Contour-based Trilateration for Indoor Fingerprinting Localization

Suining He Tianyang Hu S.-H. Gary Chan
Department of Computer Science and Engineering,
The Hong Kong University of Science and Technology, Hong Kong, China
{sheaa, thuab, gchan}@cse.ust.hk

ABSTRACT

Trilateration has been widely and successfully employed to locate outdoor mobile devices due to its accuracy. However, it cannot be directly applied for indoor localization due to issues such as non-line-of-sight measurement and multipath fading. Though fingerprinting overcomes these issues, its accuracy is often hampered by signal noise and the choice of similarity metric between signal vectors. We propose ***INTRI***, a novel, simple and effective ***in***door localization technique combining the strengths of ***tri***lateration and fingerprinting.

For a signal level received from an access point (AP) by the target, INTRI first forms a contour consisting of all the reference points (RPs) of the same signal level for that AP, taking into account the signal noise. The target is hence at the juncture of all the contours. With an optimization formulation following the spirit of trilateration, it then finds the target location by minimizing the distance between the position and all the contours. INTRI further uses an online algorithm based on signal correlation to efficiently calibrate heterogeneous mobile devices to achieve higher accuracy. We have implemented INTRI, and our extensive simulation and experiments in an international airport, a shopping mall and our university campus show that it outperforms recent schemes with much lower location error (often by more than 20%).

Categories and Subject Descriptors

C.2.1 [**Network Architecture and Design**]: Wireless Communication

General Terms

Design, Experimentation, System, Performance

Keywords

Indoor localization; contour-based trilateration; fingerprinting; device calibration; signal contour; linear programming.

1. INTRODUCTION

In traditional trilateration, a mobile device (target) first measures its distances to a number of landmarks of known location. It then estimates its position which best matches these measured distances (e.g., given by minimizing the error between the measured distances and the distances from the position to the landmarks). Such a localization technique has been widely used outdoors, with exemplary applications in GPS (Global Positioning System) and cellular positioning, where the landmarks are satellites and base stations (cell towers), respectively.

Despite of its accuracy and successful outdoor deployment, trilateration does not work well indoors because distances to landmarks cannot be estimated accurately. Such inaccuracy is mainly due to non-line-of-sight landmarks, complex indoor signal fading (due to multipath), over-simplification or parametric uncertainties in indoor propagation models, etc. Fingerprinting, on the other hand, emerges as a promising approach for indoor localization. An example is Wi-Fi fingerprinting, which is gaining popularity due to its ease of deployment without the need to install extra sensor infrastructure beyond the existing Wi-Fi network [11].

Fingerprint-based localization is usually conducted in two phases. In the first offline (survey) phase, a site survey is conducted to collect the vectors of *received signal strength indicators* (RSSIs) at known locations, the so-called "reference points" (RPs). These vectors of RSSIs are the *fingerprints* of the locations and are stored in a database. In the second online (query) phase, a user samples or measures an RSSI vector at his own position and reports it to the server (in this paper, we use "user", "mobile device" and "target" interchangeably). The server then locates the indoor user by comparing the target vector with the fingerprints using some similarity metric (such as Euclidean distance [2] between signal vectors). The target position is then estimated out of the most similar "neighbors," the set of RPs whose fingerprints closely resemble the target's RSSI.

Traditionally, the similarity metric used often treats the RSSI vector of the target as a single "entity" in comparison. This makes the neighbor selection susceptible to statistical fluctuation in signal strength and measurement noise. Due to random signals, it has been widely observed that the matching algorithm in the online phase may result in a dispersed set of neighbors (i.e., RPs which are quite distant apart in the physical space). This leads to unsatisfactory localization accuracy [11,30].

In this paper, we propose **INTRI** (**In**door **Tri**lateration using signal contours), a novel, efficient and accurate indoor

SenSys'15, November 1–4, 2015, Seoul, South Korea..
© 2015 ACM. ISBN 978-1-4503-3631-4/15/11 ...$15.00.
DOI: http://dx.doi.org/10.1145/2809695.2809703.

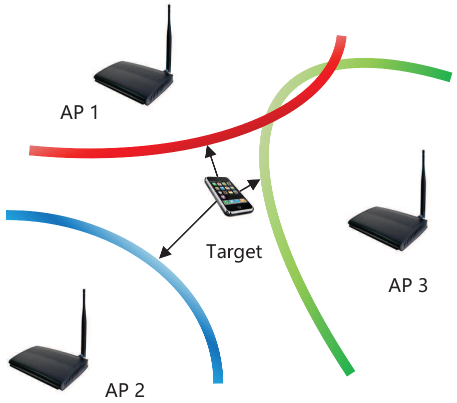


Figure 1: The basic idea of INTRI, indoor contour-based trilateration by minimizing the distance to signal contours. The contours are based on the fingerprints.

localization technique by employing the concept of trilateration in fingerprint-based environment. By treating the RSSI from each AP *individually* (instead of as a single signal vector), INTRI does not suffer from the problem of dispersed set of neighbors. It is highly robust to random signals (due to signal noise and measurement uncertainty) as it is not based on fingerprint similarity comparison.

We illustrate the basic concept of INTRI in Figure 1 with three access points (APs). Let S be the signal strength from a certain AP as measured by the target. For that AP, we can form a *contour* for S in the fingerprint region, which are the spatially distributed RPs whose signal level of the AP is S, subject to its statistical fluctuation (the figure shows a continuous contour, though in reality the signal contour consists of discrete points in space). It is clear that the target is somewhere on the contour line. Given the target's received signal vector, we can hence form the corresponding contour for each of the APs (three contours for the three APs in the figure). Following the spirit of trilateration, the target position can then be estimated by minimizing its distances to the signal contours.

INTRI integrates the highly accurate contour-based trilateration technique with indoor fingerprinting. It combines the strengths of both approaches: it does not need the positions of APs and line-of-sight (LoS) measurement, and locates the target at the junction of its measured signal levels without the dispersion problem. Though at times for concreteness our discussion in this paper is on Wi-Fi RSSI fingerprinting, INTRI is a general approach applicable to any other fingerprint signal like channel state information (CSI) [39], ZigBee [37], visible light [40] or RFID [5, 10]. It may also be a complementary module to existing sensor localization systems such as [21, 26] without modifying or adding any specialized infrastructures.

In order for INTRI to be practically deployed, we need to address the following important issues:

- *Forming contours for random signals:* Signal measurement is inherently noisy. Constructing contours needs to consider random fluctuation in order to effectively locate the target. We present how to statistically analyze the fingerprints and target RSSIs received so as to construct contours under random signals. As the target is likely to be at the region where contours meet, we discuss how to find such region.

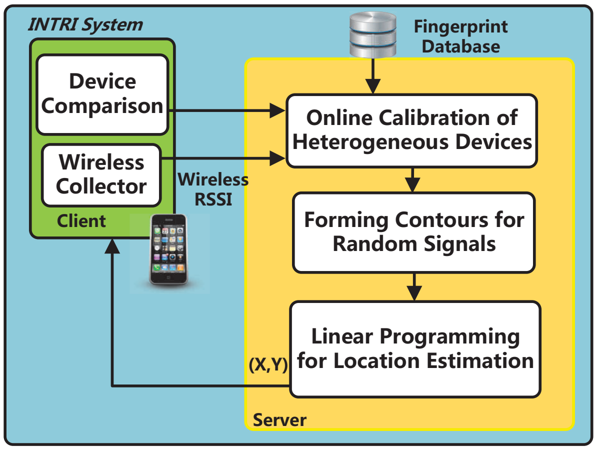

INTRI System
Fingerprint Database
Device Comparison
Wireless Collector
Client
Wireless RSSI
Online Calibration of Heterogeneous Devices
Forming Contours for Random Signals
Linear Programming for Location Estimation
(X,Y)
Server

Figure 2: System framework of INTRI.

- *Efficient contour-based localization algorithm:* We propose a novel contour-based localization algorithm based on linear programming formulation. Following the spirits of trilateration, it estimates target location with the objective to minimize the distances to the contours obtained above.

- *Adaptive online calibration for heterogeneous devices:* For the same signal, different devices may have different readings. Device readings hence need to be mapped, or "calibrated," to the corresponding signal level stored in the database so that the contours can be correctly discovered. Offline calibration for all different devices is neither efficient nor scalable. We propose an efficient and practical online scheme to calibrate devices, which adapts the target measured RSSI according to the stored fingerprints based on signal correlation. Using our efficient approach, INTRI achieves high scalability in heterogeneous devices, and robustness under noisy environment.

We show in Figure 2 the system framework of INTRI. The Wi-Fi fingerprint database is initialized by a site survey, storing <location, RSSI vector> pairs for each RP and vendor information of the devices used for data collection. The system is now ready for online estimation.

In the online phase, INTRI first checks the vendor information of the user's device. If that is different from the devices used for site survey, the target RSSI vector will be calibrated using the stored fingerprints. The calibrated RSSI vector is then used to construct the signal contours. Given signal contours, INTRI formulates a linear programming to jointly minimize the distances to the contours and estimates the user's position. The location is finally returned to the user's device.

We have conducted extensive simulation and large-scale experimental trials in the Hong Kong International Airport (HKIA), the Hong Kong Cyberport mall (HKCP) and our university campus (HKUST). Both simulation and experimental studies further confirm the high accuracy of INTRI.

The rest of this paper is organized as follows. After reviewing the related works in Section 2, we describe in Section 3 the construction of contours for noisy signals. Based on these contours, in Section 4 we present the linear programming formulation for localization based on the idea of trilateration. In Section 5, we propose an efficient online approach to calibrate different devices. Illustrative simulation and experimental results are presented in Sections 6 and 7, respectively. We finally conclude in Section 8.

2. RELATED WORK

We briefly discuss related works in this section. Pattern recognition techniques have been widely studied in Wi-Fi fingerprinting localization. RADAR [2] and Horus [42] are the two representative approaches. Recently more advanced techniques on pattern matching have been investigated [11–13, 22, 29]. Signal propagation model is also considered recently to derive RSSI at different locations [6, 14, 23]. EZ [6] utilizes rigid matching between signals and distances to determine the target location. More recent works like EZPerfect [31] and Modellet [23] further utilize the labeled fingerprints to derive signal propagation model and achieve higher accuracy.

In contrast to the above works, we combine the advantages of fingerprinting and trilateration approaches. We employ a geometric scheme (with spirits of trilateration) based on random fingerprint signals to constrain the target region. By formulating a novel linear programming, INTRI achieves much better localization accuracy without neighbor dispersion. INTRI is also compatible to emerging fingerprint update schemes like [4] (crowdsourcing), [5] (sensor-assisted) and [34] (crowdsourcing and model-based update) to achieve higher robustness in practical deployment.

Some other recent works leverage the temporal or spatial RSSI patterns for localization. These works consider location-dependent patterns such as the trend of RSSI sequences along corridors [33], order of RSSIs from different APs [18], or the unique existence of some Wi-Fi APs at some area [38] [20]. Once the target measures such patterns, its location is then mapped to the area. These patterns achieve promising results for constrained and narrow environment with well-defined user trajectories (like corridors or offices). To the contrary, the contours in INTRI are solely derived from fingerprints and are applicable to any indoor environment. Furthermore, INTRI does not need the positions of APs and LoS measurement, and is not based on indoor models (indoor environment may be too complex to model).

Calibrating different devices has been studied in recent works [7, 25, 28, 32]. Traditional offline calibration [32] causes extra manual efforts in real deployment and hence is not scalable. Given the target RSSI measurement, works like [7, 25, 28] utilize the deduction [25] (or ratio [28]) between AP signal values to calibrate the devices. However, large noise and fluctuation in signal levels can degrade the quality of the above calibration. Some learning-based approaches [9, 23] utilize expectation maximization to calibrate the signal difference. Different from these works, INTRI proposes a more efficient and robust scheme which maps the target signals to the signal space in fingerprint database.

To improve fingerprint localization, sensor fusion has attracted intensive attention recently. Using the smartphone inertial sensors, fusing motion information has been studied extensively to improve fingerprint-based positioning [15, 17, 36]. Peer-assisted localization has also been recently studied, making use of Wi-Fi Direct [19] and high-pitch sound [24] to measure the distance between peer devices [16] as extra position constraints. Different from above fusion approaches, INTRI solely relies on existing Wi-Fi measurement, achieving higher scalability in pervasive deployment. INTRI is also independent of these works, and is amendable to integrate with them for higher accuracy of mobile localization.

Table 1: Major symbols used in INTRI formulation.

Notation	Definition
N	Number of RPs in fingerprint database.
L	Number of APs in the site.
$\mathbf{f}_n$	Wi-Fi RSSI vector received at RP n.
f_n^l	Wi-Fi AP l's RSSI at RP n (dBm).
$\mathbf{g}$	Wi-Fi RSSI vector received at target.
g^l	Wi-Fi AP l's RSSI at target (dBm).
σ_n^l	AP l's standard deviation at RP n (dB).
σ^l	AP l's standard deviation at target (dB).
z_0	Uncertainty range parameter of contour.
$\mathbf{S}^l$	Signal contour from AP l.
$\mathbf{C}^l$	Indices of RPs on $\mathbf{S}^l$.
$\widehat{d}_l$	Pseudo distance from AP l.
D_n	Contour weight at RP n.
$\mathbf{R}$	Indices of selected RPs for LP localization.
$N^{\mathbf{R}}$	Number of RPs selected in $\mathbf{R}$.
Δ_n^l	Minimum distance between RP n and $\mathbf{S}^l$.
ω_n	Weight of RP n in target estimation.
$\mathbf{r}_n$	2-D coordinate of RP n.
$\widehat{\mathbf{x}}$	Estimated 2-D coordinate of target.

3. FORMING CONTOURS FOR RANDOM SIGNALS

In this section, we describe how to properly form the signal contours for later localization estimation (Section 4). In Section 3.1, we first present how to find the signal contour for each AP, given target random RSSIs. A contour (of an AP) consists of discrete RPs whose signal level is the same as the target's measured signal level, subject to statistical fluctuation. To achieve higher localization accuracy, the RPs visited by many contours of strong signals are preferred. Therefore, in Section 3.2 we propose a weighting scheme which is able to differentiate the importance of the RPs. The important RPs are kept while those unimportant ones (where target is unlikely at) are filtered. After the above steps, given that the target is likely to be in the "dense" region of selected RPs, we finally present in Section 3.3 how to find such region based on maximally connected components. The major symbols used in this paper are shown in Table 1.

3.1 Finding a Signal Contour

By evaluating the signal map of each given AP within the survey site, we may observe a set of RPs which share similar RSSI values subject to some statistical fluctuation. These RPs forms the signal contour for that AP. Then a target measuring a similar RSSI value is likely within a certain range from that contour. And the juncture of contours from these detected APs becomes the potential location of the target. Inspired by the above observations, we present as follows how to form the signal contours given fingerprints and target signals.

Let N and L be the total number of RPs and distinct APs detected in the whole survey site, respectively. Further let $\widetilde{F}_n^l$ be the random variable of the RSSI collected at RP n for AP l in the offline fingerprint collection, where $1 \leq n \leq N$ and $1 \leq l \leq L$. Multiple samples are collected at different time τ indexed $1, 2, \ldots, T_n^l$ for RP n and AP l, which are denoted as

$$\{f_n^l(\tau) | \tau = 1, \ldots, T_n^l, T_n^l > 1\}, \tag{1}$$

where T_n^l is the total number of samples collected. The unbiased estimate of $E(\widetilde{F}_n^l)$, denoted as $\widehat{\mu}_n^l$, is the mean of $f_n^l(\tau)$'s. The unbiased estimate on the variance of $\widetilde{F}_n^l$ is denoted as $\widehat{\sigma}^2(\widetilde{F}_n^l)$. Then $\widehat{\mu}_n^l$ and $\widehat{\sigma}^2(\widetilde{F}_n^l)$ are respectively given by

$$\widehat{\mu}_n^l = \frac{1}{T_n^l}\left(\sum_{\tau=1}^{T_n^l} f_n^l(\tau)\right), \widehat{\sigma}^2\left(\widetilde{F}_n^l\right) = \frac{1}{T_n^l - 1}\left(\sum_{\tau=1}^{T_n^l}\left(f_n^l(\tau) - \widehat{\mu}_n^l\right)^2\right). \tag{2}$$

Let f_n^l be the mean of the measured fingerprint signals (a random variable) at RP n for AP l, given by

$$f_n^l = \frac{1}{T_n^l}\sum_{\tau=1}^{T_n^l} F_n^l(\tau), \tag{3}$$

where $F_n^l(\tau)$'s are random variables distributed as $\widetilde{F}_n^l$ and $f_n^l(\tau)$'s are their realized values. Let $v(\tau)$ be a noise process independent from $f_n^l(\tau)$. Let α_n^l be a parameter determining the autocorrelation of samples. Then the signal time series can be represented as a first order autoregressive model [41],

$$f_n^l(\tau) = \alpha_n^l f_n^l(\tau - 1) + (1 - \alpha_n^l)v(\tau), \tag{4}$$

where α_n^l represents the correlation between successive samples ($0 \leq \alpha_n^l \leq 1$). For f_n^l, its expected value $\bar{f}_n^l$ and standard deviation σ_n^l [41] can be estimated as

$$\bar{f}_n^l = \widehat{\mu}_n^l, \quad \sigma_n^l = \left\{\frac{\widehat{\sigma}^2\left(\widetilde{F}_n^l\right)}{(T_n^l)^2}\left[\left(\frac{1 - (\alpha_n^l)^{T_n^l}}{1 - \alpha_n^l}\right)^2 + T_n^l - 1 - (\alpha_n^l)^2\frac{1 - (\alpha_n^l)^{2(T_n^l - 1)}}{1 - (\alpha_n^l)^2}\right]\right\}^{1/2}, \tag{5}$$

respectively. Here α_n^l can be approximated by autocorrelation coefficient with lag one [41] for the RSSI samples in Equation (1), i.e.

$$\alpha_n^l \approx \frac{\sum_{\tau=1}^{T_n^l - 1}\left(f_n^l(\tau) - \widehat{\mu}_n^l\right)\left(f_n^l(\tau + 1) - \widehat{\mu}_n^l\right)}{\sum_{\tau=1}^{T_n^l}\left(f_n^l(\tau) - \widehat{\mu}_n^l\right)^2}. \tag{6}$$

Such autocorrelation within sequential RSSIs may be related to the caching in Wi-Fi sampling. Wi-Fi data caching can be identified using Timing Synchronization Function (TSF) [43] of the RSSI. We can conclude that an RSSI is a cached one if its TSF is identical to another in an earlier scan result.

To illustrate that, we collect $21,000$ signal vectors at 350 different locations in the HKIA with HTC One X (Android 4.3). At each location we collect 60 RSSI samples. For each AP l at an RP n, we calculate the α_n^l using Equation (6). Figure 3 shows the histograms of all α_n^l before and after cache filtering. We can see that the autocorrelation between RSSI samples decreases if we filter the cache based on TSF.

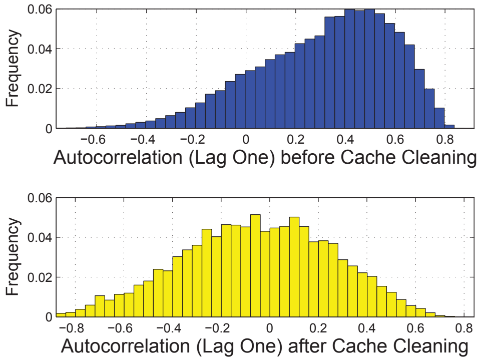


Figure 3: Lag one autocorrelation before and after cleaning Wi-Fi data cache (samples from the international airport).

Given much lower signal correlation, we can further simplify the modeling of σ_n^l [41] into

$$\sigma_n^l = \sqrt{\frac{1}{T_n^l}\widehat{\sigma}^2\left(\widetilde{F}_n^l\right)}, \tag{7}$$

However, TSF of RSSI cache is only available in the latest smartphones (supported by Android 4.2 or above). Therefore, here we implement Equations (5) and (6) to calculate σ_n^l for more general cases. Given the above, let

$$\mathbf{f}_n = [\bar{f}_n^1, \bar{f}_n^2, \ldots, \bar{f}_n^L] \tag{8}$$

be the RSSI vector (fingerprint) at RP n.

In the online query stage, denote the target measurement of AP l as g^l. We denote the RSSI vector at the target as

$$\mathbf{g} = [g^1, g^2, \ldots, g^L]. \tag{9}$$

For Equations (8) and (9), by definition, $\bar{f}_n^l = 0$ ($g^l = 0$), if AP l is not sampled at RP n (at the target).

As online g^l is also random signal, we utilize the uncertainty in offline fingerprint to characterize its variation. Specifically, let $(\sigma^l)^2$ be its variance, estimated as the mean of the variance in all the fingerprints, i.e.,

$$\left(\sigma^l\right)^2 = \frac{1}{|\mathbf{N}^l|}\left(\sum_{n \in \mathbf{N}^l}\widehat{\sigma}^2\left(\widetilde{F}_n^l\right)\right), \tag{10}$$

where $\mathbf{N}^l$ is the set of RPs detecting AP l in the site and $|\mathbf{N}^l|$ is its cardinality.

We consider the randomness in the difference between fingerprint and target RSSI, $g^l - f_n^l$ for each AP l. As g^l and f_n^l are independently measured, the variance of $g^l - f_n^l$ is therefore given by

$$\begin{aligned}\mathbb{V}(g^l - f_n^l) &= \mathbb{V}(g^l) + \mathbb{V}(f_n^l) \\ &= (\sigma^l)^2 + (\sigma_n^l)^2.\end{aligned} \tag{11}$$

A *signal contour* for AP l, denoted as $\mathbf{S}^l$, consists of a set of RPs where the target is likely within. In other words, $\mathbf{S}^l$ represents the RPs whose RSSI for AP l is likely within a certain range from g^l. Therefore, in finding the contour $\mathbf{S}^l$, we *eliminate* the RPs whose RSSI is more likely far away from the target's, i.e., if

$$\left|g^l - \bar{f}_n^l\right| > z_0\sqrt{\mathbb{V}\left(g^l - f_n^l\right)}, \; z_0 > 0, \tag{12}$$

where z_0 represents the uncertainty range. z_0 determines the sensitivity of INTRI towards the signal noise (in our experiment we will further evaluate this parameter). We hence find the signal contour for AP l as follows. Given a target RSSI g^l, we utilize Equations (11) and (12), and form the signal contour $\mathbf{S}^l$ consisting of RP $n \in \mathbf{N}^l$ such that

$$g^l - z_0\sqrt{(\sigma^l)^2 + (\sigma_n^l)^2} \leq \bar{f}_n^l \leq g^l + z_0\sqrt{(\sigma^l)^2 + (\sigma_n^l)^2}. \quad (13)$$

Given AP l, we denote the corresponding index set of RPs on contour $\mathbf{S}^l$ as $\mathbf{C}^l$, i.e.,

$$\mathbf{C}^l = \{n \in \mathbf{N}^l, \text{where } \bar{f}_n^l \text{ satisfies Equation (13)}\}. \quad (14)$$

Then we denote its cardinality as $|\mathbf{C}^l|$.

3.2 Calculation of Contour Weights

Given the found contours, an intuitive idea is to locate the target at RPs with the maximum number of contours. However, due to the indoor partitions and signal measurement uncertainty, spatially dispersed RPs may have very similar number of signal contours passing by. In this case, finding the RPs with the largest number of contours may not lead to accurate location estimation. As shown in Figure 4, a target (red diamond) measures a signal vector consisting of RSSIs from six APs, A, B, C, D, E and F. The contours of D, E and F (black circles) shift from those of A, B and C (red rectangles) due to signal fluctuation or wall partitions. Thus, the RPs in black circles may share the same number of contours as those close to target. If all these RPs are considered equally without sufficient filtering or differentiation, large location errors still exist.

Through empirical studies, we observe that the strong signals near APs provide more reliable location-dependent information than the weak ones. The stronger the RSSI, the more important the AP contour is in contributing to location estimation. This is due to the sharpness of signal strength change at the locations near the Wi-Fi APs, which differentiates RPs the most from other distant ones. Inspired by this, we consider as follows how to utilize such distinguishable RSSIs to improve location accuracy.

Take Figure 4 again as an illustration. The signals at contours of A, B and C are stronger since they are closer to the corresponding APs. If we can assign more weights on the contours with strong signals in final location decision, we can distinguish the important RPs more accurately. Thus, we propose a weighting scheme which differentiates RPs the most and finds the RPs with higher confidence. The physical intuition of the weighting scheme is based on the log-distance path loss (LDPL) model [6, 23]. We adopt it in the weighting function for signal contour differentiation, which achieves high localization accuracy.

Denote the reference power at distance d_0 as P_0^l (dBm). Let d_l be the distance between target and AP l. Then the received power at target from AP l is given by

$$g^l = P_0^l - 10\gamma^l \log_{10}\left(\frac{d_l}{d_0}\right) + X, \quad (15)$$

where γ^l denotes the decay rate of RSSI in propagation. X represents the inherent signal fluctuation and noise. Here we consider using LDPL to represent the closeness of APs for contour differentiation (not exact distance), and X has been considered separately in forming contours (Section 3.1). Based on Equation (15), we define the corresponding *pseudo distance* from AP l ($d_0 = 1$ m) as

$$\widehat{d_l} = 10^{\frac{P_0^l - g^l}{10\gamma^l}}. \quad (16)$$

Instead of indicating actual distances, we use it to represent the *confidence level* with AP signals. Specifically, the smaller $\widehat{d_l}$, the more likely the AP is nearby. Then the *contour weight* at RP n from all detected APs is defined as

$$D_n = \frac{1}{L}\sum_{l=1}^{L}\frac{1}{\widehat{d_l}}. \quad (17)$$

Using Equation (17), for each g^l at contour $\mathbf{S}^l$, we consider the potential that an AP is physically nearby. The larger D_n, the more contours of strong signals hit the RP. Such APs are more likely to be close to these RPs around the target, and we further utilize such closeness information to constrain the target region.

Then we find the RP set $\mathbf{R}$ consisting of the indices of RPs which have the highest contour weights as the potential area for final estimation, i.e.,

$$\mathbf{R} = \arg\max_n D_n. \quad (18)$$

In INTRI, we dynamically select the RPs with D_n higher than $\rho \max\{D_n\}$ ($\rho = 0.75$ in our simulation and experiment). RSSIs from an AP that is located in a region surrounded by obstacles may lead to larger γ^l than those from other APs with freer signal propagation. Here we do not assume ideal line-of-sight measurement. The external parameters (P_0^l and γ^l) in Equations (15) and (16) can be learned through gradient decent analysis over the fingerprint signals [6, 8].

To summarize, by traversing the survey site, we find the signal contours at each RP within the signal range in Equation (13), and calculate the contour weights at that RP using Equations (16) and (17). The most important RPs with strong signals will be selected to form $\mathbf{R}$.

3.3 Finding the Dense Contour Region

The selected RPs in $\mathbf{R}$ may still have "strayed RPs" due to measurement uncertainty. In Figure 5, we illustrate the spatial distribution of $\mathbf{R}$ (red rectangles), which is based on the extensive experimental observations in HKIA trials (from 1,100 target RSSI samples). We can find RPs which are physically close to each other. These RPs form a region where the target is likely at, and should be used in our localization formulation. Due to signal temporal fluctuation, some RPs (the two to the right) exist and are distant away from the target location. If included in localization, these RPs may lead to location error and unnecessary computation. Considering their spatial connectivity, we observe in Figure 5 that the RPs near the target (red diamond) form a connected component with the largest cardinality.

Based on such observation, we utilize an algorithm of finding maximally connected components, and find the region with dense contours by filtering out RPs not in the region.

We show the process of such filtering in Algorithm 1. Let $N^{\mathbf{R}}$ be the cardinality of $\mathbf{R}$. We first construct an $N^{\mathbf{R}} \times N^{\mathbf{R}}$ adjacency matrix $\mathbf{A}$, where $\mathbf{A}(i, j) = 1$ indicates that RP i and j are adjacent, and $\mathbf{A}(i, j) = 0$ otherwise (Lines 1). We set the threshold of adjacency as $\sqrt{2}$ times of the square grid width in site survey. By treating RPs in $\mathbf{R}$ as an undirected graph, we find the membership list of all connected components [35] (Lines 11 to 29). After that, we find the

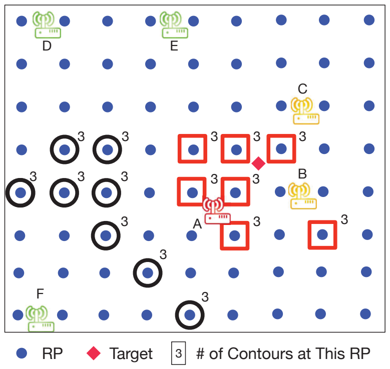


Figure 4: Illustration of differentiating signal contours.

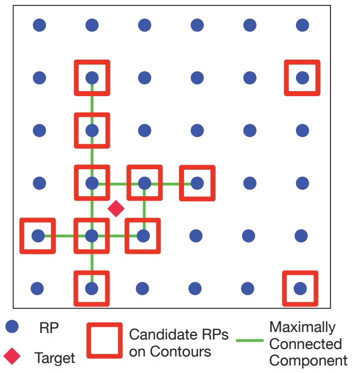


Figure 5: Illustration of maximally connected component to find the dense contour region.

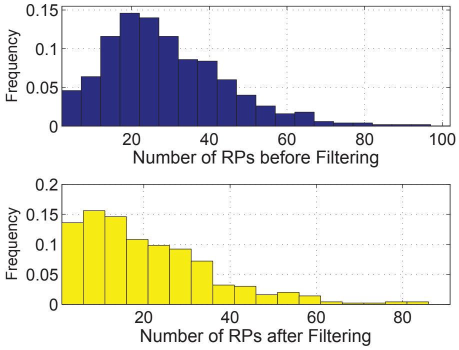


Figure 6: Histograms of $N^{\mathbf{R}}$ before and after RP filtering (the airport data).

component with the maximum number of RPs (Line 30). If multiple components have the maximum cardinality in common, we use their union for later localization.

In Figure 6, we plot the histograms of $N^{\mathbf{R}}$ from 1,100 targets in the Hong Kong International Airport before and after the proposed RP filtering. We can observe that such a scheme narrows the search scope and facilitates the final location estimation.

4. LINEAR PROGRAMMING FOR LOCATION ESTIMATION

In this section, we present the core formulation of INTRI. To formulate the objective function, we first define in Section 4.1 the physical (geographical) distance from an RP to signal contours. Then in Section 4.2, we formulate a linear programming based on weighted physical distances to those contours. We finally analyze the online computational complexity of INTRI in Section 4.3.

4.1 Defining Distances from an RP to Signal Contours

We are given a set of RPs where the contours locate. In the following we introduce how to utilize the signal contours as the objective for INTRI .

Recall that traditional trilateration estimates target position by minimizing the difference between the measured distances and the distances from the position to the landmarks. In our formulation, based on the same spirit, we use the distances to the constructed signal contours. As RPs on contours are discretely sampled in the survey site and a target is surrounded by RPs, we utilize in our formulation the distances from RPs in $\mathbf{R}$ to those on other contours. The RPs with small distances to other $\mathbf{S}^l$'s are likely to be the target region.

Let $\mathbf{r}_n = [x_n, y_n]$ be the coordinate of RP $n \in \mathbf{R}$. We calculate its distance from each RP m $(n \neq m)$ in $\mathbf{S}^l$, i.e.,

$$\delta^l_{nm} = \sqrt{(x_n - x^l_m)^2 + (y_n - y^l_m)^2},\ \forall m \in \mathbf{C}_l, \tag{19}$$

where $[x^l_m, y^l_m]$ is RP m's coordinate on signal contour l. Given all distances $\delta^l_{nm}, \forall m \in \mathbf{C}_l$, we find the minimum one, i.e.,

$$\Delta^l_n(\mathbf{g}, \mathbf{f}_n) = \min \delta^l_{nm}, \forall m \in \mathbf{C}_l, \tag{20}$$

Algorithm 1: Finding dense contour region.

```
   Input: R: indices of selected RPs; $\zeta$: threshold.
   Output: R: set of RPs with dense contours.
   /* Constructing Adjacency Matrix.                    */
1  $\mathbf{A} \leftarrow zeros(N^{\mathbf{R}}, N^{\mathbf{R}})$./* $N^{\mathbf{R}} \times N^{\mathbf{R}}$ matrix.     */
2  for $i \in \{2, \ldots, N^{\mathbf{R}}\}$ do
3      for $j \in \{1, \ldots, i\}$ do
4          if $dist(i,j) \leq \zeta$ then
               /* Symmetric Matrix                      */
5              $\mathbf{A}(i,j) \leftarrow 1$; $\mathbf{A}(j,i) \leftarrow 1$;
6          end
7      end
8  end
9  $isDiscovered \leftarrow zeros(N^{\mathbf{R}})$;
10 $mem \leftarrow \{\}$;/* Set of Components                 */
11 $nGp \leftarrow 0$; /* Number of Components              */
12 for $n \in \{1, \ldots, N^{\mathbf{R}}\}$ do
13     if $!isDiscovered(n)$ then
14         $nGp \leftarrow nGp + 1$;
15         $isDiscovered(n) \leftarrow 1$;
16         $mem[nGp].pt \leftarrow mem[nGp].pt \cup \{n\}$;
17         $ptr \leftarrow 1$;
18         while $ptr \leq sizeof(mem[nGp].pt)$ do
               /* Find Its Neighbors.                   */
19             $nbrs \leftarrow find(A(:, mem[nGp].pt(ptr)))$;
20             for $nb \in \{1, \ldots, sizeof(nbrs)\}$ do
                   /* Connected Components              */
21                 if $!isDisovered(nbrs(nb))$ then
22                     $isDisovered(nbrs(nb)) \leftarrow 1$;
23                     $mem[nGp].pt \leftarrow$
                       $mem[nGp].pt \cup \{nbrs(nb)\}$;
24                 end
25             end
26             $ptr \leftarrow ptr + 1$;
27         end
28     end
29 end
30 $\mathbf{R} \leftarrow MaxMembers(mem)$;
```

which represents the distance between RP n and contour l. Given above, the distances to all contours are aggregated as

$$\Delta_n = \sum_{l=1}^{L} \Delta^l_n(\mathbf{g}, \mathbf{f}_n), \tag{21}$$

which, in other words, approximates the residual between the target estimation's distances to the landmarks and the measured distances.

By minimizing Equation (21), we minimize the distance difference and hence extend the idea of trilateration into a contour-based scheme. As the contours are derived from fingerprints and target RSSIs, we require no explicit knowledge of AP locations or LoS measurement, and therefore combine the advantages of both fingerprinting and trilateration in our formulation. In the following, we present the formulation to find the target position.

4.2 Linear Programming Formulation

Given $\mathbf{R}$ and Δ_n, we formulate the localization problem based on linear programming (LP). For each target, denote its estimated location as $\widehat{\mathbf{x}} = [\widehat{x}, \widehat{y}]$. Let ω_n be the weight assigned to $\mathbf{r}_n$ in locating the target. As the target is surrounded by the RPs in $\mathbf{R}$, its estimated position can be expressed as

$$\widehat{\mathbf{x}} = \sum_{n=1}^{N^{\mathbf{R}}} \omega_n \mathbf{r}_n, \tag{22}$$

where $\mathbf{r}_n \in \mathbf{R}$, and the weights satisfy the normalization and non-negativity, i.e.,

$$\sum_{n=1}^{N^{\mathbf{R}}} \omega_n = 1, \quad \omega_n \geq 0, \quad \forall n \in \{1, \ldots, N^{\mathbf{R}}\}. \tag{23}$$

Based on Equation (20) and (22), we extend the idea of trilateration into finding the weights which minimize the target's weighted sum of distances to all the contours, i.e.,

$$\underset{\{\omega_n\}}{\arg\min} \sum_{n=1}^{N^{\mathbf{R}}} \omega_n \Delta_n. \tag{24}$$

In real deployment, a target is far more likely to be between two neighboring RPs (or within the square grid formed by four RPs). In order to jointly consider the neighboring RPs to the target, we set a constraint over the weight ω_n at each RP, i.e.,

$$w_n \leq W, \ \forall n \in \{1, \ldots, N^{\mathbf{R}}\}, \tag{25}$$

where W is a dynamic parameter determined by the maximum contour weight

$$W = \frac{\max D_n}{\sum_{n=1}^{N^{\mathbf{R}}} D_n}. \tag{26}$$

Through Equations (24) and (25), we can jointly consider the physical distances (denoted as $\{\Delta_n\}$) and the contour weights (denoted as $\{D_n\}$) in our formulation.

If there are indoor wall partitions in narrow space, we can include map constraints in our basic formulation. Denote the set of map constraints as $\mathbf{E}$. For each edge $e \in \mathbf{E}$, we consider the accessible area within the map constraints as

$$a_e \widehat{x} + b_e \widehat{y} + c_e \geq 0, e \in \mathbf{E}, \tag{27}$$

where a_e, b_e and c_e are the line parameters obtained from the site map in our system initialization. The formula of map edges can be easily found using the nearest map constraints. Using the above, the localization problem can therefore be formulated as a linear programming (LP):

$$\begin{array}{ll} \text{Objective:} & \text{Equation (24),} \\ \text{subject to:} & \text{Constraints (22), (23),} \\ & \text{(25), (26) and (27).} \end{array} \tag{28}$$

In other words, we opt to find the estimated position with the smallest weighted physical distances to the contours within the accessible area. INTRI returns the set of weights assigned to RPs in $\mathbf{R}$ which minimizes the distances to contours, i.e., the RPs which are closer to all contours are assigned higher weights, and vice versa.

Using some commercial optimization solver, we can solve the above LP efficiently [3]. The final solution $\{\omega_n\}$ is then used to estimate the target position according to Equation (22). We summarize INTRI in Algorithm 2.

Algorithm 2: Contour-based indoor trilateration.

```
Input: {r_n} and {f_n}: the set of N RPs and the set of
       RSSI vector at each RP; g: target RSSI vector;
       ζ: the threshold in determining maximally
       connected components; ρ: range of RPs to be
       selected.
Output: x̂: estimated locations of the target.
/* Construction of Signal Contours.                  */
$\mathbf{C}^l \leftarrow \{\}$; /* Set of RPs with Contours.     */
for $n \in \{1, \ldots, N\}$ do
    $D_n \leftarrow 0$;
    for $l \in \{1, \ldots, L\}$ do
        if $\bar{f}_n^l \geq g^l - z_0\sqrt{(\sigma^l)^2 + (\sigma_n^l)^2}$ and
        $\bar{f}_n^l \leq g^l + z_0\sqrt{(\sigma^l)^2 + (\sigma_n^l)^2}$ then
            $\mathbf{C}^l \leftarrow \mathbf{C}^l \cup \{n\}$;  $D_n \leftarrow D_n + 10^{(g^l - P_0^l)/(10\gamma^l)}$;
        end
    end
    $D_n \leftarrow D_n / L$;
end
/* Selecting Dense Contours.                         */
$\mathbf{R} \leftarrow FindHighWeight(\mathbf{\Omega}, \rho, \{D_n\})$;
/* RP Filtering Using Connectivity.                  */
$\mathbf{R} \leftarrow FindMaxComponent(\mathbf{R}, \zeta)$;
/* Calculating Dist from Contours.                   */
for $n \in \{1, \ldots, N^{\mathbf{R}}\}$ do
    for $l \in \{1, \ldots, L\}$ do
        for $m \in \{1, \ldots, |\mathbf{C}^l|\}$ do
            $\delta_{nm}^l = \sqrt{(x_n - x_m^l)^2 + (y_n - y_m^l)^2}$;
        end
        $\Delta_n^l(\mathbf{g}, \mathbf{f}_n) = \min\{\delta_{nm}^l\}$;
    end
end
LP-based Localization Using Formulation (28);
$\widehat{\mathbf{x}} \leftarrow \sum_{n=1}^{N^{\mathbf{R}}} \omega_n \mathbf{r}_n$;/* Final Estimation.   */
```

4.3 Complexity Analysis

We here analyze the computational complexity of INTRI:

1. *Forming contours with random signals:* Given N RPs and L APs, the complexity of finding signal contours and calculating contour weights is given by $\mathcal{O}(NL)$ (Section 3.1 and 3.2).

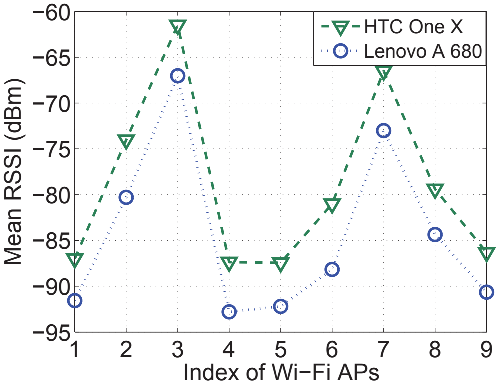


Figure 7: Device heterogeneity between two devices.

2. *Finding maximally connected components:* Given $N^{\mathbf{R}}$ selected RPs, finding the maximally connected component takes $\mathcal{O}((N^{\mathbf{R}})^2)$ [35] (Section 3.3).

3. *LP-based localization:* To prepare the objective function of LP, calculating the distances between $\mathbf{R}$ and contours takes $\mathcal{O}\left(N^{\mathbf{R}}L|\mathbf{C}^l|\right)$ (Section 4.1). In Formulation (28), there are $\mathcal{O}(N^{\mathbf{R}})$ decision variables in $\{\omega_n\}$. Thus, the LP in location estimation takes weak polynomial time, i.e., $\mathcal{O}\left((N^{\mathbf{R}})^3\right)$ [3] (Section 4.2).

To summarize, the overall online running time of INTRI is

$$\mathcal{O}\left(NL + N^{\mathbf{R}}L|\mathbf{C}^l| + \left(N^{\mathbf{R}}\right)^3\right). \qquad (29)$$

After contour differentiation and finding maximally connected components, $N^{\mathbf{R}} \ll N$. Further computation reduction can be done via AP filtering and RP cluster mapping [13]. Then we can significantly reduce the number of APs and RPs. In this way, INTRI can be integrated on existing on-board LBS systems and further support mobile targets.

5. EFFICIENT ONLINE CALIBRATION FOR HETEROGENEOUS DEVICES

Due to difference in Wi-Fi network interfaces, for the same signal different smartphones may have different measurement values [14, 25]. To illustrate this, we conduct an experiment and collect 1,000 RSSI samples using HTC One X and Lenovo A680, respectively. If such signal difference issue is not addressed, the contours (Section 3.1) cannot be found correctly. Figure 7 shows the similarity of signal trend between signals of the two smartphones despite the shift. Leveraging such similarity, we present an efficient algorithm to adapt to different mobile devices as follows.

In this section we consider efficient and scalable online calibration in order to reduce offline manual efforts. To this end, we map the target signals $\mathbf{g}$ to the signal space in fingerprint database. We first calculate the correlation between the target RSSI vector and that of each RP. The RPs with similar signal vectors can be leveraged for online signal calibration. The vector comparison is based on the correlation between $\mathbf{g}$ and fingerprint $\mathbf{f}_n$, i.e.,

$$corr(\mathbf{g}, \mathbf{f}_n) = \frac{\sum_{l=1}^{L}\left(g^l - \bar{g}\right)\left(\bar{f}_n^l - \bar{f}_n\right)}{\sqrt{\sum_{l=1}^{L}\left(g^l - \bar{g}\right)^2}\sqrt{\sum_{l=1}^{L}\left(\bar{f}_n^l - \bar{f}_n\right)^2}}, \qquad (30)$$

where $\bar{g} = \frac{1}{L}\sum_{l=1}^{L} g^l$ and $\bar{f}_n = \frac{1}{L}\sum_{l=1}^{L}\bar{f}_n^l$. The above correlation compares relative signal trend of different APs rather than the absolute RSSI values. Based on Equation (30), we can find the RPs with similar signal trend for calibration and reduce the effect of device dependency.

To mitigate the effect of random noise, we find the top several RPs with $corr(\mathbf{g}, \mathbf{f}_n) > \eta$ ($\eta = 0.95$ in our experiment) for linear calibration. For each target RSSI g^l from AP l, we find the corresponding f_n^l at RPs. Given pairs of $[g^l, f_n^l]$, we conduct the linear regression and obtain the corresponding a and b for target RSSI g^l, i.e.,

$$\widetilde{g}^l = ag^l + b. \qquad (31)$$

Note that our online calibration approach is not restricted to linear model, and is general enough to apply to any other signal mapping model (e.g. [32]). Based on Equation (31) we can conduct INTRI with calibrated $\widetilde{g}^l$. Given $\mathcal{O}(L)$ APs and $\mathcal{O}(N)$ RPs, the correlation comparison takes $\mathcal{O}(NL)$. Let λ ($\lambda \ll N$) be the number of RPs whose correlation $corr(\mathbf{g}, \mathbf{f}_n)$ is greater than η, and linear regression takes $\mathcal{O}(L^2\lambda^2)$. Therefore, the online computational complexity of 2-D linear regression is $\mathcal{O}(NL + L^2\lambda^2)$ [3].

6. ILLUSTRATIVE SIMULATION RESULTS

We develop a simulation environment ($100 \times 90m^2$) based on the setting of a local airport. We consider all the Wi-Fi transmitters and receivers are equipped with omni-directional antennas. The RSSI g^l (dBm) at distance d from AP l is given by the LDPL model [1]:

$$g^l = P^{TX} - L_0 - 10\gamma\log_{10}\left(\frac{d}{d_0}\right) + X, \qquad (32)$$

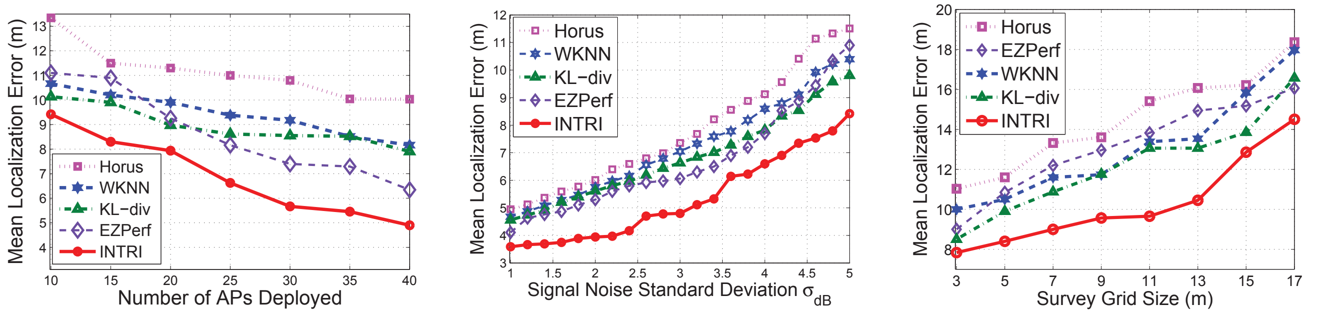
where $X \sim \mathcal{N}(0, \sigma_{dB}^2)$ is the Gaussian measurement noise. In our simulation, we set transmission power P^{TX} 25 dBm, the path loss exponent γ 4.0, the reference path loss L_0 37.7 dBm [1] and the reference distance d_0 1 m. Based on empirical observations, if $g^l < -95$ dBm, the target cannot detect the signal of this AP and we discard this measurement.

We evaluate the performance of INTRI in terms of AP number, signal noise and survey grid size in our simulation environment. We compare INTRI with four state-of-the-art schemes:

- *EZPerfect* (EZPerf) [6, 31]: a model-based Wi-Fi localization scheme which considers deriving signal propagation model from fingerprint data. Given Wi-Fi fingerprints, EZPerfect first finds the matching relationship between signals and distances from APs [6], and then locates targets with a genetic algorithm solving trilateration problem [31].

- *KL-divergence* (KL-div) [29, 30]: which utilizes the Kullback-Leibler (KL) divergence distance [30] between the signal distribution at an RP and target during comparison. Then the top k RPs with the minimum KL-divergence are utilized for final estimation.

- *Weighted k-Nearest Neighbors* (WKNN) [11, 12]: It computes the cosine similarity [12] between the fingerprints and the target RSSI vector. Then it finds the weighted average of k-Nearest Neighbors [11] of highest cosine similarity to estimate the target location.



(a)

(b)

(c)

Figure 8: Mean errors in simulation versus (a) number of APs deployed; (b) signal measurement noise; (c) survey grid size.

- *Probabilistic algorithm* (Horus [42]): It first calculates the probability distribution of the RSSI values at each RP. Given a target RSSI vector, Horus computes the overall probability of the vector at each RP and finds the top several ones with the maximum likelihood as the target location.

Unless otherwise stated, we use following baseline parameters in simulation: 15 APs are deployed; $\sigma_{dB} = 5$ dB (Equation (32)); $k = 20$ for KL-divergence, WKNN and Horus algorithm; $z_0 = 3$ (Equation (12)); survey grid size is 5 m.

Let $\mathbf{x}_m$ be target m's true location and $\widehat{\mathbf{x}}_m$ be its estimated location. The performance metric in our experiment is the mean error (unit:m) of the estimated targets in set $\mathbf{V}$:

$$\mu_e = \frac{1}{|\mathbf{V}|} \sum_{m \in \mathbf{V}} \|\mathbf{x}_m - \widehat{\mathbf{x}}_m\|. \tag{33}$$

In Figure 8(a), we show the mean localization errors versus the number of APs. The performance of all the algorithms improves as the number of APs increases. As there are more APs deployed in the site which help differentiate the RPs, the error decreases. Compared with these traditional algorithms, INTRI achieves much lower errors under different numbers of deployed APs. It is because INTRI considers differentiating contours using contour weights, which helps distinguish the RPs and prevents dispersed estimations.

In Figure 8(b), we show the mean localization errors against the signal measurement noise (σ_{dB} in Equation (32)). Clearly, the performance of all the algorithms degrades as the noise in Wi-Fi signals increases. Signal noise leads to dispersed nearest neighbors and distance measurement error, which degrades the accuracy of traditional pattern matching algorithms and EZPerfect. Especially in different indoor environments ranging from spacious halls or narrow corridors, the signal fluctuation of fingerprints can be different. Even under large fingerprint variation, INTRI achieves much better accuracy. Different from these traditional and state-of-the-art algorithms, INTRI considers the signal variation in the contour construction. Therefore, it achieves higher robustness and localization accuracy under large signal noise.

Figure 8(c) shows the mean localization errors versus survey grid size, i.e., we vary the site survey grid width (in square shape) to change the fingerprint density, given different schemes. Accuracy suffers as grid size increases, illustrating the tradeoff between survey cost and localization accuracy. INTRI considers the signal contours in the formulation to constrain the target region. By minimizing distances to contours, INTRI prevents dispersion of nearest neighbors and achieves higher accuracy.

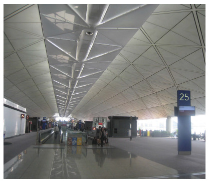


Figure 9: Survey site of boarding area at HKIA.

Figure 10: Survey site of HKUST campus hall.

Figure 11: Survey site of HKCP shopping mall.

7. EXPERIMENTAL EVALUATIONS

Besides simulation, we have conducted extensive experimental trials of INTRI in the Hong Kong International Airport (HKIA) (Figure 9), our university campus (HKUST) (Figure 10) and the Hong Kong Cyberport (HKCP) (Figure 11). We first present our experimental settings in Section 7.1. As the measured AP signals are different in the three sites, we discuss the comparative studies over these differences in Section 7.2. Then we illustrate the experimental results in HKIA (Section 7.3) followed by those in HKUST and HKCP (Section 7.4).

7.1 Experimental Settings

We use in our experimental studies the same state-of-the-art algorithms and comparison metrics as in Section 6. Besides, we compare the device calibration scheme in INTRI with two recent methods, signal strength difference (SSD) [25] and signal ratio (SR) [28]. SSD utilizes the differences between pairs of AP signal values as patterns. Similarly, SR calculates the ratio between pairs of AP signals as Wi-Fi fingerprints. Both methods aim at compensating the signal difference among heterogeneous devices.

In Figure 12, we show the corresponding survey floor plan of RPs and targets in HKIA. In the $10,000$ m^2 site, we collect 340 RPs and $1,100$ targets. We utilize HTC One X as the fingerprint collector and Lenovo A680 for target measurement. The locations of RPs are predefined on the indoor map. To balance between localization accuracy and survey cost, we use 5 m grid density in fingerprinting. At each RP, signal data is sampled from four different directions (north, west,

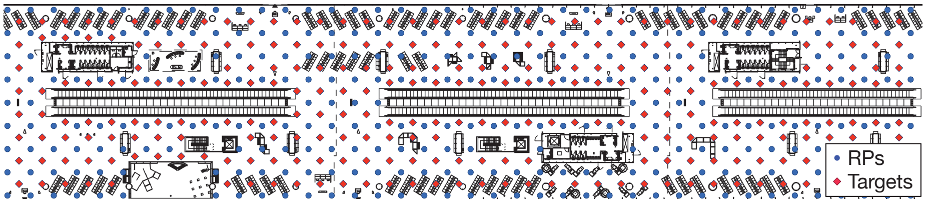


Figure 12: Indoor map of a boarding gate at the HKIA. The survey grid size is 5 m (survey conducted on July 8, 2014).

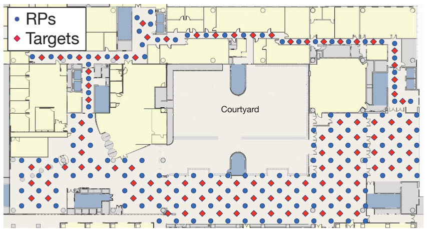


Figure 13: Hall map of HKUST campus (survey conducted on Nov. 28, 2014).

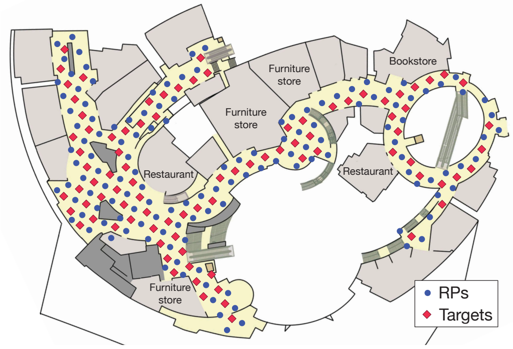


Figure 14: Indoor map of HKCP mall (survey conducted on Sept. 10, 2014).

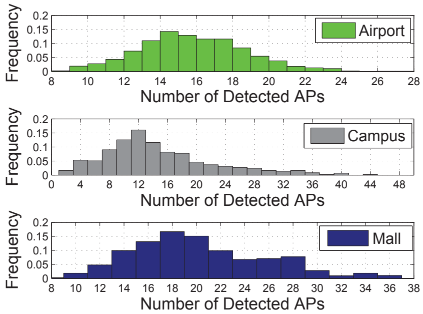


Figure 15: Histograms of detected AP number at targets in three sites.

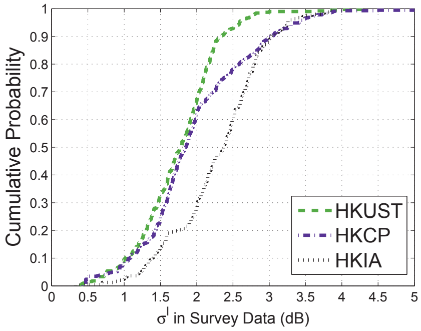


Figure 16: Average signal noise σ^l (Equation (10)) in different sites.

south and east). For each direction, 15 samples of RSSI vectors are collected. The ground truth of the target locations is also predetermined in grid form (in the testing, the surveyors find the RP or target locations from the nearest pillars, floor tiles and other noticeable indoor landmarks). Note that the data sampling is conducted with people around, and temporal fluctuation exists within the fingerprints and target signals. The time interval between samples in Wi-Fi scanning is 1 second.

Wi-Fi APs are officially pre-deployed. Thus, their number, locations and transmission power are already settled before site survey. In the data preprocessing, we filter out the mobile APs tethered by smartphones, and combine the signals of virtual APs (VAPs) [27]. Overall 360 APs are detected (each RP measures 47 APs on average). Part of these APs may be installed outside the survey site since their coverage in our site is relatively small and signals are globally weak. The target samples are collected one month later than RP collection. For schemes like KL-divergence and Horus which are device dependent, we utilize our scheme to calibrate the signals.

In the HKUST campus and the HKCP shopping mall, fingerprint collection, target sampling and data preprocessing are the same as those in the airport. Figure 13 shows the RPs and targets on campus ($100 \times 50m^2$). In campus corridor environment, we collect 250 RPs and 475 targets. Figure 14 shows the RPs and targets in the shopping mall ($150 \times 100m^2$). In the HKCP mall, we collect 680 RPs and 680 targets. In both the HKUST and HKCP, the blue dots represent RPs and the red diamonds are targets. Similar to the HKIA, we are given the officially deployed APs and we cannot manually change their settings (installation locations and TX power). In the site survey of campus corridor, overall 320 APs are detected (each RP measures 24 APs on average). In the site survey at shopping mall, overall 190 APs are detected (average 28 APs at each RP).

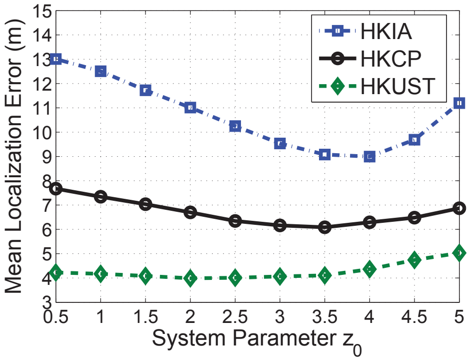


Figure 17: Mean errors versus parameter z_0 in three sites.

7.2 Comparative Studies of Different Sites

Figure 15 shows the histograms of AP number detected by targets at different sites. On average, each target can detect 16 APs on HKUST and 22 APs in HKCP. In HKIA, each target can detect 16 APs on average. Though targets in HKIA and HKUST have similar detected AP number, the survey site in HKUST is smaller and hence it has denser AP deployment. Moreover, the signal coverage of APs in our campus corridor and HKCP is constrained by the wall partitions, which helps differentiate the RPs. Therefore, we expect a better localization performance on HKUST and HKCP than in HKIA. Based on these detected APs, we evaluate the effect of different received AP numbers.

Figure 16 shows the corresponding signal noise in each survey site. We show the cumulative probability of the σ^l according to Equation (10). Clearly, we can see a smaller signal noise in HKUST and HKCP than that in HKIA. It is because the airport boarding area is large open space with

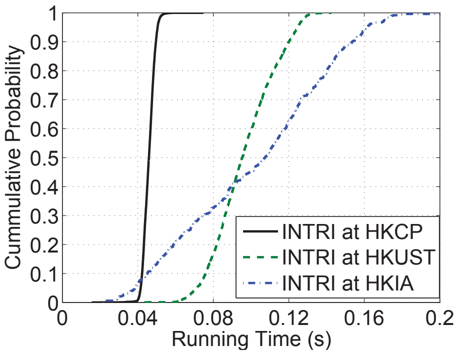


Figure 18: Cumulative probability of INTRI running time on different sites.

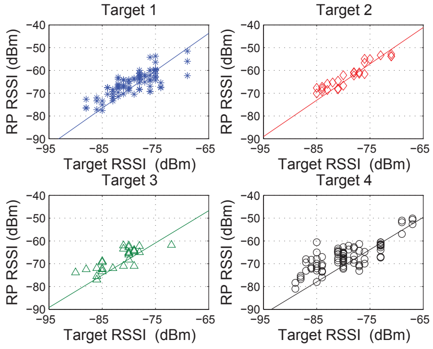


Figure 19: Calibration of target RSSI from RP signals (HKIA).

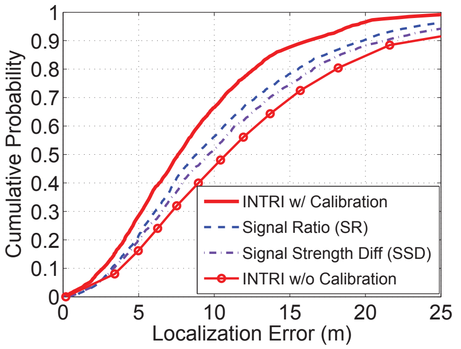


Figure 20: Performance of different device calibration methods in HKIA.

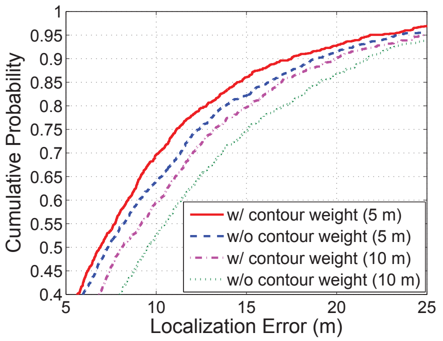


Figure 21: Performance of INTRI with and without contour weights (HKIA).

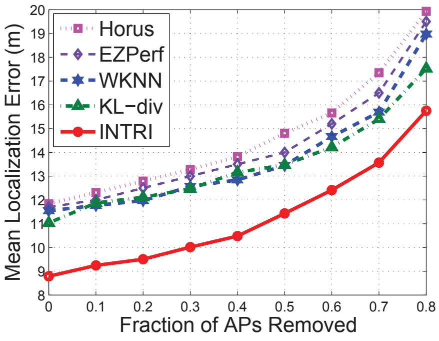


Figure 22: Mean errors versus fraction of APs removed (HKIA).

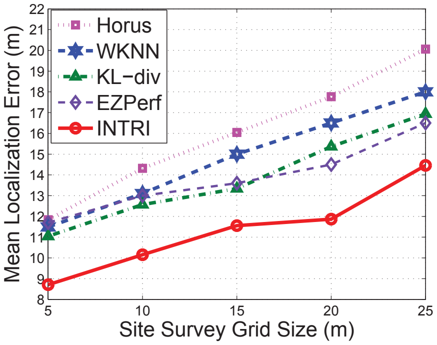
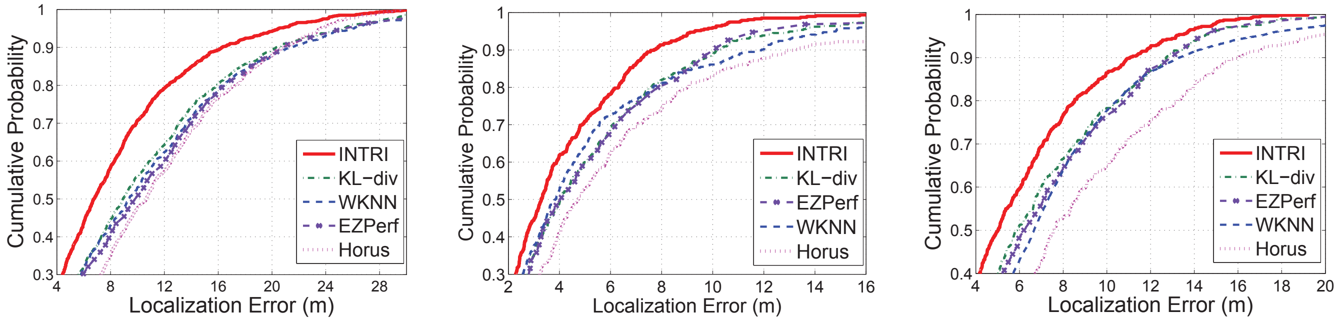


Figure 23: Mean localization errors versus site survey grid size (HKIA).



(a)

(b)

(c)

Figure 24: Cumulative probability of localization errors in (a) HKIA boarding gate; (b) HKUST campus; (c) HKCP mall.

many airline passengers, which leads to higher signal uncertainty. Based on the difference in signal noise, we adjust the parameter z_0 in Equation (12) for online localization.

Figure 17 shows the mean localization errors of INTRI in three sites given different z_0. In general, the error first decreases and then increases. This is because the localization error depends on two factors: signal noise and RP fingerprint differentiation. When z_0 is small, the tight contours cannot accommodate the large measurement uncertainty in target. Thus, the error is high. As z_0 increases, the contours can bound the target, and hence the error decreases. As z_0 further increases, the error increases because, as contours become wide, the differentiation between RPs becomes weak. Then more distant RPs are included in localization. The result shows that without sufficient RP fingerprint differentiation, wide contours would not help. Compared with HKUST and HKCP, z_0 is slightly larger in HKIA due to higher signal variance in airport (as shown in Figure 16). Thus, in our experiment, $z_0 = 4$ in HKIA, $z_0 = 3.5$ in HKCP, and $z_0 = 2.5$ on our HKUST campus.

Figure 18 shows the running time CDF of INTRI on the targets at different sites. We test INTRI on a server PC with i7 4790 (3.6GHz) CPU. It shows that at the three sites, our INTRI shows different computation time due to the difference in RP numbers and detected AP number at target side and RP side. The running time of INTRI at HKCP is much smaller than that at HKIA and HKUST. It is mainly because many of detected APs in HKCP are installed in the shop stores and have limited coverage (small contours) due to wall partitions, compared with those in HKIA and HKUST. A target can be quickly mapped to smaller areas and therefore the overall computation becomes much smaller.

7.3 International Airport Trials

Figure 19 shows the linear signal model of four different target estimations in HKIA. Target data (x-axis) are collected using Lenovo A680 while RP signals (y-axis) are from HTC One X. Based on the a and b, we obtain the calibrated signal strength, $\widetilde{g}^l$, for each given target. We can see that the linear calibration scales up the Lenovo A680 measurements (target), which matches our observation in Figure 7.

Figure 20 compares the cumulative errors of different device calibration schemes in the airport. We use uncalibrated signals in INTRI as the baseline case. The calibrated INTRI improves from the uncalibrated scheme, and also outperforms SSD and SR. It is mainly because our proposed scheme jointly considers the relative trend and the RSSI adjustment model when calibrating devices. Unlike SSD and SR, our correlation scheme mitigates the errors in signal values when INTRI calibrates target RSSI using fingerprints, which is more robust under large signal noise in the airport.

Figure 21 shows the localization error with and without differentiating the contours using HKIA data. We consider two scenarios using 5 m (default) and 10 m survey grid size. Without using contour weights, we count the number of signal contours and implement it into INTRI as the baseline case. As shown in Figure 21, simple contour counting cannot discriminate the dispersed nearest neighbors (RPs) with similar number of contours. In contrast, contour weight discriminates the fingerprints by penalizing RPs only with many weak signals. Then we mitigate the influence from the noise in the measurement. Thus, we have shown using contour weights achieves better performance than the unweighted scheme, especially under sparser survey grid size.

In Figure 22, we plot the mean localization errors of these algorithms against the number of APs used. We randomly select part of the APs detected at target side to simulate the miss of the RSSI due to crowds of people or site construction change, which generally exists in the HKIA. As the number of APs used decreases, the performance of all five algorithms degrades. INTRI is less susceptible to AP number change than other four algorithms. It is mainly because junction of contours can constrain the target in a small region and reduce the dispersed set of nearest neighbors.

Figure 23 shows the mean localization errors against the survey grid size. As the minimum grid size is five meters, columns or rows of RPs are removed to form the grid sizes with multiples of five. Clearly all five algorithms degrade as grid size increases. We can observe EZPerfect achieves slightly higher accuracy than WKNN, KL-divergence and Horus under larger grid size. It is because for EZPerfect additional distances from multiple APs constrain the target location and prevent large error. Compared with the algorithms above, INTRI has much higher localization accuracy at different grid sizes. It is because INTRI considers signal uncertainty in contours and constrains the target estimation through joint optimization. Without relying on accurate distance measurement, INTRI can still constrain the target estimation by signal contours under large grid size.

Figure 24(a) compares the cumulative errors of five algorithms in HKIA. Due to large measurement noise in the airport, WKNN's accuracy is weakened by the dispersed nearest neighbors. Horus assumes a certain distribution of signal level at each RP and therefore cannot represent real signal distribution under limited sampling. KL-divergence also requires large data sampling and dense fingerprints in signal distribution comparison. Therefore, it cannot adapt to the noisy airport environment. The large signal noise also degrades the distance accuracy of traditional trilateration in EZPerfect. In contrast to above methods, INTRI jointly considers the signal noise and distances to contours, and therefore reduces misestimation of the target.

7.4 Campus Hall & Shopping Mall Trials

Figure 24(b) and Figure 24(c) show the cumulative errors in HKUST and HKCP, respectively. Compared with the HKIA, the fingerprints and target signals in HKUST and HKCP carry less signal measurement noise. Thus, we observe the increase of localization accuracy in all the algorithms at these two sites compared with that in the airport. EZPerfect becomes slightly better than WKNN as the distance measurement becomes less noisy. Similar to Figure 24(a), INTRI achieves higher accuracy than other algorithms since it considers the signal variation in constructing contours and utilizes them to reduce the dispersed nearest neighbors. INTRI is general enough to work under different environments with markedly higher accuracy.

We also study the performance of INTRI in the HKUST and HKCP extensively. As the conclusions are qualitatively the same, for brevity we will not repeat them here.

8. CONCLUSION

Traditional trilateration has achieved much success for outdoor localization. However, it does not work well indoors due to non-line-of-sight measurement and signal fading. Fingerprinting is a promising approach for indoor localization, but its performance is vulnerable to signal noise. We propose in this work INTRI, a highly accurate algorithm which combines the advantages of both trilateration and fingerprinting to achieve much better accuracy.

Based on the spirit of trilateration, for each measured AP signal level, INTRI forms the corresponding contour, which is the set of RPs with the same signal level subject to its statistical fluctuation. The target is hence where the contour is. To estimate target's location, INTRI formulates a linear programming to minimize the distance between the location and these contours (i.e., following the spirit of trilateration). To achieve higher accuracy, INTRI addresses device heterogeneity with an efficient and scalable algorithm based on the correlation in RSSI for online signal calibration. We have conducted extensive simulation and experimental studies on INTRI in the Hong Kong International Airport, Hong Kong Cyberport mall and HKUST campus. Compared with other approaches, INTRI achieves significantly higher accuracy and robustness under signal noise (often by more than 20%).

9. ACKNOWLEDGMENTS

This work was supported, in part, by The Hong Kong R&D Center for Logistics and Supply Chain Management Enabling Technologies (ITP/034/12LP), and Hong Kong Research Grant Council (RGC) General Research Fund (610713).

10. REFERENCES

[1] N. Alsindi, R. Raulefs, and C. Teolis. *Geolocation Techniques: Principles and Applications*. Springer, 2012.

[2] P. Bahl and V. N. Padmanabhan. RADAR: An in-building RF-based user location and tracking system. In *Proc. IEEE INFOCOM*, volume 2, pages 775–784, 2000.

[3] S. P. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[4] K. Chang and D. Han. Crowdsourcing-based radio map update automation for Wi-Fi positioning systems. In *Proc. ACM SIGSPATIAL GeoCrowd*, pages 24–31, 2014.

[5] Y.-C. Chen, J.-R. Chiang, H.-h. Chu, P. Huang, and A. W. Tsui. Sensor-assisted Wi-Fi indoor location system for adapting to environmental dynamics. In *Proc. ACM MSWiM*, pages 118–125, 2005.

[6] K. Chintalapudi, A. Padmanabha Iyer, and V. N. Padmanabhan. Indoor localization without the pain. In *Proc. ACM MobiCom*, pages 173–184, 2010.

[7] S.-H. Fang, C.-H. Wang, S.-M. Chiou, and P. Lin. Calibration-free approaches for robust Wi-Fi positioning against device diversity: A performance comparison. In *Proc. IEEE VTC Spring*, pages 1–5, 2012.

[8] B. Ferris, D. Fox, and N. D. Lawrence. WiFi-SLAM using Gaussian process latent variable models. In *Proc. IJCAI*, pages 2480–2485, 2007.

[9] A. Goswami, L. E. Ortiz, and S. R. Das. WiGEM: A learning-based approach for indoor localization. In *Proc. ACM CoNEXT*, pages 3:1–3:12, 2011.

[10] X. Guo, D. Zhang, K. Wu, and L. Ni. MODLoc: Localizing multiple objects in dynamic indoor environment. *IEEE Trans. Parallel and Distributed Systems*, 25(11):2969–2980, Nov. 2014.

[11] D. Han, S. Jung, M. Lee, and G. Yoon. Building a practical Wi-Fi-based indoor navigation system. *IEEE Pervasive Computing*, 13(2):72–79, Apr. 2014.

[12] S. Han, C. Zhao, W. Meng, and C. Li. Cosine similarity based fingerprinting algorithm in WLAN indoor positioning against device diversity. In *Proc. IEEE ICC*, pages 4313–4317, 2015.

[13] S. He and S.-H. Chan. Sectjunction: Wi-Fi indoor localization based on junction of signal sectors. In *Proc. IEEE ICC*, pages 2605–2610, June 2014.

[14] S. He and S.-H. G. Chan. Wi-Fi fingerprint-based indoor positioning: Recent advances and comparisons. *IEEE Communications Surveys and Tutorials*, to appear.

[15] S. He, S.-H. G. Chan, L. Yu, and N. Liu. Calibration-free fusion of step counter and wireless fingerprints for indoor localization. In *Proc. ACM UbiComp*, 2015.

[16] S. He, S.-H. G. Chan, L. Yu, and N. Liu. Fusing noisy fingerprints with distance bounds for indoor localization. In *Proc. IEEE INFOCOM*, pages 2506–2514, 2015.

[17] S. Hilsenbeck, D. Bobkov, G. Schroth, R. Huitl, and E. Steinbach. Graph-based data fusion of pedometer and WiFi measurements for mobile indoor positioning. In *Proc. ACM UbiComp*, pages 147–158, 2014.

[18] Y. Jiang, Y. Xiang, X. Pan, K. Li, Q. Lv, R. P. Dick, L. Shang, and M. Hannigan. Hallway based automatic indoor floorplan construction using room fingerprints. In *Proc. ACM UbiComp*, pages 315–324, 2013.

[19] J. Jun, Y. Gu, L. Cheng, B. Lu, J. Sun, T. Zhu, and J. Niu. Social-Loc: Improving indoor localization with social sensing. In *Proc. ACM SenSys*, pages 14:1–14:14, 2013.

[20] Y. Kim, H. Shin, and H. Cha. Smartphone-based Wi-Fi pedestrian-tracking system tolerating the RSS variance problem. In *Proc. IEEE PerCom*, pages 11–19, Mar. 2012.

[21] S. Kumar, S. Gil, D. Katabi, and D. Rus. Accurate indoor localization with zero start-up cost. In *Proc. ACM MobiCom*, pages 483–494, 2014.

[22] A. Kushki, K. N. Plataniotis, and A. N. Venetsanopoulos. Kernel-based positioning in wireless local area networks. *IEEE Trans. Mobile Computing*, 6(6):689–705, 2007.

[23] L. Li, G. Shen, C. Zhao, T. Moscibroda, J.-H. Lin, and F. Zhao. Experiencing and handling the diversity in data density and environmental locality in an indoor positioning service. In *Proc. ACM MobiCom*, pages 459–470, 2014.

[24] H. Liu, J. Yang, S. Sidhom, Y. Wang, Y. Chen, and F. Ye. Accurate WiFi based localization for smartphones using peer assistance. *IEEE Trans. Mobile Computing*, 13(10):2199–2214, Oct 2014.

[25] A. Mahtab Hossain, Y. Jin, W.-S. Soh, and H. N. Van. SSD: A robust RF location fingerprint addressing mobile devices' heterogeneity. *IEEE Trans. Mobile Computing*, 12(1):65–77, Jan. 2013.

[26] A. T. Mariakakis, S. Sen, J. Lee, and K.-H. Kim. SAIL: Single access point-based indoor localization. In *Proc. ACM MobiSys*, pages 315–328, 2014.

[27] E. Martin, O. Vinyals, G. Friedland, and R. Bajcsy. Precise indoor localization using smart phones. In *Proc. ACM MM*, pages 787–790, 2010.

[28] Mikkel Baun Kjaergaard. Indoor location fingerprinting with heterogeneous clients. *Pervasive and Mobile Computing*, 7:31 – 43, 2011.

[29] P. Mirowski, D. Milioris, P. Whiting, and T. Kam Ho. Probabilistic radio-frequency fingerprinting and localization on the run. *Bell Labs Technical Journal*, 18(4):111–133, 2014.

[30] P. Mirowski, P. Whiting, H. Steck, R. Palaniappan, M. MacDonald, D. Hartmann, and T. K. Ho. Probability kernel regression for WiFi localisation. *Journal of Location Based Services*, 6(2):81–100, 2012.

[31] R. Nandakumar, K. K. Chintalapudi, and V. N. Padmanabhan. Centaur: Locating devices in an office environment. In *Proc. ACM MobiCom*, pages 281–292, 2012.

[32] J.-g. Park, D. Curtis, S. Teller, and J. Ledlie. Implications of device diversity for organic localization. In *Proc. IEEE INFOCOM*, pages 3182–3190, Apr. 2011.

[33] G. Shen, Z. Chen, P. Zhang, T. Moscibroda, and Y. Zhang. Walkie-Markie: Indoor pathway mapping made easy. In *Proc. USENIX NSDI*, pages 85–98, 2013.

[34] H. Shin, Y. Chon, Y. Kim, and H. Cha. MRI: Model-based radio interpolation for indoor

war-walking. *IEEE Trans. Mobile Computing*, 14(6):1231–1244, June 2015.

[35] S. S. Skiena. *The Algorithm Design Manual*. Springer Science & Business Media, 2008.

[36] W. Sun, J. Liu, C. Wu, Z. Yang, X. Zhang, and Y. Liu. MoLoc: On distinguishing fingerprint twins. In *Proc. IEEE ICDCS*, pages 226–235, July 2013.

[37] S.-H. Tsai, S.-Y. Lau, and P. Huang. WSN-based real-time indoor location system at the Taipei World Trade Center: Implementation, deployment, measurement, and experience. In *Proc. IEEE Sensors*, pages 1–4, Oct 2012.

[38] H. Wang, S. Sen, A. Elgohary, M. Farid, M. Youssef, and R. R. Choudhury. No need to war-drive: Unsupervised indoor localization. In *Proc. ACM MobiSys*, pages 197–210, 2012.

[39] K. Wu, J. Xiao, Y. Yi, D. Chen, X. Luo, and L. M. Ni. CSI-based indoor localization. *IEEE Trans. Parallel and Distributed Systems*, 24(7):1300–1309, Jul. 2013.

[40] Z. Yang, Z. Wang, J. Zhang, C. Huang, and Q. Zhang. Wearables can afford: Light-weight indoor positioning with visible light. In *Proc. ACM MobiSys*, pages 317–330, 2015.

[41] M. Youssef and A. Agrawala. Handling samples correlation in the Horus system. In *Proc. IEEE INFOCOM*, pages 1023–1031, 2004.

[42] M. Youssef and A. Agrawala. The Horus location determination system. *Wireless Networks*, 14(3):357–374, 2008.

[43] D. Zhou and T.-H. Lai. A compatible and scalable clock synchronization protocol in IEEE 802.11 ad hoc networks. In *Proc. ICPP*, pages 295–302, 2005.