SLAC: Calibration-Free Pedometer-Fingerprint Fusion for Indoor Localization

Suining He, S.-H. Gary Chan, *Senior Member, IEEE*, Lei Yu, and Ning Liu

Abstract—To improve the accuracy of fingerprint-based localization, one may fuse step counter with fingerprints. However, the walking step model may vary among people. Such user heterogeneity may lead to measurement error in walking distance. Previous works often require a step counter tediously calibrated offline or through explicit user input. Besides, as device heterogeneity may introduce various signal readings, these studies often need to calibrate the fingerprint RSSI model. Many of them have not addressed how to jointly calibrate the above heterogeneities and locate the user. We propose **SLAC**, a novel system which **s**imultaneously **l**ocalizes the user and **c**alibrates the sensors. SLAC works transparently, and is calibration-free with heterogeneous devices and users. Its novel formulation is embedded with sensor calibration, where location estimations, fingerprint signals, and walking motion are jointly optimized with resultant consistent and correct model parameters. To reduce the localization search scope, SLAC first maps the target to a coarse region (say, floor) via stacked denoising autoencoders and then executes the fine-grained localization. Extensive experimental trials at our campus and the international airport further confirm that SLAC accommodates device and user heterogeneity, and outperforms other state-of-the-art fingerprint-based and fusion algorithms by lower localization errors (often by more than 30 percent).

Index Terms—Indoor localization, joint optimization, device RSSI dependency, step counter calibration, fingerprinting, calibration-free fusion, walk detection, area identification, stacked denoising autoencoders

1 Introduction

Indoor Location-Based Service (LBS) has attracted wide attention in recent years due to its social and commercial values, with total revenue predicted to worth 10 billion US dollars by 2,020. The service quality of indoor LBS largely depends on the localization accuracy of users. Among all the current indoor localization techniques, fingerprinting [1] emerges as a well-known one, as it does not assume line-of-sight measurement and is adaptive to indoor environment without deployment of extra infrastructure.

Fingerprint-based indoor localization is usually conducted in two phases: *offline* survey phase and *online* query phase. In the offline phase, a site survey is conducted to collect the vectors of received signal strength indicators (RSSIs) from access points (APs) at the reference points (RPs) with known locations. In the offline phase, given a query with received signal strength indicator (RSSI) measurement, a target (in this paper, we use “user”, “client” and “target” interchangeably) obtains her or his location with the closely-matched signals in the database.

With the advance in smartphone sensors of accelerometers, magnetometers and gyroscopes, fusing motion with RF fingerprint has been recently studied to improve the localization accuracy. While fusing step counter (or pedometer) and fingerprinting has shown to be promising, many practical issues remain to be addressed. Among them, a critical one is system calibration for both devices and users. *Device heterogeneity* arises when different mobile devices are used to measure the same RF signal [2]. As their RSSI may not agree with each other, the RSSI difference needs to be calibrated, traditionally by offline training. *User heterogeneity* occurs when the motion sensors for different users need to be calibrated with different parameters in system operation. In step counter, the user stride length (the distance between the front foot heel and the back foot heel where they are farthest apart) is different for different users, and is related to some stride frequency model [3]. Traditional localization techniques based on RF fingerprint and step counter fusion often require explicit input of stride model parameters, or tedious intrusive training offline.

We show in Fig. 1a the traditional localization approach fusing step counter measurement and RF fingerprints. The device first needs to be calibrated to align the RSSI measurement with the fingerprint signal. The step counter measures the change in user motion with step frequency and count as output. Based on a step length model, the user displacement may be estimated by summing the stride length over all the steps. Through some probabilistic inference between RF signals and walking displacement [4], [5], the system estimates the current user location. External parameter calibrations are needed in both the device dependency model and the step length model.

Observe that when a user walks, her/his spatial location, RF signals received (as measured by the mobile device), and displacement (as measured by the step counter as stride model) are *correlated* with the walking trajectory. Specifically, distance between location estimations by fusion should be *consistent* with the measured walking displacement. Such correlation or consistency can be utilized to calculate the

- S. He and S.-H. G. Chan are with the Department of Computer Science and Engineering, Hong Kong University of Science and Technology, Kowloon, Hong Kong SAR, China. E-mail: {sheaa, gchan}@cse.ust.hk.
- L. Yu and N. Liu are with School of Data and Computer Science, Sun Yat-sen University, Guangzhou 510275, China.
E-mail: yulei5@mail2.sysu.edu.cn, liuning2@mail.sysu.edu.cn.

*Manuscript received 13 Oct. 2016; revised 8 May 2017; accepted 19 Sept. 2017. Date of publication 27 Sept. 2017; date of current version 2 Apr. 2018.
(Corresponding author: Ning Liu.)
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TMC.2017.2757023*

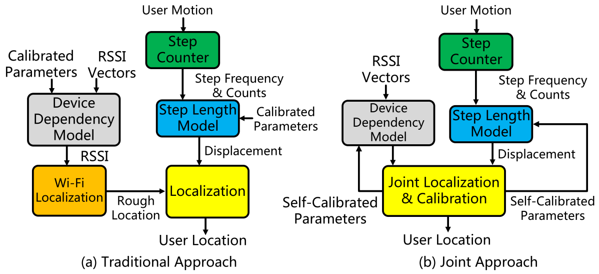

Fig. 1. Comparison between traditional and proposed fusion schemes: (a) traditional approach and (b) joint localization and calibration approach.

target location given the stored fingerprint map, leading to higher localization accuracy. Furthermore, the estimated target locations can be in turn used to calibrate the model parameters for heterogeneity at both device and user.

Armed with the above observations, we consider simultaneously localizing the target and calibrating above heterogeneities transparently without explicit user inputs, i.e., a *calibration-free* approach. As shown in Fig. 1b, the correlations between signals and walking displacements are jointly considered in the fusion algorithm. The consistency requirement between spatial measures (namely RF signals received at the users) and temporal measures (namely step frequency and counts reported by the step counter) is used as constraint to self-calibrate the sensor parameters for both the device dependency and step length models. Using such an approach, signal noise from each sensor can be mitigated, while the system can achieve high localization accuracy and calibration efficiency without explicit user participation.

We propose *SLAC* (*S*imultaneous *L*ocalization *A*nd *C*alibration), a novel system achieving fusion of step counter and RF fingerprints regardless of aforementioned device and user heterogeneities. Fig. 2 shows the work flow of SLAC system. In the *offline* phase, SLAC is initiated with a site survey, storing the RF fingerprints of reference points (RPs) into the database. In the *online* phase, the target (client) collects Wi-Fi RSSIs and measures user walking steps. In order to reduce the location search scope, the target is first mapped to a small area by the *area identification* module. After that, SLAC finds the fingerprint matching pattern via *signal difference calculation*. The server then fuses a step length model and stride frequency with the RF signals received, and conducts *convex-optimization localization & sensor calibration* to locate the user, and meanwhile adapts the parameters in the step length model. Given the estimated locations, it also finds the RSSI difference and addresses the device heterogeneity. Note that the entire calibration process is transparent to the client user.

The major contributions of SLAC are as follows:

- *Simultaneous Localization and Calibration:* SLAC is a novel system which achieves indoor localization and transparent sensor calibration simultaneously. SLAC learns the parameters in the user step model, and meanwhile calibrates RSSI measurements due to heterogeneous devices. To our best knowledge, SLAC is the first localization approach which can jointly and transparently achieve accurate location estimations, calibrate device dependency in RSSI and adapt to user heterogeneity in motion without their explicit participation.
- *Joint Optimization with Parameter Learning:* We propose a novel optimization-based localization and sensor calibration formulation. In such formulation, SLAC leverages the spatial-temporal correlation in target locations, RSSI signals and step counts. It jointly considers the correlation between location estimations and different sensors via a single optimization formulation. By solving a convex optimization problem, we accurately estimate the location of the walking target. Sequence of estimated locations on the trajectory are also used to calibrate the device RSSI, as considered in Fig. 1b, and find the system parameters with the best fit.

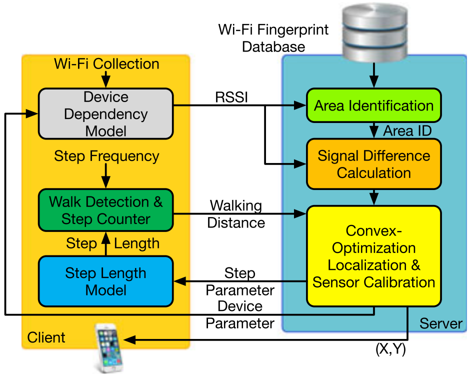

Fig. 2. The system work flow of our proposed SLAC.

- *Area Identification for Computational Efficiency:* To further improve the computational efficiency of SLAC, we propose in this work a highly accurate and efficient area identification based on a deep learning approach (*stacked denoising autoencoders* or *SDA*). Through the deep learning, the target can be first mapped to a small area (say, a small region or a floor). Then, based on the given RPs on that area, we execute SLAC with much less search computation. We have integrated it with SLAC, and the experimental trials have shown that SDA can achieve high detection accuracy and improves computational efficiency of SLAC.

We have implemented SLAC on Android platforms and conducted large-scale experiment in the Hong Kong International Airport (HKIA) and the Hong Kong University of Science and Technology (HKUST). Experimental results show that our scheme outperforms other fingerprint-based and fusion approaches (including [4], [6], [7], [8], [9]) in the localization accuracy (often by more than 30 percent), with transparent calibration of step counter (user heterogeneity) and fingerprint signals (device heterogeneity). Note that though our prototype study is on Wi-Fi fingerprints, SLAC is general enough to be applied with other emerging fingerprint signals such as RFID, channel state information (CSI) or image.

The rest of this paper is organized as follows. After reviewing the related work in Section 2, we discuss in Section 3 the preliminaries of SLAC, including the fingerprint and step measurement. Then we present the formulation of joint localization and sensor calibration in Section 4. In Section 5 we propose an area identification scheme for efficiency improvement. After that, we present in Section 6 its experimental results at two sites, and conclude in Section 7.

2 Related Work

We briefly review the related work as follows. Due to pervasive deployment based upon the existing WLAN [10], Wi-Fi fingerprinting techniques, including RADAR [6], Horus [11] and EZPerfect [2] have been widely studied in recent years. While promising results have been achieved, the above works mainly focus on Wi-Fi fingerprinting. SLAC, on the other hand, fuses user motion with the RF fingerprints and therefore can achieve much better localization accuracy. Many recent studies have proposed crowdsourcing [12], deploying external infrastructures [13] and motion-based trajectory mapping to reduce cost in RF fingerprint collection and signal map update, which are orthogonal and amendable to our studies here for more efficient deployment.

To fuse step counter and RF fingerprints, many recent LBS works like Markov model [5], [14], [15], signal pattern matching [16], conditional random field [17], SLAM [18] and particle filter [4], [8] have been studied. However, these works usually rely on specific probabilistic assumption between different noisy sensors and may work the best under narrow or constrained environment. Our scheme, in contrast, jointly utilizes the RF signals and motion sensors in a single optimization formulation, and thus can achieve higher accuracy and robustness under noisy measurements. Furthermore, our approach is more adaptive to different environments, including large open space (like the airport) or narrow corridors (like the office building).

Note that our study here is also orthogonal to the SLAM-based schemes [18], [19]. SLAM focuses on locating the target without explicit indoor map and fingerprint information. Unlike these traditional SLAM works, SLAC considers simultaneous target localization and automatic sensor calibration given above information. Existing SLAM systems such as UnLoc [19] and SemanticLoc [20] can serve as the initial map and RF fingerprint providers for our studies.

In addition, the works based on step counter and Wi-Fi fusion above often consider a pre-calibrated step counter for displacement estimation [19]. Some offline step length calibration methods on step counter have been proposed. However, they either require offline calibration [8], [21], or deploy external sensor infrastructures for walking distance estimation [22], [23]. Our work, in contrast, adaptively fuses available RF fingerprints for online step counter calibration without explicit user input, and hence can be integrated with existing smartphone-based indoor localization systems [6], [24]. Compared with R-PDR [25] which focuses on stride estimation in pedestrian dead reckoning based on life-long learning and map constraints, SLAC focuses on joint optimization with RF-fingerprint and pedometer for step length and fingerprint signal calibration, showing marked difference in terms of objectives and approaches.

Device dependency in wireless signal measurement has been studied in recent years [2], [26], [27]. The work in [26] considers a linear model to calibrate the RSSI signals offline. Online signal adaptation [2], [27] has been recently studied to facilitate the calibration. However, they are solely based on the measured RSSI and have not fully taken advantage of the correlation between fingerprint signals, locations and motion information. To our best knowledge, SLAC is the first approach to utilize step counter fusion to jointly calibrate the device RSSI, achieving much higher adaptability.

Area identification has attracted much attention recently [28], [29] in order to determine the target floor or region. SkyLoc [30] utilizes RSSIs from cell towers to locate the floor where the target is on skyscrapers. It searches against the RPs in different floors, and then finds the floor with RPs of minimum signal difference [31], which, however, is computationally heavy in practice. A more recent work, Locus [28], finds the floor where the strong signals match those in the target RSSI vector. However, due to signal fluctuation, such a heuristic may not always hold and hence estimation error exists [32]. Unlike the tedious nearest neighbor search, our proposed stacked denoising autoencoders (SDA) approach is highly efficient in area mapping and facilitates the entire localization computation. Via the offline training, the SDA scheme learns signal features to differentiate the RF fingerprints, and identifies the area with much better accuracy.

Using barometer in floor localization has attracted much attention recently [29], [33]. However, barometer readings may still suffer from thermal change and calibration efforts. Orthogonal to their studies, our work here focuses on the existing WLAN network and can be easily integrated with them for more advanced applications.

An initial study of SLAC has been conducted in [34]. This paper further advances from it in following aspects: 1) Previous SLAC considers using particle filter to address the device dependency, which is computationally expensive. In this version, we reformulate SLAC into a single joint optimization framework, which efficiently calibrates both the step and device parameters without the particle filter. The computation overhead of SLAC is significantly reduced compared with the previous particle filter version. 2) Previous SLAC has not yet considered the area identification for large-scale deployment. In the complex multi-area environment (say, multi-floor building), the overall location search scope can be large. Therefore, we propose area identification based on deep learning (stacked denoising autoencoders, or SDA). Based on the highly accurate and efficient SDA, we narrow down the search scope of localization and improve the computation efficiency of SLAC in practical deployment. 3) Previous version only considers accelerometer and gyroscope to detect step motion. However, abnormal user behavior including shaking the smartphone may lead to false alarm in detection. In the new version, we further improve the walking detection accuracy of SLAC by fusing the accelerometer, gyroscope and magnetometer. The joint decision of the sensors achieves better robustness against irregular motions. 4) We have further conducted more experimental studies on performance of the joint localization and sensor calibration, fusion-based motion detection and efficient area identification to validate their effectiveness.

3 Preliminaries of SLAC

Table 1 lists the important symbols in SLAC formulation. In this section, we present the preliminaries of SLAC. We first discuss in Section 3.1 the fingerprint measurement, followed by the device dependency of the RSSI in Section 3.2. Then we discuss the motion estimations in Section 3.3 and the user heterogeneity in user step length in Section 3.4.

3.1 Fingerprint RSSI Collection

In the offline phase of fingerprint-based localization, a site survey is conducted on overall Q reference points (RPs). Let $\mathbf{r}_q$ be the 2-D position of RP q, and $\mathbf{R} = [\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_Q]$ be a 2-by-Q matrix indicating the RP positions. Let $\mathcal{L}$ be the set of L Wi-Fi access points (APs) that cover the site.

At each RP, multiple Wi-Fi RSSI samples are collected to reduce measurement uncertainty. Denote the RSSI (in dBm) at RP q from AP l at time t as $\psi_q^l(t)$, $1 \leq t \leq T_q^l$ ($T_q^l > 1$), with T_q^l being the total number of samples collected. Let $\bar{\psi}_q^l$ be the average RSS reading over time domain for AP l, $l \in \mathcal{L}$, at RP q, and $(\sigma_q^l)^2$ be the unbiased estimate of variance in RSS time samples for AP l at RP q. Then for each RP, the mean RSSI is computed as

TABLE 1
Major Symbols Used in SLAC

Notation	Definition
$\widehat{\mathbf{x}}_m$	Estimated 2-D coordinate of the target user
δ_{mn}	Distance between temporal targets m and n (m)
$\mathcal{S}_c$	Step length at the cth step (m)
f_c	Step frequency at the cth step (Hz)
$\mathbf{r}_q$	2-D coordinate of the reference point (RP) q
Q & L	Number of RPs & APs in fingerprint database
ω_{mq}	Weight of RP q in locating the temporal target m
$\mathbf{W}_b$	Vector of indicators determining the device offsets
$\mathbf{B}$	Vector of potential device RSSI offset (dB)
ψ_q^l	RSSI at RP q from AP l (dBm)
ϕ_m^l	RSSI at the temporal target m from AP l (dBm)
ψ_q & ϕ_m	RSSI vector at RP q & at the temporal target m
$\bar{\psi}_q^l$	RSSI for AP l at RP q (dBm)
T_q^l	Number of RSSI measurements at q for AP l
σ_q^l	RSSI standard deviation at RP q for AP l (dB)
$\widehat{b}$	Parameters in the linear RSSI model between devices in the offline and online phases
$[\alpha, \beta]$	Parameters in linear step length model

$$\bar{\psi}_q^l = \frac{1}{T_q^l}\left(\sum_{t=1}^{T_q^l} \psi_q^l(t)\right), \tag{1}$$

and the corresponding RSSI variance is given by

$$\left(\sigma_q^l\right)^2 = \frac{1}{T_q^l - 1}\left(\sum_{t=1}^{T_q^l}\left(\psi_q^l(t) - \bar{\psi}_q^l\right)^2\right). \tag{2}$$

Then the Wi-Fi RSSI vector at RP q is given by

$$\psi_q = \left[\bar{\psi}_q^1, \bar{\psi}_q^2, \ldots, \bar{\psi}_q^L\right], \quad q \in \{1, 2, \ldots, Q\}. \tag{3}$$

In the online phase, the mobile device continuously measures the Wi-Fi RSSI vectors as the user walks. These vectors form the *temporal targets* with locations to be estimated. We consider a sliding window of M temporal targets for SLAC, and the Mth one is the latest measurement. Let ϕ_m^l be the RSSI value at target m $(1 \le m \le M)$ from Wi-Fi AP l, $l \in \mathcal{L}$. Similar to the RP RSSI vector, we define the sampled RSSI vector at the mth temporal target $(1 \le m \le M)$ as

$$\phi_m = \left[\phi_m^1, \phi_m^2, \ldots, \phi_m^L\right]. \tag{4}$$

By definition, if an RP or target location cannot detect signals from AP l, $\bar{\psi}_q^l = 0$ and $\sigma_q^l = 0$ (or $\phi_m^l = 0$).

3.2 Device Dependency in RSSI

Due to the difference in Wi-Fi network interfaces, for the same RF signal, different types of smartphones may have different reading values [27]. To illustrate this, we conduct an experiment and collect $1{,}000$ RSSI samples using HTC One X and Lenovo A680, respectively. Fig. 3 shows the typical linear shift between the signals of the two smartphones. Note that our work focuses on the joint optimization framework, while here we consider the linear offset model for concreteness and ease of properties. Other more advanced device dependency models can be adapted to our framework, and considered in our future work.

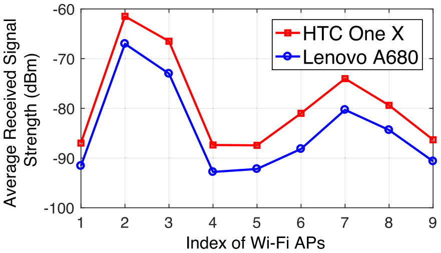

Fig. 3. RSSI vectors collected from HTC One X and lenovo A680.

To address such device dependency, there have been a lot of models used for calibration [2]. In this paper, for concreteness we implement a linear signal model to adjust the device heterogeneity. Given an RSSI ϕ_m^l (in dBm) at target m, we find the corresponding *device parameters*, denoted as k $(k > 0)$ and b, in order to satisfy $\widetilde{\phi}_m^l = k\phi_m^l + b$. Calibrated $\widetilde{\phi}_m^l$ is then utilized for signal difference calculation.

In our deployment, we have observed that k is close to 1 in practice. We have also investigated the offset device dependency within other smartphones. Similar test is conducted on several other smartphone models like Xiaomi Note 4, Samsung Note 4, S4 and Google Nexus 5. We also observe similar properties, i.e., compared with Xiaomi Note 4, the linear RSSI offsets b's of Note 4, S4 and Nexus 5 are -8.155 dB, 3.326 dB and 6.318 dB respectively, while their k's are very close to 1. Such offset observations also confirm those discussed in recent works like [2], [35], [36]. Therefore, in our formulation and experiment, we consider focusing on finding the parameter b, i.e., calibrating the model

$$\widetilde{\phi}_m^l = \phi_m^l + b, \tag{5}$$

which adapts to the device difference. Though our paper uses a linear model in device calibration for concreteness, SLAC is compatible to the previous linear model used in [34] and other more advanced models, e.g., [36].

3.3 Walk Detection and Step Counter

Besides the RF signals, the mobile device also measures the step and walking displacement of the user via the inertial sensors. A step counter usually has two modules: walking detection and step counting [37]. Walk detection classifies the current state of the target. If a user is identified as moving, the step counting measures her/his displacement with step counts and stride length.

We have studied empirically and implemented a more robust walk detection scheme compared with previous version in [34]. Beyond combining the accelerometer and gyroscope [34], it leverages the observation that magnitude of magnetic readings changes significantly when the user is walking indoors, due to the magnetic field diversity at different locations. Therefore, if drastic change in magnetometer readings is not detected, we may conclude the user has not yet move from her/his current location. By fusing decisions from accelerometer, gyroscope and magnetometer, we can reduce the influence from irregular motion (say, false detection due to shaking or rotating smartphones) and improve the walk detection accuracy.

Specifically, we denote the magnitude of linear acceleration, rotation rate and magnetic field as a_t (m/s^2), ρ_t (rad/s) and m_t (μT), respectively. Given a sliding window of $\mathcal{W}$ measured values, we consider the average magnitude of linear acceleration, denoted as h_a, the standard deviation of angular velocity, denoted as h_ρ [37], and the standard deviation of magnetic field magnitude, denoted as h_m, for motion detection

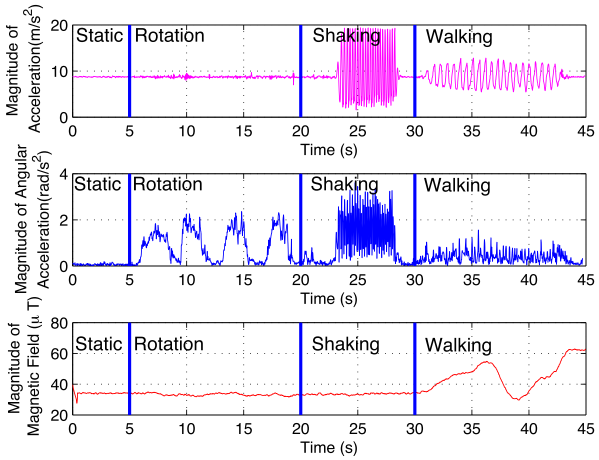

Fig. 4. Walk detection via joint decision of linear accelerometer, gyroscope, and magnetometer. All readings are measured at the same time.

$$h_a = \frac{\sum_{t=1}^{\mathcal{W}} a_t}{\mathcal{W}},$$
$$h_\rho = \frac{1}{\mathcal{W}}\sum_{t=1}^{\mathcal{W}}\left(\rho_t - \frac{\sum_{t=1}^{\mathcal{W}}\rho_t}{\mathcal{W}}\right)^2, \tag{6}$$
$$h_{\mathrm{m}} = \frac{1}{\mathcal{W}}\sum_{t=1}^{\mathcal{W}}\left(\mathrm{m}_t - \frac{\sum_{t=1}^{\mathcal{W}}\mathrm{m}_t}{\mathcal{W}}\right)^2.$$

If some or all of h_a, h_ρ and h_{m} are below certain thresholds, then the user is classified as static (not walking) [37]. Otherwise, if these measurements are all above certain values, the user is identified as moving and the step counter begins to measure the walking steps.

Fig. 4 shows the walk detection given different user behaviors. When the user is unexpectedly rotating or shaking her/his smartphone, we may observe marked fluctuation in accelerometer and gyroscope readings, which may also trigger walk detection like normal walking given certain reading threshold. Simply considering accelerometer and gyroscope readings may lead to incorrect decision, while combination of the three sensors, including the magnetometer, can filter the wrong one. However, by combining three sensors (right hand side), the walk detection can be better than simply using accelerometer and gyroscope readings [34] (further experimental evaluation is presented in Section 6).

Given the decision that the user is walking, we implement the step counting algorithm in [9], [21]. A repetitive step pattern in the accelerometer is discovered through the normalized autocorrelation [9]. Specifically, we calculate the normalized autocorrelation (NAC) given the time lag $\mathcal{W}$ and the sth accelerometer sample, i.e.,

$$\chi(s,\mathcal{W}) = \frac{\sum_{t=0}^{\mathcal{W}-1}\begin{bmatrix} a_{s+t}-\mu(s,\mathcal{W}) \\ a_{s+t+\mathcal{W}}-\mu(s+\mathcal{W},\mathcal{W})\end{bmatrix}}{\mathcal{W}\sigma(s,\mathcal{W})\sigma(s+\mathcal{W},\mathcal{W})}. \tag{7}$$

If the NAC $\chi(s,\mathcal{W})$ is above a predefined threshold, we count it as one step detection.

Note that the accelerometer readings and their thresholds are highly correlated with the gravity reading of the smartphone [37]. We have observed that different devices may show different gravity readings due to imperfection of manufacture, which shows a constant offset. Table 2 summarizes the offsets for several existing COTS smartphones.

TABLE 2
Measured Gravity Readings (m/s^2) of Different Smartphones

Smartphone Type	Gravity Value (m/s^2)
Samsung S3	8.7
Galaxy TabPro	10.6
HTC One X	9.8
Google Nexus 5	9.2

To approach such heterogeneity, we conduct self gravity calibration when the smartphone is static or left being charged with a laptop. Specifically, when the standard deviations of accelerometer, gyroscope and magnetometer are all sufficiently small, we may say that the smartphone is static (say, placed on the table), and the mean of the gravity readings inside a sliding window is then used as the calibrated one. Then we can adjust the thresholds in walking detection automatically by subtracting the offset between different smartphones. Based on the calibrated offsets, we can then calibrate gravity reading inside the devices.

3.4 User Heterogeneity in Step Length

Based on step detection and counting, we estimate the walking distance by multiplying number of steps with stride length. For ease of prototyping, we implement the linear model for step length estimation [8], [22]. Denote the step length at the cth step as $\mathcal{S}_c$ (m) and the corresponding step frequency (the inverse of time taken by a user to move a stride) as f_c (Hz). The linear relationship between them is

$$\mathcal{S}_c = \alpha f_c + \beta, \tag{8}$$

where the *step parameters*, α ($\alpha > 0$) and β, are user dependent. Then given C_{mn} steps between two locations (temporal targets) m and n with Wi-Fi measurements, the walking displacement (distance) is given by

$$\delta_{mn} = \sum_{c=1}^{C_{mn}} \mathcal{S}_c. \tag{9}$$

Note that our calibration is independent of step length model and can be generic enough to apply in more sophisticated ones [38]. To summarize, our goal in SLAC is to locate the target user given sets of signals, $\{\psi_q\}$ and $\{\phi_m\}$, while calibrating b in RSSI and $[\alpha, \beta]$ in user profile transparently without explicit user inputs.

4 Formulation of Simultaneous Localization & Sensor Calibration

Given above preliminaries in fingerprint and pedometer measurement, we present in this section how we formulate a novel joint optimization and parameter learning problem for localization and calibration in SLAC. We first present how we compare the signal difference with the fingerprint map (Section 4.1). Then we provide formulation of location estimation and joint optimization (Section 4.2). Given the formulation, we discuss the complexity analysis (Section 4.3).

4.1 Signal Difference Calculation

In SLAC formulation, we first need to characterize the difference between the target signals and those stored in the fingerprint database. In other words, the smaller the signal difference is, the better matched the target signal is with the fingerprint map. As the devices may have different RSSI reading offsets (reflected in Equation (5)), in our formulation, the signal difference function also considers the device parameters in order to jointly calibrate them with the optimization.

Specifically, in the joint RSSI calibration, we are to find the device parameter within a potential search interval, i.e.,

$$b_{min} \leq b \leq b_{max}, \tag{10}$$

where the two bounds of the offset parameter can be obtained through empirical studies [3]. During the joint optimization, SLAC estimates the most suitable b for the device calibration. In order to facilitate the optimization calculation, we discretize the search space within $[b_{min}, b_{max}]$ into a set of K_b potential offset parameters. In other words, we define the set of potential offsets by a vector $\mathbf{B}$

$$\mathbf{B} = [b_{min}, b_{min} + g, \ldots, b_{max} - g, b_{max}]^T, \tag{11}$$

where

$$g = \frac{b_{max} - b_{min}}{K_b - 1}. \tag{12}$$

Here K_b determines the number of potential parameters as well as the granularity of calibration. We further define a vector $\mathbf{W}_b$ consisting of K_b indicator variables in order to represent the potential calibrated offsets, i.e.,

$$\mathbf{W}_b = [w_1, \ldots, w_K], \quad \widehat{b} = \mathbf{W}_b\mathbf{B} = \sum_{k=1}^{K_b} w_k \mathbf{B}[k], \tag{13}$$

where $0 \leq w_k \leq 1$ and $\sum_{k=1}^{K_b} w_k = 1$. Based on above discretization, the joint optimization of SLAC finds the indicator variables inside $\mathbf{W}_b$ in order to adaptively calibrate the device RSSI. It calculates the $\widehat{b}$ by the weighted average of potential offsets and returns the optimal one.

In signal vector comparison, we jointly consider device heterogeneity in Equation (5), and the measurement uncertainty in Equation (2). Let $\mathbf{S}_m$ be the number of APs at the mth temporal target ($0 < |\mathbf{S}_m| \leq L$). Given a target's Wi-Fi RSSI ϕ_m^l (constant) from AP $l \in \mathbf{S}_m$ and the parameter b, the expected signal difference between fingerprint at RP q and the target RSSI in AP l is defined as

$$\begin{aligned}
\Delta^l(\phi_m, \psi_q) &\triangleq \mathrm{E}\left(\left(\widetilde{\phi}_m^l - \psi_q^l\right)^2\right) \\
&= \mathrm{E}\left(\left(\widetilde{\phi}_m^l\right)^2 - 2\phi_m^l\psi_q^l + \left(\psi_q^l\right)^2\right) \\
&= \left(\widetilde{\phi}_m^l\right)^2 - 2\phi_m^l\mathrm{E}\left(\psi_q^l\right) + \mathrm{E}\left(\left(\psi_q^l\right)^2\right) \\
&= \left(\widetilde{\phi}_m^l\right)^2 - 2\phi_m^l\mathrm{E}\left(\psi_q^l\right) + \mathrm{E}^2\left(\psi_q^l\right) + \left(\sigma_q^l\right)^2 \\
&= \left(\phi_m^l + \mathbf{W}_b\mathbf{B} - \bar{\psi}_q^l\right)^2 + \left(\sigma_q^l\right)^2,
\end{aligned} \tag{14}$$

where, by definition, $\Delta^l(\phi_m, \psi_q) = 0$, if AP l is not detected at either or both sides. Thus, the overall expected signal difference between ϕ_m and ψ_q is given by

$$\Gamma(\phi_m, \psi_q) \triangleq \sum_{l=1}^{|\mathbf{S}_m|} \Delta^l(\phi_m, \psi_q). \tag{15}$$

If $|\mathbf{S}_m| = 0$, RP q is not considered in estimating target m.

4.2 Convex-Optimization Localization & Sensor Calibration

Given the signal difference function, we discuss how to estimate the target's potential location using Equation (15) and the measured distance constraints from motion sensors in Equation (8). We propose a novel joint optimization formulation such that the device and user heterogeneities are calibrated during the localization. At a high level, in SLAC the optimization formulation considers the location estimation as the weighted average of RPs, and estimates the most appropriate weight variables in order to maximize the fingerprint matching with the stored signal map, and minimize the walking distance difference with the measured ones.

We first present the location estimation formulation. In the localization process, we consider finding the *weight* assigned to each of the RPs to represent the target location. Each weight variable to be optimized captures the closeness between a target and this RP given RF fingerprints and walking distances. In other words, the higher the weight for an RP is, the more likely the target is nearby. Mathematically, let $\mathbf{V} = \{1, 2, \ldots, M\}$ be the time index of each temporal target in the sliding window, and $\widehat{\mathbf{x}}_m$, $m \in \mathbf{V}$, be the location for each of them to be estimated. The RPs in $\mathbf{R}$ are used to locate these target positions. Let ω_{mq} be the weight assigned to RP q when locating target m, and we have

$$\widehat{\mathbf{x}}_m = \sum_{q=1}^{Q} \omega_{mq}\mathbf{r}_q, \tag{16}$$

where the weights $\omega_{mq}, \forall m \in \mathbf{V}$, are constrained by

$$\sum_{q=1}^{Q} \omega_{mq} = 1, \quad \omega_{mq} \geq 0, \quad \forall q \in \{1, 2, \ldots, Q\}. \tag{17}$$

We then discuss the objective of SLAC formulation. SLAC searches against the fingerprint map to find the RPs which can both minimize the signal differences and satisfy the sequential distance measurements. In order to efficiently solve this problem, we propose a joint optimization formulation. Based on Equations (15) and (19), we first present as follows the objective function of SLAC.

Recall that the measured distance between $\widehat{\mathbf{x}}_m$ and $\widehat{\mathbf{x}}_n$ as δ_{mn} for each two sequential targets m, n. The constraints over the step model parameters are given by

$$\alpha_{min} \leq \alpha \leq \alpha_{max}, \quad \beta_{min} \leq \beta \leq \beta_{max}. \tag{18}$$

To jointly localize all the temporal targets in $\mathbf{V}$, we would like to find a set of locations $\widehat{\mathbf{x}}_1, \widehat{\mathbf{x}}_2, \ldots, \widehat{\mathbf{x}}_M \in \mathbb{R}^2$ in the survey site in order to satisfy two criteria:

- Criterion I. *Minimizing the walking distance error*: In SLAC optimization, the mutual distance between the sequential location estimations should minimize the difference with the walking distance measurement (δ_{mn}), i.e.,

$$\underset{\{\omega_{mq}\},[\alpha,\beta]}{\arg\min} \sum_{m=2}^{M} \left(\|\widehat{\mathbf{x}}_m - \widehat{\mathbf{x}}_n\|_2 - \delta_{mn}\right)^2, \tag{19}$$

where $n = m - 1$. In other words, this objective function requires the location estimations to *be consistent with their sequentially measured distances*. Step parameters in Equation (8) are then calibrated via the above consistency.

Due to dynamic changes in heading direction, holding gestures and other factors, readings from motion sensors often carry noise. In order to be robust towards such measurement fluctuation, we further implement the Berhu loss function [39] to replace the squared errors in Equation (19). Berhu loss function has been widely implemented for robust fitting [39], and is defined as follows. Given the difference z, the corresponding Berhu loss, denoted as $\mathcal{B}(z)$, is

$$\mathcal{B}(z) \triangleq \begin{cases} |z|, & |z| \leq \mathrm{T}; \\ \frac{z^2+T^2}{2T}, & |z| > \mathrm{T}. \end{cases} \tag{20}$$

Here T is a tunable parameter determining the penalty range. Berhu loss means that when the difference between $\|\widehat{\mathbf{x}}_m - \widehat{\mathbf{x}}_n\|_2$ and δ_{mn} is small, the penalty grows slowly so it can tolerate small measurement fluctuation. If the difference is large, Berhu loss assigns more penalty, leading to much less importance in location estimation.

- Criterion II. *Minimizing signal difference*: Furthermore, the SLAC optimization is to meanwhile maximize the *matching with fingerprint signal map* and *calibrate the RSSI device dependency parameter*. To achieve this, we consider finding the target locations which can minimize the signal difference with the fingerprint map. Mathematically, the weights assigned to RPs should minimize the weighted signal difference, i.e.,

$$\underset{\{\omega_{mq}\},\mathbf{W}_b}{\arg\min} \sum_{q=1}^{Q}\sum_{m=1}^{M} \omega_{mq}\Gamma(\phi_m, \psi_q). \tag{21}$$

In other words, the RPs with higher signal difference $\Gamma(\phi_m, \psi_q)$'s (better matched with the target signals) are assigned smaller weights ω_{mq}'s in the optimization, while those with lower signal difference are given more weights in location estimation.

Based on above two criteria, we finally formulate a *joint objective function* [39], i.e.,

$$\begin{aligned} \underset{\{\omega_{mq}\},[\alpha,\beta],\mathbf{W}_b}{\arg\min} \quad & \gamma \sum_{m=2}^{M} \mathcal{B}(\|\widehat{\mathbf{x}}_m - \widehat{\mathbf{x}}_n\|_2 - \delta_{mn}) \\ & + (1-\gamma)\sum_{q=1}^{Q}\sum_{m=1}^{M} \omega_{mq}\Gamma(\phi_m, \psi_q), \end{aligned} \tag{22}$$

where $\forall n = m-1$, and γ $(0 \leq \gamma \leq 1)$ represents the scaling factor (tradeoff) between the fingerprint signal difference and the pedestrian distance measurement error. In our experiment, we have empirically evaluated the parameter setting of γ and by default use $\gamma = 0.6$.

To summarize, SLAC formulates an optimization problem to find the target locations $\widehat{\mathbf{x}}_m$, the device parameters $\mathbf{W}_b$ and the step parameters $[\alpha, \beta]$, such that the overall walking distance estimation errors and the weighted fingerprint signal differences are jointly minimized, i.e.,

$$\begin{aligned} &\text{Objective : Equation (22),} \\ &\text{subject to : Equations (11), (13), (16), (17) and (18).} \end{aligned} \tag{23}$$

When the user launches the application and walks, the step counts and frequencies from the pedometer, and the RF RSSIs are fed to above localization framework. After the joint optimization and sensor calibration, the user location at time M (i.e., corresponding to the latest Wi-Fi measurement) is calculated as the weighted average of RP coordinates (Equation (16)), and returned to the client smartphone. Meanwhile, the calibrated device and the step parameters are fed to the Wi-Fi collector and step counter respectively to adapt the sensor readings.

Note that these parameters can be calibrated in a crowdsourcing manner. For example, the learned device parameters can be crowdsourced to the cloud. Other clients with the same device model may benefit from the calibration. The joint localization and sensor calibration can be executed at the initialization of ILBS system. In practice, if the calibrated parameters do not change significantly during each time of estimation, we may cease the calibration and conduct traditional fusion localization [34] with the learned parameters.

4.3 Computational Complexity Analysis

We briefly analyze the computational complexity here. Given Q RPs and L APs, the computation of signal difference calculation takes $\mathcal{O}(QL)$ in total. Note that usually the number of temporal targets M (size of sliding window) is small. To summarize, solving the optimization for each single user takes $\mathcal{O}(Q^3M^3)$ [39] on the server side, which is much lower than previous fusion with particle filter [34]. Many existing commercial optimizers (such as CVXOPT [40] and JOptimizer [41]) can be easily applied here.

Further computation reduction can be conducted by AP filtering and RP cluster mapping to reduce the number of APs and RPs. By filtering those APs which do not sufficiently differentiate the RPs (reducing L), we can reduce the time in signal difference calculation. The target location can be first mapped to a small region of the floor plan. Then Q is significantly reduced and computation of SLAC decreases. Besides, the search scope of next target location given each new pair of Wi-Fi measurement and walking distance can be restricted to the neighborhood of previous estimated locations (i.e., $\widehat{\mathbf{x}}_M$ in the earlier sliding window). In this way, the localization time is not high in practice. In the following Section 5, we will further present how to reduce the search scope and computation overhead.

5 Area Identification for Large Site

To further enhance the computational efficiency of SLAC, we propose in this section a novel area identification scheme based on a deep learning algorithm. The indoor map is first partitioned into several areas. Then the deep learning approach is applied to learn the relationship between the RF fingerprints and the area labels (say, the area id). In the online phase, given a measured RSSI vector, the target location is first mapped to a specific region via an identification algorithm. Then the core algorithm of SLAC finds the exact locations of the target within that area. In this way, the entire search scope is significantly reduced, and hence the computation time for a location query is much shortened.

Note that before area identification, we first preprocess the indoor map during offline phase by partitioning the RPs into multiple areas. The indoor area partitioning can be conducted via some clustering algorithms like k-means [42] or affinity propagation [43]. Besides RP clustering, the indoor site can be also partitioned based on the building structures (floors, rooms, etc.) by nature. Then the fingerprint database stores the id of RPs according to their corresponding areas. In this section, we first discuss the formulation of stacked denoising autoencoders in Section 5.1. Then we provide the complexity analysis of SDA in Section 5.2.

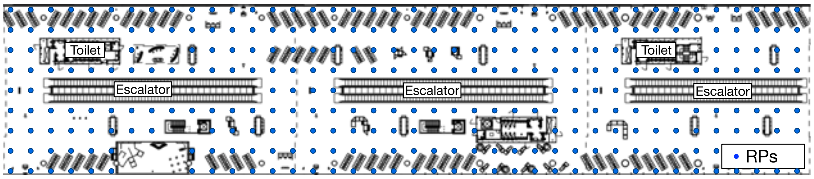

Fig. 5. The map of a boarding area (200 × 40 m^2 at each floor) in HKIA. The site survey density is 5 m.

5.1 Stacked Denoising Autoencoders

Stacked Denoising Autoencoders come from a deep network scheme which stacks multiple denoising autoencoder together to learn complicated features [44]. Each denoising autoencoder consists of three layers: input, hidden and output layers. The hidden layer and the output layer are called the encoding layer and the decoding layer, respectively. Given the input RSSI vector $\phi \in \mathcal{R}^L$, the encoding layer of denoising autoencoder first transforms it by

$$\mathbf{y} = f_e(\mathbf{W}\phi + \mathbf{b}), \tag{24}$$

where $f_e(\cdot)$ is the activation function, **W** is the input-to-hidden weights and b denotes the bias. Let $\mathbf{z} \in \mathcal{R}^L$ be the final output of the denoising encoder, given by a mapping function of

$$\mathbf{z} = f_d(\mathbf{W}'\mathbf{y} + \mathbf{b}'), \tag{25}$$

where $f_d(\cdot)$ is the decoding function ($\mathbf{W}'$ and $\mathbf{b}'$ are the corresponding weights and offsets). The objective of the denoising autoencoder is to reconstruct the input ϕ by output **z**. By minimizing the cost function, the denoising autoencoder learns the features of the signals.

In training the SDA, a certain number of input components are randomly chosen and forced to be zero [45]. The autoencoders are trained to fill in these blanks and reconstruct these corrupted inputs. In this so-called "denoising" way, we can reduce the influence from some unimportant or noisy components, and focus on retrieving the information we need, or the so-called "useful features". This is especially important for RF fingerprint-based area identification, when some of the input signals may be corrupted by the noise. Then in the stacked structure, once the first k layers are trained, the SDA scheme then leverages the outputs to train the $(k+1)$th layer. After some fine-tuning of the layers [45], we obtain the stacked denoising autoencoders.

5.2 Computational Complexity Analysis

A greedy approach has been implemented in order to train the SDA one layer at a time to reduce the offline learning time [45]. For example, the offline network training for an international airport (HKIA, totally 1,849 APs) and the university campus (HKUST, overall 2,263 APs) takes 1.06 h and 7.34 h, respectively (PC with i7 3610QM CPU). Further computation time reduction can be achieved through GPU parallel computing [46]. Once the SDA are trained, we can simply utilize the generated neural network to classify the RF fingerprints, which takes $\mathcal{O}(\#\text{Layers} \times L)$ given L APs. In the implementation, we can store the weights trained through the SDA in SLAC, and then the areas can be identified efficiently. By our area mapping, given $\mathcal{O}(C)$ partitioned areas, the computation of SLAC can be approximately reduced by factor of $\mathcal{O}(C^3)$, and final search scope with joint optimization can be significantly reduced.

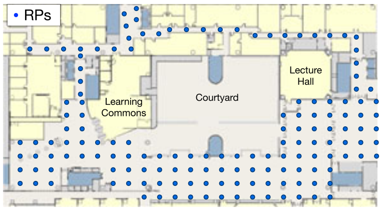

Fig. 6. A campus atrium map (100 × 50 m^2 at each floor) at the 2nd floor of HKUST.

6 Experimental Evaluations

We evaluate the SLAC prototype in the Hong Kong International Airport and campus area of our HKUST. Note that the international airport has large open space with high degree of freedom for pedestrians, while the university campus office environment includes wall and pillar partitions where the walking trajectories are more constrained and fingerprint signals are more differentiable. Therefore, the localization algorithms may experience different signal properties in these two sites, and we can correspondingly evaluate their adaptivity to the environment.

We first present the experimental settings and comparison schemes in Section 6.1. Then we present the performance of location estimation, sensor calibration and SDA-based area identification in Sections 6.2, 6.3, and Section 6.4.

6.1 Experimental Settings and Comparison Schemes

We have implemented SLAC on the Android platforms. We have conducted extensive site survey in HKIA (5,500 RPs in 3 floors) and our HKUST campus (18,000 RPs in 13 floors). The site survey is conducted in each site for over a day. Figs. 5 and 6 show the maps of their survey floor plans (8,000 m^2 and 5,000 m^2), respectively. At each RP, we take totally 80 Wi-Fi RSSI vectors using HTC One X (each sample takes 1 second). A quarter of these samples are collected when we are facing north, south, west and east, respectively. The grid size of site survey is 5 m.

During the testing phase, users collect the testing (target) data, including RSSI and INS during walking, and explicitly record the ground-truth when they pass by landmarks (like pillars, windows or doors). Locations without explicit landmarks upon them are labeled based on neighboring landmark interpolation/extrapolation and accurate CAD map information provided by site owners. Time stamps of the readings are also recorded during testing. Smartphones are held in front of the users (like internet browsing and viewing the indoor LBS map) during walking, as it is often the traditional gesture for indoor navigation service.

We have conducted trials on 5 volunteer users with different heights and weights. In the experimental trial of the international airport, we use HTC One X during our site survey and Lenovo A680 as our target devices (we use two device models for concreteness and ease of exposition only). In the trial at the campus atrium, we utilize the HTC One X as survey devices. Then in the target estimation, we use

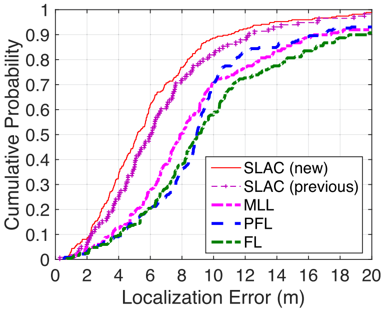

Fig. 7. CDF of localization errors using different schemes (HKIA).

Google Nexus 5 and Lenovo A680. We use these two smartphone models for ease of deployment, but our studies can be easily applied on other device models.

We compare the performance of SLAC with the following state-of-the-art localization and fusion schemes:

- *Fingerprint-based Localization* (*FL*), the classical location estimation algorithm [6], [7] which evaluates the Euclidean distance of each target RSSI vector [24] with the fingerprints at RPs and finds the top K nearest neighbors in signal space for location estimation.
- *Maximum-Likelihood-based Localization* (*MLL*), a recent positioning scheme which considers sequential probability along the walking trajectory [14]. Assuming conditional independence between sequential sensor measurements, MLL calculates the product of probability obtained through Wi-Fi and motion estimation [5], [14]. Then it finds the location with the maximum likelihood [17] as the target position.
- *Particle-Filter-based Localization* (*PFL*), a typical fusion algorithm [4], [8], [9] based on particle filter, which fuses walking distance and Wi-Fi fingerprints. The weights of particles are updated according to the Wi-Fi location estimation and the walking path [4]. Then these particles are resampled according to their weights and map constraints [9].

In our localization evaluation, we also compare our "SLAC (new)" with the previous version [34], denoted as "*SLAC (previous)*". Parameter settings of SLAC (previous) follow [34]. When addressing the device dependency, we also compare SLAC with two typical online RSSI calibration schemes, signal strength difference (*SSD*) [27] and signal strength ratio (*SSR*) [47]. SSD and SSR focus on using the pairwise RSSI deduction [27] and ratio [47] between AP signals to reduce the effect of device dependency. We also conduct offline RSSI calibration (using the linear least-squares fitting) [26] to compare with our SLAC.

For all the location estimation systems above, we utilize the same SDA to first identify the target area and then execute the corresponding localization algorithm against RPs there. When evaluating the performance of area identification, we compare our SDA scheme with other typical machine learning algorithms, including support vector machine (*SVM*) [32], [42] and the nearest neighbor (*NN*) search [30], [31]. We also implement a signal heuristic (*SH*) algorithm [28], which identifies the floors based on strong signals and common AP sets.

Unless otherwise stated, we use the following parameters as baseline: size of sliding window $M = 7$; $T = 2$ m in Equation (20). Initially, $[b_{min}, b_{max}] = [-10, 0]$ and the number of offset parameters $K_b = 20$ (determined empirically via experimental studies). Top $K = 15$ nearest neighbors are used for FL. Number of particles $P = 60$. For FL, MLL and PFL which may be device and user dependent, we utilize the same offline calibration [21], [26] to mitigate the heterogeneity effects. In PFL, 2,000 particles are generated for location estimation. In area identification evaluation, the number of training epochs for SDA is set to be 500, and the batch size is set to be 100. Sigmoid functions are used in the denoising autoencoders.

We evaluate the localization systems by the following performance metrics. Let $\mathbf{x}_i$ be target i's true position and $\widehat{\mathbf{x}}_i$ be the estimated location. The mean localization error (m) of target set $\mathbf{U}$ is given by

$$\mu_e = \frac{1}{|\mathbf{U}|} \sum_{i=1}^{|\mathbf{U}|} \|\mathbf{x}_i - \widehat{\mathbf{x}}_i\|. \tag{26}$$

In our walk detection, we evaluate the accuracy of motion identification, which is the ratio between number of correctly classified samples and the total number. Here we denote the walking as "positive" and static as "negative". We evaluate the true negative rate (TNR), the ratio of truly static (negative) cases over the total of negative decisions (similarly we define the true positive rate or TPR). In device model calibration, we evaluate the difference (in dB) in the trained RSSI offset. In step model calibration, we also evaluate the mean step length estimation error (m) as the average stride length estimation error of multiple steps along a certain trajectory.

6.2 Location Estimation Performance

We first present the localization performance of SLAC. Fig. 7 shows the CDF of localization errors using different algorithms at the baseline parameters in HKIA. Under large signal noise in the airport, target RSSI may show similar values with RPs that are distant apart. FL is hence severely influenced by the dispersed nearest neighbors during the fingerprint matching. PFL has not jointly considered the relationship between RSSI and motion information. As the airport contains large indoor open space, particles become too sparse without map constraints, thereof converging slow under large sensor noise. Similarly, MLL assumes probabilistic independence between different sensor measurements (RF signals and step counter), and also degrades in localization accuracy under noisy environment.

Compared with the above state-of-the-art algorithms, SLAC significantly reduces the location estimation errors in the indoor large open space. SLAC considers the statistical analysis over wireless RSSI and sensor uncertainty (Equations (2) and (6)). Therefore, SLAC mitigates the influence of uncertainty and thus the localization error decreases. With the joint optimization, SLAC is more robust to signal uncertainty and reduces large localization errors. Compared with the previous version, our new SLAC achieves higher localization accuracy due to the following two reasons: (1) our new SLAC applies a more robust motion detection scheme to identify the step, which reduces the influence of measurement noise; (2) our new SLAC formulates a single joint optimization framework without particle filter, which further prevents particle dispersion given random errors. Thus we can observe improvement in location estimation.

Fig. 8 shows the mean localization error versus the parameter γ (Equation (22)). We may observe two important factors over the localization accuracy: the signal difference and the walking distance error. As the γ increases from 0.2, the localization error decreases. It is because considering

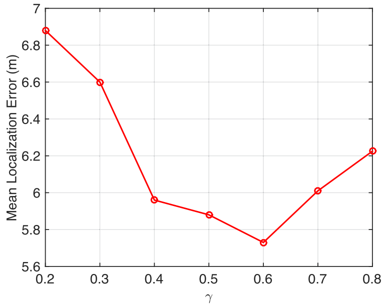

Fig. 8. Mean localization error of SLAC versus the parameter γ (HKIA).

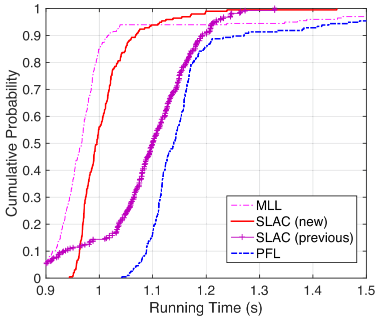

Fig. 9. CDF of running time (s) with different fusion algorithms (HKIA).

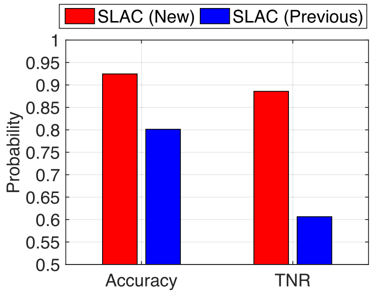

Fig. 10. Classification accuracy and true negative rate (TNR) in the walk detection.

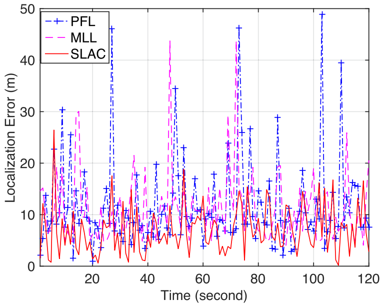

Fig. 11. Real-time localization errors of different fusion schemes over time (HKIA).

both fingerprints and walking distance differences can jointly mitigate the dispersed matching against the fingerprint signal map, and hence can improve the location estimation. When γ approaches a certain value, say, 0.6, the localization accuracy becomes optimal. As γ further increases, the location estimation error enlarges, showing a trade-off in performance. It is because only considering the walking distance difference in the objective does not provide sufficient location matching information against the fingerprint map. Based on above trade-off, in our experimental settings we empirically set 0.6 by default.

To balance between the localization accuracy and computational efficiency, we choose $M = 7$ in our default experimental settings. As the major computation is conducted on the server side in our prototype system, the major energy consumption on the smartphone mainly comes from the Wi-Fi scanning and step counter (overall 560 mW in our experiment) [48]. Therefore, the energy consumption of SLAC on the smartphone is similar to other state-of-the-art sensor fusion techniques [5]. Further energy efficiency improvement can be achieved via approaches like [49], [50] to reduce energy consumption of Wi-Fi and pedometer.

We further compare the computational time of different algorithms. In Fig. 9, we show the CDF of their running time (including previous SLAC). We can observe that the new SLAC outperforms the previous one (reduction by around 40 percent) due to the more efficient optimization formulation. Note that FL is computationally fast given the same set of Wi-Fi inputs, as it does not need extra sensor information for fusion. Compared with other fusion schemes, MLL may be faster in computation due to the simple maximum likelihood calculation. However, the accuracy of both FL and MLL degrades significantly due to large Wi-Fi signal noise, as shown in Fig. 7. PFL uses large number of particles during the location estimation, which is computationally expensive in deployment. To the contrary, SLAC not only locates the target accurately, but also adapts to user and device heterogeneity. Compared with the previous SLAC, our proposed new SLAC does not require particle learning, and hence is computationally more efficient.

We present in Fig. 10 the overall accuracy of walk detection and step detection. We have collected 3,788 sensor readings under different states of stand, walking, smartphone shaking and turning. We also compare our scheme performance with only using h_a and h_ρ in our previous version [34]. From the figure, we can observe that including the magnetic field readings in h_{m} improves the detection accuracy of the step counter. The classification accuracy and TNR comparison are 92.45 versus 80.12, and 88.58 versus 60.65 percent, respectively. It is mainly because considering the magnetic field reduces the misestimation due to device shaking and rotating when the user is not walking. Both schemes can achieve high TPR as 95.51 percent. We can conclude that the proposed new walk detection mitigates the noisy measurements under non-walking behaviors (such as the smartphone shaking and rotating). As the new SLAC outperforms the previous version in both localization accuracy and computational efficiency, in the following we focus on the new SLAC in our experimental evaluation (unless otherwise stated, we use "SLAC" to denote the new version).

Fig. 11 shows the overtime localization errors of different fusion algorithms. Under large signal noise in HKIA, PFL and MLL both degrade in localization performance. It is because the noisy measurements increase the uncertainty in particle transition and probability distribution, thereof leading to large estimation fluctuation. In contrast, SLAC achieves much smaller fluctuation as it considers the correlation among sensors and maximizes the measurement consistency via joint optimization. It hence accurately maps the target location against the fingerprint map.

Fig. 12 shows the step counter and walking distance measurement. Fig. 12a shows the step counter readings based on the repetitive patterns in the accelerometer. Based on the measurement in step counter, SLAC obtains step frequency and walking distance of the user. Fig. 12b shows the measurement error with respect to the walking distance. T can

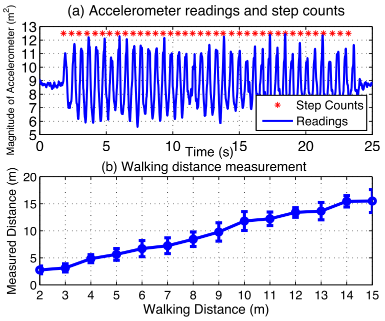

Fig. 12. (a) Step counter readings and (b) measurement accuracy versus walking distances.

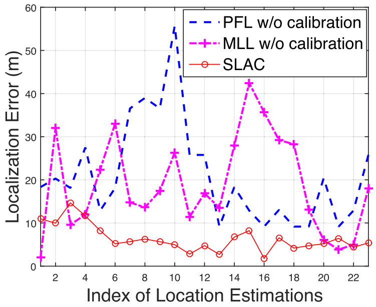

Fig. 13. Localization errors of SLAC in the initial learning process (HKIA).

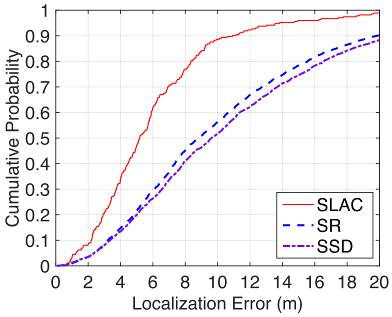

Fig. 14. Cumulative errors of algorithms approaching device dependency (HKIA).

also be obtained through statistical analysis of above errors (T is set to 2 m in our experiment). Via the calibration in SLAC, we obtain different users' step length parameters and achieve higher accuracy in walking displacement than uncalibrated step counters.

6.3 Sensor Calibration Performance

We have also evaluated the performance of inexplicit calibration process in SLAC. Fig. 13 shows the localization error with respect to time for a target using SLAC and other two schemes without sensor calibration. PFL and MLL do not consider simultaneous localization and sensor calibration. Therefore, if the measured Wi-Fi signals get no pre-calibration, their performance degrades significantly. The localization error of SLAC is high at the first few target samples due to the randomness in RSSI model parameters. Then as the incorrect parameters are filtered, SLAC effectively adapts itself to the device heterogeneity and the localization error decreases. Therefore, we can observe that SLAC can learn the heterogeneous model parameters transparently and quickly converge to high localization accuracy.

Fig. 14 shows the CDF of localization errors with different calibration approaches against device dependency. Both SSD and SSR utilize the online RSSI vectors for online calibration. However, under large signal noise, the deduction and ratio between pairwise signals is vulnerable to noise fluctuation. Different from the above approaches, SLAC utilizes the motion information to jointly find the device parameters. Therefore, it reduces the influence of noise while achieving higher localization accuracy and robustness.

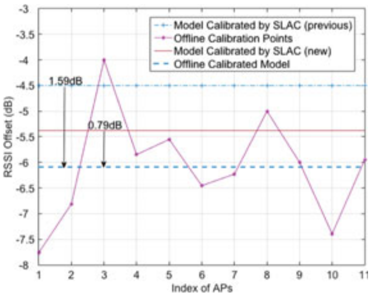

Fig. 15. Performance of device calibration using the previous and new versions (HKIA).

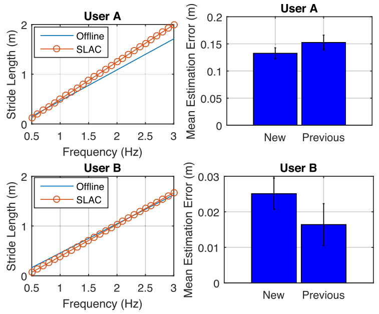

Fig. 16. Performance of step counter calibration for two different users (HKIA).

Fig. 15 illustrates an example of RSSI parameter calibration in Equation (5). As comparison, offline signal calibration (linear fitting) and the previous SLAC (particle-filter-based calibration) is also conducted between Lenovo A680 and HTC One X. We can observe that SLAC can achieve closer calibration results to the accurate but tedious offline calibration, compared with the previous version. The offset calibration error is much smaller than the previous SLAC. Therefore, based on a novel joint optimization, our new SLAC is capable of online calibration without explicit user participation and tedious offline training.

Fig. 16 shows the step length calibration of SLAC for two different users. In the left hand side, we have also shown the offline training using map information to calibrate the step length [8] as comparison. Different from these works using offline training, SLAC learns the step parameters using online readings of Wi-Fi and step counter. With simultaneous calibration and localization, SLAC obtains close parameters in the step length model with those generated from offline training. It further confirms that SLAC can effectively learn the step parameters through the proposed transparent calibration. From the right hand side, we also compare the mean step length estimation error (with standard deviation) between the new SLAC and previous version. We invite the users to walk in different step frequency and estimate the step length using trained models. We can observe that they are close in accuracy, while we have noted in the computation efficiency comparison that the new version outperforms the previous one. We have conducted extensive studies on other users and for brevity we do not repeat the similar results here.

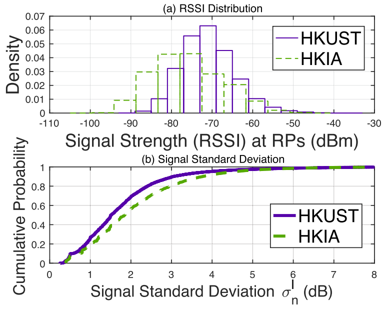

Fig. 17. PDFs of the RSSI and CDFs of σ_n^l (dB) at RPs in the two survey sites.

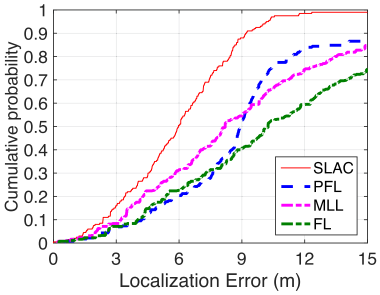

Fig. 18. The CDF of localization errors using different algorithms (HKUST).

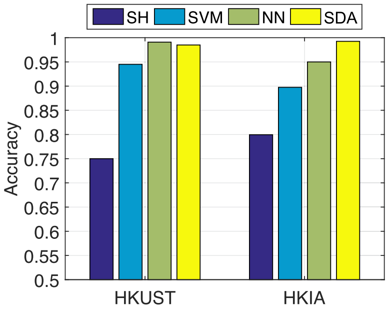

Fig. 19. Area identification accuracy of various schemes in the campus (HKUST) and the international airport (HKIA).

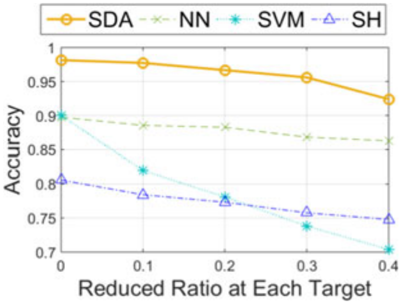

Fig. 20. Area identification accuracy versus reduced ratio of APs at each target (HKIA).

We have also conducted the experimental trials in our university atrium at HKUST. Fig. 17 compares the collected RSSIs (Equation (1)) and signal noise (σ_n^l in Equation (2)) in the HKUST campus and the international airport HKIA. The RSSI range is $[-102, -40]$ dBm in HKIA and $[-101, -30]$ dBm on campus. On average, at an RP of HKIA one may detect 20.83 ± 8.38 APs whose RSSI is above -70 dBm, while at our HKUST one may find 27.05 ± 8.04 APs above -70 dBm. Besides, due to more crowds of people and larger indoor open space, the signals in HKIA show much more fluctuations than on the university campus. As shown in Fig. 17, the signal noise in the sites can be up to 5 dB in the airport, which may introduce large measurement errors for traditional fingerprint-based localization. Under such noisy environment, SLAC can still achieve higher localization accuracy and better self-calibration.

Recall that as shown in Figs. 5 and 6, our university atrium at HKUST has more wall partitions than HKIA. Under office wall partition, the fingerprints show more differentiation. In Fig. 17, we have also observed smaller signal noise on the university campus. Thus, we can observe in Fig. 18 (CDF of localization errors) that SLAC achieves much better performance than in HKIA. Note the marked resemblance between Figs. 18 and 7. As the results are qualitatively similar, we do not repeat others for brevity.

To summarize, the new SLAC exhibits similar or even better (say, less sensor-calibration error) sensor calibration performance compared with the previous version in terms of user and device heterogeneity. As the localization accuracy and the computation efficiency are better, we may conclude the new SLAC is preferred in ILBS deployment.

6.4 Area Identification Performance

In the following, we evaluate the performance of our proposed area identification. In Fig. 19, we show the area (floor) identification accuracy in our university campus and the international airport. We can observe that our SDA-based area identification outperforms other machine learning algorithms. It is mainly because the deep structure of SDA captures the difference between signals at different floors, reduces the input features to low dimensions and distinguishes them through the weight optimization. Other schemes like SH and SVM still suffer from the signal noise in both experimental sites.

Besides localization accuracy, we have also measured the computation time with different area identification schemes. In our experiment, we observe that NN suffers from the heavy computation due to the large fingerprint database (usually more than 2 s per target). Clearly, using the nearest neighbor (NN) search severely degrades the efficiency of localization. SVM and SH take 38.3 ms and 211.5 ms respectively for each target. However, the SDA detection is much faster than above algorithms, with only 8.33 ms for each target. Using SDA can significantly reduce the computation time and improve the localization efficiency of SLAC. Given multiple areas of interests (say, floors), the search scope is then significantly reduced.

We also simulate the effect of AP miss due to crowds of people, building structure change or AP removal, and evaluate the robustness of different schemes. Fig. 20 plots the floor identification accuracy versus the reduced ratio of APs at each target in HKIA. We may observe higher accuracy and more robust performance using SDA compared with other schemes. It is because the deep structure (stacked denoising autoencoders) in SDA sufficiently captures the signal patterns to identify the floors. We implement SVM in a “one-against-all” structure, i.e., we train one SVM model over fingerprints of a floor against other floors. We adopt this structure instead of “one-against-one” due to better accuracy (89.75 percent versus 83.5 percent) in our test. However, as a binary classifier, SVM is easily influenced by imbalance between the binary classes (say, different number of fingerprints at various floors), and is sensitive to missing signals. Given more altered online measurement (say, AP reduction), the suboptimal models trained offline in the SVM may not accurately classify the signals, leading to fast error increase.

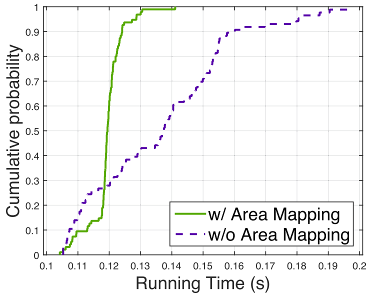

Fig. 21. CDF of SLAC running time (s) with (w/) and without (w/o) area mapping (HKUST).

Finally, we show the efficiency improvement of SLAC using our proposed area identification. In Fig. 21, we show the CDF of SLAC running time with and without the area (floor) mapping in our university campus. From the figure, we can observe that the computation time is significantly reduced after the area mapping (by more than 30 percent reduction in computation time), as the search scope is significantly reduced from multi-floor space to a single floor. Therefore, we can see that out proposed area identification can improve the efficiency and the user experience of SLAC.

7 Conclusion

Step counter has been used to obtain user step frequency and counts, which serve as inputs to a user-based step model to estimate user displacement. The motion model parameters need to be calibrated for different users due to their stride length difference, the so-called user heterogeneity. Given an RSSI signal, different devices may report different readings. Such device heterogeneity needs to be calibrated to align the RSSI measurements. To address the above user and device heterogeneity when fusing step counter with fingerprint signals, the traditional localization approach is to explicitly pre-calibrate the user model and device reading, which is tedious and inconvenient in practice.

We propose SLAC, a novel calibration-free localization system which simultaneously locates the target and calibrates the user and device heterogeneity. In SLAC, an optimization framework jointly considers the RSSI calibration and step counter measurement to locate a target with high accuracy. Furthermore, SLAC utilizes the correlation in location estimations, RSSI readings and step information to transparently calibrate the heterogeneity in device and user models. To further reduce the search scope in location estimation, we leverage the stacked denoising autoencoder to identify the area where target is likely at. We have conducted extensive experimental trials in the Hong Kong International Airport and our HKUST atrium. Our results show that SLAC can significantly improve the localization accuracy while learning the models for step counter and device RSSI.

Acknowledgments

This work was supported, in part, by the National Natural Science Foundation of China (61472455) and the Natural Science Foundation of Guangdong Province (2014A030313154), and the Guangzhou Science Technology and Innovation Commission (GZSTI16EG14/201704030079).

References

[1] S. He and S. H. G. Chan, "Wi-Fi fingerprint-based indoor positioning: Recent advances and comparisons," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 1, pp. 466–490, Jan.–Mar. 2016.

[2] 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. 9th ACM Annu. Int. Conf. Mobile Comput. Netw.*, 2014, pp. 459–470.

[3] D.-K. Cho, M. Mun, U. Lee, W. J. Kaiser, and M. Gerla, "AutoGait: A mobile platform that accurately estimates the distance walked," in *Proc. 13th IEEE Int. Conf. Pervasive Comput. Commun.*, Mar. 2010, pp. 116–124.

[4] Y. Gao, et al., "XINS: The anatomy of an indoor positioning and navigation architecture," in *Proc. 1st Int. Workshop Mobile Location-Based Service*, 2011, pp. 41–50.

[5] W. Sun, J. Liu, C. Wu, Z. Yang, X. Zhang, and Y. Liu, "MoLoc: On distinguishing fingerprint twins," in *Proc. IEEE 33rd Int. Conf. Distrib. Comput. Syst.*, Jul. 2013, pp. 226–235.

[6] P. Bahl and V. N. Padmanabhan, "RADAR: An in-building RF-based user location and tracking system," in *Proc. IEEE Conf. Inf. Comput. Commun.*, 2000, vol. 2, pp. 775–784.

[7] D. Han, S. Jung, M. Lee, and G. Yoon, "Building a practical Wi-Fi-based indoor navigation system," *IEEE Pervasive Comput.*, vol. 13, no. 2, pp. 72–79, Apr.–Jun. 2014.

[8] 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. 12th ACM Int. Conf. Ubiquitous Comput.*, 2014, pp. 147–158.

[9] A. Rai, K. K. Chintalapudi, V. N. Padmanabhan, and R. Sen, "Zee: Zero-effort crowdsourcing for indoor localization," in *Proc. ACM 8th Annu. Int. Conf. Mobile Comput. Netw.*, 2012, pp. 293–304.

[10] J. Xiao, Z. Zhou Youwen Yi, and L. M. Ni, "A survey on wireless indoor localization from the device perspective," *ACM Comput. Surv.*, vol. 49, no. 2, pp. 25:1–25:31, Jun. 2016.

[11] M. Youssef and A. Agrawala, "The Horus WLAN location determination system," in *Proc. ACN 1st Int. Conf. Mobile Syst. Appl Serv.*, 2005, pp. 205–218.

[12] C. Wu, Z. Yang, and Y. Liu, "Smartphones based crowdsourcing for indoor localization," *IEEE Trans. Mobile Comput.*, vol. 14, no. 2, pp. 444–457, Feb. 2015.

[13] M. Atia, A. Noureldin, and M. J. Korenberg, "Dynamic online-calibrated radio maps for indoor positioning in wireless local area networks," *IEEE Trans. Mobile Comput.*, vol. 12, no. 9, pp. 1774–1787, Sep. 2013.

[14] J. Seitz, J. Jahn, J. G. Boronat, T. Vaupel, S. Meyer, and J. Thielecke, "A hidden Markov model for urban navigation based on fingerprinting and pedestrian dead reckoning," in *Proc. 13th Int. Conf. Inf. IEEE Fusion*, Jul. 2010, pp. 1–8.

[15] H. Wen, Z. Xiao, N. Trigoni, and P. Blunsom, "On assessing the accuracy of positioning systems in indoor environments," in *Wireless Sensor Networks*. Berlin Heidelberg: Springer, 2013, vol. 7772.

[16] G. Shen, Z. Chen, P. Zhang, T. Moscibroda, and Y. Zhang, "Walkie-Markie: Indoor pathway mapping made easy," in *Proc. 10th USENIX Conf. Netw. Syst. Des. Implementation*, 2013, pp. 85–98.

[17] Z. Xiao, H. Wen, A. Markham, and N. Trigoni, "Indoor tracking using undirected graphical models," *IEEE Trans. Mobile Comput.*, vol. 14, no. 11, pp. 2286–2301, Nov. 2015.

[18] B. Ferris, D. Fox, and N. Lawrence, "WiFi-SLAM using gaussian process latent variable models," in *Proc. 25th Int. Joint Conf. Artif. Intell.*, 2007, vol. 7, pp. 2480–2485.

[19] H. Wang, S. Sen, A. Elgohary, M. Farid, M. Youssef, and R. R. Choudhury, "No need to war-drive: Unsupervised indoor localization," in *Proc. Int. Conf. Mobile Syst. Appl Serv.*, 2012, pp. 197–210.

[20] H. Abdelnasser, et al., "SemanticSLAM: Using environment landmarks for unsupervised indoor localization," *IEEE Trans. Mobile Comput.*, vol. 15, no. 7, pp. 1770–1782, Jul. 2016.

[21] F. Li, C. Zhao, G. Ding, J. Gong, C. Liu, and F. Zhao, "A reliable and accurate indoor localization method using phone inertial sensors," in *Proc. ACM 12th Int. Conf. Ubiquitous Comput.*, 2012, pp. 421–430.

[22] M. Uddin and T. Nadeem, "SpyLoc: A light weight localization system for smartphones," in *Proc. 11th Annu. IEEE Int. Conf. Sensing Commun. Netw.*, Jun. 2014, pp. 72–80.

[23] L. Zhang, et al., "Montage: Combine frames with movement continuity for realtime multi-user tracking," in *Proc. IEEE Conf. Inf. Comput. Commun.*, Apr. 2014, pp. 799–807.

[24] F. Lemic, et al., "Toward extrapolation of WiFi fingerprinting performance across environments," in *Proc. ACM 17th Int. Workshop Mobile Comput. Syst. Appl.*, 2016, pp. 69–74.

[25] Z. Xiao, H. Wen, A. Markham, and N. Trigoni, "Robust indoor positioning with lifelong learning," *IEEE J. Select. Areas Commun.*, vol. 33, no. 11, pp. 2287–2301, Nov. 2015.

[26] C. Laoudias, D. Zeinalipour-Yazti, and C. G. Panayiotou, "Crowdsourced indoor localization for diverse devices through radiomap fusion," in *Proc. Int. Conf. Indoor Positioning Indoor Navigation*, Oct. 2013, pp. 1–7.

[27] 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 Comput.*, vol. 12, no. 1, pp. 65–77, Jan. 2013.

[28] P. Bhargava, S. Krishnamoorthy, A. Shrivastava, A. K. Nakshathri, M. Mah, and A. Agrawala, "Locus: Robust and calibration-free indoor localization, tracking and navigation for multi-story buildings," *J. Location Based Serv.*, vol. 9, no. 3, pp. 187–208, 2015.

[29] D. Banerjee, S. K. Agarwal, and P. Sharma, "Improving floor localization accuracy in 3D spaces using barometer," in *Proc. ACM Int. Symp. Wearable Comput.*, 2015, pp. 171–178.

[30] A. Varshavsky, A. LaMarca, J. Hightower, and E. de Lara, "The SkyLoc floor localization system," in *Proc. 13th IEEE Int. Conf. Pervasive Comput. Commun.*, 2007, pp. 125–134.

[31] H. X. Liu, B. A. Chen, P.-H. Tseng, K.-T. Feng, and T.-S. Wang, "Map-aware indoor area estimation with shortest path based on RSS fingerprinting," in *Proc. IEEE 81st Veh. Technol. Conf. Spring*, May 2015, pp. 1–5.

[32] S. He, J. Tan, and S.-H. G. Chan, "Towards area classification for large-scale fingerprint-based system," in *Proc. 12th ACM Int. Conf. Ubiquitous Comput.*, 2016, pp. 232–243.

[33] X. Shen, Y. Chen, J. Zhang, L. Wang, G. Dai, and T. He, "BarFi: Barometer-aided WiFi floor localization using crowdsourcing," in *Proc. IEEE 12th Int. Conf. Mobile Ad Hoc Sensor Syst.*, Oct. 2015, pp. 416–424.

[34] 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. 12th ACM Int. Conf. Ubiquitous Comput.*, 2015, pp. 897–908.

[35] H. Xu, Z. Yan, Z. Zhou, L. Shangguan, K. Yi, and Y. Liu, "Enhancing WiFi-based localization with visual clues," in *Proc. 12th ACM Int. Conf. Ubiquitous Comput.*, 2015, pp. 963–974.

[36] J.-G. Park, D. Curtis, S. Teller, and J. Ledlie, "Implications of device diversity for organic localization," in *Proc. IEEE Conf. Inf. Comput. Commun.*, Apr. 2011, pp. 3182–3190.

[37] A. Brajdic and R. Harle, "Walk detection and step counting on unconstrained smartphones," in *Proc. 12th ACM Int. Conf. Ubiquitous Comput.*, 2013, pp. 225–234.

[38] B. Huang, G. Qi, X. Yang, L. Zhao, and H. Zou, "Exploiting cyclic features of walking for pedestrian dead reckoning with unconstrained smartphones," in *Proc. 12th ACM Int. Conf. Ubiquitous Comput.*, 2016, pp. 374–385.

[39] S. P. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge University Press, 2004.

[40] CVXOPT: Python software for convex optimization. (2017). [Online]. Available: http://cvxopt.org/

[41] JOptimizer - Java convex optimizer. (2017). [Online]. Available: http://www.joptimizer.com/

[42] R. W. Ouyang, A. K. S. Wong, C.-T. Lea, and M. Chiang, "Indoor location estimation with reduced calibration exploiting unlabeled data via hybrid generative/discriminative learning," *IEEE Trans. Mobile Comput.*, vol. 11, no. 11, pp. 1613–1626, Nov. 2012.

[43] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Sci.*, vol. 315, no. 5814, pp. 972–976, 2007.

[44] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *J. Mach. Learn. Res.*, vol. 11, pp. 3371–3408, Dec. 2010.

[45] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proc. 25th Int. Conf. Mach. Learn.*, 2008, pp. 1096–1103.

[46] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Netw.*, vol. 61, pp. 85–117, 2015.

[47] M. B. Kjargaard, "Indoor location fingerprinting with heterogeneous clients," *Pervasive Mobile Comput.*, vol. 7, no. 1, pp. 31–43, 2011.

[48] L. Zhang, et al., "Accurate online power estimation and automatic battery behavior based power model generation for smartphones," in *Proc. IEEE/ACM/IFIP Int. Conf. Hardw./Softw. Codesign Syst. Synthesis (CODES+ISSS)*, 2010, pp. 105–114.

[49] N. Brouwers, M. Zuniga, and K. Langendoen, "Incremental Wi-Fi scanning for energy-efficient localization," in *Proc. 13th IEEE Int. Conf. Pervasive Comput. Commun.*, Mar. 2014, pp. 156–162.

[50] T. O. Oshin and S. Poslad, "ERSP: An energy-efficient real-time smartphone pedometer," in *Proc. IEEE Int. Conf. Syst. Man Cybern.*, Oct. 2013, pp. 2067–2072.

Suining He received the PhD degree from the Department of Computer Science and Engineering, The Hong Kong University of Science and Technology (HKUST), in 2016. He is currently working as a postdoctoral research fellow with the Real-Time Computing Lab (RTCL), Department of Electrical Engineering and Computer Science, the University of Michigan, Ann Arbor, MI. His research interest includes indoor localization, smartphone, sensing, and mobile computing.

S.-H. Gary Chan (S'89-M'98-SM'03) received the BSE (highest honor) degree in electrical engineering from Princeton University, Princeton, NJ, with certificates in applied and computational mathematics, engineering physics, and engineering and management systems, and the MSE and the PhD degrees in electrical engineering from Stanford University, Stanford, CA, with a minor in business administration, 1993, 1994, and 1999, respectively. He is currently a professor in the Department of Computer Science and Engineering, The Hong Kong University of Science and Technology (HKUST), Hong Kong. He is also the director of the Entrepreneurship Center, and chair of the Committee on Entrepreneurship Education Program, Center for Education Innovation, HKUST. His research interest includes multimedia networking, mobile computing, data analytics, and IT entrepreneurship. He has been an associate editor of the *IEEE Transactions on Multimedia (2006-11)*, and a vice-chair of the Peer-to-Peer Networking and Communications Technical sub-committee of the IEEE Comsoc Emerging Technologies Committee (2006-2013). He is and has been guest editor of *Elsevier Computer Networks* (2017), the *ACM Transactions on Multimedia Computing, Communications, and Applications* (2016), the *IEEE Transactions on Multimedia* (2011), the *IEEE Signal Processing Magazine* (2011), the *IEEE Communication Magazine* (2007), and *Springer Multimedia Tools and Applications* (2007). He was the TPC chair of the IEEE Consumer Communications and Networking Conference (IEEE CCNC) 2010, Multimedia Symposium of IEEE Globecom (2007 and 2006), IEEE ICC (2007 and 2005), and the Workshop on Advances in Peer-to-Peer Multimedia Streaming in ACM Multimedia Conference (2005). He has co-founded several startups deploying his research results. Due to their innovations and commercial impacts, his projects have received local and international ICT awards (2012-2015). He is the recipient of the Google Mobile 2014 Award (2010 and 2011) and Silver Award of Boeing Research and Technology (2009). He has been a visiting professor and researcher at Microsoft Research (2000-2011), Princeton University (2009), Stanford University (2008-2009), and University of California at Davis (1998-1999). He was undergraduate programs coordinator in the Department of Computer Science and Engineering (2013-2015), director of the Sino Software Research Institute (2012-2015), co-director of the Risk Management and Business Intelligence Program (2011-2013), and director of the Computer Engineering Program (2006-2008) with HKUST. He was a William and Leila Fellow at Stanford University (1993-1994), and the recipient of the Charles Ira Young Memorial Tablet and Medal, and the POEM Newport Award of Excellence at Princeton (1993). He is a chartered fellow of the Chartered Institute of Logistics and Transport Hong Kong (CILTHK), and a member of honor societies Tau Beta Pi, Sigma Xi, and Phi Beta Kappa. He is a senior member of the IEEE.

Lei Yu received the bachelor's degree in software engineering from Sun Yat-sen University. He is currently working toward the double master's degree in electrical and computer engineering at Carnegie Mellon University and Sun Yat-sen University. He received many prizes during his undergraduate period, such as the National Scholarship of China, Google Excellence Scholarship, etc.

Ning Liu received the BS degree in computational mathematics from Northwest University, Xian, China, and the PhD degree in computer science from Sun Yat-sen University (SYSU), Guangzhou, China, in 1996 and 2004, respectively. He is currently an associate professor with Sun Yat-sen University. He has served as a reviewer for several important conferences and journals. His current research interests include computer vision, peer-to-peer networks, and machine learning algorithms.