truth floor plan  $P' = (V', E')$ , we define the floor plan mapping as a function  $p: V \rightarrow V'$ . In WILL, we set the numbers of virtual rooms is at least equal to the number of physical areas; i.e.,  $|V| \geq |V'|$ .

We propose a *subsection mapping method* (SSMM) which contains three stages: skeleton mapping, branch-knot mapping and the correction. The virtual rooms with higher betweenness are in prior mapped in skeleton mapping while the rest are

mapped using bipartite matching in branch-knot mapping. The initial mapping results are slightly adjusted in the correction stage.

**Skeleton mapping.** *Betweenness centrality* [22] is a measure of a vertex's centrality within a graph. Vertices that occur on many shortest paths between other vertices have higher betweenness than those do not. Formally, for a graph  $G = (V, E)$ , let  $C_B(v)$  denote the betweenness centrality of a vertex  $v$ , then  $C_B(v)$  is calculated by the following equation:

$$C_B(v) = \sum_{s \neq v \neq t \in V'} \frac{\sigma_{st}(v)}{\sigma_{st}} \quad (4)$$

where  $\sigma_{st}$  is the number of shortest paths from  $s$  to  $t$ , and  $\sigma_{st}(v)$  is the number of shortest paths from  $s$  to  $t$  that pass through vertex  $v$ . As shown in Figure 6, the vertices in the center (labeled C1, C2, C3, and C4) apparently have higher betweenness than others.

Let  $c$  be a constant. In skeleton mapping stage, the  $c$  vertices which have the highest betweenness in  $P$  are mapped to other  $c$  vertices with highest betweenness in  $P'$ . Here the mapping goal is to minimizing the total difference of betweenness for all matching pairs. The constant  $c$  is adjusted according to the graph structures. In WILL, this parameter  $c$  is simply set equal to the number of corridor segments in physical graph because those corridor segment vertices surely have higher betweenness under our physical graph model.

**Branch-knot mapping.** The rest of vertices in  $P$  are mapped using the sum of shortest paths length as weights. In other words, for each vertex  $v$  in graph  $P$ , its weight  $w(v)$  equals to the sum of all shortest path lengths from  $v$  to all other vertices in  $P$ , namely,  $w(v) = \sum_{u \in P, u \neq v} d(v, u)$  where  $d(v, u)$  is the length of the shortest path from  $v$  to  $u$ . The weight of each vertex in  $P'$  is calculated in the same way. Then the mapping goal is to minimize the total weight difference, say,  $W(p) = \sum_{v \in V'} |w(v) - w(p(v))|$ .

We formalize the branch-knot mapping as a *weighted minimum bipartite matching* (WMBM) problem where every vertex in  $P$  is matched to another vertex in  $P'$ , resulting in a *perfect matching*. The WMBM problem is then performed using the Kuhn-Munkras (KM) [23] algorithm as follows. Put all *unmatched* vertices in  $P$  to a set  $L$ , and all vertices in  $P'$  into a set  $R$ . To achieve perfect matching,  $|L| - |R|$  *fake vertices* are added to  $R$  if  $|L| > |R|$ ; otherwise  $|R| - |L|$  fake vertices are added to  $L$  (Without loss of generality, we assume  $|L| > |R|$  below). Construct a bipartite graph  $G_B = (V_B, E_B)$ , where  $V_B = L \cup R$ . For each vertex  $v \in V_B$ , the weight  $w(v)$  is the same as in  $P$  or  $P'$  if  $v \in V \cup V'$ ; Otherwise  $w(v) = \infty$  when  $v$  is a fake vertex.  $L$  and  $R$  are disjoint and all edges in  $E_B$  go between  $L$  and  $R$ . For vertices  $u \in L$  and  $v \in R$ , edge  $(u, v) \in E_B$  if  $u$  could be matched to  $v$  and the weight of  $(u, v)$  is set to be  $w(u, v) = |w(u) - w(v)|$ . As each vertex in  $P$  is possible to be matched to any vertex in  $P'$ , the resulting bipartite graph  $G_B$  is a complete bipartite graph  $K_{|L|, |L|}$ .

Afterwards, the KM algorithm is performed on  $G_B$  and returns a matching result of  $M = [p(v_1), p(v_2), \dots, p(v_{|L|})]$  where  $p(v_i)$  indicates  $v_i \in L$  is matched to  $p(v_i) \in R$ . As there may be  $|L| - |R|$  fake vertices in  $R$  (if  $|L| > |R|$ ), the matching result  $M$  needs to be modified to remove those fake vertices. For each vertex  $v_i \in V$ , if  $p(v_i) \notin V'$ , then  $p(v_i)$  is changed to  $v'$  such that  $w(v_i, v') =$

---

### Algorithm 1 Correction

---

```

correction = false
for each vertex  $s$  in  $S_P$ 
  if  $ND(N_P(s), N_M(s)) > |N_M(s)|/2$  then
    find  $v \in S_{P'}$  minimizing  $ND(N_P(s), N_M(v))$ 
    map  $s$  to  $v$ 
    correction = true
  else
    for each vertex  $v$  in  $N_P(s) \cup N_I(s) - \{s\}$ 
      if  $v \notin N_P(s)$  and  $v \in N_I(t)$  and  $v \notin S_{P'}$  then
        remove mapping from  $s$  to  $v$ 
        correction = true
      else if  $v \in N_P(s)$  and  $v$  is not mapped then
        find  $u \in N_M(t)$  minimizing  $|w(v) - w(p(u))|$ 
        set  $p(v) = p(u)$ 
        correction = true
    end if
  end for
end if
end for

```

---

$\min\{w(v_i, v'_j), v'_j \in V'\}$ . By branch-knot mapping, every vertex in  $V$  unmapped upon skeleton mapping is matched to one and only one vertex in  $V'$ . The details of KM algorithm are described in [23] and we omit them in this paper.

Combining the result of skeleton mapping and branch-knot mapping, an original mapping is obtained. Let  $M_I = [p(v_1), p(v_2), \dots, p(v_n)]$  be the initial mapping result, where  $p(v_i)$  denotes  $v_i \in V$  is mapped to  $p(v_i) \in V'$ . Figure 11(a) and (b) show the result of skeleton mapping and branch-knot mapping, respectively. Evident from Figure 11, mapping errors could exist in the initial mapping result. We perform the correction stage of SSMM to fix some error mapping.

**Correction.** Sort all vertices in  $P$  in descending order of betweenness. Find a watershed value  $b$  of betweenness such that vertices with betweenness higher than  $b$  are apparently larger than those with betweenness lower than  $b$ . Formally, let  $v_h$  be the vertex with the smallest betweenness higher than  $b$  and  $v_l$  be the vertex with the largest betweenness lower than  $b$ . Then  $b$  is determined such that  $C_B(v_h) - C_B(v_l) = \max\{|C_B(v_i) - C_B(v_{i+1})|, v_i, v_{i+1} \in V\}$ . Define a *skeleton set*  $S_P$  of  $P$  as below:  $S_P = \{v: C_B(v) > b, v \in V\}$ . For each *skeleton vertex*  $s \in S_P$ , define a *neighboring tree*  $T(s)$  of  $s$ .  $T(s)$  is similar to a depth-first tree with  $s$  as root, but with a little difference. In the depth-first search (DFS) procedure starting from  $s$ , the DFS stops and traces back when encountering a skeleton vertex. Let  $N_P(s)$  denote the vertices set in neighboring tree  $T(s)$ . Then  $N_P(s)$  is the *neighboring set* (NS) of skeleton vertex  $s$ . The skeleton set  $S_{P'}$  of  $P'$  and its NS are defined and generated in the same way as in  $P$ .

For each  $v \in V'$ , define a *clan*  $C(v)$  of  $v$  as  $C(v) = \{u: p(u) = v, u \in V'\}$ . Thus  $V'$  contains  $|V'|$  clans. For each  $s \in S_P$ , let  $t = p(s)$  and define a mapping NS of  $s$  as  $N_M(s) = \bigcup_{v \in N_{P'}(t)} C(v)$  where  $N_{P'}(t)$  is the NS of  $t$ . For a skeleton vertex  $s \in S_P$ , define the *neighboring distance* (ND) of  $N_P(s)$  and  $N_M(s)$  as:  $ND(N_P(s), N_M(s)) = |N_M(s)| - |N_P(s) \cap N_M(s)|$ . In other words,  $ND(N_P(s), N_M(s))$  indicates the number of vertices which are in  $N_M(s)$  but

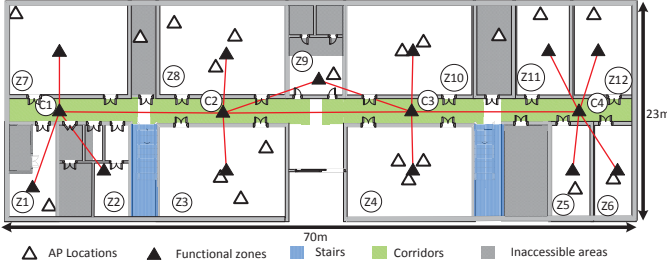


Figure 5: Floor plan of the building in Tsinghua University. APs with unknown locations are not marked.

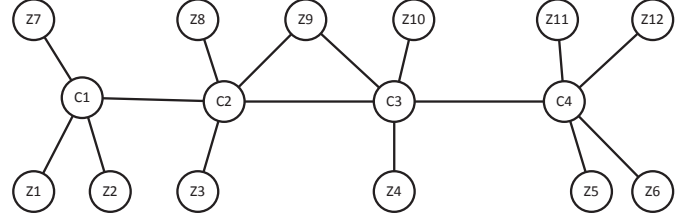


Figure 6: Ground truth floor plan graph. Vertices marked with ‘Z’ denote physical rooms and vertices marked with ‘C’ denote corridor segments.

not in  $N_P(s)$ . Then the correction of initial mapping is performed according to the value of  $ND$ , as outlined by Algorithm 1.

Repeat Algorithm 1 until no correction occurs. Then the final mapping result  $M = [p(v_1), p(v_2), \dots, p(v_n)]$ ,  $v_i \in V$ , is achieved. The ultimate results of SSMM on two graphs are depicted in Figure 11(c).

## VI. LOCALIZATION USING WILL

We have constructed the fingerprint database and the floor plan database during the training phase of WILL. The association between these two databases is also established. In this section, we describe the entire working process of WILL when it receives location queries, which corresponds to the localization engine module in WILL system.

### A. Localization

Recall Section IV, we mark each virtual room  $R$  with a representative fingerprint  $F[R]$  after they are generated from the fingerprints. We use the mean value of all fingerprints in virtual room  $R$  as  $F[R]$ . Formally,  $F[R]$  can be calculated by the following formulas:

$$F[R] = \frac{1}{|\mathcal{F}_R|} \sum_{F_i \in \mathcal{F}_R} F_i \quad (5)$$

where  $\mathcal{F}_R$  is the set of fingerprints that belong to  $R$ .

When one user visits a building where WILL is deployed, the user queries WILL server for his/her current location with a record  $D_t = \langle F, A \rangle$ , where  $t$  is the timestamp,  $F$  and  $A$  indicate WiFi signals and accelerometer values, respectively. The localization engine of WILL first determines the virtual room  $F$  belongs to, and then consults the floor plan database to obtain the mapped physical room, which is the response to be sent back to the user.  $F$  is estimated to be in the virtual room which has the shortest distance to  $F$  among all virtual rooms. Formally, the virtual room localization defers to the below rule:  $F$  belongs to virtual room  $R_i$  if the dissimilarity of  $F$  and  $F[R_i]$  satisfies

$$\phi(F, F[R_i]) = \min \{ \phi(F, F[R_j]), R_j \in \mathcal{R} \} \quad (6)$$

and

$$\phi(F, F[R_i]) < \xi \quad (7)$$

where  $\phi$  is the dissimilarity defined by Equation (2),  $\mathcal{R}$  is the set of all virtual rooms and  $\xi$  denotes the dissimilarity threshold of rooms. Fingerprints beyond above two equations are treated

as outliers and discarded. Assume that virtual room  $R_i$  is mapped to a physical area  $R'_i$ , then the user location is estimated as zone  $R'_i$  and the result is sent back to users with, if possible, the floor plan.

We design the localization engine as lightweight as possible for the purpose of better user experience on mobile phones and making WILL easily scalable.

### B. Database Update

The floor plan and the fingerprint database can be updated over time to capture environment dynamics. In addition, the data collected in the training phase of WILL may contain deviations. They might not roundly reflect the overall situation of the building. As a result, updating is necessary to remedy such deviations. We execute two types of update operations in WILL: minor update and major update.

Minor update, being triggered frequently, deals with newly collected fingerprints. When user queries arrive, the attached fingerprints are not only used for localization, but also for updating virtual room features, including the representative fingerprints and dissimilarity thresholds.

Major update is carried out occasionally for a large amount of new data, resulting in large modifications in the previous database. For instance, if huge data are collected through a long-term running, especially when enough continuous data are included, the floor plan is corrected using the updated logical floor plan.

## VII. EXPERIMENTATION AND EVALUATION

### A. Experimental Methodology

We developed the client of WILL on the increasingly popular Android OS. WiFi signals are recorded with the frequency of around twice per second when measuring. Accelerometers work in two frequencies: when detecting motions, they record sensory data with short intervals of 50 milliseconds; otherwise a relatively long interval of one second is adopted.

We implemented our prototype on two Google Nexus S phones, which support WiFi and contain accelerometer sensors. We deployed WILL system in one floor of an office building covering over 1600m<sup>2</sup> in Tsinghua University, which contains 16 offices, of which 5 are large rooms of 142m<sup>2</sup>, 7 are small ones with different sizes and the other 4 are inaccessible. The floor plan is shown in Figure 5, where every physical zone is marked with a sequence number. Most rooms are installed with one or more APs while some have none. Totally,  $n=26$  APs are

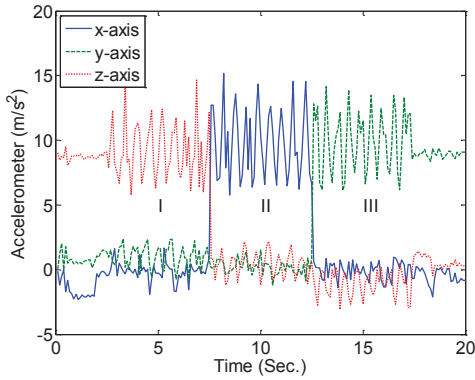


Figure 7: Accelerometer over different postures. I: the phone is horizontally placed; II: the phone is sideways up; III: the phone is vertically placed

installed in the floor, of which 20 are with known locations and are denoted in Figure 5. Note that the walls of the experimental building are constituted by steel keels wrapped in two wooden clapboards instead of reinforced concrete, which reduces the walls' shielding effects of wireless signals.

The ground truth floor plan is modeled as a physical graph. As depicted in Figure 5, the black triangles indicate physical functional zones and the edges show their connected relationships. In our evaluation, each physical room is modeled with a vertex while the corridor is divided into 4 segments. Each segment's length is roughly in line with the length of the largest room it connects. As a result, there are total 16 functional zones in the physical graph.

To evaluate WILL, we need the accurate room of each user when the location query is submitted. We require location samples, especially those close to the walls, to evaluate the localization performance. To obtain these location-labeled data, we set a data acquisition point every  $4m^2$  and collected 30 records with manually reported locations at each point. The data records for evaluation are extended to be  $D_i = \langle F, A, L \rangle$  where  $L$  is an additional tuple, location. We collected 16336 records (*dataset #1*) on one phone and 14271 records (*dataset #2*) on another. All data are evenly collected from accessible areas in the floor.

Space-continuous data, say, the mobility data collected during user movements, consist of two parts in our experiments. One part are collected from real user traces, the other are generated from the discontinuous data. To collect continuous data, mobile phone records the accelerometer and WiFi data during users' natural movements. Totally, 30 real traces are extracted and additional 118 traces are generated from those location-labeled data. Different traces have various lengths and cover different areas of interests. Note that the generated traces are realistic because the experimental data contains manually labeled accurate location information.

### B. Performance

In this section, we evaluate WILL using dataset #1 as training data for building databases and dataset #2 as querying data to localize.

#### User Trace Detection

Though the users kept their mobile phones in hand when collecting continuous data in our experiments, we find that the

rhythmic acceleration signatures in human walking patterns are evident no matter what postures the mobile phones are. As depicted in Figure 7, although the most remarkable acceleration variation caused by walking appears on different axes, the tri-axial accelerometer captures rhythmic fluctuations finely whenever the mobile phone is placed horizontally in hand, sideways up, or vertically held, corresponding to the segments I, II and III in Figure 7 respectively.

#### Virtual Room Generation

For all virtual rooms, we mark each of them with the label of a physical zone where the largest portion of fingerprints within this virtual room come. The assignment error rate (AE) is used for evaluation of virtual room generation which is referred to the percent of fingerprints tagged with a virtual room taking a physical zone label different from the zone where the fingerprints are actually collected.

As illustrated in Figure 8, we notice that all clustering approaches can achieve a fairly good accuracy of over 80% on virtual rooms. Particularly, the KMeans approach can reach an accuracy of 93% when the virtual room number is set to be 16 (equal to the physical functional zones number), which outperforms the best performance achieved by SurroundSense [16], a mobile phone localization system using many kinds of fingerprints relying on site survey. We are delighted even more that such improvement is made while fewer kinds of fingerprints (actually only WiFi here) are involved. The results benefit from the proposed feature of RSS stacking difference and the concept of virtual room. Figure 8 further shows that partitional clustering approaches (KMeans) achieve better performance than others like density-based clustering (EM) and hierarchical clustering methods (FarthestFirst).

Both physical rooms and corridor segments can be partitioned well. As shown in Figure 9, AE of partition on physical rooms is lower than 9%. As we expected, partition on corridor segments is less accurate. Nevertheless, the error is smaller than 19%, which we think is acceptable because fingerprints in corridors are farraginous. In addition, there are no walls or other obstructions between corridor segments, which enlarges the fingerprint similarities between different corridor segments.

It is also indicated that some virtual rooms may be indistinguishable. As illustrated in Figure 10, when virtual room number increases, we observe that AE caused by some specific rooms always keep relatively large. On the other hand, the special building structure and materials of the building, as described above, add to the difficulty of distinguishing rooms, which results in larger AE. We believe WILL would work better in typical modern buildings with walls of reinforced cement.

#### Localization Accuracy

The final localization accuracy is affected by two factors: the virtual room estimation accuracy and the floor plan mapping results. We present the mapping results and evaluate the ultimate localization performance using accuracy of virtual room localization (VRL) and physical room localization (PRL) in the following.

We use the virtual room results generated by KMeans with virtual room number of 16 for following evaluation. The original mapping results of the proposed SSMM on the logical

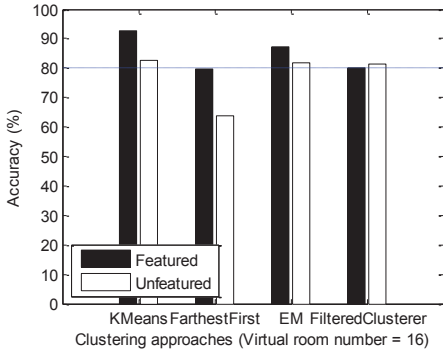


Figure 8: Accuracy of virtual room generation using RSS stacking difference vs. raw fingerprints.

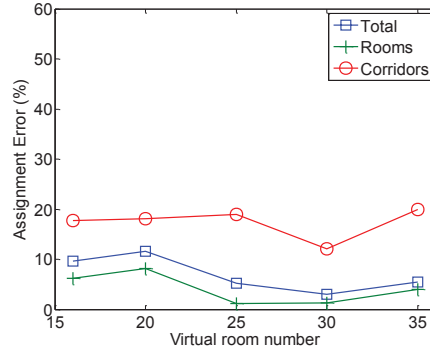


Figure 9: Assignment error with virtual room number (using KMeans)

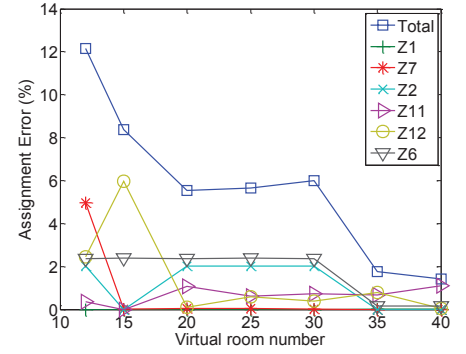


Figure 10: Assignment error caused by different rooms. Rooms causing no AE are not indicated.

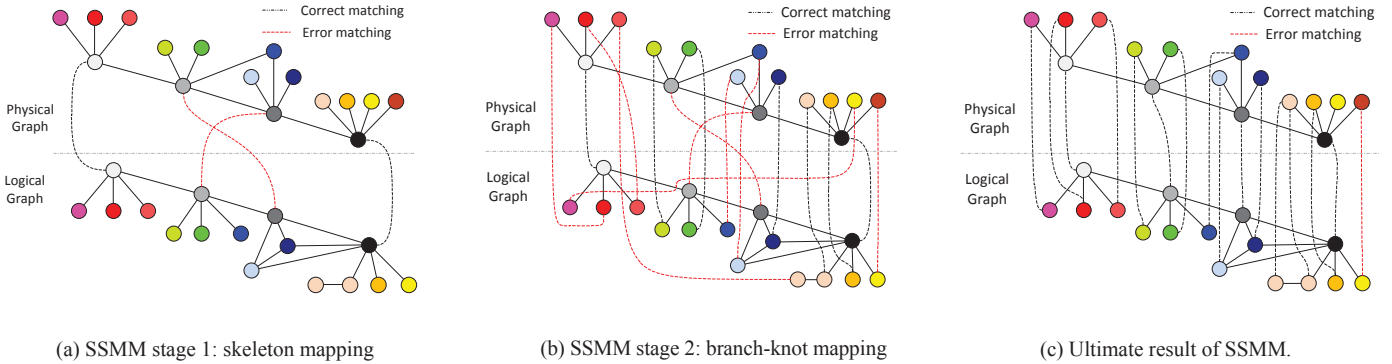


Figure 11: Floor plan mapping. Vertices colored the same indicate ground truth mapping pairs.

and physical graphs are displayed in Figure 11(a) and (b). Some of the mapping errors are corrected in the correction stage of SSMM, as shown in Figure 11(c). As two virtual rooms are marked with the same physical room label, one room in the physical floor plan is not mapped with any virtual rooms. As a final result, 15 out of 16 virtual rooms are correctly mapped.

We evaluate the location estimation accuracy based on the mapping result illustrated in Figure 11(c). To understand the localization accuracy of each room, we plot the cumulative distribution function (CDF) in Figure 12. 75% of physical rooms can achieve localization accuracy of 80% or more. The median accuracies of VRL and PRL are 89% and 90% and the average accuracies of them are 81% and 86%, respectively. Such encouraging results show competitive performance of WILL comparing with traditional site survey based methods.

Furthermore, to evaluate the practical user experience of WILL, we simulated 100 virtual users and observe localization accuracy of their queries. Each user is assigned to a set of 100 to 200 fingerprints, randomly selected from data set #2. The results are depicted in Figure 13, where we see that accuracy of VRL is much higher than that of PRL. Concretely, all users acquire the accuracy of 80% or more for both VRL and PRL. Furthermore, around 60% of these virtual users can enjoy an accuracy of over 90%. The median and average accuracy of per user are both around 90% of PRL.

## VIII. CONCLUSION

Previous indoor localization approaches mostly rely on labor-intensive site survey over every location. In this paper, we presented WILL, an indoor logical localization approach without site survey or knowledge of AP locations and power settings. The main idea is to combine WiFi fingerprints with user movements. Fingerprints are partitioned into different virtual rooms and a logical floor plan is accordingly constructed. Localization is achieved by finding a matching between logical and ground truth floor plan. We implement WILL in a typical office building and it achieves an average room-level accuracy of 86%, which is competitive to existing designs. We believe WILL demonstrates its advantage on low human cost, a long-standing and universal will in wireless indoor localization. Future research in physical floor plan construction, sophisticated floor plan mapping as well as user behavior detection should make WILL a ubiquitous indoor positioning system.

## IX. ACKNOWLEDGMENTS

This work is supported in part by the NSFC Major Program 61190110, NSFC under grant 61171067 and 61133016, National High-Tech R&D Program of China (863) under grant No. 2011AA010100, and China Postdoctoral Science Foundation under grant 2011M500331.



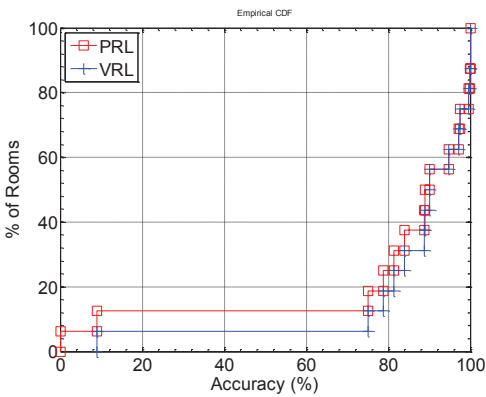


Figure 12: CDF of per room accuracy.

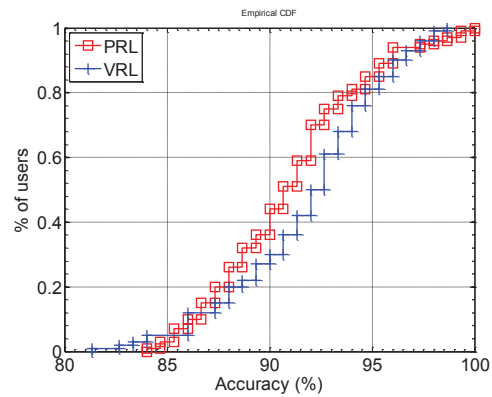


Figure 13: CDF of per user accuracy.

## REFERENCES

- [1] X. Wang, L. Fu, and C. Hu, "Multicast Performance With Hierarchical Cooperation," *IEEE/ACM Transactions on Networking*, vol. 20, no. 3, pp. 1–1, 2011.
- [2] J. Yang, S. Sidhom, G. Chandrasekaran, T. Vu, H. Liu, N. Cekan, Y. Chen, M. Gruteser, and R. P. Martin, "Detecting driver phone use leveraging car speakers," in *Proceedings of the ACM MobiCom*, 2011, pp. 97–108.
- [3] S. Mathur, T. Jin, N. Kasturirangan, J. Chandrasekaran, W. Xue, M. Gruteser, and W. Trappe, "ParkNet: drive-by sensing of road-side parking statistics," in *Proceedings of the ACM MobiSys*, New York, NY, USA, 2010, pp. 123–136.
- [4] 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.
- [5] R. Want, A. Hopper, V. Falcão, and J. Gibbons, "The active badge location system," *ACM Transactions on Information Systems (TOIS)*, vol. 10, no. 1, pp. 91–102, 1992.
- [6] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan, "The cricket location-support system," in *Proceedings of the ACM MobiCom*, 2000, pp. 32–43.
- [7] D. Niculescu and B. Nath, "VOR base stations for indoor 802.11 positioning," in *Proceedings of the ACM MobiCom*, 2004, pp. 58–69.
- [8] M. Youssef, A. Youssef, C. Rieger, U. Shankar, and A. Agrawala, "Pinpoint: An asynchronous time-based location determination system," in *Proceedings of the ACM MobiSys*, 2006, pp. 165–176.
- [9] P. Bahl and V. N. Padmanabhan, "RADAR: An in-building RF-based user location and tracking system," in *Proceedings of the IEEE INFOCOM*, 2000, vol. 2, pp. 775–784.
- [10] M. Youssef and A. Agrawala, "The Horus WLAN location determination system," *Wireless Networks*, vol. 14, no. 3, pp. 357–374, 2008.
- [11] W. G. Griswold, P. Shanahan, S. W. Brown, R. Boyer, M. Ratto, R. B. Shapiro, and T. M. Truong, "ActiveCampus: experiments in community-oriented ubiquitous computing," *Computer*, vol. 37, no. 10, pp. 73–81, 2004.
- [12] Y. C. Cheng, Y. Chawathe, A. LaMarca, and J. Krumm, "Accuracy characterization for metropolitan-scale Wi-Fi localization," in *Proceedings of the ACM MobiSys*, 2005, pp. 233–245.
- [13] J. Park, B. Charrow, D. Curtis, J. Battat, E. Minkov, J. Hicks, S. Teller, and J. Ledlie, "Growing an organic indoor location system," in *Proceedings of the ACM MobiSys*, 2010, pp. 271–284.
- [14] E. S. Bhasker, S. W. Brown, and W. G. Griswold, "Employing User Feedback for Fast, Accurate, Low-Maintenance Geolocationing," in *Proceedings of the IEEE PerCom*, Los Alamitos, CA, USA, 2004, pp. 111 - 120.
- [15] A. Varshavsky, E. de Lara, J. Hightower, A. LaMarca, and V. Otsason, "GSM indoor localization," in *Proceedings of the IEEE PerCom*, vol. 3, no. 6, pp. 698–720, 2007.
- [16] M. Azizyan, I. Constandache, and R. Roy Choudhury, "Surroundsense: mobile phone localization via ambience fingerprinting," in *Proceedings of the ACM MobiCom*, 2009, pp. 261–272.
- [17] H. Lim, L. C. Kung, J. C. Hou, and H. Luo, "Zero-configuration indoor localization over IEEE 802.11 wireless infrastructure," *Wireless Networks*, vol. 16, no. 2, pp. 405–420, 2010.
- [18] K. Chintalapudi, A. Padmanabha Iyer, and V. N. Padmanabhan, "Indoor localization without the pain," in *Proceedings of the ACM MobiCom*, New York, NY, USA, 2010, pp. 173–184.
- [19] Y. Ji, S. Biaz, S. Pandey, and P. Agrawal, "ARIADNE: a dynamic indoor signal map construction and localization system," in *Proceedings of the ACM MobiSys*, 2006, pp. 151–164.
- [20] D. Madigan, E. Einahrawy, R. P. Martin, W. H. Ju, P. Krishnan, and A. S. Krishnakumar, "Bayesian indoor positioning systems," in *Proceedings of the IEEE INFOCOM*, 2005, vol. 2, pp. 1217–1227.
- [21] O. Woodman and R. Harle, "Pedestrian localisation for indoor environments," in *Proceedings of the ACM UbiComp*, 2008, pp. 114–123.
- [22] L. C. Freeman, "A set of measures of centrality based on betweenness," *Sociometry*, pp. 35–41, 1977.
- [23] J. Munkres, "Algorithms for the assignment and transportation problems," *Journal of the Society for Industrial and Applied Mathematics*, pp. 32–38, 1957.