Pers Ubiquit Comput (2017) 21:489–505
DOI 10.1007/s00779-017-1011-7

ORIGINAL ARTICLE

GraphLoc: a graph-based method for indoor subarea localization with zero-configuration

Yuanyi Chen[1] · Minyi Guo[1] · Jiaxing Shen[2] · Jiannong Cao[2]

Received: 8 October 2016 / Accepted: 30 December 2016 / Published online: 20 February 2017

Abstract Indoor subarea localization can facilitate numerous location-based services, such as indoor navigation, indoor POI recommendation and mobile advertising. Most existing subarea localization approaches suffer from two bottlenecks, one is fingerprint-based methods require time-consuming site survey and another is triangulation-based methods are lack of scalability. In this paper, we propose a graph-based method for indoor subarea localization with zero-configuration. Zero-configuration means the proposed method can be directly employed in indoor environment without time-consuming site survey or pre-installing additional infrastructure. To accomplish this, we first utilize two unexploited characteristics of WiFi radio signal strength to generate logical floor graph and then formulate the problem of constructing fingerprint map as a graph isomorphism problem between logical floor graph and physical floor graph. In online localization phase, a Bayesian-based approach is utilized to estimate the unknown subarea. The proposed method has been implemented in a real-world shopping mall, and extensive experimental results show that the proposed method can achieve competitive performance comparing with existing methods.

Keywords Subarea localization · Zero-configuration · Graph-based matching · WiFi radio signal strength

✉ Yuanyi Chen
cyyxz@sjtu.edu.cn

Minyi Guo
guo-my@cs.sjtu.edu.cn

Jiaxing Shen
csjshen@comp.polyu.edu.hk

Jiannong Cao
csjcao@comp.polyu.edu.hk

[1] Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China

[2] Department of Computing, The Hong Kong Polytechnic University, Hung Hom, Hong Kong

1 Introduction

With the increasing number of mobile devices, indoor location-based services, i.e., indoor advertising [29], patient activity monitoring [27] and indoor check-in services [30], are expected to witness a significant growth in the next decade. Recent years have witnessed an increasing attention on indoor subarea localization in view of its importance to indoor location-based services, such as:

- *Indoor advertising* [29], which aims to reach out to a specific section of customers based on their shopping preferences. The key of indoor advertising is uncovering customer's preference, and traditional ways [1, 16] are predominantly field surveys thus are not effective as they need time-consuming and labor intensive survey. Indoor subarea localization can facilitate indoor advertising in terms of customer's check-in activities (e.g., the check-in frequency and stay time in a store) imply their preference, and such kind of check-in activities can be extracted by continuous subarea localization.
- *Mobile localization analytic* [9, 24], which aims to in-depth analyses and utilizes user's location information, which is seen as essential context information and can provide deep insight about people's behavior. Similar to online behavior analysis in the Internet, location can be seen people's physical footstep, which is valuable for understanding people's behavior patterns, for

example, how much time people spend in visiting shops, how people come to these visiting shops (e.g., go directly or just random visit), which interactions with different kinds of shops.

- *Patient activity monitoring* [27, 28], which aims to track bed-ridden patients in hospital environments and report anomalous behavior and emergency. Indoor subarea localization can facilitate patient activity monitoring in terms of two aspects [28]: (1) indoor subarea localization can serve as reliable and accurate asset tracking systems, compared to manual tracking system which are prone to human error; (2) indoor subarea localization can be used to locate the position of patients and medical professionals in real time with room-level accuracy.

Since traditional GPS positioning technique is infeasible in indoor environment and the positioning accuracy of cellular-based method is not enough, localization methods based on radio signal strength (RSS) have attracted enormous research from both academia and industry. Existing localization methods using RSS either require time-consuming site survey or huge costs for deploying additional infrastructure. Therefore, indoor subarea localization remains an unsolved problem according to the report from Microsoft indoor localization competition [19]. Existing localization methods using RSS consist of two categories: infrastructure-based methods and infrastructure-free methods. Infrastructure-based methods require pre-installed hardware for localization, such as UWB [18], ZigBee [8] or wearable sensor [25], which make this kind of system unsalable to large-scale environment. To address this drawback, many infrastructure-free localization methods [12, 33, 38] without requiring additional hardware have been proposed. One of the most promising infrastructure-free localization approach is using WiFi RSS, which is mainly attributed to the widespread deployment of WLAN infrastructure.

Existing localization methods based on WiFi RSS include geometric-based scheme and fingerprint-based scheme. Geometric-based scheme utilizes geometry relation between the unknown location and more than two reference locations for localization, such as TOA [34], TDOA[22] and AOA [13]. Geometric-based scheme requires prior knowledge of WiFi access point (AP) and indoor radio signal propagation model. However, there is not a ubiquitous radio signal propagation model due to complex phenomena (e.g., multi-path fading, shadowing) in indoor environment. Moreover, the performance of geometric-based scheme is sensitive to many factors, such as layout changes or crowd walking. On the contrary, fingerprinting-based scheme is more robust since it does not depend on radio signal propagation model. Typically, fingerprinting-based scheme consists of two phases: (1) offline constructing fingerprint map, which firstly divides indoor space into a few cells and manually associates each cell with the scanned RSS values from surrounding APs; (2) online localization, which estimates the unknown location by matching the scanned RSS values with the fingerprinting map. The main bottleneck of fingerprint-based scheme is that manually constructing fingerprint map is time-consuming and labor intensive. For instance, the deployment overhead for a 300 m^2 environment is more than 7 h [19]. Additionally, the fingerprinting map needs to be updated dynamically for maintaining localization accuracy.

For a practical subarea localization system, several requirements are necessary: reasonable localization accuracy; no additional hardware components on user's side; scalable to large-scale deployment. On this basis, we propose a graph-based indoor subarea localization method with zero-configuration, which is infrastructure-free and constructing fingerprint map by passive crowdsourcing. Specifically, we firstly generate logical floor graph by utilizing two inherent characteristics of WiFi RSS in indoor environment and then we formulate the problem of constructing fingerprinting map as a graph mapping problem between logical floor graph and physical floor graph. Finally, we utilize a Bayesian-based approach to estimate the unknown location.

The rest of this paper is structured as follows. Section 2 surveys related studies on indoor subarea localization. Section 3 describes the proposed method in detail. Section 4 reports and discusses our experimental results. Finally, we present our conclusion and future work in Sect. 5.

2 Related work

In this section, we survey previous related works about indoor subarea localization and discuss how these works differ from our work. In general, existing studies on this topic can be divided into two categories:

2.1 Infrastructure-based localization system

Infrastructure-based localization systems estimate unknown location based on the information from additional infrastructure or external equipment, such as WiFi signals, Bluetooth signals and ZigBee signals. For instance, the beacon frames from multiple Bluetooth APs [20] are used to localization the room, ZigBee interface [8] is used to collect WiFi RSS for room localization, wearable wrist sensors [25] is used to detect a person. The main drawback

of infrastructure-based system is lack of scalability since costly infrastructure pre-deployment is necessary. Moreover, the performance of infrastructure-based systems is limited by disturbances and errors caused by indoor obstacles (e.g., walls, ceiling and furniture). Another challenge of infrastructure-based systems is how to design optimal configurations to trade-off the deployment cost and localization performance. Hossain and Soh [11] analyzed the localization performance and deployment issues by revealing localization error trends with geometric configurations and concluded the optimal configuration is regular polygon where the vertices represent the RSS APs.

2.2 Infrastructure-free localization system

In contrast, infrastructure-free localization systems utilize user's mobile device or existing infrastructure (e.g., WiFi [14, 21, 33, 38, 39], magnetic field [2]) to estimate an unknown location without deploying additional hardware.

Infrastructure-free localization system relies on mobile device usually calculates user's current location according to the previously determined position by built-in sensors (e.g., gyroscope, accelerometer and compass) of mobile devices, which is also called dead reckoning positioning. However, dead reckoning relies on the initial location and will suffer from cumulative error, and continually collecting data from multi-sensor is energy-consuming.

Typically, infrastructure-free localization system consists of geometric-based method and fingerprint-based method. Geometric-based method utilizes triangulation principle to estimate the unknown location based on radio propagation model, such as TOA [34], TDOA [22] and AOA [13]. However, there is not a ubiquitous radio propagation model in indoor environment, since the radio signal propagation would be strongly affected by multi-path effect. In addition, specific devices for measuring TOA or AOA are costly. Fingerprint-based method utilizes the RSS values collected from a specific location as its fingerprint for labeling location. The localization process of this scheme includes two phases: offline construction fingerprint map and online localization. For example, Castelli et al. [6] utilized fingerprint-based method with WiFi RSS to obtain room-level localization for visualizing indoor energy consumption. Hida et al. [10] proposed an subarea detection method using WiFi RSS. Biehl et al. [5] proposed a more robust location fingerprint for localization using the RSS relative ordering of each pair of APs. For reducing erroneous estimation, Hotta et al. [12] utilized the RSS characteristics when passing through a boundary point to calibration. However, previous fingerprinting-based method is infeasible because constructing fingerprint map is time-consuming and labor intensive [19].

Recently, several studies have been proposed to automatically construct fingerprinting map without time-consuming site survey. For instance, Jiang et al. [14] proposed an indoor floor plan construction method with leveraging WiFi RSS and user's motion information, which can be utilized to automatically construct fingerprinting map. WILL [33] automatically construct fingerprint map by utilizing RSS characteristics and user motions too. WicLoc [21] records user motions as well as WiFi signals for constructing fingerprint map. However, these methods need user's active participation when constructing fingerprint map. In contrast, the proposed method only utilizes WiFi RSS to automatically construct fingerprint map, which can be done by passive crowdsourcing.

3 Graph-based localization method

In this section, we first introduce the key data structures and notations used in the proposed subarea localization method and then present the problem definition and solution.

3.1 Problem definition

For ease of the following presentation, we define the key notations used in the proposed method. Table 1 lists the relevant notations used in this paper.

Definition 1 (*RSS record*) A RSS record is a triple $o(u, t, R)$ that means the collected WiFi RSS values by user u at time t. R is a K-dimensional vector and denotes by

Table 1 Notations used in indoor subarea localization

Symbol	Description
N, K	The number of subareas, WiFi APs
M, F	The number of RSS traces, floors
S, D	The set of subareas, RSS traces
H	The set of Histogram bins
r^i	The RSS value from ap_i
R	The RSS values from all WiFi APs
$o(u, t, R)$	The RSS record collected by u at time t
$L_i, traj(L_i)$	A WiFi RSS trace, a virtual trajectory
f_{si}	The fingerprint of subarea s_i
v_i	Virtual subarea with similarity fingerprint
Y	The fingerprint map
G_p, G_f	The physical floor graph
G_f	The logical floor graph
σ, τ	User-specific thresholds

$(r^1, \ldots, r^i, \ldots, r^K)$, r^i means the scanned WiFi RSS value from AP ap_i, K is the number of WiFi APs in indoor space and $1 \leqslant i \leqslant K$.

Definition 2 (*WiFi RSS trace*) We define a WiFi RSS trace as a sequence of RSS records and denote by $L = \{o_1, \ldots, o_i, \ldots, o_T\}$, o_i represents the collected RSS record at time t_i, $1 \leqslant i \leqslant T$.

Definition 3 (*Indoor subarea*) $S = \{s_1, s_2, \ldots, s_N\}$ denotes the set of subareas, N is the num of subareas and a subarea s_i refers to a region that makes up part of indoor space. Typically, subareas are rectangle, such as rooms and corridors, but not necessary

Definition 4 (*Subarea fingerprint*) The feature of subarea s_i is defined as a $H \times K$ matrix $f_{si} = \{p_1, p_2, \ldots, p_K\}$, H is the histogram bins and p_j represents the histogram of scanned RSS values from ap_j in s_i, $1 \leqslant i \leqslant N$ and $1 \leqslant j \leqslant K$.

We split the RSS values range into H bins and then p_j denote by a H-dimensional vector; a bin-based method is used to calculate the p_j of subarea s_i, as shown in Eq. 1.

$$p_j = \prod_{h=1}^{H} \frac{\sum_{j=1}^{K} C_{ij}^h}{C_i} \tag{1}$$

where $\sum_{j=1}^{K} C_{ij}^h$ is the num of collected RSS values from ap_j belongs to the h-th bin in total collected RSS values, C_i means the total collected RSS values in subarea s_i.

Definition 5 (*Fingerprint similarity*) The fingerprint similarity of subarea s_i and s_j is calculated by cosine similarity, as shown in Eq. 2.

$$Sim(f_{si}, f_{sj}) = \frac{1}{K} \sum_{n=1}^{K} \frac{Row_n(f_{si}) \cdot Row_n(f_{sj})}{||Row_n(f_{si})|| \times ||Row_n(f_{sj})||} \tag{2}$$

where $Row_n(f_{si})$ and $Row_n(f_{sj})$ represent the n-th row vector of f_{si} and f_{sj}, respectively.

Definition 6 (*Fingerprint map*) The fingerprint map is a set of tuples by associating physical subarea and its fingerprint and denote by $Y = \{(s_1, f_{s1}, \ldots, (s_i, f_{si}), \ldots, (s_N, f_{sN}))\}$.

Definition 7 (*Physical floor graph*) We denote the physical floor graph by $G_p = \langle V_p, E_p \rangle$, where $V_p = \{v_1, v_2, \ldots, v_N\}$ and v_i represents subarea s_i, $E_p \subseteq V \times V$ correspond to the directly reachable of subareas in indoor space.

Based on the above definitions, we formulate the problem of indoor subarea localization as: Given: (1) indoor subarea set $S = \{s_1, s_2, \ldots, s_N\}$. (2) WiFi RSS Trace set $D = \{L_1, L_2, \ldots, L_M\}$ collected by passive crowdsourcing. (3) physical floor graph $G_p = \langle V_p, E_p \rangle$. (4) a user localization request $o_?(u, t, R)$; Objective: find the correspond subarea s_i when scanning RSS record $o_?(u, t, R)$.

Our solution for this problem consists of two phases: (1) construct fingerprint map by graph mapping; (2) estimate the unknown subarea with a Bayesian approach.

3.2 Construct fingerprint map

In this subsection, we first give a high-level overview of the graph-based method for constructing fingerprint map (as shown in Fig. 1) and then present the details of the method.

Unlike existing fingerprint-based methods, our method automatically constructs fingerprint map without manual site survey. First, we collect RSS traces by crowdsourcing (e.g., when participants go shopping, drink a coffee or relaxing). Then, after obtaining enormous RSS traces, the fingerprint map is constructed by the following three steps: modeling physical floor plan to an undirected graph, generate logical floor graph, and mapping logical floor graph to physical floor graph.

3.2.1 Modeling physical floor plan

Motivated by indoor robots pursuit/evasion research [15], we model the indoor floor plan with a undirected graph $G_p = \langle V_p, E_p \rangle$ by decomposing the indoor floor plan into a collection of convex subareas and further reduce the indoor space to a graph by discretization. Specifically, the discretization includes two steps:

- *Step 1* decomposing the indoor floor plan into a set of convex subareas based on critical visibility events and association vertex v_i to subarea s_i;
- *Step 2* adding edges between vertices which are directly connected in the original indoor floor plan.

For example, the indoor floor plan of our experimental environment is shown in Fig. 2a, which consisting of 27 rooms and covering over 2000 m^2. Then, we decompose the floor plan into a set of subareas and add edges between directly connected vertices and finally model the indoor floor plan as an undirected graph as shown in Fig. 2b.

3.2.2 Generate logical floor graph

Typically, large indoor space (e.g., urban shopping mall, museum and airport) is multi-floor environment with hundreds of WiFi APs to provide WiFi service. For generating logical floor graph, we need to cluster the RSS records that collected from the same floor.

(1) *Cluster RSS records to the same floor* Several factors can influence the propagation of WiFi radio signal in indoor environment, such as people walking, layout change

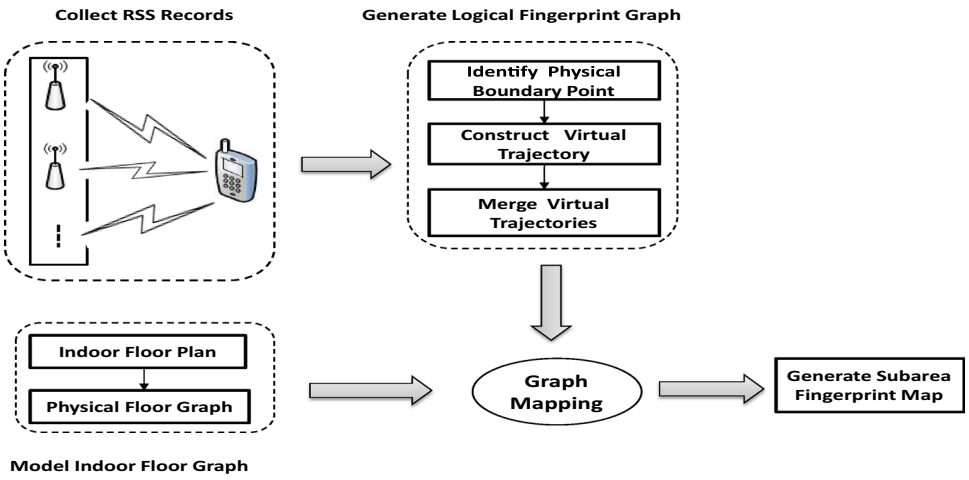

Fig. 1 High-level overview of constructing fingerprint map

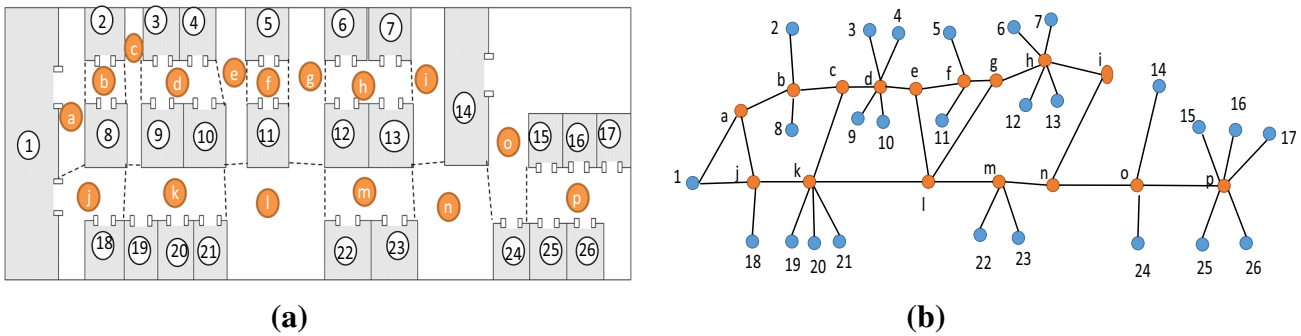

Fig. 2 Modeling physical floor plan as a undirected graph, **a** indoor floor plan, **b** physical floor graph

and multiple diffraction from window frames. According to [17], one floor may weaken WiFi RSS values between 15dBm and 35dBm. Therefore, the range of RSS values from a specific AP is useful for floor recognition [36].

Formally, let $R = (r^1, \ldots, r^i, \ldots, r^K)$ denote the scanned RSS record from surround WiFi APs, $L(x, y, f)$ denote the location coordinate where the RSS record is collected, where (x, y) is the two-dimensional coordinate of location and f is the floor. Ideally, each location can correspond to a unique WiFi RSS record, and we can cluster WiFi RSS records to the same floor by reducing high-dimensional RSS record to a 3-dimensional vector. However, WiFi RSS record is very unstable [18, 19] even at the same location due to a few factors, such as heterogeneous devices, environmental change or crowd walking. We cluster RSS records to the same floor by the following two steps:

- *Step 1* using Laplacian Eigenmaps [4] to reduce the RSS values with high-dimension to d-dimension vector ($d > 2$).
- *Step 2* clustering the d-dimension vectors to F classes by k-means algorithm, where F is the number of floors. In the clustering process, we use the Euclidean distance to measure the closeness of two vectors.

(2) *Construct logical floor graph* A few factors can influence the propagation of radio signal in indoor environment, such as multiple diffraction, reflection of scattered signals from adjacent walls and crowd walking. By investigating spatial–temporal characteristics of indoor radio signal propagation, we observe two valuable characteristics can be exploited to subarea localization.

The first observation is physical obstacles, such as walls and stairs, will make WiFi RSS values jump dramatically. In order to investigate the physical obstacles effect on radio signal propagation, we collected 200 RSS records from three APs in room 1 and room 2, where AP1 and AP2 are located in room 1 and AP3 is located in room 2. Statistical information of RSS values is shown in Table 2, and we can observe that the range of RSS values from the same AP significantly differ in different rooms.

Therefore, this characteristic can reflect the indoor floor plan to a certain degree and can be used to distinguish two subareas, which is also demonstrated in [7]. Based on this characteristic, we design a robust subarea fingerprint using RSS histogram as shown in Definition 4. In order to distinguish different subareas, we further define the similarity of subarea fingerprint as shown in Definition 5.

Table 2 The RSS values scanned from three WiFi APs at different rooms

Range	AP1 at Room 1	AP1 at Room 2	AP2 at Room 1	AP2 at Room 2	AP3 at Room 1	AP3 at Room 2
$[-55,-40]$	115	0	93	1	0	120
$[-70,-55)$	72	3	81	5	4	63
$[-85,-70)$	10	11	21	17	21	15
$[-100,-85)$	3	23	5	39	42	2

Take RSS values of Table 2 as an example, split the range of RSS values into 4 bins: $\{(-40,-55],(-55,-70],(-70,-85],(-85,-100]\}$, the fingerprint of room 1 and room 2 can be calculated as f_{s1} and f_{s2}, respectively.

$$f_{s1} = \begin{bmatrix} 0.575 & 0.36 & 0.05 & 0.015 \\ 0.465 & 0.405 & 0.105 & 0.025 \\ 0 & 0.0597 & 0.3134 & 0.6269 \end{bmatrix} \quad (3)$$

$$f_{s2} = \begin{bmatrix} 0 & 0.0811 & 0.2973 & 0.6216 \\ 0.0161 & 0.0806 & 0.2742 & 0.629 \\ 0.6 & 0.315 & 0.075 & 0.01 \end{bmatrix} \quad (4)$$

The second observation is the WiFi RSS values will jump dramatically when passing a physical boundary point, such as room entrances and corners. For example, we collect a sequence of RSS values from three APs when walking from room 1 to room 2, as shown in Fig. 3a. Specifically, $\{t1,t2,t3,t4,t5\}$ are collected in room 1, $\{t6,t7,t8\}$ are collected when passing the entrance, $\{t9,t10,t11,t12\}$ are collected in room 2, as shown in Fig. 3b. We find that the "jump" range can reach 15dBm-30dBm. However, the RSS values should change smoothly in a small continuous area according to indoor empirical propagation model [26]. Therefore, the RSS "jump" characteristic when passing boundary points can be utilized to identify subarea entrance.

Based on the two spatial–temporal characteristics of radio signal propagation in indoor environment, we generate logical floor graph by three stages, as shown in Fig. 4. Specifically, we first identify all physical boundary points based on the RSS "jump" characteristic when passing a physical boundary point and remove false identification using subarea fingerprint similarity. Then, we partition a WiFi RSS trace into a virtual trajectory according to physical boundary points, as shown in Fig. 4. Finally, we merge all virtual trajectories to generate logical floor graph, as shown in Fig. 5.

Identify physical boundary points Based on the observation that the WiFi RSS values will jump significantly when walking through a physical boundary point, we utilize the fluctuation of RSS values in a small time window to identify physical boundary points. Formally, given a WiFi RSS trace $L = \langle o_1,\ldots,o_i,\ldots,o_T\rangle$, we define $Var(t_i,\tau)$ to represent the RSS fluctuation in time window $(t_i-\tau/2, t_i+\tau/2)$, as shown in Eq. 5.

$$Var(t_i,\tau) = \frac{1}{K}\sum_{i=1}^{K} Var(ap_i) \quad (5)$$

where K is the number of WiFi APs, $Var(ap_i)$ is the variation of RSS values from ap_i during the time window, as calculated in Eq. 6.

$$Var(ap_i) = \frac{1}{\tau-1}\sum_{j=t_i-\tau/2}^{t_i+\tau/2}(r_j^i - \overline{r^i})^2 \quad (6)$$

where $\overline{r^i}$ is the average RSS values from ap_i in time window $(t_i-\tau/2, t_i+\tau/2)$, r_j^i is the RSS value from ap_i at time t_j.

If the RSS fluctuation in time window $(t_i-\tau/2, t_i+\tau/2)$ is significantly higher than average, we can infer the user is walking through a physical boundary point at time

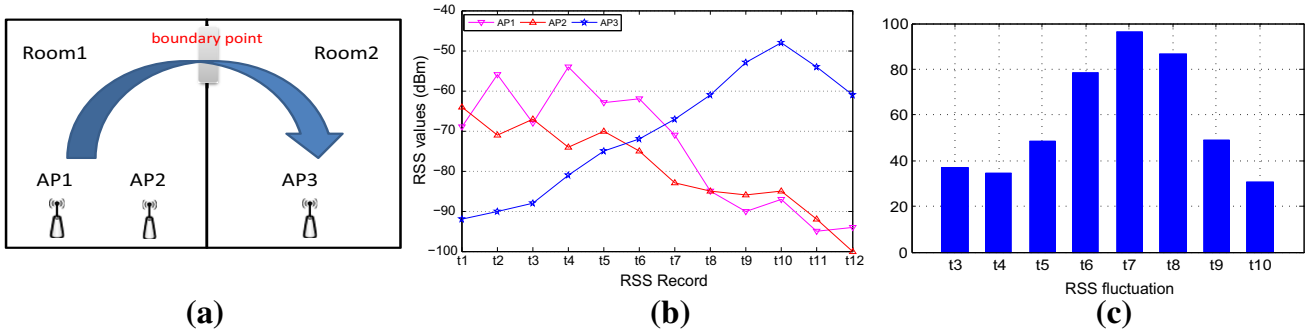

Fig. 3 The "jump" characteristic when passing physical boundary points

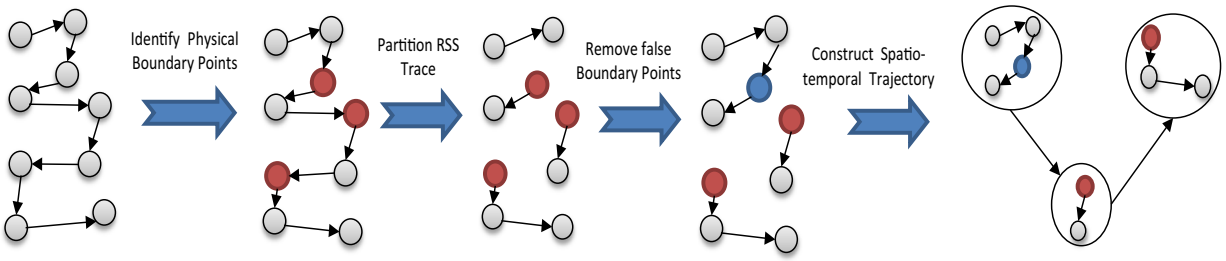

Fig. 4 Construct virtual trajectory of WiFi RSS trace

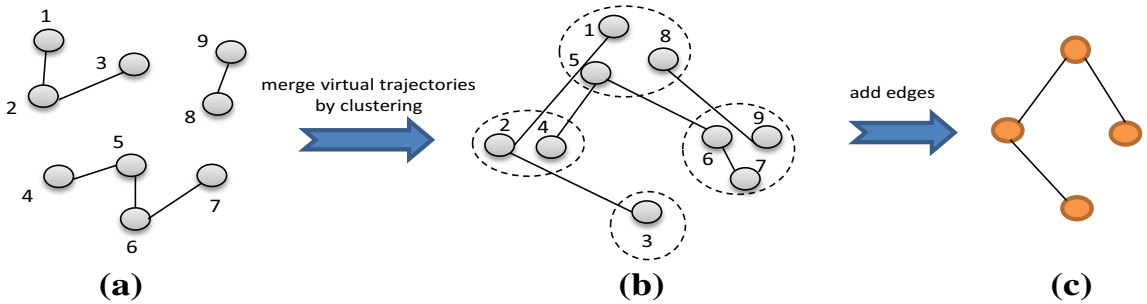

Fig. 5 Constructing logical floor graph. **a** The virtual trajectories. **b** Using K-means to cluster all elements of virtual trajectories. **c** Adding edges between two clusters if they are reachable

Table 3 The variation of RSS values from three WiFi APs

time window	($t1$, $t5$)	($t2$, $t6$)	($t3$, $t7$)	($t4$, $t8$)	($t5$, $t9$)	($t6$, $t10$)	($t7$, $t11$)	($t8$, $t12$)
AP1	46.5	31.8	42.3	137.5	162.7	143.5	80.8	18.7
AP2	14.7	10.3	36.7	40.3	48.7	20.2	11.7	42.3
AP3	49.7	61.7	66.3	58.2	77.8	96.7	55.3	31.3

t_i. Formally, we use variation coefficient α to quantify the degree of RSS "jump", as shown in Eq. 7.

$$\alpha = \frac{\tau \times Var(t_i, \tau)}{\sum_{j=t_i-\tau/2}^{t_i+\tau/2} Var(t_j, \tau)} \tag{7}$$

For example, set time window size $\tau = 5$ and variation coefficient as $\alpha = 1.3$, the variation of RSS values from three APs in Fig. 3b is calculated as shown in Table 3. We further calculate the RSS fluctuation:

$$V = \{36.97, 34.6, 48.4, 78.67, 96.4, 86.8, 49.27, 30.77\}$$

as shown in Fig. 3c and infer the user is passing a physical boundary point in time $\{t_6, t_7, t_8\}$.

Remove false identification As mentioned above, we identify physical boundary points according to the RSS "jump" characteristic. However, this method may bring some false positives, since other factors (e.g., crowd passing and furniture layout change) may create similar RSS "jump". However, subarea fingerprint using RSS histogram is stable and robust according to the first observation. On the basis, we remove false positives based on the similarity of subarea fingerprint.

Formally, after obtaining time set $\Omega = \{t_p, t_{p+1}, \ldots, t_q\}$ that users may walk through physical boundary points according to RSS "jump" characteristic, we partition RSS trace L into a subsequence set $L = \{o(t_1 : t_p), o(t_p : t_{p+1}), \ldots, o(t_{q-1} : t_q), o(t_q : t_T)\}$, $o(t_p : t_{p+1})$ is the RSS subsequence collected from t_p to t_{p+1}. Then, we calculate the fingerprint of each RSS subsequence as denote by $F = \{f_p, f_{p+1}, \ldots, f_q\}$, f_p represents the fingerprint of RSS subsequence $o(t_1 : t_p)$. Finally, we use a threshold-based approach to remove false positives, which means t_{p+1} is a false positive if the fingerprint similarity between f_p and f_{p+1} is greater than a threshold δ, as shown in Eq. 8.

$$Sim(f_p, f_{p+1}) > \delta \tag{8}$$

Construct virtual trajectory After removing false identification of physical boundary points, we repartition the RSS trace L into a subsequence set $L = \{o(t_1 : t_p), o(t_p : t_{p+1}), \ldots\}$ and map each RSS subsequence $o(t_p : t_{p+1})$ to a virtual subarea v_{p+1}. A virtual subarea is a container which consists of fingerprint with high similarity. Finally, we construct the virtual trajectory of RSS trace L as $traj(L) = < v_p \rightarrow v_{p+1} \rightarrow \cdots >$, as shown in Fig. 4.

Generate logical floor graph After constructing virtual trajectory for each RSS trace, we generate logical floor graph $G_f(V_f, E_f)$ by merging all virtual trajectories $\{traj(L_1), traj(L_2), \ldots, traj(L_M)\}$. Specifically, the merge process consists of two steps:

- *Step 1* using K-means algorithm to cluster all elements of virtual trajectories $\{traj(L_1), traj(L_2), \ldots, traj(L_M)\}$ into P classes, and mapping class center π_i of cluster P_i to vertex v_i of logical floor graph, as shown in Fig. 5b. Since traditional K-means algorithm is sensitive to initial cluster centers, the selection of initial cluster centers directly affects the accuracy and stability of the clustering results. To solve this problem, we utilize the elements density distribution to optimize the selection of initial cluster center.

Definition 8 (*Elements distance*) For two elements x_i and x_j of virtual trajectories, we calculate their distance as:

$$d(x_i, x_j) = 1 - Sim(f_{xi}, f_{xj}) \tag{9}$$

where $Sim(f_{xi}, f_{xj})$ is calculated as Equation 2.

Definition 9 (*Element density*) For an element x_i, we select its k-nearest neighbor elements Ω_i according to elements distance. Then, we define the density of x_i as:

$$dens(x_i) = \frac{1}{k} \sum_{x_j \in \Omega_i} d(x_i, x_j) \tag{10}$$

Algorithm 1 formally describes the framework of the proposed method for selecting initial cluster centers of k-means. First, as shown in Lines 2−5, we calculate the density for all elements of virtual trajectories and sort the elements according to element density. Then, as depicted in Line 7−12, we choose a unvisited element with the highest density and generate its k-nearest neighbors. Finally, we select the gravity center of the k-nearest neighbors as a cluster center. In the clustering process, we use the fingerprint similarity (See in Definition 5) to measure the closeness of two elements.

Algorithm 1 Density-based algorithm for selecting initial cluster centers of k-means

Require: 1) Element set of all virtual trajectories: $X = \{x_1, x_2, ...\}$; 2) the number of local neighbors: K; 3) the number of cluster centers: p

Ensure: The initial cluster centers: $\pi = \{\pi_1, \pi_2, ...\pi_k\}$

1: Label all elements of X as unvisited, $\pi = \emptyset$
2: **for** $\forall x_i \in X$ **do**
3: Select its k-nearest neighbor set Ω_i according to elements distance.
4: Calculate the local density $dens(x_i)$ according to Equation. 10.
5: **end for**
6: Sort X according to the local density of element, denote as X'.
7: **while** the number of π is less than p **do**
8: **for** $\forall x_j \in X'$ and x_j is unvisited **do**
9: Calculate the cluster center: $\pi_j = \frac{1}{k} \sum_{x_i \in \Omega_j} x_i$,
10: Label x_j and elements of Ω_j as visited, add π_j to π.
11: **end for**
12: **end while**
13: **return** cluster center set π.

- *Step 2* adding an edge between v_i and v_j if cluster P_i and cluster P_j is reachable, which means that there is at least one pair of adjacent virtual subareas $\langle v_i \rightarrow v_j \rangle$ for $\forall v_i \in P_i$ and $\forall v_j \in P_j$, as shown in Fig. 5c.

3.2.3 Mapping logical floor graph to physical floor graph

For automatically constructing fingerprint map, we need to associate virtual subarea v_i to the corresponding subarea s_j by mapping logical floor graph to physical floor graph. Formally, given logical floor graph $G_f = \langle V_f, E_f \rangle$ and physical floor graph $G_p = \langle V_p, E_p \rangle$, find a mapping function $\tau : V_f \rightarrow V_p$ for $\forall e(u, v) \in E_f, e(\tau(u), \tau(v)) \in E_p$. Obviously, this is a subgraph isomorphism problem and can be solved by Ullman algorithm [31].

Graph matching Ullman algorithm utilizes a depth-first search strategy to enumerate all subgraphs of G_f that matching G_p. For ease of understanding, Fig. 6c is the search tree for mapping G_f (Fig. 6a) to G_p (Fig. 6b), the i-th layer of search tree represents mapping u_i of G_f to each node of G_p, and a path from root node to leaf node represents a subgraph matching between G_p and G_f. A subgraph matching is correct if the adjacency relationship of u_i in G_f is the same as its mapping node v_j in G_p.

Since we have mapped each virtual subarea v_i to the corresponding physical subarea s_j, we further compute the fingerprint of s_j according to Eq. 1. Then, we construct subarea fingerprint map with associating s_j to the calculated fingerprint.

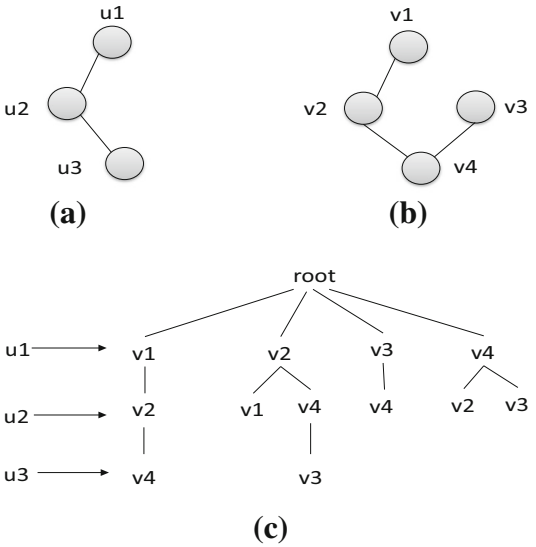

Fig. 6 Mapping logical floor graph to physical floor graph. **a** Logical floor graph: G_f, **b** physical floor graph: G_p, **c** search tree for mapping G_f to G_p

Correction As shown in Fig. 6, the existence of symmetric subgraphs may lead to matching errors when mapping logical floor graph to physical floor graph. We perform the correction stage to fix some error mapping. Indoor space can typically be divided into places with different functionalities, such as smart houses usually consist of kitchens, bedrooms and seminar rooms, shopping malls provide various leisure and food facilities (e.g., cafes, game centers and theaters, as shown in Fig. 7). The basic idea is the unique relationship between WiFi RSS and different types of subareas due to signal reflection, refraction and diffraction. In other words, different subareas vary in internal structures and human activities that can

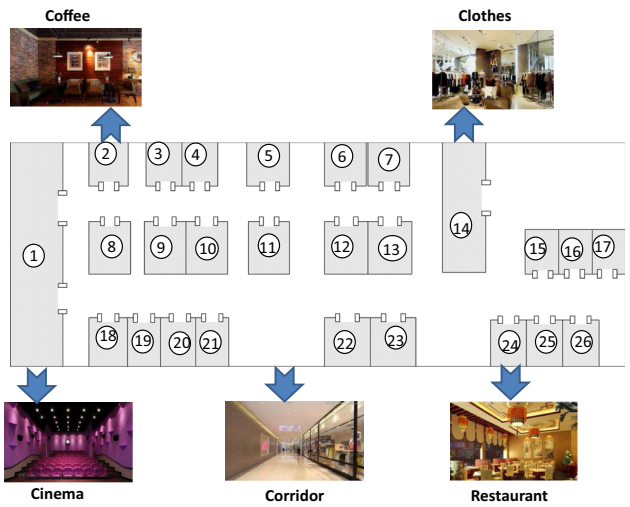

Fig. 7 A shopping mall with coffee, restaurant, clothes, cinema and corridor, etc

be reflected by RSS characteristics. Based on these observations, we extract two kinds of semantic features to correct some mapping errors:

- *Average stay time* After constructing the virtual trajectories, we can extract the average stay time of all check-ins in each subarea. Figure 8a reports the distribution of average stay time with different types of subareas using a real-world dataset collected over 33 days. As shown in Fig. 8a, the average stay time have very different distribution pattern in terms of different types of subareas. For instance, the average stay time for subareas belong to *Clothes* is mainly between 0 and 30 min, while subareas with the stay time is more than 60 min have a high probability belong to *Cinema*. Therefore, we consider the average stay time to be a very useful feature for distinguishing different types of subareas.
- *Temporal distribution* To show the temporal pattern of different types of subareas, we aggregate the number of virtual trajectories at a specific timestamp. In particular, the opening hours of the shopping mall are 10:00 a.m.–10:00 p.m. and we empirically divided a day into 5 timestamps: (1) Morning—hours between 10 a.m. and 12 p.m.; (2) Noon–hours between 12 p.m. and 2 p.m.; (3) Afternoon—hours between 2 p.m. and 5 p.m.; (4) Dinnertime—hours between 5 p.m. and 7 p.m.; (5) Night—hours between 7 p.m. and 10 p.m. From Fig. 8b, we can observe two very different temporal patterns corresponding to two kinds of subareas (i.e., Restaurant and Cinema). For instance, the subareas belong to *Restaurant* have clearly two peak periods, corresponding to lunch and dinner time, respectively. On the contrary, for subareas belong to *Cinema*, there is one peak period (from 7:00 p.m. to 10:00 p.m.). Therefore, the temporal distribution is discriminative for the classification of subareas such as belong to *Restaurant* and *Cinema*.

In our approach, we combine the ratio of average stay time in different ranges and the temporal distribution together as inputs for training a SVM classifier for different types of subareas, which can further correct mapping errors. For instance, the virtual subarea u_1 may be mapped to physical subarea v_1 or v_2 in Fig. 6, if more than 70% of check-ins in u_1 are less than 15 min, we should map u_1 to v_1 rather than v_2 (as v_1 is a subarea belongs to corridor, while v_2 is a restaurant given by the shopping mall owner).

3.3 Online localization

At the online localization part, user sends localization request with submitting the scanned RSS record $o(u, t, R)$, $R = \{r^1, r^2, \ldots, r^K\}$, our method estimates the subarea of

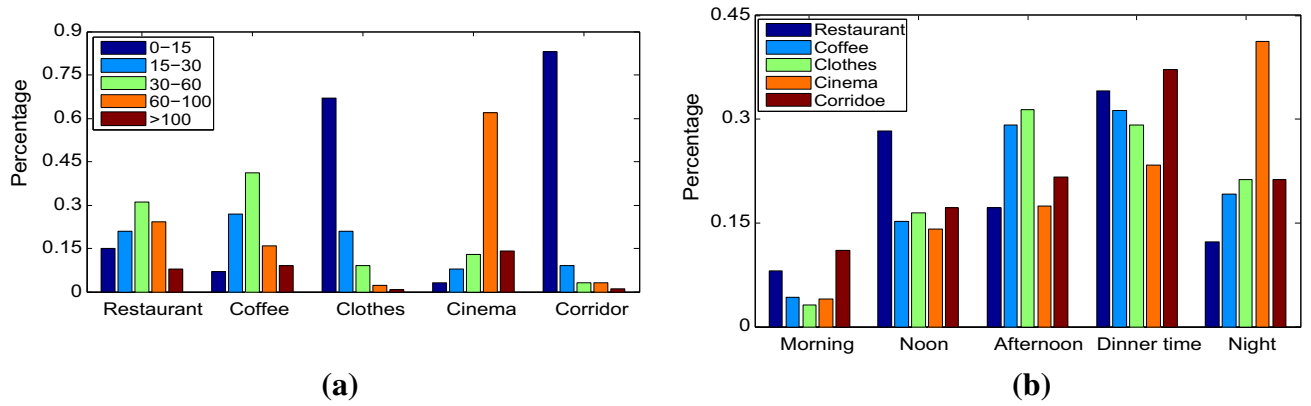

Fig. 8 The semantic features of subareas for correcting mapping errors. **a** Average stay time of different types of subareas. **b** Temporal distribution of different types of subareas

his/her current location using a Bayesian approach. According to Bayesian inference, the posterior probability $P(s_i|R)$ can be calculated as Eq. 11.

$$P(s_i|R) = \frac{P(R|s_i)P(s_i)}{P(R)} \tag{11}$$

Since the prior probability that user is located in each subarea is equal and the RSS values from different WiFi APs are independent, the posterior probability $P(s_i|R)$ can further be simplified as Eq. 12.

$$P(s_i|R) \propto \prod_{j=1}^{K} P(r^j|s_i) \tag{12}$$

For a given subarea s_i, the prior probability $P(r^j|s_i)$ can be calculated by the normalized histogram of ap_j in this subarea. We partitioned the RSS values range into H bins when constructing fingerprint map, suppose r^j belongs to the h-th bin, $P(r^j|s_i)$ is equal to $f_{si}(h,j)$. Then, the localization result for RSS record $o(u, t, R)$, $R = \{r^1, r^2, \ldots, r^K\}$ can be estimated by Eq. 13.

$$\widehat{s} = \underset{s_i \in S}{argMax} \prod_{j=1}^{K} f_{si}(h,j) \tag{13}$$

Algorithm 2 formally describes the framework of our proposed method for indoor subarea localization. First, as shown in Lines 2−3, we first cluster the RSS records to the same floor. Then, we generate the logical floor graph based on two unexploited RSS characteristics in indoor space as shown in Lines 5−10. Finally, as depicted in Line 11−12, we construct subarea fingerprint map by mapping logical floor graph to physical floor graph. At the online localization part, we calculate the posterior probability for each subarea based by Bayesian inference, as shown in Line 14−17. Finally, we select the subarea with the maximum posterior probability as the localization result.

Algorithm 2 Graph-based method for indoor subarea localization

Require: 1) The RSS traces set $D = \{L_1, L_2, ..., L_M\}$; 2) Subarea set $S = \{s_1, s_2, ..., s_N\}$; 3) The number of floors: F; 4) Physical floor graph G_p; 5) user-specific threshold: τ, α, δ, d; 6)The RSS record of user's localization request: $o < u, t, R >$ and $R = \{r^1, r^2, ..., r^K\}$.
Ensure: The subarea s_u of user's current location
1: ∗∗∗Phase 1: Cluster RSS Records∗∗∗
2: reduce the RSS values d-dimension vector by Laplacian Eigenmaps.
3: cluster the d-dimension vectors to F classes by k-means algorithm.
4: ∗∗∗Phase 2: Construct Fingerprint Map∗∗∗
5: **for** $\forall L_i \in D$ **do**
6: Identify physical boundary points according to Equation. 7.
7: Remove false identification according to Equation. 8.
8: Construct virtual trajectory $traj(L_i)$.
9: **end for**
10: Generate logical floor graph G_f by merging virtual trajectories $\{traj(L_1), traj(L_2), ..., traj(L_M)\}$.
11: Map logical floor graph G_f to physical floor graph G_p.
12: Construct subarea fingerprint map $Y = \{(s_1, f_{s1}, ..., (s_i, f_{si}), ..., (s_N, f_{sN}))\}$.
13: ∗∗∗Phase 3: online localization∗∗∗
14: **for** $\forall (s_i, f_{si}) \in Y$ **do**
15: Otain the histogram bin h that r^j belongs to.
16: Calculate the probability $P(s_i|R) = \prod_{j=1}^{K} f_{si}(h,j)$
17: **end for**
18: **return** s_u=arg $\underset{s_i \in S}{Max} P(s_i|R)$.

4 Experiment evaluation

In this section, we first describe the experimental setting and dataset for evaluation. Then, we report the results of a series of experiments conducted to evaluate the

performance of our proposed method for indoor subarea localization, followed by discussions.

4.1 Experimental datasets

Our experimental environment is a large indoor shopping mall with four floors, and each floor is about 55 m × 30 m.

4.1.1 Dataset for floor clustering

To evaluate the method for clustering the RSS records to the same floor, we need to label the floor that the RSS record is collected. Finally, we collect 3948 RSS records with floor information with a sampling rate of 1 Hz in total; more details about this dataset are shown in Table 4. After the analysis, there are 275 different WiFi APs; then we extend each RSS sample to a 275 dimensional vectors and set -110 dBm as default value for WiFi AP without collecting RSS values.

4.1.2 Dataset for subarea localization

We evaluate the proposed subarea localization algorithm at one floor with 26 shops and 7 corridors. Each shop is regarded as a subarea and corridors are partitioned to 16 subareas, so there are 42 subareas in total. The floor plan and subarea partition is shown in Fig. 2.

To evaluate our subarea localization method, we need to record two labeled information: the subarea and whether the location is a physical boundary point of each WiFi RSS record. We develop a mobile application to collect WiFi RSS samples with a sampling rate of 1 Hz, and each sample is represented by a tuple: $\langle L, o\rangle$. Specifically, $L = \{s_i, 0|1\}$ is the label information: floor, subarea and whether is a physical boundary point, o is the scanned RSS record from surround WiFi APs and represented by a triple $(M, t, \langle r^1, r^2, \ldots, r^K\rangle)$, M is the MAC address of collection device and t is the collection time, r^1 is the scanned RSS values from AP1. Note that we collect RSS information with a sampling rate of 1 Hz at the offline phase for constructing fingerprint map and users only need to submit the single RSS sample in online localization without continuously submitting RSS information.

We collect 117 WiFi RSS traces for experiment evaluation by 25 participants (including students and shop workers) over 33 days, in which one RSS trace includes an average of 10 subareas and 1532 RSS records, and each subarea has been visited by at least three participants. Statistically, there are 123 different WiFi APs and 179241 WiFi records. For constructing subarea fingerprint and calculating fingerprint similarity, we extent each RSS sample to a 127 dimensional vectors, as shown in Table 5. For WiFi AP without collecting RSS values, we set −110 dBm as default value, and one example of RSS samples is shown in Table 6.

Table 4 Dataset for floor clustering

	Floor 1	Floor 2	Floor 3	Floor 4
# of RSS records	1005	1471	760	712
# of different WiFi APs	81	72	56	66

4.2 Experimental results

4.2.1 Cluster RSS records

We use Fowlkes–Mallows index [32] to evaluate the performance of cluster algorithm. Let *TP* denote the number of true positives, *FP* denote the number of false positives, and *FN* denote the number of false negatives, the Fowlkes–Mallows index (FMI) is calculated by:

$$\mathrm{FMI} = \sqrt{\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}} \cdot \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}} \tag{14}$$

Tuning parameters of cluster algorithm, such as the number of clusters and the dimension of RSS records after reduction, are critical to the performance of clustering RSS records to the same floor. Figure 9a reports the clustering performance (FMI) with different number of clusters and different dimensions of RSS records. From this figure, we observe: (1) the best clustering performance is achieved when setting the number of clusters equal to 4, which is the number of floors. For example, the FMI is 43.6% using the raw RSS records when the number of clusters equal to 4; (2) the clustering performance using low-dimension vectors after reducing have an obvious improvement compared to use the raw RSS records, showing the advantages of using Laplacian Eigenmaps to reduce the raw RSS records to low-dimension vector. For instance, the best clustering performance is achieved when reducing the raw RSS records to three-dimensional vector and setting the number of clusters equal to 4. The reason is dimension reduction based on Laplacian Eigenmaps can find the manifold structure of raw RSS records.

In Fig. 9b, we compare the clustering time of using raw RSS records and low-dimension vectors after reducing, the clustering time is obtained after repeating the experiments 10 times on Intel's Core i5 based computer. It can be seen from this figure that clustering with raw RSS records consumes much more time than with low-dimension vectors after reducing. The reason is large indoor space with multi-floor usually has hundreds of available WiFi APs (e.g., there are 275 WiFi APs in our experiment). Thus,

Table 5 The RSS sample format

001	002	...	123	124	125	126	127
RSS value	RSS value	...	RSS value	Timestamp	Phone ID	Boundary point flag	Subarea ID

Table 6 One example of RSS sample

[001]	[002]	...	[123]	[124]	[125]	[126]	[127]
−73	−65	...	−87	2015-12-07 15:28:15	1	0	1

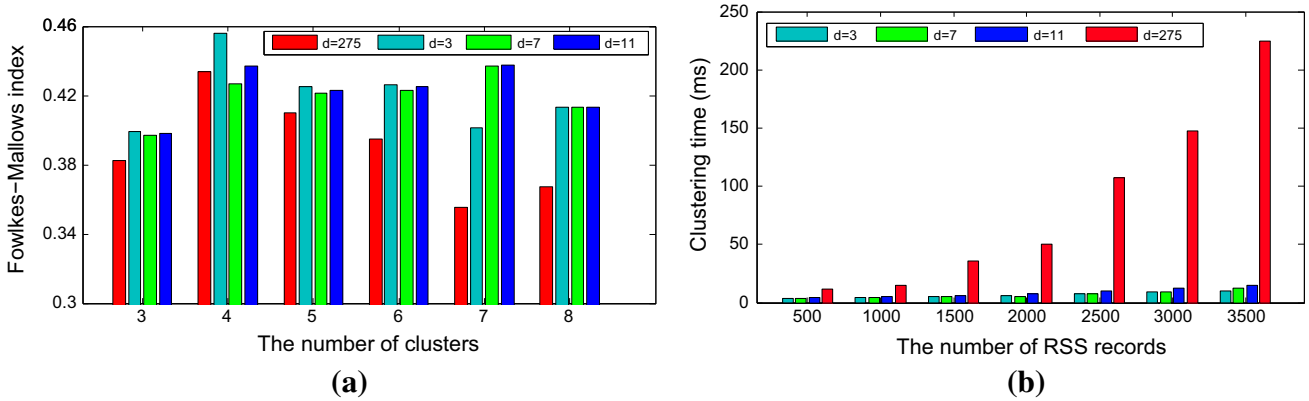

Fig. 9 The performance of clustering RSS records to the same floor. **a** The Fowlkes–Mallows index with different dimensions of RSS records. **b** The clustering time with different dimensions of RSS records

dimension reduction based on Laplacian Eigenmaps can effectively reduce the clustering time.

4.2.2 *Identify physical boundary points*

Three parameters in our algorithm need to be determined for identifying physical boundary points: time windows size τ, variation coefficient α for recognition boundary points, user-specific threshold δ for removing false identification. The three parameters directly impact the accuracy of identifying physical boundary points. We use a cluster-based method to select δ. Specifically, we first cluster all WiFi RSS records to N classes by KNN; N is the number of subareas. Then, we calculate the fingerprint of each class and further obtain the similarity for each pair of fingerprints. Finally, we select the average similarity as δ for removing false identification.

For calculating the subarea fingerprint, we partition the range of RSS values into 4 bins which is in line with typical RSS quality partition [3, 23]: (1) bin-1, which represents WiFi signal is excellent and the RSS values are in range $[-55, 0]$; (2) bin-2, which represents WiFi signal is good and the RSS values are in range $[-70, -55)$; (3) bin-3, which represents WiFi signal is poor and the RSS values are in range $[-85, -70)$; (4) bin-4, which represents WiFi signal is bad and the RSS values are in range $[-100, -85)$.

Table 7 shows the accuracy of identifying physical boundary points with time window size τ and variation coefficient α. From this table, we observe: (1) the accuracy drops sharply when the user-specific threshold of variation coefficient α is lower than 1.2 or greater than 1.5; (2) Set $\alpha = 1.5$, the accuracy increases with time window size increasing from 1 to 5, and slightly decrease when the time window size is larger than 5 due to the RSS fluctuation between physical boundary point and other location will be smaller for a large time window size. Finally, the best performance (83%) is achieved when $\alpha = 1.5$ and $\tau = 5$.

Table 7 Parameter tuning for identifying physical boundary points

α	τ					
	3	4	5	6	7	8
1.1	0.23	0.29	0.43	0.37	0.30	0.24
1.2	0.37	0.48	0.56	0.45	0.37	0.29
1.3	0.44	0.51	0.61	0.51	0.44	0.32
1.4	0.47	0.59	0.75	0.70	0.59	0.48
1.5	0.52	0.71	0.83	0.79	0.67	0.54
1.6	0.37	0.64	0.76	0.57	0.55	0.38
1.7	0.29	0.48	0.70	0.46	0.39	0.31
1.8	0.24	0.41	0.63	0.47	0.35	0.21
1.9	0.19	0.21	0.53	0.33	0.20	0.16

Figure 10a, b shows the identification accuracy as a function of variation coefficient and time window size, respectively. From the two figures, we observe: (1) the method using

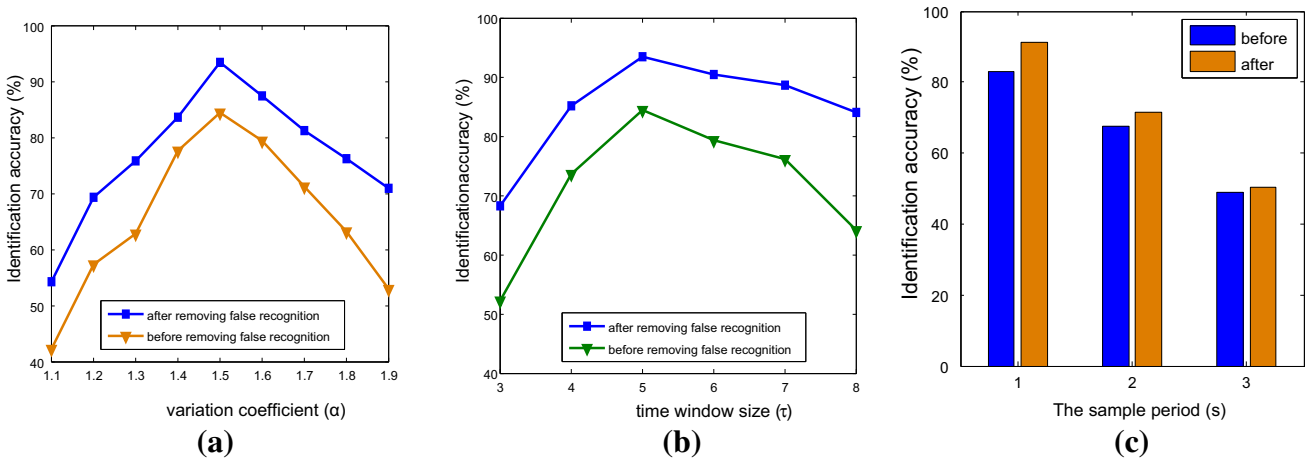

Fig. 10 The accuracy with different parameters. **a** The accuracy with different variation coefficients. **b** The accuracy with different time window sizes. c The accuracy with different sampling rate

subarea fingerprint similarity can effectively remove false recognition; (2) Set the time window size $\tau = 5$, the accuracy declines sharply when variation coefficient α is greater than 1.6 or lower than 1.4 and achieve the best accuracy when $\alpha = 1.5$; (3) Set $\alpha = 1.5$, the identification accuracy increases with the increasing number of time window size between 3 and 5 and slightly decrease when the time window size is larger than 5; (4) the performance of removing false identification decreases slightly with increasing time window size, due to the difference of RSS fluctuation between physical boundary point and normal location will be smaller with increasing time window size.

Set $\tau = 5$ and $\alpha = 1.5$, we investigate the identification accuracy with different sampling rate in Fig. 10c. As shown in Fig. 10c, we can see the identification accuracy drops significantly with increasing the sampling rate. For example, the identification accuracy is only 50.3% after removing false recognition when setting the sampling rate as 3. The reason is that if collecting RSS values with a relatively long period (e.g., 3s), the RSS values of both physical boundary point and normal location will fluctuate wildly when users moving thus cannot effectively identify physical boundary points

4.2.3 *Construct fingerprint map*

We utilize mapping accuracy to evaluate the performance for constructing fingerprint map. The mapping accuracy (MA) is defined in Eq. 15. We define s_i as the ground truth subarea label of record o_i, $\widehat{s_i}$ is the mapping subarea label.

$$\mathrm{MA} = \frac{\sum_{i=1}^{Te} I(si, \widehat{s_i})}{Te} \tag{15}$$

where $I(si, \widehat{s_i})$ is an indicator function that return 1 if $\widehat{s_i} = s_i$, Te is the test RSS records for evaluation.

One parameter needs to be determined for constructing fingerprint map: the cluster number K_f for generating logical floor graph. Figure 11 reports the performance of constructing fingerprint map with different cluster numbers (K_f), where K_f is in the range [30, 33, ... 51]. In this Fig. 11, we compare the performance of K-means and the improved K-means (DBKM) when constructing logical floor graph. As previously mentioned, DBKM utilizes a density-based algorithm to select initial cluster centers of k-means. From Fig. 11, we can see the proposed clustering method (DBKM) always outperforms K-means (for example, the FMI of DBKM for all subareas is about 91.3% when K_f = 42, and the performance is improved by 3% compared with k-means), showing the advantages of selecting initial cluster centers using density-based algorithm can achieve better clustering performance.

From the three Fig. 11a–c, we can see that the mapping accuracy for rooms increases gradually when K_f increases from 30 to 42 and then drops when K_f is greater than 42, the highest mapping accuracy of DBKM is 94.1% when K_f equals to 42 (the number of physical subareas). Another observation is the mapping accuracy for subareas located in corridor is lower about 20 percent than rooms, which shows there no obvious RSS "jump" characteristic for two connected subareas in corridor because there are no walls or physical boundary points can significantly weakened the radio signal strength.

To investigate the performance of correcting mapping errors using semantic features of subareas, we further compare the mapping accuracy of the proposed method (DBKM) after correcting in Fig. 12. From Fig. 12a–c, we observe: (1) the mapping accuracy for both rooms and subareas located in corridor has improved after correcting, showing the efficiency of distinguishing different types of subareas using semantic features. For example, the mapping accuracy for

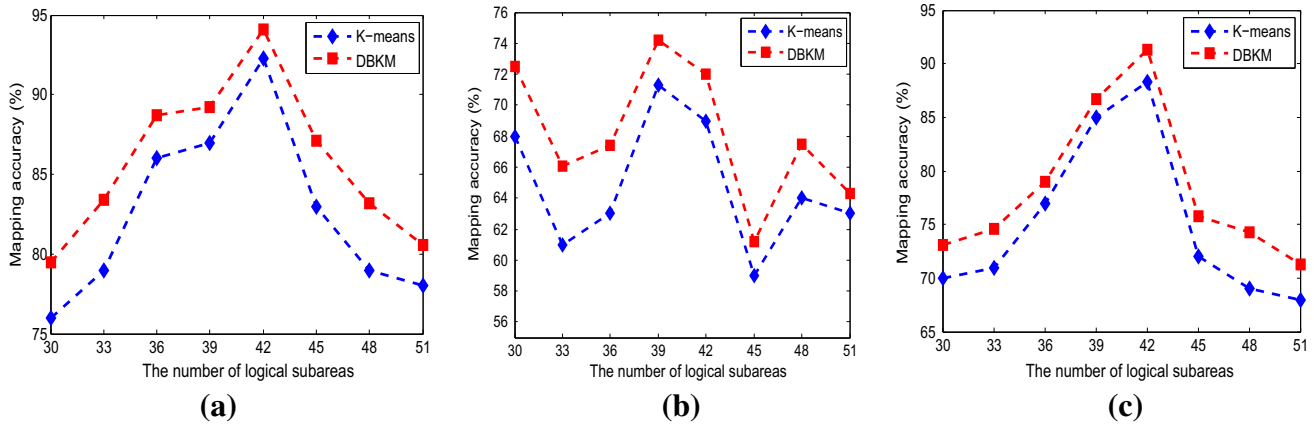

Fig. 11 The mapping accuracy with different virtual subareas, **a** rooms, **b** corridors, **c** total

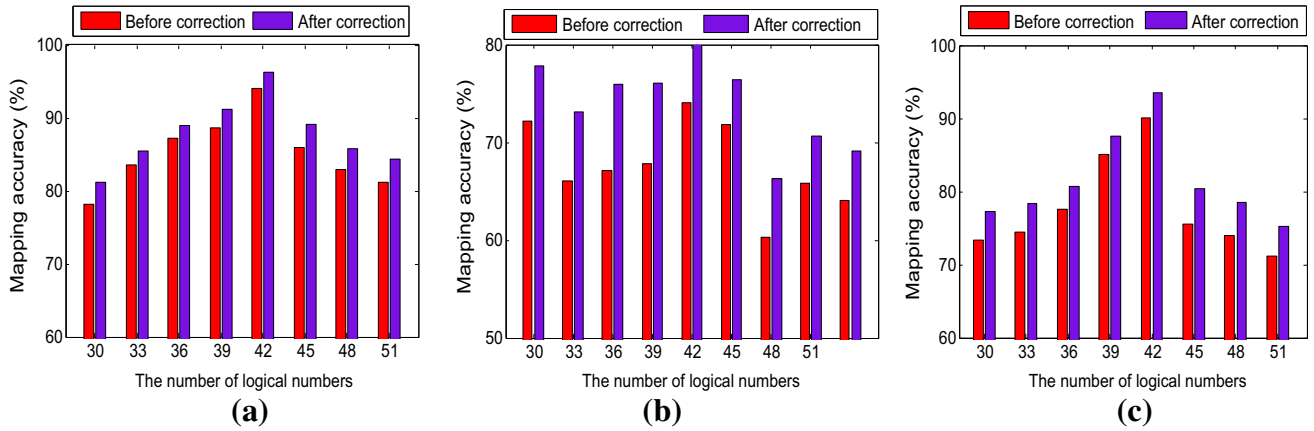

Fig. 12 The mapping accuracy with different virtual subareas after correcting, **a** rooms, **b** corridors, **c** total

rooms increases 2.3% after correcting when the number of logical numbers equals to 39; (2) the performance improvement is more obvious for subareas located in corridor than rooms. For example, the performance improvement for subareas located in corridor is 9.5% when the number of logical numbers equals to 39, while 2.3% for rooms. The reason is that the semantic feature is sufficient to distinguish subareas located in corridor and other types of subareas, since more than 84% of check-ins in subareas belong to corridor are less than 15 min.

Figure 13 reports the performance of constructing fingerprint map as a function of number of WiFi RSS records per subarea. We can see that the mapping accuracy is relatively stable when RSS records of each subarea is more than 400, which shows our algorithm for constructing the fingerprint map will converge quickly and has a low crowdsourcing data requirement. Moreover, the performance of constructing fingerprint map will improve with increasing collected data.

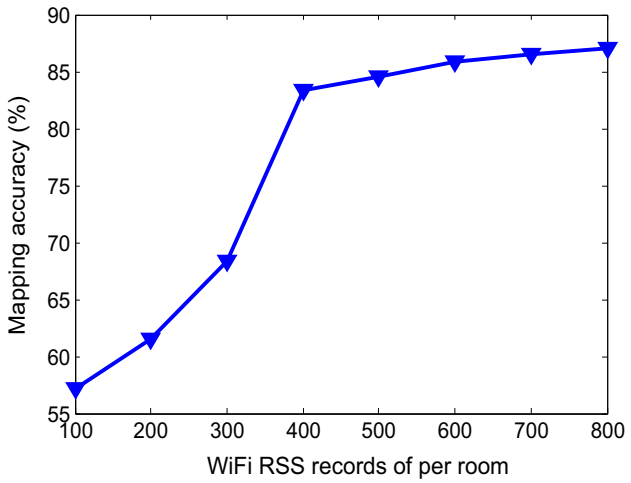

Fig. 13 The mapping accuracy with different RSS records of per subarea

4.2.4 Localization accuracy

We evaluate the performance of the proposed localization method by comparing with two well-known subarea localization methods. We first introduce the experimental dataset and parameters setting and then detail the comparative localization techniques. Finally, we report and discuss the experimental results.

Dataset We randomly select 70% RSS records of each subarea as training dataset to construct fingerprint map and the rest 30% as testing dataset for evaluation localization accuracy.

Parameters setting Tuning algorithm parameters, such as the time window size for identification physical boundary points and the clusters for constructing logical floor map, are critical to the performance of localization. According to the experience of previous experiments, our algorithm empirically set parameters as: $\{\tau = 5, \alpha = 1.5, K_f = 42\}$, for constructing fingerprint map.

Comparative methods We compare our method with the following two methods that have been widely used in subarea localization: (1) RSS-NN [39], which constructs fingerprint map by manual site survey and estimates subarea using KNN classification; (2) RSS-Bayesian [12], which also constructs fingerprint map by site survey and estimates subarea using Bayesian inference.

Results and analysis We investigate the impact of different mark-off rates (30%, 70% and 100%) to the performance of GraphLoc, RSS-NN and RSS-Bayesian. Mark-off rate is the ratio of RSS APs to construct fingerprint map, for instance, 30% mark-off rate means we utilize the RSS records from 30% WiFi APs by randomly selecting to construct fingerprint map and online localization. For each case, we repeat the experiments 10 times and report the average performance.

As shown in Fig. 14a–c, the performance of the three methods all degrade to some extent as the mark-off rate increases. Nevertheless, RSS-NN shows the best performance consistently over all mark-off rates as it needs to manually construct fingerprint map. While the performance of our proposed method drops significantly when the mark-off rate equals to 30%, for example, the localization accuracy of the proposed method drops 14% compared to RSS-NN. This is because our proposed method requires enough WiFi APs for automatically constructing fingerprint map, since less WiFi APs will degrade the accuracy of identifying physical boundary points.

Figure 14c shows the localization accuracy of the three methods using all WiFi APs. It can be seen that the performance for open subarea (subareas in the corridor) and closed subarea (room) is significantly different for all methods. As shown in Fig. 14c, the localization accuracy of rooms is more than 87% for the three methods, but lower than 85% for open subareas in corridor, which shows RSS values of two connected open subareas are too similar to distinguish. RSS-NN achieves the best performance for both closed subareas (92%) and open subareas (83%). Another observation is the average localization accuracy rate is 88.2% for our method, which is 0.8% less than RSS-NN. Therefore, our method can obtain considerable performance compared with previous methods with labor intensive and time-consuming site survey.

5 Discussion

GraphLoc involves more user efforts for collecting WiFi RSS records by crowdsourcing during the phase of constructing fingerprint map, and it provides infrastructure-free subarea localization without time-consuming site survey. We have invited 20 participants to collect data and

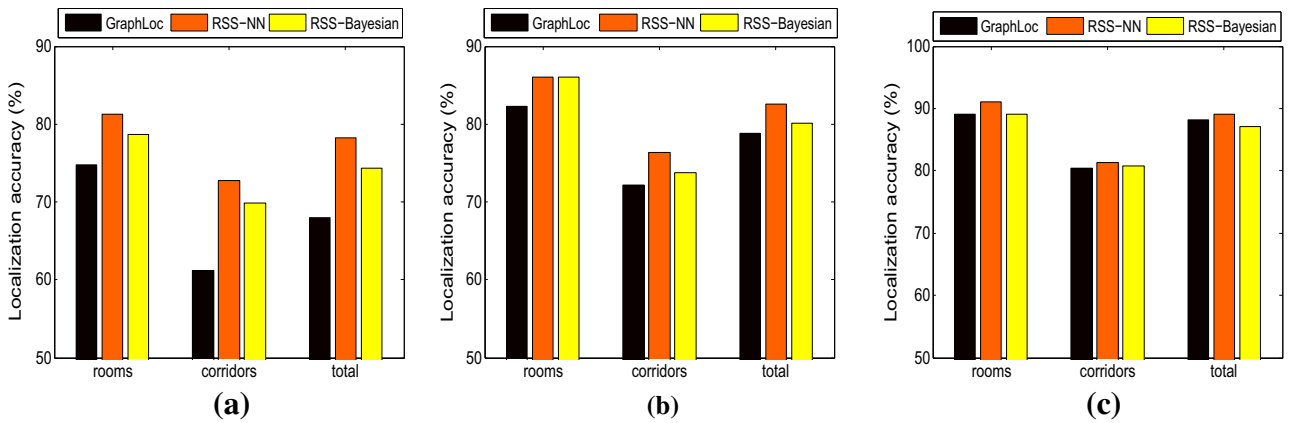

Fig. 14 The localization accuracy with different mark-off rates. **a** Mark-off rate (30%). **b** Mark-off rate (70%). **c** Mark-off rate (100%)

found that if a user can freely use WiFi service, he is willing to spend a few minutes practising data collection in the shopping journey. The collection of WiFi RSS information data costs some energy, but it is quite low according to specifications of mainstream smart phones [35], since the main energy-consuming component is scanning and associating to WiFi APs without any transmitting. Additionally, the sampling rate for collecting RSS values has a significant impact on identifying physical boundary points, as the RSS values of both physical boundary point and normal location will fluctuate wildly when users moving thus cannot effectively recognize physical boundary points. The RSS variance caused by heterogeneous devices and dynamic environmental status will degrade the positioning accuracy for fingerprint-based methods. Our previous studies [37, 38] have proposed some effective solutions for the RSS variance problem, while it has little effect on room-level localization. For example, the RSS-NN can achieve 90% localization accuracy using the raw RSS values.

The existence of symmetric subgraphs may lead to matching errors when mapping logical floor graph to physical floor graph, which may significantly degrade the performance of constructing fingerprint map. We extract two kinds of semantic features to correct the mapping errors: average stay time and temporal distribution. The basic idea is the relationship between WiFi RSS and different types of subareas due to signal reflection, refraction and diffraction, since different subareas vary in internal structures and human activities that can be reflected by RSS characteristics. Even after the correcting stage, the proposed approach cannot completely correct mapping errors. Another limitation is the correcting method cannot work in some indoor environment without obvious semantic feature, such as academic building. We plan to investigate existing methods [14, 33] by fusing internal sensors to remove the mapping errors. Another problem of mapping logical floor graph to physical floor graph is the Ullman algorithm is NP-complete. However, the Ullman algorithm is fast enough for practical use since there is usually only a few dozen nodes in one logical floor graph or physical floor graph.

6 Conclusion

This paper has proposed a ready-to-deploy method for indoor subarea localization with zero-configuration, since the proposed method is infrastructure-free and does not need time-consuming site survey. The main idea is to generate logical floor graph based on two characteristics of WiFi RSS in indoor space and automatically construct fingerprint map by mapping logical floor graph to physical floor graph. The proposed method has been implemented and deployed in a real-world shopping mall with an average localization accuracy of 88.2%, which is competitive to traditional approaches. For indoor space with multi-floors, the proposed method firstly clusters RSS records to the same floor using two steps: dimension reduction using laplacian eigenmaps and clustering using k-means. The advantages on infrastructure-free and automatically constructing fingerprint map make our method can be widely used in indoor environment.

As future work, we plan to implement some valuable indoor location-based services (e.g., indoor POI recommendation or hotspot detecting) based on the proposed subarea localization method.

Acknowledgements This work is sponsored by the National Basic Research 973 Program of China (No. 2015CB352403), the National Natural Science Foundation of China (NSFC) (61261160502, 61272099), the Program for National Natural Science Foundation of China/Research Grants Council (NSFC/RGC)(612191030), the Program for Changjiang Scholars and Innovative Research Team in University (IRT1158, PCSIRT), the Scientific Innovation Act of STCSM (13511504200), and EU FP7 CLIMBER Project (PIRSES-GA-2012-318939).

References

1. Ahmed ZU, Ghingold M, Dahari Z (2007) Malaysian shopping mall behavior: an exploratory study. Asia Pacific J Mark Logist 19(4):331–348
2. Angermann M, Frassl M, Doniec M, Julian BJ , Robertson P (2012) Characterization of the indoor magnetic field for applications in localization and mapping. In: 2012 international conference on indoor positioning and indoor navigation (IPIN). IEEE, pp 1–9
3. Azini AS, Kamarudin MR, Jusoh M (2015) Transparent antenna for WiFi application: RSSI and throughput performances at ism 2.4 GHZ. Telecommun Syst 61:1–9
4. Belkin M, Niyogi P (2001) Laplacian eigenmaps and spectral techniques for embedding and clustering. NIPS 14:585–591
5. Biehl JT, Cooper M, Filby G, Kratz S (2014) Loco: a ready-to-deploy framework for efficient room localization using wi-fi. In: Proceedings of the 2014 ACM international joint conference on pervasive and ubiquitous computing. ACM, pp 183–187
6. Castelli N, Stevens G, Jakobi T, Ogonowski C (2014) Placing information at home: using room context in domestic design. In: Proceedings of the 2014 ACM international joint conference on pervasive and ubiquitous computing: adjunct publication. ACM, pp 919–922
7. De Francisco R (2010) Indoor channel measurements and models at 2.4 GHZ in a hospital. In: Global telecommunications conference (GLOBECOM 2010), 2010 IEEE. IEEE, pp 1–6
8. Gao Y, Niu J, Zhou R, Xing G (2013) Zifind: exploiting cross-technology interference signatures for energy-efficient indoor localization. In: INFOCOM, 2013 Proceedings IEEE. IEEE, pp 2940–2948
9. Gassen M, Fhom HS (2016) Towards privacy-preserving mobile location analytics. In: Proceedings of the workshops of the EDBT/ICDT 2016 joint conference, EDBT/ICDT workshops 2016, Bordeaux, France, March 15, 2016

10. Hida K, Bin C, Hada Y, Mori S (2014) Evaluation of area detection method using machine learning. In: Multimedia, distributed, cooperative, and mobile symposium
11. Hossain AKMM, Soh W-S (2010) Cramer-rao bound analysis of localization using signal strength difference as location fingerprint. In: INFOCOM, 2010 Proceedings IEEE. IEEE, pp 1–9
12. Hotta S, Hada Y, Yaginuma Y (2012) A robust room-level localization method based on transition probability for indoor environments. In: 2012 international conference on indoor positioning and indoor navigation (IPIN). IEEE, pp 1–8
13. Hou Y, Xue Y, Chen C, Xiao S (2015) A rss/aoa based indoor positioning system with a single led lamp. In: 2015 international conference on wireless communications and signal processing (WCSP). IEEE, pp 1–4
14. Jiang Y, Xiang Y, Pan X, Li K, Lv Q, Dick RP, Shang L, Hannigan M (2013) Hallway based automatic indoor floorplan construction using room fingerprints. In: Proceedings of the 2013 ACM international joint conference on Pervasive and ubiquitous computing. ACM, pp 315–324
15. Kehagias A, Hollinger G, Singh S (2009) A graph search algorithm for indoor pursuit/evasion. Math Comput Model 50(9):1305–1317
16. Kim EY, Kim Y-K (2005) The effects of ethnicity and gender on teens' mall shopping motivations. Cloth Text Res J 23(2):65–77
17. Komar C, Ersoy C (2004) Location tracking and location based service using ieee 802.11 wlan infrastructure. In: European wireless. Citeseer, pp 24–27
18. Leitinger E, Fröhle M, Meissner PL, Witrisal K (2014) Multipath-assisted maximum-likelihood indoor positioning using uwb signals. In: 2014 IEEE international conference on communications workshops (ICC). IEEE, pp 170–175
19. Lymberopoulos D, Liu J, Yang X, Choudhury RR, Sen S, Handziski V (2015) Microsoft indoor localization competition: experiences and lessons learned. GetMobile Mobile Comput Commun 18(4):24–31
20. Martin P, Ho B-J, Grupen N, Munoz S, Srivastava M (2014) An ibeacon primer for indoor localization: demo abstract. In: Proceedings of the 1st ACM conference on embedded systems for energy-efficient buildings. ACM, pp 190–191
21. Niu J, Wang B, Cheng L, Rodrigues JJPC (2015) Wicloc: an indoor localization system based on wifi fingerprints and crowdsourcing. In: 2015 IEEE international conference on communications (ICC). IEEE, pp 3008–3013
22. Pang S, Trujillo R (2013) Indoor localization using ultrasonic time difference of arrival. In: Southeastcon, 2013 Proceedings of IEEE. IEEE, pp 1–6
23. Peltonen E, Lagerspetz E, Nurmi P, Tarkoma S (2015) Energy modeling of system settings: a crowdsourced approach. In: 2015 IEEE international conference on pervasive computing and communications (PerCom). IEEE, pp 37–45
24. Rallapalli Sati, Ganesan A, Chintalapudi K, Padmanabhan VN, Qiu L (2014) Enabling physical analytics in retail stores using smart glasses. In: Proceedings of the 20th annual international conference on mobile computing and networking. ACM, pp 115–126
25. Ranjan J, Whitehouse K (2015) Object hallmarks: identifying object users using wearable wrist sensors. In: Proceedings of the 2015 ACM international joint conference on pervasive and ubiquitous computing. ACM, pp 51–61
26. Rappaport TS et al (1996) Wireless communications: principles and practice, vol 2. Prentice Hall PTR, New Jersey
27. Rashidi P, Mihailidis A (2013) A survey on ambient-assisted living tools for older adults. IEEE J Biomed Health Inform 17(3):579–590
28. Santoso F, Redmond SJ (2015) Indoor location-aware medical systems for smart homecare and telehealth monitoring: state-of-the-art. Physiol Meas 36(10):R53
29. She J, Crowcroft J, Fu H, Li F (2014) Convergence of interactive displays with smart mobile devices for effective advertising: a survey. ACM Trans Multimed Comput Commun Appl (TOMM) 10(2):17
30. Shin H, Chon Y, Kim Y, Cha H (2015) A participatory service platform for indoor location-based services. Pervasive Comput IEEE 14(1):62–69
31. Ullmann JR (1976) An algorithm for subgraph isomorphism. J ACM 23(1):31–42
32. Wagner S, Wagner D (2007) Comparing clusterings: an overview. Universität Karlsruhe, Fakultät für Informatik Karlsruhe
33. Wu C, Zheng Y, Yunhao L, Wei X (2013) Will: Wireless indoor localization without site survey. IEEE Trans Parallel Distrib Syst 24(4):839–848
34. Xiong J, Sundaresan K, Jamieson K (2015) Tonetrack: leveraging frequency-agile radios for time-based indoor wireless localization. In: Proceedings of the 21st annual international conference on mobile computing and networking. ACM, pp 537–549
35. Yao D, Yu C, Dey AK, Koehler C, Min G, Yang LT, Jin H (2014) Energy efficient indoor tracking on smartphones. Future Gener Comput Syst 39:44–54
36. Zheng Z, Chen Y, Chen S, Sun L, Chen D (2016) Bigloc: a two-stage positioning method for large indoor space. Int J Distrib Sens Netw 2016:1289013
37. Zheng Z, Chen Y, He T, Li F, Chen D (2015) Weight-rss: a calibration-free and robust method for wlan-based indoor positioning. Int J Distrib Sens Netw 2015:55
38. Zheng Z, Chen Y, He T, Sun L, Chen D (2015) Feature learning for fingerprint-based positioning in indoor environment. Int J Distrib Sens Netw 2015:180
39. Zhou S, Wang B, Mo Y, Deng X, Yang LT (2013) Indoor location search based on subarea fingerprinting and curve fitting. In: 2013 IEEE 10th international conference on high performance computing and communications and 2013 IEEE international conference on embedded and ubiquitous computing (HPCC_EUC). IEEE, pp 2258–2262