sensors

MDPI

Article

Indoor Localization Based on Weighted Surfacing from Crowdsourced Samples

Junhong Lin [1], Bang Wang [1,*], Guang Yang [1] and Mu Zhou [2]

[1] School of Electronic Information and Communications, Huazhong University of Science and Technology (HUST), Wuhan 430074, China; junhonghust@hust.edu.cn (J.L.); guangyang@hust.edu.cn (G.Y.)

[2] Chongqing Key Lab of Mobile Communications Technology, School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China; zhoumu@cqupt.edu.cn

* Correspondence: wangbang@hust.edu.cn; Tel.: +86-27-8754-3236

Received: 9 August 2018; Accepted: 3 September 2018; Published: 7 September 2018

Abstract: Fingerprinting-based indoor localization suffers from its time-consuming and labor-intensive site survey. As a promising solution, sample crowdsourcing has been recently promoted to exploit casually collected samples for building offline fingerprint database. However, crowdsourced samples may be annotated with erroneous locations, which raises a serious question about whether they are reliable for database construction. In this paper, we propose a cross-domain cluster intersection algorithm to weight each sample reliability. We then select those samples with higher weight to construct radio propagation surfaces by fitting polynomial functions. Furthermore, we employ an entropy-like measure to weight constructed surfaces for quantifying their different subarea consistencies and location discriminations in online positioning. Field measurements and experiments show that the proposed scheme can achieve high localization accuracy by well dealing with the sample annotation error and nonuniform density challenges.

Keywords: fingerprinting localization; sample crowdsourcing; sample weighting; surface fitting

1. Introduction

Fingerprinting has been extensively researched for indoor localization systems in the last decade [1–4]. The basic idea is based on the assumption that each indoor location can be identified by a unique signal feature, called *fingerprint*. The widely used fingerprint is a vector of the *received signal strengths* (RSS) from the *access points* (AP) of wireless local access networks. The location of a test fingerprint can be estimated to a known location with minimal signal difference. One of the key challenges to support such fingerprinting localization is to construct an indoor *radio map* in the offline training phase [5–8]. Normally, the indoor environment is divided into non-overlapping grid cells. *Site survey* is often used to collect RSS samples at each grid center by surveyors for training one *grid fingerprint* for each grid. However, this scheme suffers from the time-consuming and labor-intensive site survey for radio map construction.

Recently, fingerprint crowdsourcing has been promoted to relieve or even eliminate the burdensome site survey by exploiting casually collected RSS samples [6–9]. Although not collected at specified locations, crowdsourced RSS samples still need to be annotated with some location information for fingerprint database construction. To this end, a common approach is to extract RSS samples from pedestrian movement trajectory [10–12]. As long as a trajectory can be correctly matched to one physical route, each step position can be obtained from the floor plan to annotate the corresponding step RSS sample.

Although fingerprint crowdsourcing seems a promising approach, care must be taken to deal with the samples with erroneously annotated locations. Compared with the site survey, such erroneous location annotation of crowdsourced samples could lead to an inaccurate radio map and degrade the performance of fingerprinting-based localization. Besides annotation errors, another challenge lies in that crowdsourced samples may not be uniformly distributed in the whole environment.

In this paper, we study the indoor localization through constructing radio propagation surfaces from crowdsourced samples. For each AP, its surface takes a location as input and outputs an estimated RSS of this location. To deal with annotation errors, we propose a *cross-domain cluster intersection* algorithm to assign each sample a reliability weight, which exploits the sample clustering results from both the physical and signal space. We next select a subset of weighted samples to fit each AP a surface from polynomial primary functions and construct subarea fingerprints by sampling AP surfaces. Furthermore, we compute two weights for each AP surface for describing its subarea consistency and location discrimination capability in online positioning. A two-step positioning algorithm is proposed to first determine the belonging subarea for a test sample, and then a weighted surface search is exploited to estimate its location within the subarea. We conducted field measurements and experiments. Compared with the peer schemes, results validate the effectiveness and robustness of the proposed scheme in terms of its lower localization error when facing sample annotation error and nonuniform density challenges.

The rest of the paper is organized as follows. Section 2 briefly reviews the most related work as well as the proposed system. The proposed offline surface fitting algorithm is presented in Section 3. Section 4 presents our online localization algorithm. Field measures are used for experiments and the results are provided in Section 5. Finally, Section 6 concludes the paper.

2. Related Work and System Overview

2.1. Related Work

Several fingerprinting systems based on sample crowdsourcing have been proposed for indoor localization in previous studies [13–18]. For example, Chen and Wang [13] proposed using a density-based clustering technique to group crowdsourced samples to generate a cluster fingerprint and using a matching algorithm to assign each cluster fingerprint to one subarea for room-level localization. Liu et al. [14] also applied crowdsourced samples for room-level localization yet with an improved energy-efficient sampling approach. Chang et al. [15] applied a local Gaussian process to construct grid fingerprints from crowdsourced samples. Jung et al. [16] adopted a hybrid global-local optimization scheme to determine the location of fingerprint sequences based on the constraint of the indoor structure, rather than using labeled fingerprints. They also proposed an unsupervised learning method to calibrate the localization model.

In the literature, many have proposed to extract crowdsourced samples from pedestrian trajectories. The core idea is to match a trajectory to one physical route such that each sample on a trajectory can be labeled with one location in the route [19–25]. For example, Kim et al. [19] combined the lightweight site survey and fingerprint crowdsourcing for radio map construction. They first constructed an initial radio map according to the lightweight site survey and use the *pedestrian dead reckoning* (PDR) to match the the war-walking paths into the radio map. Huang et al. [20] exploited layout landmarks such as the cross points of corridors for matching pedestrian trajectories to physical routes. Zhou et al. [23] proposed to transform the indoor layout into a semantic graph to map with activity sequences contained within the trajectories. Zhou et al. [25] applied a density-based spatial clustering algorithm to determine hotspots which are then mapped to physical subareas.

For crowdsourced samples, the conventional approach is to construct a radio map for grid fingerprints. In Ref. [26], Wang et al. proposed using polynomial functionals to fit a propagation surface for each AP based on a few reference fingerprints with correct location annotations. Ye and Wang [27] applied the surfacing method to deal with the problem of non-uniformly distributed

crowdsourced samples, with the objective of composing grid fingerprints for radio map construction. Unlike these approaches, this paper proposes to exploit crowdsourced samples for fitting radio propagation surfaces. As crowdsourced samples normally have inaccurate location labels, how to construct a reliable surface is rather challenging. In this paper, we propose a sample weighting algorithm and apply weighted samples to fitting surfaces.

2.2. System Overview

We divided an indoor environment into several distinct subareas, such as rooms, corridors, etc., according to their functional layout by inherent obstructions and partitions such as concrete walls. We assumed that each crowdsourced sample has been annotated with some location, though possibly with annotation errors. We attributed each sample to one subarea according to its annotated location. The proposed system also consists of the offline and online phases.

The offline phase consists of four steps: Weighting crowdsourced samples assigns each crowdsourced sample a *reliability weight* based on our proposed *cross-domain cluster intersection* algorithm. Fitting radio surfaces constructs a radio propagation surface for each AP based on the weighted samples. Weighting fitted surfaces further assigns each fitted surface with two weights for discriminating their contributions for online localization. Constructing subarea fingerprints creates an RSS fingerprint for each subarea from its fitted and weighted surfaces.

The online localization consists of two steps: Subarea determination first locates an online test fingerprint into one subarea according to our proposed weighted signal distance. Location search searches the coordinate for the test fingerprint based on the gradient search on the constructed surfaces. Figure 1 presents the main flowchart of the proposed system, and Table 1 lists the symbols used in this paper as well as their notations.

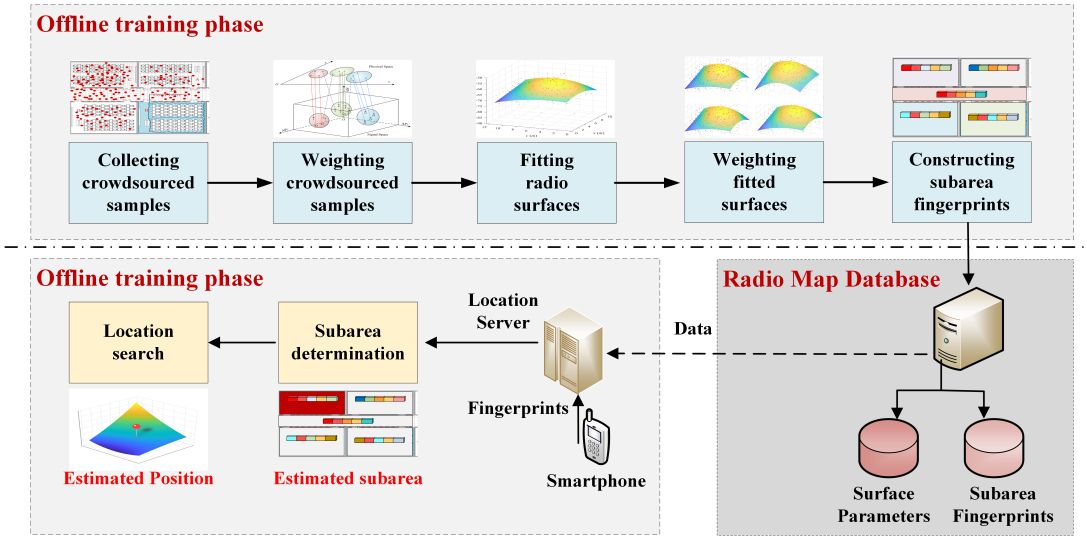

Figure 1. The flowchart of the proposed system: In the offline phase, crowdsourced samples are each weighted according to our algorithm. For each access point and for one subarea, its radio propagation surface is firstly fitted and also weighted from those selected and weighted samples. Subarea fingerprints are then composed from fitted surfaces. In the online phase, a test sample is first compared with subarea fingerprints to determine its belonging subarea, and then a gradient search is used to estimate its exact location.

Table 1. Table of symbols.

Symbol	Definition
$\mathcal{S}$	A set of crowdsourced samples in one subarea.
M	The number of crowdsourced samples in $\mathcal{S}$, $M = \|\mathcal{S}\|$.
$\boldsymbol{s}_i$	The ith crowdsourced sample in $\mathcal{S}$.
$\vec{l}_i$	The annotated location of the ith crowdsourced sample.
$\vec{r}_i$	The RSS vector of the ith crowdsourced sample.
N	The maximum number of hearable AP in $\mathcal{S}$.
K	The number of clusters.
$\mathcal{C}^p$	The set of clusters in the physical space.
$\mathcal{C}^s$	The set of clusters in the signal space.
γ_i	The cross-domain cluster coefficient of the ith sample.
ω_i	The reliability weight of the ith sample.
$\phi(x, y)$	The RSS surface function.
p_{th}	The percentile threshold in sample selection method.
ω_{th}	The weight threshold in sample selection method.
$\vec{\omega}$	The increasing order of sample reliability weight.
ω_k	The reliability weight at the p_{th} percentile in $\vec{\omega}$.
$\mathcal{S}'$	The set of select samples.
$\mathcal{A}$	The set of hearable Aps by samples in $\mathcal{S}'$.
α_{ij}	The surface coefficient of the RSS surface function.
$\mathcal{R}$	The set of RSS values from an AP in $\mathcal{S}'$.
$\bar{r}_i$	The normalized elements in $\mathcal{R}$.
η	The entropy-like quantity for each AP in $\mathcal{A}$.
ρ_n^{sub}	The surface weight of nth AP in $\mathcal{A}$ for subarea determination.
ρ_n^{loc}	The surface weight of nth AP in $\mathcal{A}$ for location search.
$\vec{f}$	Subarea fingerprint.
$\mathcal{G}$	The set of grid cells in one subarea.
G	The number of grids in $\mathcal{G}$, $G = \|\mathcal{G}\|$.
$\vec{f}_t$	The RSS vector of a test sample.
$\vec{f}_s$	The sth subarea fingerprint.
$\mathcal{A}_{int}$	The set of hearable APs by both $\vec{f}_t$ and $\vec{f}_s$.
D_s	The weighted signal distance between the test sample and a subarea.
M_g	The number of grid cells.
σ	The standard deviation of location offset.
$\mathcal{S}_{site}$	The set of samples from site survey.
$\mathcal{S}_{walk}$	The set of samples from pedestrian trajectories.

3. The Offline Weighted Surfacing Algorithm

3.1. Weighting Crowdsourced Samples

In one subarea, e.g., a room, let $\mathcal{S} = \{\boldsymbol{s}_i, ..., \boldsymbol{s}_M\}$ denote its set of M crowdsourced samples. A sample $\boldsymbol{s}_i = (\vec{l}_i, \vec{r}_i)$ consists of two parts: $\vec{l}_i = (x_i, y_i)$ is its annotated location; and $\vec{r}_i = (r_{i1}, r_{i2}, ...r_{iN})$ the received RSS vector where N is the maximum number of hearable APs in one subarea. For one sample $\boldsymbol{s}_i$, it is possible that not all the N APs could be heard, that is, some r_{ij} ($j < N$) might not be available in $\boldsymbol{s}_i$. In this case, to allow the clustering and surfacing algorithm to run normally, we simply set it to a very small RSS value, $r_{min} = -90$ dBm, which is the lower bound of the collected signal strength, during the following sample clustering and weighting process.

A crowdsourced sample $\boldsymbol{s}_i$ may not be reliable in that its annotated location $\vec{l}_i$, RSS measurement $\vec{r}_i$, or both might have some errors. However, among a large number of such samples, we conjecture that some statistical relations could be extracted from the similarities between the physical and signal space. Consider the following example of two samples $\boldsymbol{s}_i$ and $\boldsymbol{s}_j$. Let $d_{ij}^p \triangleq \|\vec{l}_i - \vec{l}_j\|$ and $d_{ij}^s \triangleq \|\vec{r}_i - \vec{r}_j\|$ denote the distance between the two samples in the physical space and signal space, respectively. Suppose that d_{ij}^p is small, indicating that $\boldsymbol{s}_i$ and $\boldsymbol{s}_j$ are close to each other according to their annotated

locations. For a small d_{ij}^s, we could conjecture that both samples are reliable or both samples are unreliable. Although we could not determine which is the real case for only two samples, we might be able to infer the statistic relations from a large number of samples to discriminate unreliable samples. Motivated from such considerations, we next present a *cross-domain cluster intersection* (CCI) algorithm to assign each sample a *reliability weight*.

In both the physical and signal space, we group all samples $\boldsymbol{s}_i \in \mathcal{S}$ into K clusters by the classic *K-means* clustering algorithm. Let $\mathcal{C}^p = \{C_1^p, ..., C_K^p\}$ and $\mathcal{C}^s = \{C_1^s, ..., C_K^s\}$ denote the set of clusters in the physical and signal space, respectively. Notice that a sample $\boldsymbol{s}_i$ is within one of the clusters in $\mathcal{C}^p$ and $\mathcal{C}^s$ simultaneously. We define a *cross-domain cluster coefficient* for such a sample $\boldsymbol{s}_i$ based on the cluster intersection between C_a^p and C_b^s as follows:

$$\gamma_i = \frac{|C_a^p \bigcap C_b^s|^2}{|C_a^p| \times |C_b^s|}. \tag{1}$$

If $C_a^p = C_b^s$, i.e., the two clusters contain the same set of samples, then all such samples have the same coefficient and $\gamma_i = 1$. According to the *K-means* clustering, all samples in C_a^p are closer to this cluster center than to other cluster centers. This is also the case for samples in C_b^s in the signal space in terms of their RSS vector similarities. Therefore, $|C_a^p \bigcap C_b^s|$ describes how many samples are close to each other in both the physical and signal space. A small value of γ_i indicates that $\boldsymbol{s}_i$ is not similar to the majority of the two clusters, which might suggest its unreliability. As the surface fitting is done in the signal space, we further normalize γ_i to assign the sample weight based on the signal space clusters. For each sample $\boldsymbol{s}_i \in C_b^s$, we compute its reliability weight by

$$\omega_i = \frac{\gamma_i}{\max\{\gamma_j | \boldsymbol{s}_j \in C_b^s\}}, \tag{2}$$

where the denominator is the maximum cross-domain cluster coefficient of the samples in the cluster. Figure 2 illustrates the CCI algorithm and computes reliability weights for some samples.

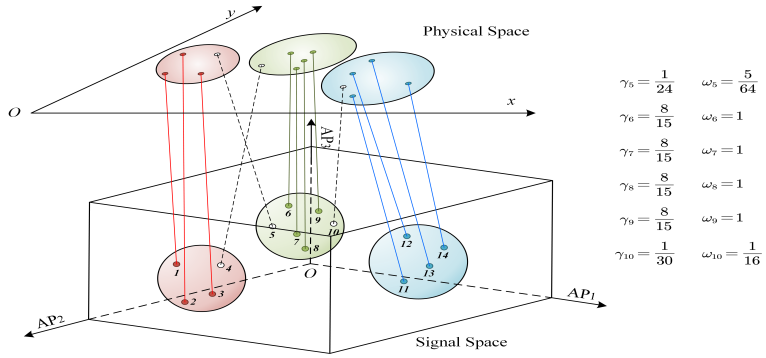

Figure 2. Illustration of the cross-domain cluster intersection algorithm: In the physical space, samples are clustered according to their annotated coordinates. In the signal space, samples are clustered according to the RSS distances. The weight of a sample is determined by the common samples between its belonged physical cluster and signal cluster.

3.2. Fitting Radio Surfaces

In one subarea, we construct a radio propagation surface for each hearable AP based on the weighted samples $(\boldsymbol{s}_i, w_i)$. A surface function $\phi(x, y)$ takes a location as its input and outputs an estimated RSS at this location. Notice that the number of crowdsourced samples could be large and keep increasing. To reduce computational complexity and alleviate surface overfitting, we propose a *percentile weight partition* (PWP) method to select only a subset of weighted samples.

Define p_{th} and ω_{th} as the percentile and weight threshold, respectively, and $p_{th}, \omega_{th} \in [0,1]$. The objective is to ensure that more than p_{th} samples have weights larger than ω_{th}. We first sort samples according to their weights in an increasing order, denoted by $\vec{\omega}$. Let ω_k denote the kth sample whose weight is at the p_{th} percentile of $\vec{\omega}$. If $\omega_k \geq \omega_{th}$, then no samples will be removed. Otherwise, we remove the first $\lceil \frac{M-k\omega_{th}}{1-\omega_{th}} \rceil$ samples from $\vec{w}$. After the sample selection, let $\mathcal{S}'$ denote the set of select samples and let $\mathcal{A}$ denote the set of hearable APs by samples in $\mathcal{S}'$.

In this paper, we adopt a polynomial function to fit a radio propagation surface for each AP in $\mathcal{A}$ as follows:

$$\phi_n(x,y) = \sum_{i=1}^{p}\sum_{j=1}^{q} a_{ij}x^{i-1}y^{j-1}, \text{ for all } n \in \mathcal{A}, \tag{3}$$

where a_{ij}s are fitting coefficients. The objective of weighted surface fitting is to

$$\text{minimize } H \equiv \sum_{i=1}^{|\mathcal{S}'|} \omega_i^2 \left(\phi_n\left(x_i, y_i\right) - r_{in}\right)^2 \tag{4}$$

To compute one fitting coefficient a_{er}, we equate its partial derivative to zero to minimize H.

$$\begin{aligned}
\frac{\partial H}{\partial a_{er}} &= \frac{\partial}{\partial a_{er}} \sum_{i=1}^{n} \omega_i^2 \left[\phi_n\left(x_i, y_i\right) - r_{in}\right]^2 \\
&= \sum_{i=1}^{n} \left\{ 2\omega_i^2 \left[\phi_n\left(x_i, y_i\right) - r_{in}\right] \frac{\partial}{\partial a_{er}} \left[\phi\left(x_i, y_i\right)\right] \right\} \\
&= \sum_{i=1}^{n} \left\{ 2\omega_i^2 \left[\phi_n\left(x_i, y_i\right) - r_{in}\right] x_i^{e-1} y_i^{r-1} \right\} \\
&= 0
\end{aligned} \tag{5}$$

From the equation above, we can derive

$$\sum_{i=1}^{n} 2\omega_i^2 x_i^{e-1} y_i^{r-1} \phi_n\left(x_i, y_i\right) = \sum_{i=1}^{n} 2\omega_i^2 x_i^{e-1} y_i^{r-1} r_{in} \tag{6}$$

$$\sum_{i=1}^{n} 2\omega_i^2 x_i^{e-1} y_i^{r-1} \sum_{c=1}^{p}\sum_{d=1}^{q} a_{cd} x_i^{c-1} y_i^{d-1} = \sum_{i=1}^{n} 2\omega_i^2 x_i^{e-1} y_i^{r-1} r_{in} \tag{7}$$

We define

$$u_{cd}\left(e,r\right) = \sum_{i=1}^{n} \left(2\omega_i^2 x_i^{c-1} y_i^{d-1} x_i^{e-1} y_i^{r-1} \right) \tag{8}$$

$$v\left(e,r\right) = \sum_{i=1}^{n} 2\omega_i^2 x_i^{e-1} y_i^{r-1} r_{in} \tag{9}$$

Thus, we can rewrite the equation as:

$$\sum_{c=1}^{p}\sum_{d=1}^{q} a_{cd} u_{cd}\left(e,r\right) = v\left(e,r\right),\ e = 1, \cdots, p,\ r = 1, \cdots, q \tag{10}$$

The matrix form of equation above is:

$$\begin{bmatrix} u_{11}\left(1,1\right) & \cdots & u_{pq}\left(1,1\right) \\ \vdots & \ddots & \vdots \\ u_{11}\left(p,q\right) & \cdots & u_{pq}\left(p,q\right) \end{bmatrix} \begin{bmatrix} a_{11} \\ \vdots \\ a_{pq} \end{bmatrix} = \begin{bmatrix} v\left(1,1\right) \\ \vdots \\ v\left(p,q\right) \end{bmatrix} \tag{11}$$

Then, by $\mathbf{A} = \mathbf{U}^{-1}\mathbf{V}$, the surface coefficient can be calculated.

3.3. Weighting Fitted Surfaces

Each AP surface is constructed based on its weighted samples. Different AP surfaces could contribute differently for describing the whole signal space. We next assign two weights to each AP surface via an entropy-like quantity computed from its samples: one is used for subarea determination and the other for location search in our online positioning.

For each AP in $\mathcal{A}$, let $\mathcal{R} = \{r_1, ..., r_R\}$ denote its set of RSS values extracted from the weighted samples in $\mathcal{S}'$. As the samples are assumed to be crowdsourced randomly from different locations, the set $\mathcal{R}$ is also expected to contain the RSS values from different locations. If all elements in $\mathcal{R}$ have similar values, then this AP might not be very helpful for discriminating different locations in one subarea. On the other hand, such an AP may be seen as a good indication of this subarea for its RSS consistency. Motivated by such considerations, we propose to weight AP surfaces for their different *subarea consistencies* and *location discriminations* from an entropy-like viewpoint.

We first normalize the elements in $\mathcal{R}$ by

$$\bar{r}_i = \frac{r_i - \min(\mathcal{R})}{\max(\mathcal{R}) - \min(\mathcal{R})}, \text{ for all } r_i \in \mathcal{R}. \tag{12}$$

We next compute an entropy-like quantity η for each AP in $\mathcal{A}$ to describe its RSS distribution property by

$$\eta = -\frac{\sum_{i=1}^{R} p_i \ln(p_i)}{\ln(R)}, \text{ where } p_i = \frac{\bar{r}_i}{\sum_{j=1}^{R} \bar{r}_j}. \tag{13}$$

For our two-step online positioning, we compute two surface weights for each AP:

$$\rho_n^{sub} = \frac{\eta_n}{\sum_{j=1}^{|\mathcal{A}|} \eta_j}, \ \rho_n^{loc} = \frac{1 - \eta_n}{\sum_{j=1}^{|\mathcal{A}|} (1 - \eta_j)}. \tag{14}$$

ρ_n^{sub} is used in the subarea determination, while ρ_n^{loc} is used in the location search in one subarea.

3.4. Constructing Subarea Fingerprints

For each subarea, we construct a *subarea fingerprint* $\vec{f}$ based on its weighted surfaces ϕ_n ($n \in \mathcal{A}$). We adopt a grid lattice approach to sample each surface ϕ_n uniformly in the physical space. Let $\mathcal{G}$ denote such a grid structure. For the gth grid, let $f_{gn} = \phi_n(g_x, g_y)$ denote a sampled grid RSS value from the nth surface, where (g_x, g_y) is the coordinate of the grid center. Then, $\vec{f}$ consists of subarea-averaged RSS values for all hearable APs

$$\vec{f} = \left(\frac{1}{|\mathcal{G}|} \sum_{g \in \mathcal{G}} f_{g1}, ..., \frac{1}{|\mathcal{G}|} \sum_{g \in \mathcal{G}} f_{gN'} \right), \tag{15}$$

where $N' = |\mathcal{A}|$ is the number of hearable APs in $\mathcal{A}$.

4. The Online Positioning Algorithm

The online positioning consists of two phases: subarea determination and location search.

Subarea Determination: Let $\vec{f}_t$ denote the RSS vector of a test sample, and $\vec{f}_s$ the sth subarea fingerprint. Let $\mathcal{A}_{int}$ denote the set of hearable APs by both $\vec{f}_t$ and $\vec{f}_s$. We compute the weighted signal distance between $\vec{f}_t$ and $\vec{f}_s$ as:

$$D_s = \frac{1}{|\mathcal{A}_{int}|} \sqrt{\sum_{n \in |\mathcal{A}_{int}|} (\rho_n^{sub} \times (f_{sn} - f_{tn}))^2}, \tag{16}$$

where f_{sn} and f_{tn} are the RSS values from the nth hearable AP in $\vec{f}_s$ and $\vec{f}_t$, respectively. The test sample is then localized into a subarea with the minimum D_s.

Location Search: Assume that the sth subarea is selected in the first phase. We next search a space point $(\hat{x}, \hat{y})$ in this subarea to minimize the weighted signal difference between $\vec{f}_t$ and subarea surfaces:

$$(\hat{x}, \hat{y}) = \arg\min_{(x,y)} \sum_{n \in \mathcal{A}_{int}} [\rho_n^{loc}(\phi_n(x,y) - f_{tn})]^2 \tag{17}$$

In this paper, we use the gradient descent search method. Instead of randomly choosing a start point, we use the localization result of a simple *nearest neighbor* (NN) algorithm as the initial searching point, where the grid fingerprints are spatially sampled from the fitted surfaces. We then calculate weighted signal difference as the cost function and its partial derivation to determine the search direction. The cost function is defined as

$$J(l_t) = \sum_{n \in \mathcal{A}_{int}} \left[\rho_n^{loc}(\phi_n(x,y) - f_{tn})\right]^2 \tag{18}$$

The search iteration is defined by

$$l_{t+1} = l_t + \alpha_d d_t, \text{ where } d_t = -\nabla J(l_t) \tag{19}$$

$$\nabla J(l_t) = \left[\frac{\partial J(l_t)}{\partial x}, \frac{\partial J(l_t)}{\partial y}\right]^T, \tag{20}$$

where α_d is the search step. We substitute Equation (3) into Equation (18):

$$J(l_t) = \sum_{n \in \mathcal{A}_{int}} \left[\rho_n^{loc}\left(\sum_{i=1}^{p}\sum_{j=1}^{q} a_{ij} x^{i-1} y^{j-1} - f_{tn}\right)\right]^2 \tag{21}$$

Next, we compute the partial derivation of this cost function to gain the gradient and update the search iteration.

$$\begin{aligned} \frac{\partial J(l_t)}{\partial x} = {} & \sum_{n \in \mathcal{A}_{int}} 2(\rho_n^{loc})^2 \left[\phi_n(x,y) - f_{tn}^0\right] \sum_{i=1}^{p}\sum_{j=1}^{q} a_{ij}(i-1) \\ & x^{j-2} y^{j-1} \end{aligned} \tag{22}$$

$$\begin{aligned} \frac{\partial J(l_t)}{\partial y} = {} & \sum_{n \in \mathcal{A}_{int}} 2(\rho_n^{loc})^2 \left[\phi_n(x,y) - f_{tn}^0\right] \sum_{i=1}^{p}\sum_{j=1}^{q} a_{ij} x^{i-1} \\ & (j-1) y^{j-2} \end{aligned} \tag{23}$$

The gradient search will stop when the d_t is too small to update the search position for the next iteration.

5. Field Measurements and Experiments

5.1. Experiment Settings

Figure 3 plots the indoor layout of our field measurements in a typical lecture building with total area of 482 m^2. In our work, we did not place our own APs. Instead, we employed the existing Wi-Fi infrastructure with APs deployed by different parties, such as individual laboratories, telecom operators and campus authorities. Indeed, the total number of hearable AP in our experimental environment was more than 400, while, for each sample, normally >70 APs could be heard. We note that emplying the existing Wi-Fi infrastructure makes our proposed scheme ready to be implemented in many practical scenarios.

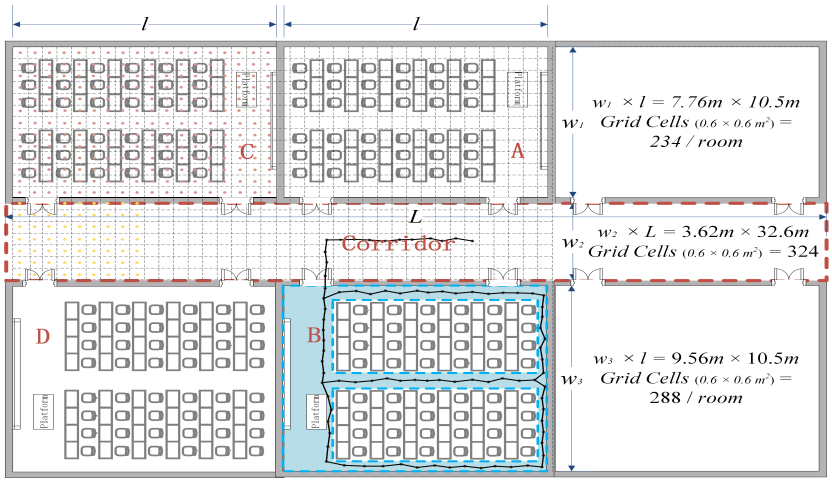

Figure 3. The layout of the indoor environment. A grid lattice has been used to collect samples, with in total 1368 grid cells each with size 0.6×0.6 m^2. Besides, pedestrian trajectories have also been used to collect samples for the corridor and walkable pathways in each room.

A Huawei Honor 3C smartphone was used to collect RSS samples. We conducted two batches of sample collection: The first batch $\mathcal{S}_{site}$ was based on the site survey approach, containing in total 13,670 samples each collected at one grid center. The second batch $\mathcal{S}_{walk}$, containing in total 13,370 samples, was extracted from movement trajectories restricted to only those walkable routes, as illustrated by the colored area in one room in Figure 3. Note that the samples in both $\mathcal{S}_{site}$ and $\mathcal{S}_{walk}$ are firstly annotated true location information at collection. To emulate annotation errors, we again annotate each sample into a new location with a *location offset* randomly drawn from a Gaussian distribution with zero mean and σ standard deviation. The test set $\mathcal{S}_{test}$ contains 5600 samples uniformly distributed in the whole environment.

Experiment Schemes: According to their annotated locations, crowdsourced samples can be assigned into different grids to construct grid fingerprints. Similarly, they can also be grouped into different clusters in the signal space to obtain cluster fingerprints. We tested the following peer localization schemes to examine these typical approaches.

- FGrid emulates the traditional site-survey fingerprinting based on grid fingerprints, which divides the subarea into several non-overlapping grid cell to contain samples, and assigns each new sample into its nearest grid cell. For each grid cell, a *grid fingerprint* is composed by averaging all samples located within the grid cell, and the location of the grid fingerprint is annotated as the grid center. In the online phase, we used the *nearest neighbor* algorithm.
- SGrid is similar to the FGrid to obtain grid fingerprints. We then constructed surfaces based on these fingerprints in the offline phase. In the online phase, we used the same surface search method as the one in our proposed SWSample.
- SRaw retains the original position of every crowdsourced sample and fits propagation surfaces based on them. In the online phase, we used the same surface search method as the one in our proposed SWSample.
- SCluster clusters the samples in signal domain only. For each cluster, we obtained a *cluster fingerprint*, which is the average of its cluster members' RSS vectors. The location of a cluster fingerprint is the geometric center of the cluster members. We fitted the propagation surfaces for every AP based on these cluster fingerprints. In the online phase, we used the surface search method the same as the one in our proposed SWSample.
- SWSample is the proposed scheme.

In all the above schemes, we set the cluster number equal to the number of grids used in *FGrid* for a fair comparison. We also adopted the proposed two-step online positioning algorithm. We noticed that, from our experiments, the *subarea hitting rate* of all these schemes is not smaller than 99.58%,

i.e., almost all test samples can be correctly determined to its belonging subarea. Thus, we do not report this result again in the following.

5.2. Surface Fitting Examples

Figures 4–7 plot the fitted surfaces for the four surfacing-based schemes. We chose one AP for Room A and fit its surface from 1800 samples randomly drawn from $\mathcal{S}_{site} \bigcup \mathcal{S}_{walk}$. For the SWSample fitted surface in Figure 7, we also color the sample weight as shown by the weight color spectrum alongside the graph. It can be seen that the proposed scheme could produce a smoother surface, compared with other schemes. If we assume that this AP is located at the coordinate around the highest RSS value, then we could observe that the surface in Figure 7 is more like an attenuated sphere centered at the AP. The Keenan–Motley path loss model has been widely adopted to characterize the radio propagation in mobile cellular networks. If such a model could still be applicable in a small and open space such as a room, then our fitted surface resembles the most to this model, which might also help to explain the effectiveness of our weighted surface fitting.

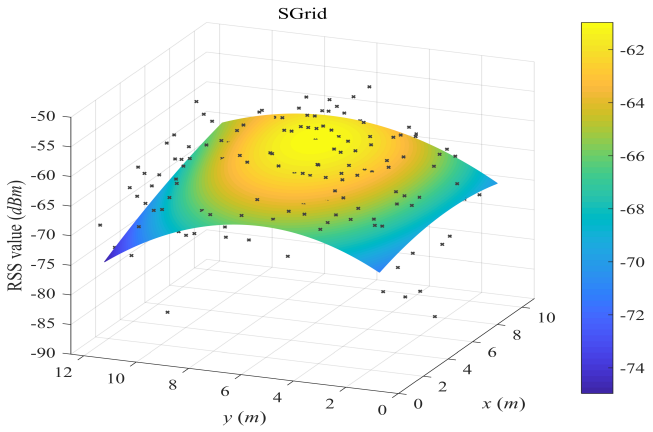

Figure 4. Illustration of fitted surface by SGrid. We choose one AP for Room A and fit its surface from 1800 samples randomly drawn from $\mathcal{S}_{site} \bigcup \mathcal{S}_{walk}$. Crowdsourced samples are assigned to grid cells. A grid fingerprint is composed by averaging all samples in the grid cell, and its location is the grid center. The fitted surface is based on the grid fingerprints.

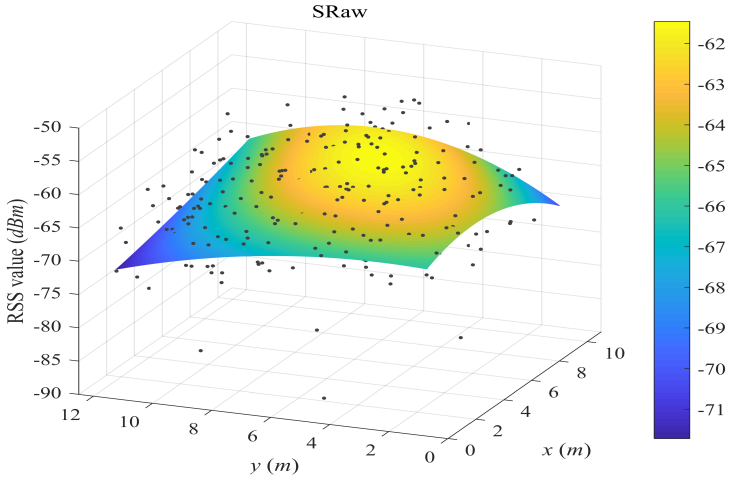

Figure 5. Illustration of fitted surface by SRaw. We choose one AP for Room A and fit its surface from 1800 samples randomly drawn from $\mathcal{S}_{site} \bigcup \mathcal{S}_{walk}$. All crowdsourced samples are used for surface fitting, without sample weighting and selection.

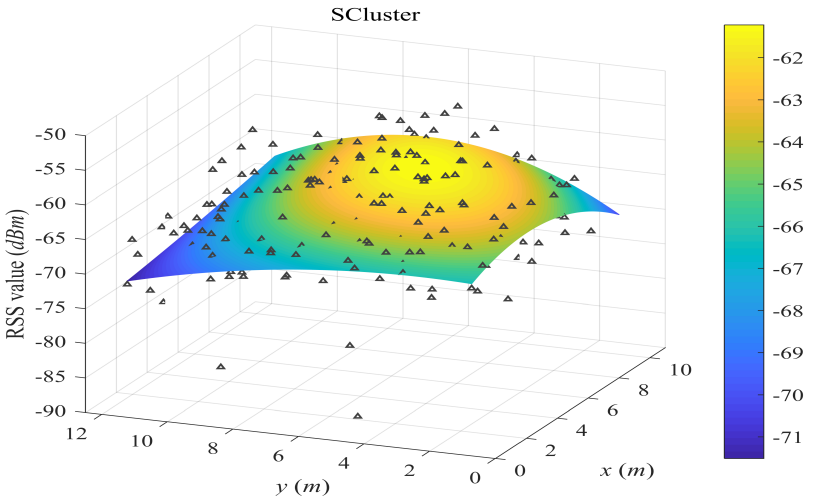

Figure 6. Illustration of fitted surface by SCluster. We choose one AP for Room A and fit its surface from 1800 samples randomly drawn from $\mathcal{S}_{site} \bigcup \mathcal{S}_{walk}$. All crowdsourced samples are first clustered in the signal space. For each cluster, a cluster fingerprint is composed by averaging the RSS vectors of its cluster members, and its location is the geometric center of the cluster members. The fitted surface is based on the cluster fingerprints.

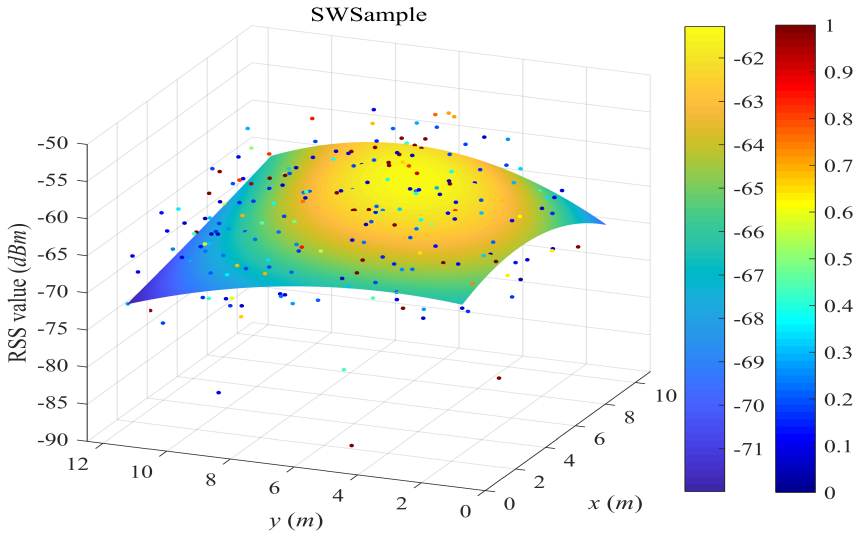

Figure 7. Illustration of fitted surface by our proposed SWSample. We choose one AP for Room A and fit its surface from 1800 samples randomly drawn from $\mathcal{S}_{site} \bigcup \mathcal{S}_{walk}$. Crowdsourced samples are weighted and selected for surface construction. The sample weight is illustrated by the dot color in the figure.

5.3. Experiment Results

Uniformly distributed samples: We first considered the scenario that all crowdsourced samples are uniformly distributed in the experiment environment, that is, we used crowdsourced samples from $\mathcal{S}_{site} \bigcup \mathcal{S}_{walk}$. Figure 8 plots the *average localization error* (ALE) against the number of crowdsourced samples randomly drawn from $\mathcal{S}_{site} \bigcup \mathcal{S}_{walk}$. It was first observed that all the surfacing schemes outperform the grid fingerprinting FGrid, which validates the effectiveness of using fitted radio propagation surfaces for localization. When the number of samples increases, from about $0.33M_g$ to $20M_g$ with M_g the number of total grid cells, the ALE of the surfacing schemes first decreases and then increases. At first, the number of samples is not large enough to well fit actual surfaces. In this case, our scheme SWSample has a slightly higher ALE than other surfacing schemes (see the first two

points in Figure 8) due to its sample selection. On the other hand, if the noisy samples are too many, the surfaces may be overfitted for unreliable samples. However, ours presents a decent degradation and the ALE of using all 27,040 samples is 1.54 m, slightly higher than the best case of 1.45 m of using 3605 samples. The positioning accuracy improvement of our scheme are 36.71% over FGrid and 9.41% over SRaw, respectively. Compared with the SRaw scheme, the improvement can be attributed to our sample weighting and selection algorithm, which only chooses those reliable samples for weighted surface fitting, leading to a more accurate radio map and better positioning results.

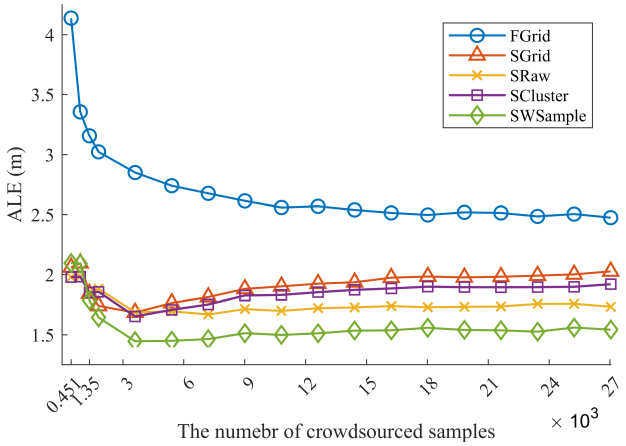

Figure 8. Comparison of localization performance. The average localization error (ALE) vs. the number of crowdsourced samples M_{all}, when using crowdsourced samples from $\mathcal{S}_{site} \bigcup \mathcal{S}_{walk}$. The standard deviation of location offset $\sigma = 1.2$ m.

Figure 9 presents the ALE against the standard deviation σ of location offset. Notice that $\sigma = 0$ indicating no annotation errors. It is not unexpected to see that all schemes suffer from the increasing of σ, i.e., the annotated locations farther away from true locations. However, our scheme SWSample still performs the best. Figure 10 plots the *cumulative distribution function* (CDF) of localization error. It is worth noting that, besides a low median localization error of 1.51 m, our SWSample has a low 90% percentile error of only 2.64 m. To provide the exact numbers, Table 2 summarizes the localization error results for three situations, namely, $\sigma = 0.6$ m, $\sigma = 0.9$ m, and $\sigma = 1.2$ m, respectively.

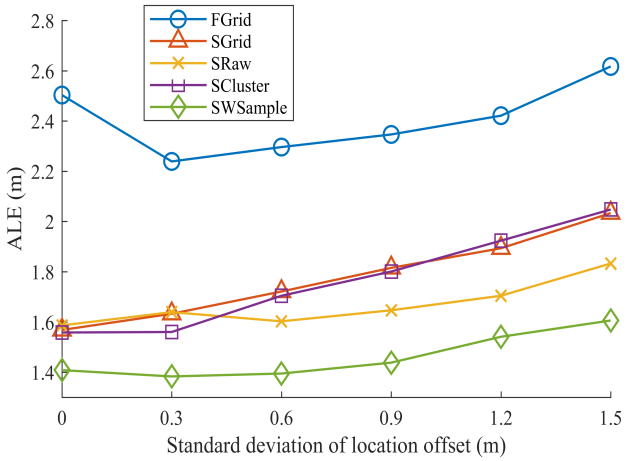

Figure 9. Comparison of localization performance. The average localization error (ALE) vs. the standard deviation σ of location offset, where $M_{all} = 27,040$.

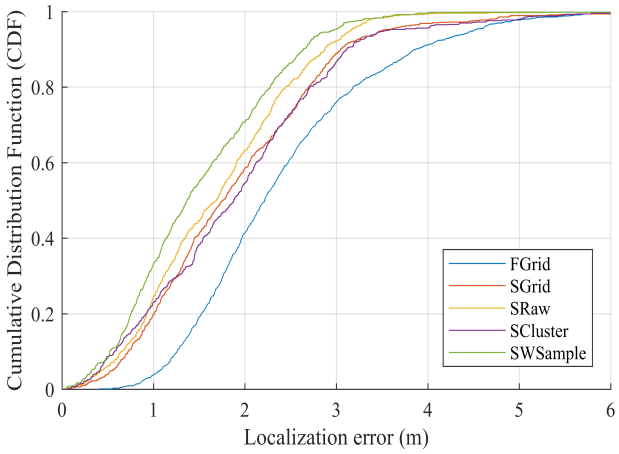

Figure 10. Comparison of cumulative distribution function (CDF) localization error, where M_{all} = 27,040 and σ = 1.2 m.

Table 2. Comparison of mean, 50% and 90% localization error.

Error (m)		σ = 0 m			σ = 0.6 m			σ = 1.2 m		
		Mean	**50%**	**90%**	**Mean**	**50%**	**90%**	**Mean**	**50%**	**90%**
Uni.	FGrid	2.479	2.448	3.672	2.284	2.086	3.744	2.421	2.217	3.868
	SGrid	1.571	1.353	2.595	1.726	1.630	2.884	1.898	1.757	3.048
	SRaw	1.575	1.370	2.645	1.618	1.524	2.694	1.711	1.688	2.873
	SCluster	1.552	1.364	2.550	1.708	1.657	2.875	1.916	1.879	3.111
	SWSample	1.373	1.124	2.413	1.374	1.243	2.470	1.513	1.366	2.640
Non-uni.	FGrid	2.897	2.776	3.672	2.982	2.813	4.477	3.059	2.932	4.502
	SGrid	2.164	1.691	3.522	2.086	1.679	3.402	2.169	1.795	3.499
	SRaw	2.155	1.713	3.459	2.221	1.732	3.594	2.322	1.898	3.647
	SCluster	2.063	1.602	3.497	2.009	1.584	3.287	2.144	1.752	3.477
	SWSample	1.854	1.497	3.172	1.951	1.472	3.217	2.043	1.625	3.242

Non-uniformly distributed samples: It is also often the case that crowdsourced samples are not uniformly distributed in the whole environment. To examine this *nonuniform density* issue, we only use the samples from $\mathcal{S}_{walk}$ to fit surfaces. That is, the subregion of chairs and desks in each room do not contain crowdsourced samples. However, as we intentionally include location annotation errors, some samples may still be annotated to locations within such a vacant subregion. As shown in Figures 11 and 12, it is not unexpected to observe that all schemes suffer from such a nonuniform density situation, comparing with the results in Figure 8. However, our SWSample scheme can still outperform other schemes in most of cases. The positioning accuracy improvements are 36.85% over FGrid and 18.79% over SRaw, respectively. Furthermore, the median and 90% localization errors in Figure 13 are 2.04 m and 3.24 m, respectively, which are comparable to the uniform density case. Table 2 summarizes the localization error results from three situations for non-uniformly distributed samples. It can be observed that our proposed scheme has great potential to obtain a better result in this non-uniformly distributed case, which illustrates its robustness for tackling the nonuniform density challenge.

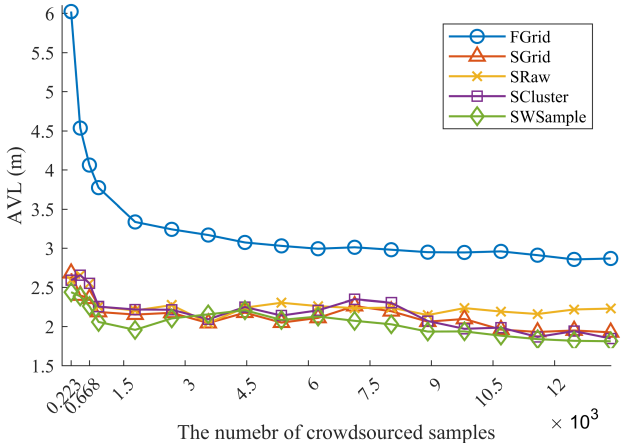

Figure 11. Comparison of localization performance, when using crowdourced samples only from $\mathcal{S}_{walk}$. The average localization error (ALE) vs. the number of crowdsourced samples M_{all}, where $\sigma = 1.2$ m.

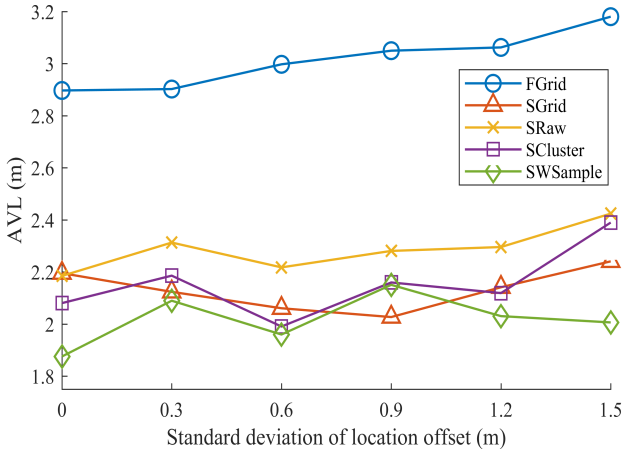

Figure 12. Comparison of localization performance, when using crowdsourced samples only from $\mathcal{S}_{walk}$. The average localization error (ALE) vs. the standard deviation σ of location offset, where $M_{all} = 4456$.

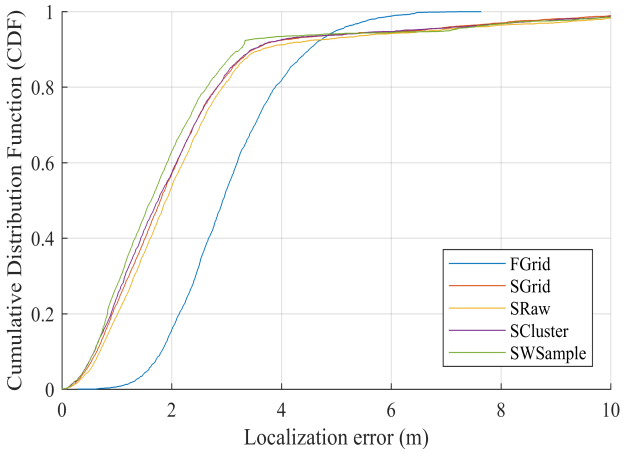

Figure 13. Comparison of cumulative distribution function (CDF) localization error with $M_{all} = 4456$ and $\sigma = 1.2$ m, when using crowdourced samples only from $\mathcal{S}_{walk}$.

6. Concluding Remarks

This paper has studied the problem of constructing radio propagation surfaces from unreliable crowdsourced samples with annotation errors. We have proposed a cross-domain cluster intersection to weight each sample reliability and an entropy-like approach to further weight the constructed surfaces. Field experiments have validated its effectiveness and robustness for dealing with the nonuniform density challenge. Our proposed method contributes to indoor localization society in its high accuracy and easy implementation.

We close this paper with some discussions about future work. This paper has applied polynomial functions for fitting radio propagation surfaces in the offline phase. Indeed, the propagation surfaces may take different forms and there could exist many other primary functions or stochastic kernels for surface fitting. How to intelligently choose the most suitable primary functions or stochastic kernels and automatically adjust their fitting parameters are worthy of further research. In this paper, we have applied the commonly used deterministic positioning algorithm in the online phase. Using some probabilistic positioning algorithms, especially when the radio propagation surfaces are modelled as stochastic processes, is also worthy of further investigation.

Author Contributions: B.W. provided problem conceptualization and revised the paper; B.W. and J.L. investigated the problem solutions and proposed the algorithm; J.L. implemented the algorithm and drafted the paper; J.L. and G.Y. experimented the algorithm and analyzed the results; M.Z. reviewed and revised the paper.

Funding: This work is partly supported by the National Natural Science Foundation of China with grant number 61771209 and the Fundamental Research Funds for the Central Universities with grant number 2018KFYYXJJ136.

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

References

1. Liu, H.; Darabi, H.; Banerjee, P.; Liu, J. Survey of wireless indoor positioning techniques and systems. *IEEE Trans. Syst. Man Cybern. C* **2007**, *37*, 1067–1080. [CrossRef]
2. Yassin, A.; Nasser, Y.; Awad, M.; Al-Dubai, A.; Liu, R.; Yuen, C.; Raulefs, R.; Aboutanios, E. Recent advances in indoor localization: A survey on theoretical approaches and applications. *IEEE Commun. Surv. Tutor.* **2016**, *19*, 1327–1346. [CrossRef]
3. He, S.; Chan, S.-H.G. Wi-Fi fingerprint-based indoor positioning: Recent advances and comparisons. *IEEE Commun. Surv. Tutor.* **2016**, *18*, 466–490. [CrossRef]
4. Wang, B.; Zhou, S.; Liu, W.; Mo, Y. Indoor localization based on curve fitting and location search using received signal strength. *IEEE Trans. Ind. Electron.* **2015**, *62*, 572–582. [CrossRef]
5. Bahl, P.; Padmanabhan, V.N. Radar: An in-building RF-based user location and tracking system. In Proceedings of the IEEE INFOCOM 2000. Conference on Computer Communications. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (Cat. No.00CH37064), Tel Aviv, Israel, 26–30 March 2000; Volume 2, pp. 775–784.
6. Hossain, A.; Soh, W.-S. A survey of calibration-free indoor positioning systems. *Comput. Commun.* **2015**, *66*, 1–13. [CrossRef]
7. Wang, B.; Chen, Q.; Yang, L.T.; Chao, H.-C. Indoor smartphone localization via fingerprint crowdsourcing: Challenges and approaches. *IEEE Wirel. Commun.* **2016**, *23*, 82–89. [CrossRef]
8. Zhou, X.; Chen, T.; Guo, D.; Teng, X.; Yuan, B. From one to crowd: A survey on crowdsourcing-based wireless indoor localization. *Front. Comput. Sci.* **2018**, *12*, 423–450. [CrossRef]
9. He, S.; Ji, B.; Chan, S.-H.G. Chameleon: Survey-free updating of a fingerprint database for indoor localization. *IEEE Pervasive Comput.* **2016**, *15*, 66–75. [CrossRef]
10. Abdelnasser, H.; Mohamed, R.; Elgohary, A.; Alzantot, M.F.; Wang, H.; Sen, S.; Choudhury, R.R.; Youssef, M. Semanticslam: Using environment landmarks for unsupervised indoor localization. *IEEE Trans. Mob. Comput.* **2016**, *15*, 1770–1782. [CrossRef]
11. Zhou, M.; Zhang, Q.; Wang, Y.; Tian, Z. Hotspot ranking based indoor mapping and mobility analysis using crowdsourced Wi-Fi signal. *IEEE Access* **2017**, *5*, 3594–3602. [CrossRef]

12. Wu, C.; Yang, Z.; Xiao, C. Automatic radio map adaptation for indoor localization using smartphones. *IEEE Trans. Mob. Comput.* **2018**, *17*, 517–528. [CrossRef]
13. Chen, Q.; Wang, B. Finccm: Fingerprint crowdsourcing, clustering and matching for indoor subarea localization. *IEEE Wirel. Commun. Lett.* **2015**, *4*, 677–680. [CrossRef]
14. Liu, X.; Zhan, Y.; Cen, J. An energy-efficient crowd-sourcing-based indoor automatic localization system. *IEEE Sens. J.* **2018**, *18*, 6009–6022. [CrossRef]
15. Chang, Q.; Li, Q.; Shi, Z.; Chen, W.; Wang, W. Scalable indoor localization via mobile crowdsourcing and gaussian process. *Sensors* **2016**, *16*, 381. [CrossRef] [PubMed]
16. Jung, S.; Moon, B.; Han, D. Unsupervised learning for crowdsourced indoor localization in wireless networks. *IEEE Trans. Mob. Comput.* **2016**, *15*, 2892–2906. [CrossRef]
17. Zhou, M.; Tang, Y.; Tian, Z.; Geng, X. Semi-supervised learning for indoor hybrid fingerprint database calibration with low effort. *IEEE Access* **2017**, *5*, 4388–4400. [CrossRef]
18. Jung, S.; Han, H. Automated construction and maintenance of Wi-Fi radio maps for crowdsourcing-based indoor positioning systems. *IEEE Access* **2017**, *6*, 1764–1777. [CrossRef]
19. Kim, Y.; Shin, H.; Chon, Y.; Cha, H. Crowdsensing-based Wi-Fi radio map management using a lightweight site survey. *Comput. Commun.* **2015**, *60*, 86–96. [CrossRef]
20. Huang, Z.; Xia, J.; Yu, H.; Guan, Y.; Gan, X.; Liu, J. Fusing fixed and hint landmarks on crowd paths for automatically constructing Wi-Fi fingerprint database. *China Commun.* **2015**, *12*, 11–24. [CrossRef]
21. Zhou, B.; Li, Q.; Mao, Q.; Tu, W.; Zhang, X.; Chen, L. Alimc: Activity landmark-based indoor mapping via crowdsourcing. *IEEE Trans. Intell. Transp. Syst.* **2015**, *16*, 2774–2785. [CrossRef]
22. Yu, N.; Xiao, C.; Wu, Y.; Feng, R. A radio-map automatic construction algorithm based on crowdsourcing. *Sensors* **2016**, *16*, 504. [CrossRef] [PubMed]
23. Zhou, B.; Li, Q.; Mao, Q.; Tu, W. A robust crowdsourcing-based indoor localization system. *Sensors* **2017**, *17*, 864. [CrossRef] [PubMed]
24. Li, W.; Wei, D.; Lai, Q.; Li, X.; Yuan, H. Geomagnetism-aided indoor Wi-Fi radio-map construction via smartphone crowdsourcing. *Sensors* **2018**, *18*, 1462. [CrossRef] [PubMed]
25. Zhou, M.; Wang, Y.; Tian, Z.; Zhang, Q. Indoor pedestrian motion detection via spatial clustering and mapping. *IEEE Sens. Lett.* **2018**, *2*, 1–4. [CrossRef]
26. Wang, B.; Zhou, S.; Yang, L.T.; Mo, Y. Indoor positioning via subarea fingerprinting and surface fitting with received signal strength. *Pervasive Mob. Comput.* **2015**, *23*, 43–58. [CrossRef]
27. Ye, Y.; Wang, B. RMapCS: Radio map construction from crowdsourced samples for indoor localization. *IEEE Access* **2018**, *6*, 24224–24238. [CrossRef]