Article

A Radio-Map Automatic Construction Algorithm Based on Crowdsourcing

Ning Yu *, Chenxian Xiao, Yinfeng Wu and Renjian Feng

Key Laboratory of Education Ministry for Precision Opto-Mechatronics Technology, School of Instrumentation Science and Opto-Electronics Engineering, Beijing University of Aeronautics and Astronautics (Beihang University), Beijing 100191, China; xiaochenxian@buaa.edu.cn (C.X.); yfwu@buaa.edu.cn (Y.W.); rjfeng@buaa.edu.cn (R.F.)

* Correspondence: nyu@buaa.edu.cn; Tel.: +86-10-8231-5132

Academic Editor: Leonhard M. Reindl

Received: 28 January 2016; Accepted: 30 March 2016; Published: 9 April 2016

Abstract: Traditional radio-map-based localization methods need to sample a large number of location fingerprints offline, which requires huge amount of human and material resources. To solve the high sampling cost problem, an automatic radio-map construction algorithm based on crowdsourcing is proposed. The algorithm employs the crowd-sourced information provided by a large number of users when they are walking in the buildings as the source of location fingerprint data. Through the variation characteristics of users' smartphone sensors, the indoor anchors (doors) are identified and their locations are regarded as reference positions of the whole radio-map. The AP-Cluster method is used to cluster the crowdsourced fingerprints to acquire the representative fingerprints. According to the reference positions and the similarity between fingerprints, the representative fingerprints are linked to their corresponding physical locations and the radio-map is generated. Experimental results demonstrate that the proposed algorithm reduces the cost of fingerprint sampling and radio-map construction and guarantees the localization accuracy. The proposed method does not require users' explicit participation, which effectively solves the resource-consumption problem when a location fingerprint database is established.

Keywords: indoor localization; smartphone sensors; crowdsourcing; radio-map

1. Introduction

Wireless sensor networks (WSNs), which are multi-hop self-organized networks with wireless communication, are used to achieve information collection and processing by deploying a large number of tiny sensor nodes in a surveillance area. One of the fundamental functions in a wireless sensor network is to acquire location information of users, as the effectiveness of sensor monitoring depends mostly on whether the location information is accurate or not [1]. In the outdoor environment, the global positioning system (GPS) [2] is usually used to locate static or mobile users. As for the complex indoor environments, static or mobile users can be located by the received signal strength (RSS) [3], time of arrival (TOA) [4], time difference of arrival (TDOA) [5] and angle of arrival (AOA) [6] based methods.

In recent years, with the continuous development of wireless communication technology, WiFi is now deployed in many public places and has become a necessary infrastructure in people's daily life [7,8]. Thus, WiFi-based indoor localization methods with location fingerprints [9–13] are commonly used for indoor localization. The WiFi information received by sensors at certain indoor location is called a "fingerprint", which is used to identify its corresponding location. The traditional fingerprint-based localization methods have two phases: an offline phase for acquiring a fingerprint database and an online phase for localization. In the offline phase, sampling points are set at regular distances in

the indoor area, and the corresponding WiFi information, including media access control (MAC) addresses of access points (AP) and their corresponding RSS values at every sampling point is received and recorded. Each piece of WiFi information and its corresponding location coordinates constitute a location fingerprint, and all the location fingerprints constitute a radio-map of the indoor area. In the online phase, when a user sends a localization enquiry, the user's current WiFi information is compared with the WiFi information in the radio-map. The location in the radio-map which WiFi information is most similar to the user's current WiFi information is estimated to be the user's current location.

The quantity of data in a radio-map directly impacts the indoor localization accuracy since the radio-map is a bridge connecting the offline phase and the online phase. During the offline phase, multiple measurement samples are required from a single sampling point in order to get a better localization accuracy. The manual data sampling method is resource-consuming and a practical bottleneck in radio-map construction. Every time a new localization demanding area is added, some new APs join in the network, or a certain AP moves to another new place [14], repetitive manual sampling work is needed to ensure the localization accuracy.

Generally, people carry out their daily work with smartphones in hand in indoor spaces. With the development of embedded technology and wireless communication technology, existing mobile phones are equipped with various sensors (like acceleration sensors, electronic compasses, gyroscopes, etc.) and good communication transceiver modules [15], which provide an important information interface connecting the wireless network with the actual environment, and reflect information about human activities in the environment to the wireless network. If all smart phones carried by ordinary users implicitly take part in the construction of a radio-map and contribute data to the acquisition of indoor fingerprints, the sampling workload can be significantly reduced. Therefore, in order to solve the poor extensibility and flexibility of radio-map construction in traditional fingerprint-based indoor localization algorithms, a novel automatic radio-map construction algorithm based on crowdsourcing (RACC) is proposed in this paper. The main contributions are summarized as follows.

- (1) Data is collected by exploiting crowdsourcing technologies. The daily activities of users are represented by their walkable mobile phone path in the indoor building. The WiFi location information that a user has passed by, user's walking acceleration, walking direction and rotational angular velocity are recorded by phones automatically. The recorded data provided by a large number of users are named crowdsourced data.
- (2) Through the variation characteristics of users' smartphone sensors, the indoor anchors (doors) are identified and their locations are regarded as reference positions of the radio-map. The crowdsourced data are clustered to acquire the representative fingerprints. According to the reference positions and the similarity between fingerprints, the representative fingerprints are linked to their corresponding physical locations and the radio-map is generated.
- (3) Online localization is realized based on the constructed radio-map. Extensive simulations show that RACC can automatically generate radio-maps with acceptable accuracy and efficiently solve the resource-consuming problem in traditional radio-map-based indoor localization methods.

The remainder of the paper is organized as follows: Section 2 discusses two categories of methods in radio-map-based indoor localization algorithms. In Section 3, methodology and system architecture of the proposed algorithm is given. Section 4 presents the detailed procedures of RACC. Section 5 shows the experimental results and a performance evaluation of RACC. Finally, Section 6 concludes the paper.

2. Related Works

2.1. Offline Fingerprinting-Based Localization

RADAR [16] is an indoor localization system based on RSS fingerprints proposed by Microsoft Research in 2000. It is a deterministic localization method. In the offline phase, the mean value of

received signal strength at every sampling point is recorded into the radio-map; in the online phase, the RSS information is matched to calculate the user's current location.

Castro *et al.* [17] proposed a probabilistic location service for a WiFi networked environment called Nibble. Unlike RADAR, Nibble infers location information from the SNR measures received from the WiFi network, builds a probability distribution of signal space using Bayesian network and chooses the location with the maximum *a posteriori* probability to be the estimated location. The service can achieve a room-level localization accuracy. Roos *et al.* [18] considered two probabilistic approaches—kernel method and histogram method—to realize indoor localization. The RSS data at each reference point are recorded for a period of time and stored in the radio-map. The probability distribution characteristics of received signal strength can be respectively fitted by kernel function and histogram, and then user's location is calculated. Compared with the deterministic localization method, probabilistic method has better noise immunity, and can largely reduce the effects of noise on the localization accuracy. Youssef *et al.* [19] designed the Horus location determination system. In the data collection phase, the mean value and standard deviation of received signal strength at every reference point are recorded. The collected data are modeled by a Gaussian probability distribution and the locations that receive the same AP signals are grouped using the clustering method to improve the online search efficiency.

Brunato *et al.* [20] presented a statistical learning theory for mobile user location estimation. The offline sampling data are set as input vectors, the corresponding coordinates are output vectors, and thus the linear decision function is constructed by support vector machine (SVM) in high dimensional space. The user's location can be obtained by inputting the user's RSS information to the linear function. Battiti *et al.* [21] considered a multi-layer perceptron (MLP) architecture for building a flexible mapping between the raw signal measurements collected offline and the locations of the sampling points. Laoudias *et al.* [22] proposed an indoor localization algorithm based on clustered RBF (cRBF). It applied a clustering technique into the reference fingerprints, reduced dramatically the number of neurons in the hidden layer and then improved the training performance of the offline fingerprinting data.

Yin *et al.* [23] proposed a novel method to adapt the temporal radio-maps for indoor location estimation by offsetting the varying environmental factors. It continuously adjusts the fingerprint information of reference points through a regression analysis, providing more accurate data references for online localization. Pan *et al.* [24] developed a graph-based semi-supervised learning method to learn mobile users' locations. A manifold-based model is firstly built from the data recorded in the offline training phase and then in the online localization phase the weighted k-nearest neighbor method is used to localize mobile clients. Au *et al.* [25] proposed an indoor tracking system based on compressive sensing (CS). The system clusters the RSS fingerprints and selects fingerprint representatives after building the radio-map offline. The online phase is a two-step method, where first a coarse localization step is performed to estimate the approximate location of the user by matching user's RSS information to the fingerprint representatives, and then the accurate location is estimated in the fine localization step formulating the localization problem into a sparse-natured signal recovery problem, for which the CS theory can be applied. Liu *et al.* [26] proposed a localization solution based on visible light communication. It employed LED lights as beacons and constructed Bayesian framework through Gaussian process to realize the low-cost localization.

All of the methods mentioned above require manpower to do the offline sampling work. The differences are that they employ different matching methods to realize the online localization on the basis of generated radio-map.

2.2. Offline Fingerprinting-Free Localization

Considering the problem that manual offline fingerprint acquisition is resource-consuming and easily affected by environmental changes, scholars have put forward many alternatives to realize radio-map construction.

Based on users' point of view, some researchers allow users to join in the process of radio-map construction directly. Bolliger *et al.* [27] proposed an adaptive zero-configuration indoor localization

system through user collaboration. The style that Redpin uses is similar to the Wikipedia concept for information collection, that is, a single locally-knowlegeable user is allowed to associate any text string with a place's fingerprint and many visitors can then rely on the database of fingerprints. Lee *et al.* [28] developed a crowdsourcing approach for indoor place recognition. An open participatory system through which general users can contribute fingerprints is built and the place recognition service can be provided as the database expands to certain degree. To deal with the volatility of RSS data and the radio-map redundancy, Yang *et al.* [29] proposed a localization method FreeLoc, that maintained only one fingerprint for a location in a radio-map, regardless of any number of uploaded data sets for a given location. It provides consistent localization accuracy in an environment where there exist device heterogeneity and the multiple surveyor problems. Park *et al.* [30] grew an organic indoor location system called OIL, which explicitly incorporated site surveys by users. The organic localization start with an empty database which is gradually populated with user-provided fingerprints. On the basis of OIL, Ledlie *et al.* [31] described a scalable user-generated localization engine named Molé. It arranges places uploaded by a number of users in a hierarchy and utilizes the Maximum Overlap (MAO) to estimate the place as the one whose fingerprint is most similar to the user's fingerprint. Sorour *et al.* [32] gave a reduced-effort radio-map generation algorithm based on dimensionality reduction transfer learning. The location of unlabeled crowdsourced signal observations can be estimated through manifold alignment and thus the indoor radio-map is constructed, which significantly reduces the large calibration load.

Other researchers think that user motions can be applied to connect previously independent RSS fingerprints since the increasing number of sensors on mobile phones offer new opportunities for indoor localization. Azizyan *et al.* [33] proposed SurroundSense, which is a mobile phone-based system that explores localization via ambience fingerprinting. It assumed that the combined effect of ambient sound, light and color can be unique enough for locating certain place. SVMs and color clustering are used to extract simple features from the collected data and then the fingerprint database is created. Rai *et al.* [14] developed a zero-effort indoor localization system called Zee. It leverages the sensors in smartphones carried by users to track them as they are moving in the indoor building and the user's steps and orientations can be estimated. Armed with a map of the indoor space of interest, Zee employs WiFi and sensor measurements crowdsourced from users to construct a WiFi training set and then uses augmented particle filtering to simultaneously infer the user's location. Wu *et al.* [34] designed WILL, an indoor localization method based on off-the-shelf WiFi infrastructure and mobile phones. WILL partitions the crowdsourced fingerprints into different virtual rooms and a logical floor plan is accordingly constructed. Finding a matching between logical and ground-truth floor plan is a key step for localization. Liu *et al.* [35] presented an indoor localization approach named LiFS, in which the calibration of fingerprints is crowdsourced. LiFS leverages the user motions to link multiple separated RSS fingerprints and builds the mapping relation between the fingerprints space and the stress-free floor plan through multidimensional scaling (MDS), which leads to the automatic establishment of fingerprint database.

The indoor localization approaches mentioned in this section take advantage of auxiliary information such as user collaboration, sensor measurements and actual indoor maps to annotate RSS fingerprints with their corresponding location labels, and thus generate the radio-map without any offline manual fingerprinting work. However, the accuracy of radio-maps automatically generated by existing indoor localization systems still needs to be improved when it is compared with the accuracy of manually constructed ones. In this paper, a novel radio-map automatic construction algorithm based on crowdsourcing (RACC) is proposed. For the promise that the localization accuracy is acceptable, it simply employs the WiFi and sensor data crowdsourced by users' smartphones, to construct the radio-map automatically without explicit volunteers to bind RSS fingerprints to a location.

3. Methodology & System Architecture

The proposed algorithm RACC consists of three parts: crowdsourced data collection, radio-map automatic construction and online localization, as is shown in Figure 1. The emphasis is on radio-map automatic construction.

Figure 1. Schematic diagram of proposed algorithm RACC.

The crowdsourced data include WiFi information (also called RSS fingerprints), user's walking acceleration, walking direction and rotational angular velocity. The radio-map automatic construction part is divided into five sections: doors' fingerprint recognition, door's fingerprint matching, crowdsourced fingerprints clustering, fingerprints partition and radio-map construction. The online localization part covers three sections: location enquiry, fingerprint matching and location estimation.

3.1. Crowdsourced Data Collection

Unlike the traditional manual sampling way, RACC gathers general users' contributions to obtain raw fingerprints, namely using crowdsourcing. When users walk in the indoor space for their daily activities, the smartphones carried by them can collect the WiFi features of the locations that users have passed. Simultaneously, a suite of sensor measurements like user's walking acceleration, walking direction and rotational angular velocity, which reflect users' walking characteristics, are also recorded by the sensors (acceleration sensor, electronic compass, gyroscope, etc.) in smartphones. The collection of the RSS fingerprints and sensor data is performed automatically in the background of phones and is transparent to the users, but has no impact on the users' daily activities. The crowdsourcing way makes full use of the crowd's effort and therefore successfully accomplishes the cost reduction in manpower.

The floor plan where the data are crowdsourced is shown in Figure 2. In the indoor area, there are several APs deployed in the rooms and their locations are unknown. The patterns that users walk in or out the indoor area, or traverse in the indoor area are displayed in Figure 3. The triangle, square and circle in Figure 3 all represent the locations where the crowdsourced data are recorded during the period users are moving. Since the GPS signal would be lost or appear temporarily once users set foot inside or outside the indoor area [36], the obvious change can be used to pick out the data collected in the indoor area. Meanwhile, their fingerprints in different building entrances are acquired and recorded in advance to make a distinction among them for the later work.

Figure 2. Floor plan of experimental areas.

Figure 3. Users' moving pattern in indoor areas.

The smartphones would record the RSS fingerprints, user's walking acceleration, walking direction and rotational angular velocity at fixed sampling period (e.g., every one second) automatically in the background, which are formatted in Table 1.

- (1) User ID. Every user has a unique ID, which is utilized to identify the crowdsourced data provided by different users.
- (2) RSS fingerprint. RSS fingerprint includes the MAC addresses of APs $\{ap_1, ap_2, \dots\}_{ti}$ and their corresponding RSS values $\{rss_{i1}, rss_{i2}, \dots\}$.
- (3) Walking acceleration. User's walking acceleration $acc = \{acc_X, acc_Y, acc_Z\}$ is measured by an acceleration sensor, where acc_X , acc_Y and acc_Z are respectively the acceleration components in the X, Y and Z direction.
- (4) Walking direction. The embedded electronic compass of smartphone measures three components: the angle relative to the magnetic north, the angle between x-axis orientation of smartphones and the horizontal plane, the angle between y-axis orientation of smartphones and the horizontal plane. On account of different placement of smartphones when users are moving, only the angle relative to the magnetic north is recorded into the crowdsourced data.

- (5) Rotational angular velocity. Users' rotational angular velocity $rot = \{rot_X, rot_Y, rot_Z\}$ is measured by the gyroscope in smartphones. The rotational angular velocity in X, Y and Z orientation would all get a change when users make a turn due to the different placement of smartphones, but one of the three parameters changes more evidently than the other two.

Table 1. The recording format of crowdsourced data.

Crowdsourced Data						
User ID		ID				
Time	Data Type	RSS Fingerprints		Acceleration	Walking Direction	Rotational Angular Velocity
		AP	RSS			
t_1		$\{ap_1, ap_2, \dots\}_{t_1}$	$\{rss_{11}, rss_{12}, \dots\}$	acc_{t_1}	cp_{t_1}	rot_{t_1}
t_2		$\{ap_1, ap_2, \dots\}_{t_2}$	$\{rss_{21}, rss_{22}, \dots\}$	acc_{t_2}	cp_{t_2}	rot_{t_2}
...		...	...	...	...	...
t_s		$\{ap_1, ap_2, \dots\}_{t_s}$	$\{rss_{s1}, rss_{s2}, \dots\}$	acc_{t_s}	cp_{t_s}	rot_{t_s}

3.2. Automatic Construction of Radio Map

The goal of RACC is to extract several representative RSS fingerprints from the crowdsourced RSS fingerprints and to annotate them with physical locations according to some rules, then automatically construct a radio-map of the indoor space. In general, users are more concerned about the paths to their destinations rather than the activity locations which users have already reached. For example, customers care more for how to reach places (such as clothing stores, coffee shops or washrooms) from the building entrances or their current positions in the shopping mall. They usually don't need positioning guide after getting to the destination where their activities are more random and subjective based on their interests. Therefore our algorithm aims to generate a radio-map of indoor public walking paths (like corridors), that is, the fingerprints in a radio-map merely cover the corridor area. When users arrive at rooms (guided by the constructed radio map) after walking pass corridors from the entrance or certain location in corridor, the location guidance in rooms is no longer carried out.

The automatic construction of radio-maps goes through five steps: recognizing doors' fingerprints, labeling doors' fingerprints, clustering crowdsourced fingerprints, partitioning fingerprint representatives and constructing a radio-map. Some features which imply users' physical locations exist in the crowdsourced data, for instance, the walking direction would make a big turn when users move through a door of the room, walking in or out the room. On this basis, the doors can be recognized and fingerprints near doors are recorded. Furthermore, since the positions of room doors hardly change, it can provide a fixed reference (anchors) for the automatic matching of other fingerprints. The locations that users walk in daily life almost cover all the indoor paths (corridors), therefore, a huge volume of crowdsourced data are produced. The crowdsourced fingerprints is clustered in the clustering step based on the similarities among RSS fingerprints and the representative fingerprints are selected. On the basis of the fixed reference (doors' locations and their corresponding fingerprints) and the similarity between fingerprints, the representative fingerprints can be matched to the corresponding physical locations in the area, and then the radio-map of the whole area is constructed.

3.3. Online Localization

When users query their current positions, users' current RSS fingerprints will be searched and matched with RSS fingerprints in the radio-map to find the most similar (nearest) fingerprints, and then the location information of the most similar (nearest) fingerprints are sent back to the user to finalize the localization. In RACC, the weighted k-nearest neighbor method is utilized to realize the online localization. When the user sends a location query RSS_{query} , similar fingerprints in the radio-map will be searched. Here given that there is a fingerprint chain with k similar RSS fingerprints

that are considered to be similar with RSS_{query} , that is, $\{RSS_1, RSS_2, \dots, RSS_k\}$. The weight of each RSS fingerprint in the chain is assigned according to its similarity with RSS_{query} . The more similar they are, the larger the weight is. The weight can be calculated as:

$$\omega_i = \frac{\frac{1}{\|RSS_i - RSS_{query}\|_2 + \varepsilon}}{\sum_{i=1}^k \frac{1}{\|RSS_i - RSS_{query}\|_2 + \varepsilon}} \quad (1)$$

where ε is a near-zero positive. It is used to make the calculation effectively when Euclidean distance between RSS fingerprints $\|RSS_i - RSS_{query}\|_2 = 0$. The geographical location of RSS_{query} can be obtained as:

$$C_{query} = \sum_{i=1}^k \omega_i \cdot C_i \quad (2)$$

where C_i ($i = 1, 2, \dots, k$) is the corresponding geographical location of k similar fingerprints.

4. RACC Algorithm

4.1. Doors' Fingerprint Recognition

Users' walking direction and rotational angular velocity would change greatly when users traverse the door in or out of a room. The variation feature helps us to recognize when users are just at the position of certain door and what are the corresponding recorded RSS fingerprints. The statistic variation tendencies of electronic compass and gyroscope when users walk through doors in different directions are displayed in Figures 4 and 5.

Figure 4. The variation tendencies of electronic compass. (a) Walk in room from the right; (b) Walk out of room from the right; (c) Walk in room from the left; (d) Walk out of room from the left.

Figure 5. The variation tendencies of gyroscope. (a) Walk in room from the right; (b) Walk out of room from the right; (c) Walk in room from the left; (d) Walk out of room from the left.

Figures 4a and 5a demonstrate the changing electronic compass and gyroscope signals when users walk into a room from the right; Figures 4b and 5b show the changing electronic compass and gyroscope signals when users walk out of the room from the right; Figures 4c and 5c show the electronic compass and gyroscope changes when users walk into the room from the left; Figures 4d and 5d show the changing electronic compass and gyroscope signals when users walk out of the room from the left. From Figure 4, it can be seen that the electronic compass measurements would fluctuate within a small range when users are moving straight, but they change dramatically when users are walking through doors. The change angle is about 70° to 110° when users walk through doors because of the vertical intersecting relationship between corridor and door. Figure 5 manifests the fact that the rotational angular velocity measured by a gyroscope also shows a sudden change when users walk through doors, and the change when users walk from the right is the opposite from that when users walk from the left, so the moments user walk in or out of rooms can be accurately recognized based on the changing features in crowdsourced electronic compass and gyroscope measurements, and the corresponding RSS fingerprints recorded at those moments can be obtained at the same time.

According to the variation characteristics discussed above, we make the following rules to realize the recognition of doors and extraction of their RSS fingerprints:

Rule 1: The difference of walking direction Δcp_{12} and the difference of rotational angular velocity Δrot_{12} between two adjacent moments t_1 and t_2 ($t_2 > t_1$) respectively are:

$$\Delta cp_{12} = \|cp_{t_2} - cp_{t_1}\|_2 \quad (3)$$

$$\Delta rot_{12} = \|rot_{t_2} - rot_{t_1}\|_2 \quad (4)$$

If Δcp_{12} and Δrot_{12} both satisfy the conditions $\Delta cp_{12} > \varepsilon_{cp}$ and $\Delta rot_{12} > \varepsilon_{rot}$, t_2 is considered to be the moment when users walk through doors, where ε_{cp} and ε_{rot} are the predefined thresholds which are

determined by the change of walking direction and rotational angular velocity respectively. The RSS fingerprint at t_2 is considered to be the candidate fingerprint of a certain room door.

Rule 2: For the two doors' candidate fingerprints $RSS_1 = \{rss_{11}, rss_{12}, \dots, rss_{1n}\}$ and $RSS_2 = \{rss_{21}, rss_{22}, \dots, rss_{2n}\}$, define the RSS similarity as:

$$\Delta RSS_{12} = ||RSS_1 - RSS_2||_2 = \sqrt{\frac{1}{n} \sum_{i=1}^n (rss_{1i} - rss_{2i})^2} \quad (5)$$

Given that ϵ_{RSS} is the predefined threshold. If $\Delta RSS_{12} < \epsilon_{RSS}$, the candidate fingerprints RSS_1 and RSS_2 are similar and they are labeled with the same door. If $\Delta RSS_{12} > \epsilon_{RSS}$, the candidate fingerprints RSS_1 and RSS_2 are dissimilar and they are labeled with different doors.

ϵ_{RSS} represents the average maximum dissimilarity of RSS fingerprints in adjoining locations which is feasible to distinguish RSS fingerprints at different positions. The MAC addresses of APs and the corresponding RSS values recorded by two similar RSS fingerprints have few differences. Smartphones carried by multiple users would acquire similar RSS fingerprints when they walk through the same door. Following Rule 2, multiple similar fingerprints are assigned to the same room door. In order to facilitate the matching between door's location with its corresponding fingerprints, the averaged value of multiple similar fingerprints are used to represent the door's fingerprints, abbreviated as D-fingerprint. Rule 3 gives the calculation.

Rule 3: For m similar candidate fingerprints labeled with the same room door: $RSS_1 = \{rss_{11}, rss_{12}, \dots, rss_{1n}\}$, $RSS_2 = \{rss_{21}, rss_{22}, \dots, rss_{2n}\}$, $\dots$, $RSS_m = \{rss_{m1}, rss_{m2}, \dots, rss_{mn}\}$, define their averaged fingerprint as D-fingerprint:

$$RSS_D = \{Avg\{rss_{11}, rss_{21}, \dots, rss_{i1}, \dots, rss_{m1}\}, Avg\{rss_{12}, rss_{22}, \dots, rss_{i2}, \dots, rss_{m2}\}, \dots, Avg\{rss_{1j}, rss_{2j}, \dots, rss_{ij}, \dots, rss_{mj}\}, \dots, Avg\{rss_{1n}, rss_{2n}, \dots, rss_{in}, \dots, rss_{mn}\}\} \quad (6)$$

where $Avg\{\cdot\}$ is the average function.

The doors' fingerprints can be extracted from the crowdsourced data based on Rules 1 and 2. However, there are two exceptions that need further explanation. One is that users could occasionally or deliberately make turns when they are walking straight, the other is that the electronic compass and gyroscope in smartphones would make the similar aforementioned change when users walk through the corridor intersections. The following describes these two cases.

If users make a turning action randomly someplace, it is difficult for the turning actions to appear in the same place, which means that the probability that the users make turning actions randomly at the same position is relatively low. Only when the number of users reached some degree and their walking direction and rotational angular velocity all change at the similar RSS fingerprints, it can be indicated that the users all have walked through a certain door. Thus, the occasional changing points of sensor measurements can be eliminated and have no influence on the identification of doors' fingerprints.

As for the similar change of walking direction and rotational angular velocity when users walk through the corridor intersections, a method to differentiate doors from corridor intersections is given as follows: when users walk into rooms, they generally slow down or remain static due to the space constraint, and thus the acceleration tends to be gentle or stable. On the other hand, when users walk through corridor intersections, their acceleration remains unchanged on account of users' continued walking. Therefore, the statistic changing tendencies in crowdsourced acceleration measurements can be used to discriminate the turning actions occurring at room doors from those occurred at corridor intersections. Figure 6 shows the changing rules of acceleration during the periods before and after users walk in or out of rooms.

Figure 6. The variation tendencies of accelerometer. (a) Walk in room door on the right; (b) Walk out room door on the right; (c) Walk in room door on the left; (d) Walk out room door on the left.

From Figure 6, we can conclude that the acceleration is transformed from an obvious regular fluctuation to little fluctuation when users walk into rooms and become static. On the contrary, the acceleration changes from an initial steady state to a large range of fluctuation when users walk out of the rooms. Rule 4 is set to make the distinction of turnings between doors and corridor intersections.

Rule 4: Given t is the moment when walking direction and rotational angular velocity simultaneously make a great change. If the time period Δt_{static} that users remain static or nearly static satisfies the condition $\Delta t_{\text{static}} > T$ after the moment t , then t is recognized as the moment when users walk into the rooms, where T is predefined time threshold. If the time period Δt_{static} when users remain static or nearly static satisfies the condition $\Delta t_{\text{static}} > T$ before the moment t , then t is recognized as the moment when users walk out the rooms. Otherwise t is recognized as the moment when users go through corridor intersections.

One case that needs to be specified is that individual users might stop walking immediately after passing through the corridor intersections. This situation would not affect the judgement for Rule 4 because of the use of crowdsourced data. Since Rule 4 is concluded statistically from the same characteristics in a large amount of users' walking paths, some unreasonable small probability trajectories of individual users can be removed effectively.

Based on the above rules, all doors' fingerprints in the indoor area are extracted from the crowdsourced data. However, the doors' fingerprints have not been annotated with their specific door number yet. It is necessary to match the D-fingerprints with their corresponding door number in order to get D-fingerprints' physical locations.

4.2. Doors' Fingerprint Matching

In practical conditions, users' walking paths are constricted by the indoor architecture, and geographically there are distinct topology relationships between the doors. As is seen in Figure 3, the building entrances lie on the left side of floor plan. No matter which room users want to reach when

they set foot in the indoor area from the entrances, they first have to go past the room door that is closer to the entrances and later go past the room door that is far from the entrances. The two neighboring doors are in a sequential constraint relationship in users' walking path. As mentioned, different building entrances can be identified by the changes in GPS signal and the RSS fingerprints calibrated previously at entrances. Hence, we start fingerprint matching from the building entrances. In a single user's walking path, the time and steps spent from the entrance to the door which is the closest to entrance are the minimum among those from the entrance to any other door. Taking advantage of this feature, the D-fingerprint of the nearest door to entrance is labeled first. In sequence, the D-fingerprint of the nearest door to the door which is successfully matched in the last matching round is labeled in the same way.

The movement acceleration will generate a peak with every step user takes and the users' walking steps can be calculated by setting a threshold [35]. Rules 5 and 6 are drafted to fulfill the doors' fingerprint matching in accordance with the principle that walking past two adjacent doors spends the minimum time and steps in a single user's trajectory.

Rule 5: D-fingerprint' matching of the nearest door to entrance. Given that E is one of the building entrances. The time $\{t_1, t_2, \dots, t_s\}$ and steps $\{p_1, p_2, \dots, p_s\}$ spent from E to each D-fingerprint in every users' trajectory are computed. Taking the minimum of the time and steps as:

$$t_{\min} = \min\{t_1, t_2, \dots, t_s\} \quad (7)$$

$$p_{\min} = \min\{p_1, p_2, \dots, p_s\} \quad (8)$$

If $t_{\min}$ and $p_{\min}$ derived by all users correspond to the same D-fingerprint RSS_{D_1} , then RSS_{D_1} is labeled with the door D_1 which is closest to the entrance E .

Rule 6: Recursive D-fingerprint' matching of adjacent doors. Given that the doors $\{D_1, D_2, \dots, D_k\}$ have already been matched with D-fingerprints. The time $\{t_1, t_2, \dots, t_s\}$ and steps $\{p_1, p_2, \dots, p_s\}$ spent from the door D_k to any other D-fingerprint (not include the D-fingerprints $\{rss_1, rss_2, \dots, rss_k\}$ already labeled with the doors $\{D_1, D_2, \dots, D_k\}$) in every users' trajectory are computed. Taking the minimum of the time and steps:

$$t_{\min} = \min\{t_{k+1}, t_{k+2}, \dots, t_s\} \quad (9)$$

$$p_{\min} = \min\{p_{k+1}, p_{k+2}, \dots, p_s\} \quad (10)$$

If $t_{\min}$ and $p_{\min}$ derived by all users correspond to the same D-fingerprint $RSS_{D_{k+1}}$, then $RSS_{D_{k+1}}$ is labeled with the adjacent door D_{k+1} which is closest to the door D_k . Figure 7 illustrates the D-fingerprint matching of the nearest door to entrance and Figure 8 illustrates the recursive D-fingerprint matching of adjacent doors.

(1) D-fingerprint' matching of the nearest door to the entrance

In every user's trajectory that passes through the entrance, the D-fingerprint which cost the least time and minimum steps is confirmed according to Rule 5. This D-fingerprint corresponds to the room door that is closest to the entrance and its physical position is the location of the door that is closest to the entrance, like door D_1 shown in Figure 7.

(2) Recursive D-fingerprint' matching of adjacent doors

In every user's trajectory that passes through the last matched door, the next D-fingerprint which costs the least time and minimum steps to arrive is confirmed according to Rule 6. This D-fingerprint corresponds to the door that is closest to the last matched door D_1 and its physical position is the location of the door that is closest to the last matched door D_1 , like the door D_2 shown in Figure 8.

It should be noted that Rules 5 and 6 avoid precise computation of every users' walking distance, and only the least time and minimum steps from the current D-fingerprint to the next D-fingerprint need to be found in order to match the door with the next D-fingerprints. Although different users

spend different times and steps walking past two similar adjacent doors, the two doors are still the nearest neighbor in each individual user's trajectory, accordingly, the time and steps of walking pass the two doors are the minimum. Based on crowdsourced data, two doors are confirmed to be the nearest neighbor if an overwhelming majority of users' trajectories indicate so. Rule 5 and 6 exploit the local extremum method to bind D-fingerprints with their corresponding locations sequentially. Although Rules 5 and 6 do not necessarily ensure that $t_{\min}$ and $p_{\min}$ derived by all users should correspond to the same D-fingerprint, they are still workable when $t_{\min}$ and $p_{\min}$ of most users' trajectories correspond to the same D-fingerprint based on the crowdsourcing principle.

Figure 7. The D-fingerprint's matching of the nearest door to the entrance.

Figure 8. The recursive D-fingerprints' matching of adjacent doors.

As a building has multiple entrances, the adjacent topologies of all the D-fingerprints formed from different entrances are overlapped or complementary, which ensures the accuracy of doors' fingerprint matching. Figure 9 shows two special cases of D-fingerprints' matching. One case is that two doors are very close and the other case is that two doors are oppositely located. The two cases are analyzed respectively.

- (1) For the case that D_1 is close to D_2 , if D_1 and D_2 are closer than the coverage distance of a fingerprint (1~2 m), the fingerprints generated by D_1 and D_2 can be merged into one fingerprint, that is, they are calibrated as the same fingerprint. If D_1 is far from D_2 , two different fingerprints will be generated separately, and the corresponding location of these two fingerprints will be identified according to their users' sequential trajectories.
- (2) For the case that D_3 is opposite to D_4 , since walking direction and rotational angular velocity are different when users take turns in front of them, the fingerprints of D_3 and D_4 can be identified according to users' trajectories.

These special location features, such as D_1 and D_2 , or D_3 and D_4 , provide more specific and deterministic references (anchors or landmarks) for the whole radio-map. The changes of the

crowdsourced data in special locations are more characteristic, which is beneficial to the identification and matching of D-fingerprints.

Figure 9. The D-fingerprints' matching of doors at special positions.

4.3. Crowdsourced Fingerprint Clustering

Based on D-fingerprints and their physical locations identified in Section 4.2, the locations of other RSS fingerprints in a radio-map can be obtained through a certain method. However, since the crowdsourced RSS fingerprints are collected from a large number of users, and some users would walk through the same place at different times, it may happen in one case that more than one RSS fingerprint corresponds to the same or a neighboring physical location. The calculation cost to acquire all crowdsourced fingerprints' geographical locations is huge, and moreover, the generated radio-map might be redundant and inaccurate, so this paper proposes to cluster the crowdsourced fingerprints by means of the similarity between RSS fingerprints, and pick out some RSS fingerprints to represent the RSS fingerprints in a neighboring small area. Considering the huge number of crowdsourced fingerprints, how many clusters that crowdsourced fingerprints are clustered into and how to select the fingerprint representatives optimally are the key points. Here the Affinity Propagation Cluster (AP-Cluster) method [37] is used to solve these problems. Compared with other cluster methods, AP-Cluster does not need the assignment of the number of clusters, which is determined globally by the similarities among data. The cluster representatives are selected from the original data and can adequately represent their subordinate data to some degree. The properties of AP-Cluster method meet our requirements of crowdsourced fingerprints clustering. AP-Cluster method assumes that all the data points form a network in which each data point can evaluate the others and deliver evaluation parameters. Every data point is considered to be the cluster representative in the initialization phase. In the process of clustering, every data point has a parameter to evaluate its support degree, and simultaneously assess whether it is suitable to be the representative combining with the support degree evaluation given by other data points. The data points who get the maximum support and suitability evaluation are chosen to be the cluster representatives.

According to the AP-Cluster method, all RSS fingerprints are assumed to be a network and each RSS fingerprint calculates the support and suitability evaluation of any fingerprint serving as a representative. The AP-Cluster method has two inputs. The similarity between a pairs of fingerprint $s(RSS_i, RSS_k) = -\Delta RSS_{ik}$ (ΔRSS_{ik} see Rule 2 in Section 4.1), also called clustering similarity, is one of the input parameters. The preference degree of data points being cluster representative P_r ($P_r < 0$) is another input parameter, which influences the number of clusters. The suitability evaluations of fingerprints are initialized to zero: $a(RSS_i, RSS_k) = 0$. The similarity of one fingerprint to itself $s(RSS_k, RSS_k)$ is set to P_r . The four formulas below illuminate how RSS fingerprints generate support and suitability evaluation:

$$r(RSS_i, RSS_k) = s(RSS_i, RSS_k) - \max_{k' \neq k} \{a(RSS_i, RSS_{k'}) + s(RSS_i, RSS_{k'})\} \quad (i \neq k) \quad (11)$$

$$r(RSS_k, RSS_k) = s(RSS_k, RSS_k) - \max_{k' \text{ s.t. } k' \neq k} \{a(RSS_k, RSS_{k'}) + s(RSS_k, RSS_{k'})\} \quad (12)$$

$$a(RSS_i, RSS_k) = \min\{0, r(RSS_k, RSS_k) + \sum_{i' \text{ s.t. } i' \notin \{i, k\}} \max\{0, r(RSS_{i'}, RSS_k)\}\} (i \neq k) \quad (13)$$

$$a(RSS_k, RSS_k) = \sum_{i' \text{ s.t. } i' \notin \{i, k\}} \max\{0, r(RSS_{i'}, RSS_k)\} \quad (14)$$

where both $RSS_{k'}$ and RSS_k are candidate fingerprints to compete to be the cluster representative, $RSS_{i'}$ is other fingerprints except RSS_i and RSS_k . After all fingerprints have been implemented the support evaluation and suitability evaluation by other fingerprints in the network, the cluster representatives and their subordinates can be identified through the following conditions:

- (1) Fingerprint RSS_k is chosen to be cluster representative when $r(RSS_k, RSS_k) + a(RSS_k, RSS_k) > 0$;
- (2) Fingerprint RSS_i is subordinate to RSS_k when RSS_k maximizes $r(RSS_i, RSS_k) + a(RSS_i, RSS_k) (i \neq k)$.

Every fingerprint makes the support evaluation of candidate representative taking into account the clustering similarity between itself and other RSS fingerprint representative candidates, along with the suitability evaluation of other RSS fingerprints as representatives. Every fingerprint will consider other fingerprints' support evaluations to determine its suitability evaluation as a candidate fingerprint representative. Finally, if $r(RSS_k, RSS_k) + a(RSS_k, RSS_k) > 0$, fingerprint RSS_k can be the representative. Those fingerprints that are not representatives would calculate the clustering similarities between themselves and representatives, and then decide the representative they belong to according to the maximum similarity principle. To restrict the overall computation cost, the evaluation range of a RSS fingerprint is limited to those RSS fingerprints that have at least 20% identical Aps, so by using the AP-Cluster method and similarities between RSS fingerprints, the crowdsourced fingerprints are clustered and RSS fingerprint representatives can be obtained efficiently, which facilitates the construction of a radio-map afterwards.

Note that the doors' RSS fingerprints are included in the acquired clustering results. Since the doors of the indoor area can be considered as the anchors and their geographical locations are known, the fingerprint representatives of those fingerprints recorded in a small neighboring area of room door are adjusted to D-fingerprints. The other fingerprint representatives in clustering results which are unaware of their physical locations are referred to as C-fingerprints.

Rule 7: Given that $RSS_i (i = 1, 2, \dots, h)$ are the representative fingerprints after clustering the crowdsourced fingerprints and some of the representative fingerprints $RSS_j (j \leq h)$ are similar to the D-fingerprints (Rule 2). Except for the fingerprints $RSS_j (j \leq h)$, the remaining representative fingerprints are the C-fingerprints.

4.4. Fingerprint Partition

To ensure the accuracy of the constructed radio-map, we narrow down the geographical scope to determine the physical locations of C-fingerprints, and all fingerprints including D-fingerprints and C-fingerprints are partitioned into a few parts. Since the physical locations of D-fingerprints are determined already, the corresponding geographical corridor areas including certain D-fingerprints can be confirmed as well. The RSS fingerprints are similar when their corresponding physical locations are in close proximity to each other. The fingerprints whose corresponding physical locations are close can be divided into the same part. The fingerprints partition would not care about the integrity of geographical corridor of every part. For example, a straight corridor could be partitioned into several small areas, or the public area of two vertical crossing corridors could be assigned to one area. K-means method [38] can classify the data into the specified number of parts. The number of parts can be set and employed to realize the fingerprints partition. Figure 10 gives the schematic drawing of fingerprints partition for all D-fingerprints and C-fingerprints. As seen from Figure 10, every part contains multiple D-fingerprints, and the physical locations of D-fingerprints determine the corresponding geographical corridor areas of the parts. The C-fingerprints in each part are not labeled with their physical locations.

Figure 10. The schematic drawing of fingerprints partition.

4.5. Radio-Map Construction

According to the similarity between fingerprints, all the D-fingerprints and C-fingerprints in a fingerprint part can be arranged into an orderly fingerprint chain. The positions of C-fingerprints in fingerprint chain can be determined successively according to their similarity degree to D-fingerprints in the fingerprint chain, as depicted in Figure 11. In Figure 11, there are three C-fingerprints RSS_{C1} , RSS_{C2} and RSS_{C3} between D-fingerprints RSS_{D3} and RSS_{D4} . The position of C-fingerprints RSS_{C1} and RSS_{C3} in the fingerprint chain can firstly be confirmed because among the C-fingerprints between RSS_{D3} and RSS_{D4} , RSS_{C1} and RSS_{C3} are most similar to RSS_{D3} and RSS_{D4} , respectively. Later, the position of C-fingerprint RSS_{C2} in the fingerprint chain can be decided. The sequential relationships between C-fingerprints in the generated fingerprint chain are the same as the sequential relationships between their corresponding physical locations.

Figure 11. The generation of fingerprint chain.

The physical locations of C-fingerprints can be determined by using the physical locations of D-fingerprints, the physical distances between two adjacent doors as well as the similarities between C-fingerprints and D-fingerprints. The steps are as follows.

- (1) Calculate the physical distance d between two doors whose D-fingerprints are both nearest to C-fingerprint based on the reference of fingerprint chain (e.g., the physical distance d_{D3D4} between door D_3 and door D_4 in Figure 11).
- (2) Calculate the similarity δ between the two doors' D-fingerprints (e.g., the similarity δ_{D3D4} between door D_3 's D-fingerprint RSS_{D3} and door D_4 's D-fingerprint RSS_{D4}).
- (3) Calculate the similarity δ' between C-fingerprint and the D-fingerprint that is closest to C-fingerprint in fingerprint chain (e.g., the similarity δ_{C1D3} between RSS_{C1} and RSS_{D3}).
- (4) Calculate the physical distance Δd between C-fingerprint and the D-fingerprint that is closest to the C-fingerprint in the fingerprint chain (e.g., the distance Δd_{C1D3} between C-fingerprint RSS_{C1} and D-fingerprint RSS_{D3}):

$$\Delta d = \frac{\delta'}{\delta} \times d \quad (15)$$

According to the C-fingerprints' sequence in fingerprints chain and the physical distance Δd , the location of C-fingerprints is finally obtained.

To sum up, the doors' D-fingerprints are identified from the crowdsourced fingerprints including RSS fingerprints, walking acceleration, changing tendencies of walking direction and rotational angular velocity when users walk through doors firstly. The doors' D-fingerprints are matched according to the feature that the least time and minimum steps passing through two adjacent doors in a single user's trajectory in the second step. Thirdly, the crowdsourced fingerprints are clustered to representative fingerprints based on the AP-Cluster method. Then, the fingerprints are partitioned based on the K-means method in the fourth step. Finally, the radio-map is constructed on the basis of the similarity of fingerprints. The whole process is summarized as Figure 12. The RACC algorithm performs the characteristic mining and matching to millions of crowdsourced data, and extracts hundreds of representative RSS fingerprints labeled with physical locations, and provides a reference to online localization.

Figure 12. The procedure of radio-map automatic construction.

4.6. An Example of RACC

The main idea of RACC is to find the doors' fingerprints in crowdsourced fingerprints firstly (based on the changing tendencies of users' smartphone sensors) and their corresponding physical locations. Once the doors' locations are confirmed, the anchors of the whole radio-map are fixed. Then, according to the D-fingerprints and their locations, the representative fingerprints (C-fingerprints) clustered from the crowdsourced fingerprints are placed between D-fingerprints based on the similarity. The whole radio-map is constructed. Here is an example of RACC, as shown in Figure 13.

Figure 13. A simple example of RACC algorithm.

The steps of RACC is briefly described below.

- (1) D-fingerprints are recognized. Users' walking direction and rotational angular velocity will change when users traverse doors $D_1 \sim D_6$. Therefore, doors $D_1 \sim D_6$ are identified and the fingerprints (D-fingerprints) near them are recorded, as the red dots show in Figure 13.

- (2) D-fingerprints are matched to their physical locations. According to the geographical constraints between doors $D_1 \sim D_6$ and users' walking trajectories, $D_1 \sim D_6$'s fingerprints can be matched to their corresponding physical locations.
- (3) C-fingerprints are selected. The crowdsourced fingerprints collected in the area are clustered using the AP-Cluster method to obtain the representative fingerprints (C-fingerprints), as the blue dots shown in Figure 13.
- (4) Fingerprints are partitioned. All fingerprints (D-fingerprints and C-fingerprints) in the area are partitioned to three parts: Q_1 , Q_2 and Q_3 .
- (5) The radio-map is constructed. For every part, C-fingerprints' physical locations can be calculated based on the similarity between D-fingerprints and C-fingerprints. When all C-fingerprints are linked to their corresponding physical locations, the radio map is constructed.

5. Performance Evaluation

In this section, the experiments are conducted to study the performance of RACC on one floor of a typical office building covering 3600 m², in which corridors cover 385 m² and the total number of room doors is 29, as illustrated in Figures 2 and 14a. Figure 14a shows a certain corridor in the experimental area, which contains three doors (marked by red boxes).

Figure 14. Preparation for experiment. (a) A corridor in the experimental area; (b) The crowdsourced data collection interface; (c) The interface of test points' fingerprints collection.

A number of APs are randomly deployed in the experiment area and these APs' physical locations are unknown. Since some deviations would occur in the RSS fingerprints and sensor measurements at the same location from using different smartphones, the crowdsourced data of RACC are all collected on the same type of smartphone (Xiaomi, Android OS 4.2) which supports WiFi and is fully equipped with sensors. A software app is designed and installed on the smartphone to collect and record WiFi information of locations that users walk past, users' walking direction and rotational angular velocity, as shown in Figure 14b. The sampling frequency is 1 Hz. Ten volunteers took part in the data collection. They all carry a smartphone and walk in the experimental area while performing their daily activities, in the meantime, each smartphone carries out its data collection for a total five workdays (from Monday to Friday) during the period of 8:00~18:00. The experiment configuration is shown in Table 2.

Table 2. The experiment configuration of RACC.

Floor Area (m ²)	3600	Sampling Frequency	1 Hz
Corridor area (m ²)	385	Sampling time	8:00~18:00
Number of volunteers	10		

In totally 194 APs were scanned and 546 effective user traces were collected. These traces cover most corridor areas and several room areas. To validate the effectiveness of the automatically constructed radio-map, locations in corridors are selected as the test points for online localization. At every test point, the received APs' MAC addresses, their corresponding RSS value as well as actual locations are recorded. Figure 14c shows the software interface that can manually mark the test points' physical locations.

5.1. Smartphone Measurement Analysis

Since the change tendencies of the electronic compass and gyroscope in users' smartphones are used to realize the D-fingerprints' recognition, the measurement sensitivity of users' smartphone sensors was analyzed.

The measurement sensitivity of the electronic compass was evaluated first. Figure 15 shows the measurement error of the electronic compass when smartphones rotate 90° manually. For 60 measurements, the errors range between $-2^\circ \sim +2^\circ$ (the actual measurements are $90 \pm 2^\circ$), which has little effect on the fingerprint recognition.

Figure 15. The measurement sensitivity of the electronic compass in smartphones.

Figure 16 depicts the angular variation of the electronic compass when users traverse through doors or walk normally in corridors. When users make a turn to walk in or out of doors, the angular variation range of the electronic compass is $60^\circ \sim 115^\circ$ and its median value is approximately 90° . When users walk normally in corridors, the angular variation of electronic compass is less than 15° and the median value is approximately 5° . Therefore, users' turns at doors can be clearly identified by the angular variation of the electronic compass.

Figure 16. The angular variation of the electronic compass (making a turn *vs.* walking normally in corridors).

Figure 17 illustrates the angular velocity variation of the gyroscope when users traverse through doors or walk normally in corridors. When users make a turn to walk in or out doors, the angular velocity variation of the gyroscope fluctuates from 1.2 rad/s to 1.5 rad/s and its median value is approximately 1.3 rad/s. When users walk normally in corridors, the angular velocity fluctuation of gyroscope is less than 0.3 rad/s and the median value is approximately 0.2 rad/s. Thus, users' turning at doors can also be clearly identified by the angular velocity variation of the gyroscope.

Figure 17. The angular velocity variation of the gyroscope (making a turn *vs.* walking normally in corridors).

Therefore, if changes happen in both the electronic compass and gyroscope, users' turning moments can be effectively determined. The predefined thresholds of ε_{cp} and ε_{rot} in Rule 1 could be set as 70° and 1.2 rad/s, respectively, as well in this case.

The fluctuation of RSS fingerprint measurements is studied too. As shown in Figure 18, 500 fingerprints within a 1.2 m area in front of a door are sampled. The collected fingerprints almost cover the shadow area. In addition, a fixed point measurement at point A in front of the door is executed, and data from 120 fingerprints are acquired. In total 16 APs are scanned and the RSS values of the APs range between -60 dBm~ -95 dBm. The analysis of RSS fingerprint measurements is given in Figure 19. For the 120 RSS fingerprints measured at the same fixed point, the RSS similarity (Rule 2) is mainly distributed in the range of 2.5~3.8 dB and the maximum similarity difference is less than 5.3. For the 500 fingerprints sampled within a 1.2 m area in front of the door, the similarity between any two fingerprints mainly fluctuates within the range of 3.8~5.7 and the maximum fingerprint similarity difference is less than 9. The average value of 500 fingerprints is also calculated as D-fingerprints (Rule 3). The similarity between any of the 500 fingerprints and the average value mostly ranges between 2.3~4.3 and the maximum similarity difference is less than 6.6. So when users make a turn to walk in or out of the door, the similarity between any collected fingerprint in front of the door is less than 9. The RSS fingerprint dissimilarity threshold ε_{RSS} (Rule 2) can be set to 9 in this case and the fingerprints sampled within 1.2 m area in front of the door can be merged into the D-fingerprint.

Figure 18. The fingerprint sampling in front of a door.

Figure 19. The analysis of RSS fingerprint measurements.

5.2. Results of Doors' Fingerprint Recognition and Matching

When the walking direction and rotational angular velocity in crowdsourced sensor measurements change dramatically, the number of user trajectories passing through the corresponding RSS fingerprints are analyzed statistically according to Rule 1 and 2 mentioned in Section 4.1. Table 3 gives the results in descending order.

Table 3. Statistical analysis of the number of user traces when walking through the changing points.

Fingerprint Groups Number	Number of User Traces	Fingerprint Groups Number	Number of User Traces	Fingerprint Groups Number	Number of User Traces	Fingerprint Groups Number	Number of User Traces
1	54	11	38	21	24	31	16
2	52	12	36	22	24	32	6
3	52	13	32	23	22	33	5
4	50	14	28	24	22	34	3
5	50	15	28	25	22	35	2
6	46	16	28	26	20	36	2
7	44	17	26	27	18	37	1
8	44	18	26	28	18	38	1
9	42	19	26	29	18	39	1
10	40	20	26	30	18	40	1

From Table 3, there are 40 groups of RSS fingerprints that meet the requirements of Rules 1 and 2. The more the number of user trajectories that pass through a certain RSS fingerprint group is, the higher the possibility of this RSS fingerprint group corresponds to a room door because of the statistical characteristics of our crowdsourced data. The top 31 RSS fingerprint groups are determined to correspond to room doors and corridor intersections. The remaining RSS fingerprint groups are then regarded as the outliers through which users walk abnormally. The No. 4 RSS fingerprint group and No. 9 RSS fingerprint group are recognized to correspond to corridor intersections based on Rule 4 in Section 4.1. Figure 20 shows the recognition results of doors' fingerprint groups and Figure 21 shows the doors' D-fingerprints. It should be noted that Figures 20 and 21 merely illustrate the extraction of doors' fingerprint groups or D-fingerprints from crowdsourced fingerprints. Each fingerprint group or D-fingerprint in Figure 20 or Figure 21 are not annotated to a specific room door. Figure 22 displays the matching results of D-fingerprints.

As seen from Figures 20–22 doors' RSS fingerprints can be efficiently identified and labeled following the recognition and matching procedures in RACC. Assigning physical locations to D-fingerprints provides references for the computation of other RSS fingerprints' physical positions.

Figure 20. The recognition of doors' fingerprint groups.

Figure 21. Doors' D-fingerprints.

Figure 22. The matching of D-fingerprint.

5.3. Results of Radio-map Construction

From Section 4.3, the input preference P_r would influence the number of RSS fingerprint clusters:

$$P_r = \gamma \cdot \frac{\sum_{i=1}^n \sum_{k=1}^n s(RSS_i, RSS_k)}{n} (i \neq k) \quad (16)$$

where γ is the preference coefficient, n is the number of crowdsourced fingerprints, $s(RSS_i, RSS_k)$ is the clustering similarity between RSS_i and RSS_k .

Figure 23 investigates the relationship between the preference coefficient and the number of fingerprint clusters when $0 < \gamma \leq 1$. It can be concluded from Figure 23 that the number of fingerprint clusters would reduce gradually with the increasing preference coefficient. Since the number of fingerprint clusters directly determines the size of the radio-map, the bigger the radio-map is, the more reference information it provides for online localization.

Figure 23. The number of fingerprint clusters *versus* preference coefficient.

The fingerprints partition needs to set the number of parts in advance and every part must contain at least two doors' D-fingerprints. The locations of C-fingerprints in every part can be estimated by referencing the locations of D-fingerprints according to the steps in Section 4.5. The total number of doors in the experimental area is 29, and each part has to contain at least 2 doors, thus the maximum number of parts is $29/2 = 14.5$. The parts number k can be set to the integer between 1~14. On this basis, we pick two typical cases $k = 6$ and $k = 9$ as the number of parts. In accordance with the clustering, partitioning and calculating procedures of RACC, the radio-map construction results when $\gamma = 0.1$, $k = 6$; $\gamma = 0.1$, $k = 9$; $\gamma = 0.2$, $k = 6$ and $\gamma = 0.2$, $k = 9$ are given in Figures 24–27 respectively.

From Figures 24–27 it can be observed that the crowdsourced RSS fingerprints are clustered and partitioned by utilizing the AP-Cluster method and K-means method. The RSS fingerprints in small neighboring areas can be gathered into a cluster and the selected RSS fingerprint representatives can reflect the fingerprint features of their subordinates.

Figure 24. The automatic constructed radio map when $\gamma = 0.1$, $k = 6$.

Figure 25. The automatic constructed radio map when $\gamma = 0.1$, $k = 9$.

Figure 26. The automatic constructed radio map when $\gamma = 0.2$, $k = 6$.

Figure 27. The automatically constructed radio-map when $\gamma = 0.2$, $k = 9$.

5.4. Localization Results

5.4.1. Localization Based on Parts of Radio-Map

To validate the performance of RACC, the online localization is carried out based on the weighted k-nearest neighboring method at 110 test points in the experiment area. The calculation of localization error is:

$$LE = \sqrt{(x_k^e - x_k^r)^2 + (y_k^e - y_k^r)^2} \quad (17)$$

where $P_k^r(x_k^r, y_k^r)$ is the actual physical location of test point k and $P_k^e(x_k^e, y_k^e)$ is the estimated physical location by RACC. The localization result of every part in four radio maps are respectively depicted in Figures 28–31.

From Figures 28 and 29 when $\gamma = 0.1$, $k = 6$, the localization error of each part in the radio-map is mostly in the range of 2.5 m~4.5 m, and especially the localization error of the Q5 part is larger than that of other parts. When $\gamma = 0.1$, $k = 9$, the localization error of each part in the radio-map ranges roughly from 2.5 m to 4 m, and the localization error of Q8 part in particular is larger than that of other parts. The reason is that the Q5 part and Q8 part correspond to the same area in the physical environment in which only two room doors are located, so less location information is available for reference. Furthermore, APs are sparsely distributed in this area resulting in a tiny difference between RSS fingerprints. Accordingly, since the fingerprint chains are generated with relative deviation among fingerprints, the accuracy of radio map decreases, which leads to an increased localization error.

Figure 28. The localization results of parts in radio map when $\gamma = 0.1$, $k = 6$.

Figure 29. The localization results of parts in radio map when $\gamma = 0.1$, $k = 9$.

Figure 30. The localization results of parts in radio map when $\gamma = 0.2$, $k = 6$.

Figure 31. The localization results of parts in radio map when $\gamma = 0.2$, $k = 9$.

From Figures 30 and 31 when $\gamma = 0.2$, $k = 6$, the localization error of each part in the automatically constructed radio-map is nearly from 4 m to 7 m, while it is almost in the scope of 4 m~6 m when $\gamma = 0.2$, $k = 9$. This indicates that properly increasing the number of parts would improve the accuracy of the generated radio-map if each part contains at least two room doors' D-fingerprints. The reason is that each part will have less fingerprints when the number of parts increases, thus more uncertain

aspects are removed when the fingerprint chain is ordered, and a more accurate RSS fingerprint chain can be obtained accordingly.

One thing that needs to be pointed out is that the localization error of RACC from 2.5 m to 4 m ($\gamma = 0.1, k = 9$) or from 4 m to 6 m ($\gamma = 0.2, k = 9$) is not a very accurate localization. For some parts of the constructed radio-map, the accuracy is no better than that of the manually constructed ones. However, considering the combination of expenditure in fingerprint collection and benefit in localization accuracy, RACC is more practical than traditional data sampling methods. RACC can construct the radio-map of indoor areas automatically without participation of sampling technicians, which dramatically reduces the cost of fingerprint database construction. This is of great significance for the application and promotion of indoor fingerprint-based localization techniques.

From the comparison between Figures 28 and 30 (or Figures 29 and 31), it can also be learned that the average localization performance when $\gamma = 0.1$ is better than that when $\gamma = 0.2$, which means the setting of preference coefficient in fingerprint clustering has a great impact on the localization accuracy. The smaller the preference coefficient is, the more RSS fingerprints a radio-map contains, and the more accurately users are located. The reason is that the radio-map serves as a database which offers online searching and matching, so user's current fingerprints can be matched to a higher degree if the radio-map contains more fingerprints. However, the size of the radio-map should not be too large, otherwise it would occupy too much storage space and take more matching time, which results in online localization latency.

5.4.2. Impact of Parameters on Localization

The relationship between the localization error and number of users' walking trajectories is shown in Figure 32. The localization precision of RACC will reduce with the decrease of the number of users' walking trajectories. The averaged localization error is no more than 5.8 m when the number of users' walking trajectories is no less than 300, whereas the averaged localization error is about 16 m when the number of users' walking trajectories is only 200. It is because that RACC is a localization method based on crowdsourced data, it needs a lot of ergodic crowdsourced data to construct a radio-map automatically. Therefore, when the number of users' walking trajectories reduces to a threshold, the ergodicity of crowdsourced data will be weakened correspondingly. For instance, if there are some doors where users seldom or never walk by, major deviations will occur during the matching process of these doors in the RACC algorithm, which will also cause an increase in the average localization error. Thus it is necessary to ensure the ergodicity of crowdsourced data in order to guarantee the effectiveness of RACC.

Figure 32. Relationship between the number of users' traces and localization error.

When the preference coefficient γ is picked as 0.1, 0.2, 0.3, 0.4 and 0.5, the number of fingerprint clusters are 171, 96, 57, 37 and 30, respectively, and the corresponding density of the sampling position

of fingerprints are 1.1 m, 2 m, 3 m, 5.7 m and 8.3 m apart. The relationship between the localization error and the fingerprint density is shown in Figure 33.

Figure 33. Relationship between the fingerprint density and localization error.

The localization error increases with the reduction of the fingerprint density. The reason is that effective reference information is reduced when the number of the representative fingerprints decreases, and then the localization precision is influenced.

RACC uses the AP-Cluster method to cluster the crowdsourced fingerprints. By controlling the preference coefficient γ in the AP-Cluster method, a suitable radio-map fingerprint density can be obtained and thus the localization accuracy is improved. Since the number of clustered RSS fingerprints determines the size of the radio-map, the more fingerprints the radio-map has, and the more reference information it provides, the higher the localization accuracy will be. On the other hand, the matching calculation in online localization will increase simultaneously.

5.4.3. Comparison with Manual Calibration

The localization performance of RACC is compared with that of a manual fingerprints calibration. In corridors of the experimental area, sampling points are set every 2 m, and 60 fingerprints are collected at each point. The average value of 60 fingerprints is considered as the sampling points' RSS fingerprint. A total of 96 points are sampled and the manually calibrated radio-map is generated containing 96 RSS fingerprints. The radio-map constructed by RACC when $\gamma = 0.2$, $k = 9$ also has 96 fingerprints and the average distance between fingerprints is 2 m. Detailed information is shown in Table 4.

Table 4. The information of radio map constructed by RACC ($\gamma = 0.2$, $k = 9$).

Part Number	Part Length (m)	Number of Doors	Number of RSS Fingerprints after Clustering	Average Distance between Fingerprints (m)
Q1	20	4	9	2.22
Q2	16	2	7	2.29
Q3	39	4	21	1.86
Q4	15	2	6	2.50
Q5	20	4	11	1.82
Q6	15	3	8	1.88
Q7	19	4	10	1.90
Q8	25	2	10	2.50
Q9	26	4	14	1.86
SUM	195	29	96	2.03

For the two radio-maps constructed by manual calibration and RACC, we utilize the test points to perform the online localization based on the weighted kNN method (Section 3.3). The localization results are shown in Figure 34.

Figure 34. Localization results of manual calibration and RACC.

The 60%ile and 80%ile localization errors of RACC are respectively 3 m and 4.3 m, while the 60%ile and 80%ile localization errors of manual calibration are respectively 2.2 m and 3.1 m, which means the localization accuracy of RACC can reach about 70% of the localization accuracy of manual calibration. However, as an automatic construction method, RACC reduces greatly the cost of fingerprint sampling and radio-map construction, effectively solving the resource-consumption problem when a location fingerprint database is established.

5.4.4. Comparison with Other Localization Methods

The localization performance of RACC are compared with that of Zee [14] and LiFS [35], as shown in Table 5. We can see from the Table 5 that RACC takes advantage of users' smartphone sensors (accelerometer, gyroscope and compass) and produces comparable localization accuracy for indoor corridors.

Table 5. Comparison with other localization methods.

Method	Sensors	Floor Plan	Experimental Areas	Reported 50%ile Accuracy	Reported 80%ile Accuracy	Fingerprint Density
Zee	Accelerometer	Yes	Corridor	7 m (ZeeHorus)	13 m (ZeeHorus)	-
	Gyroscope			6 m (ZeeEZ)	13 m (ZeeEZ)	
	Compass			1.2 m (ZeeHorus-Path Only)	3 m (ZeeHorus-Path Only)	
LIFS	Accelerometer	Yes	Corridor & Room	1.0 m (ZeeEZ-Path Only)	2.3 m (ZeeEZ-Path Only)	
RACC	Accelerometer	Yes	Corridor	4.5 m (LIFS)	7 m (LIFS)	/4 m ²
	Gyroscope			1.5 m (LIFS-TM)	3.5 m (LIFS-TM)	
	Compass					
RACC	Accelerometer	Yes	Corridor	1.7 m	2.2 m	/1 m
	Gyroscope					
	Compass			2.9 m	4.3 m	/2 m

The key feature of RACC is that it uses the changing tendencies of users' smartphone sensors to identify the indoor anchors (doors) and regards the locations of doors as the fixed reference of the whole radio-map. The representative fingerprints from the crowdsourced data can be obtained by

the AP-Cluster method and the radio-map is constructed after linking the representative fingerprints to their corresponding physical locations on the basis of the fixed reference and the similarity between fingerprints.

The complex environment in indoor buildings can provide a lot of potential available anchor information. The GPS signal strength would greatly decrease when users enter the building, which will help to determine that users are at a certain building entrance. The accelerometer in users' smartphones would fluctuate considerably when the elevator starts or stops, which will help determine that the current location is a certain elevator's location. At some places in corridors, there might exist a distinctive turning angle, or a four-star strength GPS signal can be received, or no WiFi or GSM signal can be detected. Those characteristics can be used as possible anchors for radio-map construction. Increasing the number of anchors will provide more reference information, reduce the radio-map construction errors and thus improve the localization accuracy.

All in all, the proposed algorithm RACC utilizes crowdsourced WiFi information and sensor measurements to construct the radio map automatically. It greatly reduces the resource consumption of offline data collection while ensuring an acceptable localization accuracy, and thus has the merits of flexibility and extensibility over traditional methods.

6. Conclusions

In this paper, an automatic radio-map construction algorithm based on crowdsourcing is proposed to solve the problem of high cost and poor extensibility of offline fingerprint collection in traditional fingerprint-based localization algorithms. Based on the proposed RACC algorithm, the crowdsourced WiFi information and sensor measurements can be implicitly collected through the smartphones carried by users, and doors' RSS fingerprints can be identified according to their statistical features when users walk through doors. Doors' fingerprints can be annotated with their corresponding physical locations according to the proposed adjacent recursive matching method. After fingerprints are clustered and partitioned based on AP-Cluster method and K-means method, the radio-map is finally constructed in accordance with the similarity. The experimental results show that RACC can efficiently solve the resource-consuming problem and achieve a competitive localization accuracy.

Acknowledgments: The authors are grateful to the anonymous reviewers for their industrious work and insightful comments. This work is supported by the National Natural Science Foundation of China under grant No. 61201317, No. 61473021 and No. 61421063.

Author Contributions: Ning Yu and Chenxian Xiao contributed to the idea and designed the algorithm. Yinfeng Wu and Renjian Feng designed and analyzed the experimental results. All the authors involved in writing the paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Han, G.; Xu, H.H.; Duong, T.Q.; Jiang, J.F.; Hara, T. Localization algorithms of wireless sensor networks: A survey. *Telecommun. Syst.* **2013**, *52*, 2419–2436. [[CrossRef](#)]
2. Ikeda, Y.; Hatano, H.; Fujii, M.; Ito, A.; Watanabe, Y. A study on GPS positioning method with assistance of a distance sensor. In Proceedings of the Fourteenth International Conference on Networks, Barcelona, Spain, 19–24 April 2015; pp. 109–114.
3. Denkovski, D.; Angelichinoski, M.; Atanasovski, V.; Gavrilovska, L. RSS-based self-localization framework for future wireless networks. *Wirel. Pers. Commun.* **2014**, *78*, 1755–1776. [[CrossRef](#)]
4. Shen, J.; Molisch, A.F.; Salmi, J. Accurate passive location estimation using TOA measurements. *IEEE Trans. Wirel. Commun.* **2012**, *11*, 2182–2192. [[CrossRef](#)]
5. Singh, P.; Agrawal, S. TDOA based node localization in WSN using neural networks. In Proceedings of the 2013 International Conference on Communication Systems and Network Technologies, Gwalior, India, 6–8 April 2013; pp. 400–404.

6. Shao, H.J.; Zhang, X.P.; Wang, Z. Efficient closed-form algorithms for AOA based self-Localization of sensor nodes using auxiliary variables. *IEEE Trans. Signal Proc.* **2014**, *62*, 2580–2594. [[CrossRef](#)]
7. Liu, H.; Darabi, H.; Banerjee, P.; Liu, J. Survey of wireless indoor positioning techniques and systems. *IEEE Trans. Syst. Man Cybern.* **2007**, *37*, 1067–1080. [[CrossRef](#)]
8. Zhou, M.; Tian, Z.; Xu, K.; Yu, X.; Wu, H. Theoretical entropy assessment of fingerprint-based Wi-Fi localization accuracy. *Expert Syst. Appl.* **2013**, *40*, 6136–6149. [[CrossRef](#)]
9. Kanaris, L.; Kokkinis, A.; Raspopoulos, M.; Liotta, A.; Stavrou, S. Improving RSS fingerprint-based localization using directional antennas. In Proceedings of the 8th European Conference on Antennas and Propagation, Hague, Holland, 6–11 April 2014; pp. 1593–1597.
10. Stella, M.; Russo, M.; Begušić, D. Fingerprinting based localization in heterogeneous wireless networks. *Expert Syst. Appl.* **2014**, *41*, 6738–6747. [[CrossRef](#)]
11. Ding, G.; Tan, Z.; Wu, J.; Zhang, J.B. Efficient indoor fingerprinting localization technique using regional propagation model. *IEICE Trans. Commun.* **2014**, *97*, 1728–1741. [[CrossRef](#)]
12. Ma, R.; Guo, Q.; Hu, C.; Xue, J. An improved WiFi indoor positioning algorithm by weighted fusion. *Sensors* **2015**, *15*, 21824–21843. [[CrossRef](#)] [[PubMed](#)]
13. Sun, Y.; Liu, M.; Meng, M.Q.H. WiFi signal strength-based robot indoor localization. In Proceedings of the ICIA, Hailar, China, 28–30 July 2014; pp. 250–256.
14. Rai, A.; Chintalapudi, K.K.; Padmanabhan, V.N.; Sen, R. Zee: Zero-effort crowdsourcing for indoor localization. In Proceedings of the 18th annual international conference on Mobile computing and networking, Istanbul, Turkey, 22–26 August 2012; pp. 293–304.
15. Su, X.; Tong, H.; Ji, P. Activity recognition with smartphone sensors. *Tsinghua Sci. Technol.* **2014**, *19*, 235–249.
16. Bahl, P.; Padmanabhan, V.N. RADAR: An in-building RF-based user location and tracking system. In Proceedings of the IEEE INFOCOM, Tel Aviv, Israel, 26–30 March 2000; pp. 775–784.
17. Castro, P.; Chiu, P.; Kremenek, T.; Muntz, R. A probabilistic room location service for wireless networked environments. In Proceedings of the Ubicomp'01, Atlanta, GA, USA, 30 September–2 October 2001; pp. 18–34.
18. Roos, T.; Myllymäki, P.; Tirri, H.; Misikangas, P.; Sievänen, J. A probabilistic approach to WLAN user location estimation. *Int. J. Wirel. Inf. Netw.* **2002**, *9*, 155–164. [[CrossRef](#)]
19. Youssef, M.; Agrawala, A. The Horus location determination system. *Wirel. Netw.* **2008**, *14*, 357–374. [[CrossRef](#)]
20. Brunato, M.; Battiti, R. Statistical learning theory for location fingerprinting in wireless LANs. *Comput. Netw.* **2005**, *47*, 825–845. [[CrossRef](#)]
21. Battiti, R.; Le, N.T.; Villani, A. Location-aware computing: A neural network model for determining location in wireless LANs. *Int. Semicond. Conf.* **2002**, *4*, 175–179.
22. Laoudias, C.; Eliades, D.G.; Kemppi, P. Indoor localization using neural networks with location fingerprints. In Proceedings of the ICANN '09, Limassol, Cyprus, 14–17 September 2009; pp. 954–963.
23. Yin, J.; Yang, Q.; Ni, L. Adaptive temporal radio maps for indoor location estimation. In Proceedings of the PerCom '05, Auai Island, HI, USA, 8–12 March 2005; pp. 85–94.
24. Pan, J.J.; Pan, S.J.; Yin, J.; Ni, L.M.; Yang, Q. Tracking mobile users in wireless networks via semi-supervised co-localization. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 587–600. [[CrossRef](#)] [[PubMed](#)]
25. Au, A.W.S.; Chen, F.; Valaee, S.; Reyes, S.; Sorour, S.; Markowitz, S.N.; Gold, D.; Gordon, K.; Eizenman, M. Indoor tracking and navigation using received signal strength and compressive sensing on a mobile device. *IEEE Trans. Mob. Comput.* **2013**, *12*, 2050–2062. [[CrossRef](#)]
26. Qiu, K.; Zhang, F.; Liu, M. Visible light communication-based indoor localization using Gaussian Process. In Proceedings of the IROS, Hamburg, Germany, 28 September–2 October 2015; pp. 3125–3130.
27. Bolliger, P. Redpin: Adaptive, zero-configuration indoor localization through user collaboration. In Proceedings of the MELT '08, San Francisco, CA, USA, 19 September 2008; pp. 55–60.
28. Lee, M.; Yang, H.; Han, D.; Yu, C. Crowdsourced radiomap for room-level place recognition in urban environment. In Proceedings of the 8th International Conference on PERCOM Workshops, Mannheim, Germany, 29 March–2 April 2010; pp. 648–653.
29. Yang, S.; Dessai, P.; Verma, M.; Gerla, M. FreeLoc: Calibration-free crowdsourced indoor localization. In Proceedings of the INFOCOM, Turin, Italy, 14–19 April 2013; pp. 2481–2489.
30. Park, J.; Charrow, B.; Curtis, D.; Battat, J. Growing an organic indoor location system. In Proceedings of the 8th international conference on Mobile Systems, Applications and Services, San Francisco, CA, USA, 15–18 June 2010; pp. 271–284.

31. Ledlie, J.; Park, J.G.; Curtis, D.; Cavalcante, A.; Camara, L.; Costa, A.; Vieira, R. Molé: A scalable, user-generated WiFi positioning engine. *J. Locat. Based Serv.* **2012**, *6*, 21–23. [[CrossRef](#)]
32. Sorour, S.; Lostanlen, Y.; Valaee, S. Reduced-effort generation of indoor radio maps using crowdsourcing and manifold alignment. In Proceedings of the IST '12, Tehran, Iran, 6–8 November 2012; pp. 354–358.
33. Azizyan, M.; Constandache, I.; Choudhury, R.R. SurroundSense: Mobile phone localization via ambience fingerprinting. In Proceedings of the MoCom '09, Beijing, China, 20–25 September 2009; pp. 261–272.
34. Wu, C.; Yang, Z.; Liu, Y.; Xi, W. WILL: Wireless indoor localization without site survey. *IEEE Trans. Parallel Distrib. Syst.* **2012**, *24*, 64–72.
35. Wu, C.; Yang, Z.; Liu, Y. Smartphones based crowdsourcing for indoor localization. *IEEE Trans. Mob. Comput.* **2015**, *14*, 444–457. [[CrossRef](#)]
36. Alzantot, M.; Youssef, M. CrowdInside: Automatic construction of indoor floorplans. In Proceedings of the ACM SIGSPATIAL GIS'12, Redondo Beach, CA, USA, 6–9 November 2012; pp. 99–108.
37. Frey, B.J.; Delbert, D. Clustering by passing messages between data points. *Science* **2007**, *315*, 972–976. [[CrossRef](#)] [[PubMed](#)]
38. Sssikumar, P.; Khara, S. K-means clustering in wireless sensor networks. In Proceedings of the IEEE Fourth International Conference on Computational Intelligence and Communication Networks, Mathura, India, 3–5 November 2012; pp. 140–144.

© 2016 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC-BY) license (<http://creativecommons.org/licenses/by/4.0/>).