

Article

## Collaborative WiFi Fingerprinting Using Sensor-Based Navigation on Smartphones

Peng Zhang <sup>1</sup>, Qile Zhao <sup>1,\*</sup>, You Li <sup>1,2</sup>, Xiaoji Niu <sup>1</sup>, Yuan Zhuang <sup>2</sup> and Jingnan Liu <sup>1</sup>

<sup>1</sup> GNSS Research Center, Wuhan University, No.129 Luoyu Road, Wuhan 430079, China; E-Mails: fenix@whu.edu.cn (P.Z.); liyou@whu.edu.cn (Y.L.); xjniu@whu.edu.cn (X.N.); jnliu@whu.edu.cn (J.L.)

<sup>2</sup> Department of Geomatics Engineering, University of Calgary, Calgary, AB T2N1N4, Canada; E-Mail: zhy.0908@gmail.com

\* Author to whom correspondence should be addressed; E-Mail: zhaoql@whu.edu.cn; Tel.: +86-27-6877-8971.

Academic Editor: Vittorio M.N. Passaro

Received: 3 June 2015 / Accepted: 10 July 2015 / Published: 20 July 2015

---

**Abstract:** This paper presents a method that trains the WiFi fingerprint database using sensor-based navigation solutions. Since micro-electromechanical systems (MEMS) sensors provide only a short-term accuracy but suffer from the accuracy degradation with time, we restrict the time length of available indoor navigation trajectories, and conduct post-processing to improve the sensor-based navigation solution. Different middle-term navigation trajectories that move in and out of an indoor area are combined to make up the database. Furthermore, we evaluate the effect of WiFi database shifts on WiFi fingerprinting using the database generated by the proposed method. Results show that the fingerprinting errors will not increase linearly according to database (DB) errors in smartphone-based WiFi fingerprinting applications.

**Keywords:** WiFi; indoor positioning; MEMS sensors; training; PDR

---

### 1. Introduction

Mobile location-based services (LBS) are attracting the attention of many mobile device companies due to their potential applications in a wide range of personalized services [1]. As a core technology of

LBS, navigation (*i.e.*, determination of the position, velocity, and attitude of the mobile device) is evidently vital. A highly demanding issue is to provide trustable navigation solutions in real time, since mobile devices are carried by users almost anywhere and anytime [2].

While Global Navigation Satellite Systems (GNSS) based outdoor navigation has greatly advanced over the past few decades, positioning and navigation in indoor scenarios and deep urban areas are still an open issue [3]. The challenges include unavailable or degraded GNSS signals, complex indoor environments, the necessity of using low-grade devices, *etc.* To navigate in indoor or urban areas, various indoor positioning technologies based on Radio Frequency (RF) signals have been broadly researched, such as Wireless Local Area Net (WLAN, also known as Wi-Fi) [4], cellular networks [5], Bluetooth [6], Radio Frequency identification (RFID) tags [7], ZigBee [8], Ultra Wideband Beacons (UWB) [9], high-sensitivity GNSS [10], and pseudo-satellites (pseudolites, also known as Locata) [11]. These RF-based technologies can provide a long-term accurate absolute position, but require the creation and maintenance of a network [12]. WiFi (WLAN based on 802.11 standards) does not need a specific network, since there are WiFi infrastructures in public areas such as malls, airports, and universities. Considering the fact that the WiFi receivers commonly exist in consumer devices such as smartphones, it is feasible to implement WiFi positioning in these public areas. The common WiFi positioning approaches include triangulation [13], fingerprinting [14–18], and their combination [19]. Triangulation is a method that determines the relative positions of objects using the geometry of triangles; therefore, a signal propagation model is needed to convert the received signal strength (RSS) to calculate the distance between WiFi access points (APs) and user devices [20]. However, it is difficult to obtain an accurate signal propagation model in indoor environments, since transmitted signals can suffer obstructions and reflections [21]. To mitigate this issue, several enhanced models considering multipath effects [22] or Rayleigh fading effects [23] have been studied. WiFi fingerprinting approaches based on WiFi RSS have gained a large amount of attention, as they can provide a relatively high indoor positioning accuracy in a multipath indoor environment [24]. Fingerprinting is commonly achieved in two steps (phases): the offline training (pre-survey) step and the online positioning step. The training phase is conducted to build or update a <RSS, location> database (DB) that consists of a set of reference points (RPs) with known coordinates and the RSS from available WiFi APs, while the positioning step is implemented to finding the closest match between the features of the RSS and those stored in the DB. The main barrier for the broad adoption of WiFi fingerprinting is that most current DB training methods are tedious and labor-intensive [25]. Another challenge is the non-stationarity of WiFi signal distribution, which can be attributed to radio signal propagation effects induced by environment changes [20,26–28]. Therefore, the training phase needs to be rerun periodically to keep the system up-to-date [24].

Different WiFi DB training approaches have been researched due to various requirements. The research [29,30] have proposed the idea of a metropolitan-scale WiFi localization based on vehicles with equipment such as GNSS receivers and high-end inertial navigation systems. This method is highly efficient, but is mainly used in outdoor urban areas. To train indoor WiFi DBs, the first approach is to survey at every RP and record their fingerprints. This method can also improve DB reliability by averaging the RSS at each RP [31] and even providing a coarse estimate of orientation [20,32]. However, it is time- and labor-consuming when dense RPs are selected to cover an entire area of interest, and a surveyor needs to mark the position of all RPs (labels) manually [33]. The training phase can take up to several hours even for a small building [34].

To minimize the number of labels needed for training, another approach is used based on landmarks (*i.e.*, points with known coordinates) or floor plans (*i.e.*, the true position of corners and intersections, and the true orientation of corridors), and constant-speed assumption. To implement this method, a surveyor has to walk with constant speed along each link between landmarks, such as corners or intersections, over the known path. The position of landmarks can be determined by marking them manually on a digital map, while the position of other RPs on the links between landmarks can be calculated by the arrival time and the distance between two landmarks. This approach is significantly more time-effective than measuring the position of all RPs on the digital map; however, it requires the user to walk straight with a constant speed between landmarks. To break the limitation of the constant-speed assumption, scholars generate the WiFi fingerprint DB between landmarks by leveraging a dead-reckoning solution from sensors [35]. Sensors can provide a step-based dead-reckoning approach, which detects steps, predicts their length and direction based on sensor readings [24].

There is also literature on the removal of the training process by updating the WiFi DB automatically while navigating based on sensors. WiFi SLAM (simultaneous location and mapping) is an advanced algorithm which trains WiFi DB while navigating [36]. The limitation of the SLAM algorithm is the increased computational cost as the scenarios become larger [37]. There are other approaches based on crowd-sourcing. The research [38] estimates the location of WiFi APs or other radio beacons using pedestrian dead-reckoning with high-quality foot-mounted IMUs, while [34,39–41] propose similar systems or approaches using handheld smartphones. Based on this idea, it is possible for mobile users to collect WiFi fingerprints automatically in daily life by conducting sensor-based navigation. The crowd-sourcing approaches have the potential to provide daily-life navigation solutions due to the increasing popularity of Micro-electromechanical Systems (MEMS) sensor in consumer electronics [42,43]. The shortcoming is that MEMS sensor errors change with time and are significantly dependent on environmental factors such as temperature [44]. Therefore, MEMS inertial sensors provide only short-term accuracy but suffer from accuracy degradation over time [45]. Although the errors of horizontal attitudes (*i.e.*, the roll and pitch) can be controlled by accelerometer measurements, the heading error will grow when there is no additional information from other sensors or techniques [46]. Magnetometers can assist the heading estimation by sensing the Earth's magnetic field; however, the local magnetic field is susceptible to interference from man-made infrastructures when the user enters urban or indoor environments [47], which makes magnetometer measurements unreliable. Therefore, MEMS sensors have to be integrated with other positioning technologies to provide reliable long-term navigation solutions.

In this paper, we propose an approach that utilizes similar ideas as [34,39,40], *i.e.*, training the WiFi DB using the navigation data from the users, and utilizing different strategies to control sensor-based navigation errors and in turn control the drifts of the generated WiFi DBs. First, we use a strategy that combines different navigation trajectories that move in and out of a building to make up the DB; also, we restrict the time length of available indoor navigation trajectories, *i.e.*, the trajectories from the last epoch that receives GNSS signal before entering an indoor area to the first epoch that receives GNSS signal after walking out of the indoor area. To be specific, we only use the navigation data which is shorter than a time threshold (e.g., 10 min). This strategy can help control the sensor-based navigation errors, but will exclude the majority of navigation trajectories. However, since there will be massive pedestrian navigation data in real-world application, even a small number of reliable navigation

trajectories will be enough to generate the WiFi DB; also, it is necessary to use only the most valuable data and exclude others. Moreover, sometimes aiding sources such as landmarks can update the navigation solution, which divide long-term trajectories into short-term trajectories.

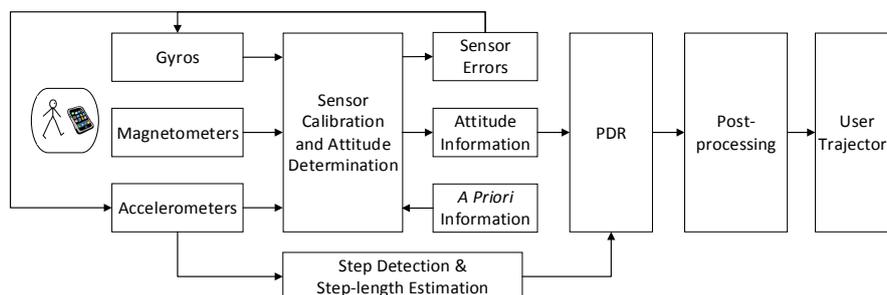
Another strategy is implemented based on the fact that users will always walk both into and out of an indoor area. Therefore, we conduct post-processing (smoothing) to improve the sensor-based navigation solution, since there are accurate GNSS position updates on the start and end points of indoor trajectories. Post-processing is common for survey applications such as mobile mapping [48,49], but is used relatively little in real-time navigation [50]. To train the WiFi DB, post-processing is both affordable and worthwhile. Test results show that the navigation solutions can be significantly improved through post-processing, but still suffer from errors during the middle part of the trajectories, which will in turn cause DB drifts.

Furthermore, we evaluate the effect of DB shifts on WiFi fingerprinting. Therefore, the difference between the WiFi fingerprinting results with the proposed DB (*i.e.*, the DB generated by the proposed approach) and the reference DB was not significant when comparing with the WiFi fingerprinting errors with smartphones. Therefore, although constructing a high-quality DB is vital since fingerprinting accuracy highly depends on DB quality [31], the fingerprinting errors will not increase linearly according to DB errors because there are other error sources in smartphone-based WiFi fingerprinting.

This paper is organized as follows: Section 2 provides the methodology of proposed sensor-based navigation methods, including attitude determination, pedestrian dead-reckoning (PDR), and smoothing. Section 3 explains WiFi fingerprinting and focuses on training the WiFi DB; Section 4 outlines the tests and results; and Section 5 draws the conclusions.

## 2. Sensor-Based Navigation

The sensor-based navigation algorithm consists of three modules: multi-sensor based attitude determination, position tracking, and post-processing. The main components of the proposed algorithm are shown in Figure 1. The following will introduce these modules separately.



**Figure 1.** Main components of sensor-based navigation algorithm.

### 2.1. Multi-Sensor Based Attitude Determination

In this part, we use the inertial navigation systems (INS) mechanization to calculate continuous attitude angles, and utilize the information from multiple sensors and *a priori* information as updates to estimate the attitude errors through an attitude determination Kalman filter. The details of INS

mechanization has been well described in [48]. The following will describe the Kalman filter models, including the system model and the measurement models.

### 2.1.1. Kalman Filter System Model

A simplifies from of the error model detailed in [48] is applied as the continuous state equation in the Kalman filter [51]

$$\begin{bmatrix} \delta \dot{\mathbf{r}}^n \\ \delta \dot{\mathbf{v}}^n \\ \dot{\Psi} \end{bmatrix} = \begin{bmatrix} -\boldsymbol{\omega}_{en}^n \times \delta \mathbf{r}^n + \delta \mathbf{v}^n \\ -(2\boldsymbol{\omega}_{ie}^n + \boldsymbol{\omega}_{en}^n) \times \delta \mathbf{v}^n + \mathbf{f}^n \times \Psi + \mathbf{C}_b^n \delta \mathbf{f}^b \\ -(\boldsymbol{\omega}_{ie}^n + \boldsymbol{\omega}_{ec}^n) \times \Psi - \mathbf{C}_b^n \delta \boldsymbol{\omega}_{ib}^b \end{bmatrix} \quad (1)$$

where  $\delta r^n$ ,  $\delta v^n$  and  $\Psi$  are the position error, the velocity error, and the attitude error.  $C_b^n$  is the Direction Cosine Matrix from b-frame (*i.e.*, the body frame) to n-frame (*i.e.*, the navigation frame).  $C_b^n$  is the specific force vector projected to n-frame, and  $\boldsymbol{\omega}_{ie}^n$  and  $\boldsymbol{\omega}_{en}^n$  represent the angular velocity of the earth and the angular rate vector of n-frame with respect to e-frame (*i.e.*, the Earth frame), both projected to n-frame. The symbol “ $\times$ ” denotes cross product of two vectors.  $\delta \mathbf{f}^b$  and  $\delta \boldsymbol{\omega}_{ib}^b$  are the accelerometer error and gyro error vectors.

### 2.1.2. Kalman Filter Measurement Models

We use multiple kinds of constraints to build the measurement model to enhance the attitude determination. These constraints include the pseudo-observations and the measurements from magnetometers and accelerometer.

The pseudo-position and pseudo-velocity observations are proposed based on the fact that the range of the position and linear velocity of the IMU are within a limited scope [51]. They can be used to compose the measurement vectors of the Kalman filter, that is

$$\hat{r}^n - \hat{v}_0^n = \delta r^n + n \quad (2)$$

with  $\hat{v}_0^n = \text{constant}$  and

$$\hat{v}^n = \delta \mathbf{v}^n + \mathbf{n}_v \quad (3)$$

where  $\hat{r}^n$  and  $\hat{v}^n$  are INS-derived position and velocity vectors;  $\hat{v}_0^n$  is the observation vector of the proposed pseudo-position;  $\delta r^n$  and  $\delta v^n$  are the position errors and velocity errors.  $\mathbf{n}_r$  and  $\mathbf{n}_v$  are the measurement noises (*i.e.*, the inaccuracy) of pseudo-position and pseudo-velocity. The pseudo-velocity is directly set as zero; the pseudo-position can be set as a random constant value and this will not influence the estimation of attitude and gyro errors [51].

Accelerometers and magnetometers are well used to update attitude estimation in many applications of attitude and heading reference systems (AHRS) [52]. The details of using the accelerometer and magnetometer measurements can refer to [53]. The measurement uncertainties on the magnetometer measurements are different from that on the accelerometer signals. The latter is commonly high-frequency and alternating; however, the perturbation on the magnetometer measurements is low-frequency because the local magnetic field (LMF) changes due to the existence of external magnetic

bodies such as man-made infrastructures. Among various kinds of perturbations to LMF, a common type is that both the direction and the strength of LMF are changed, but the change is stable during short periods. This period during which the LMF is stable is called quasi-static magnetic field (QSMF) period. We only use the magnetometer measurements in QSMF environment, which follows [47].

### 2.1.3. Initial Alignment

The initial direction cosine matrix  $\hat{C}_b^n(t_0)$  can be determined by using the accelerometer and magnetometer measurements as [54]

$$\tilde{C}_b^n = \left( \begin{bmatrix} \mathbf{f}^n & \mathbf{m}^n & \mathbf{I}^n \end{bmatrix}^T \right)^{-1} \begin{bmatrix} \tilde{\mathbf{f}}^b & \tilde{\mathbf{m}}^b & \tilde{\mathbf{I}}^b \end{bmatrix}^T \quad (4)$$

where  $\mathbf{f}^n$  and  $\mathbf{m}^n$  are the specific force and local magnetic field in the navigation frame,  $\mathbf{I}^n = \mathbf{f}^n \times \mathbf{m}^n$ ;  $\tilde{\mathbf{f}}^b$  and  $\tilde{\mathbf{m}}^b$  are the accelerometer and magnetometer measurements, and  $\tilde{\mathbf{I}}^b = \tilde{\mathbf{f}}^b \times \tilde{\mathbf{m}}^b$ . Since the purpose of this paper is to use sensor-based solution to build the WiFi DB, we start data collection from outdoor environment, where GNSS positioning results are used to provide initial position and heading.

## 2.2. Position Tracking

### 2.2.1. Pedestrian Dead-Reckoning

PDR is the relative means of determining of a new position from the previous known position using current heading and step length information [55]. The coordinates  $(\varphi_{k+1}, \lambda_{k+1})$  of a new position with respect to a previously known position  $(\varphi_k, \lambda_k)$  can be computed as follows:

$$\begin{bmatrix} \varphi_{k+1} \\ \lambda_{k+1} \end{bmatrix} = \begin{bmatrix} \varphi_k + (s_k \cos \psi_{k-1} + w_N) / (R_m + h) \\ \lambda_k + (s_k \sin \psi_{k-1} + w_E) / [(R_n + h) \cos \varphi_k] \end{bmatrix} \quad (5)$$

where  $\varphi$ ,  $\lambda$ ,  $\psi$ , and  $s$  are the latitude, longitude, heading and step length,  $R_m$  and  $R_n$  are the radius of curvature of meridian and curvature in the prime vertical, and  $h$  is the altitude. The subscript  $k$  and  $k + 1$  indicate the count of steps.

In practice, the linear error Model (6) is used to implement the PDR algorithm. This is because when other information such as GNSS or WiFi is available, it can be used to update the PDR through a position tracking Kalman filter. The Kalman filter system model is

$$\hat{\mathbf{x}}_{k+1} = \Phi_k \mathbf{x}_k + \mathbf{w}_k \quad (6)$$

with

$$\hat{\mathbf{x}}_k = [\delta\varphi_k \quad \delta\lambda_k \quad \delta\psi_k \quad \delta s_k \quad b_k]^T \quad (7)$$

$$\Phi_k = \begin{bmatrix} 1 & 0 & -s_k \sin \psi_k / (R_m + h) & \cos \psi_k / (R_m + h) & 0 \\ 0 & 1 & s_k \cos \psi_k / [(R_n + h) \cos \varphi_k] & \sin \psi_k / [(R_n + h) \cos \varphi_k] & 0 \\ 0 & 0 & 1 & 0 & \Delta t \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (8)$$

$$\mathbf{w}_k = \left[ w_{N,k} / (R_m + h) \quad w_{E,k} / [(R_n + h) \cos \varphi] \quad w_\psi \quad w_s \quad w_b \right]^T \quad (9)$$

where  $\hat{\mathbf{x}}$  is the state vector,  $\Phi_k$  is the transition matrix, and  $\mathbf{w}_k$  is the noise vector.  $\delta\varphi$ ,  $\delta\lambda$ ,  $\delta\psi$ ,  $\delta s$ , and  $b$  are the errors of latitude, longitude, heading, step length, and the vertical gyro bias component.

### 2.2.2. Step Length Model

Step detection and step length estimation have been well introduced in [56]. The basic idea is that the acceleration shows a cycle pattern responding to every step when a pedestrian is walking. The specific force signals are smoothed to reduce the influence of noise by

$$\bar{f}_k = \sum_{i=k-L}^{k+L} f_i \quad (10)$$

where  $(2L + 1)$  is the length of smooth array. The smoothed specific force is computed and then checked periodically in a sliding window. If it is a peak and the magnitude is bigger than a threshold, a new step is detected.

The step length varies with walk velocity, terrain scope, etc. It can be related to the step frequency with a linear model [57]

$$s_k = Af_k + B + w_s \quad (11)$$

with  $f_k = 1 / (t_k - t_{k-1})$ .  $f_k$  is the walking frequency from time  $t_{k-1}$  to  $t_k$ .  $A$  and  $B$  are coefficients that can be trained,  $w_s$  is noise.

### 2.3. Post-Processing

To build WiFi DB, it is affordable and worthwhile to employ post-processing to provide a better navigation solution. In general, smoothing can be performed by combining the forward and backward filter solution as follows [48]:

$$\hat{\mathbf{x}}_{sm,k} = \mathbf{P}_{sm,k} (\mathbf{P}_{f,k}^{-1} \mathbf{x}_{f,k} + \mathbf{P}_{b,k}^{-1} \hat{\mathbf{x}}_{b,k}) \quad (12)$$

$$\mathbf{P}_{sm,k} = (\mathbf{P}_{f,k}^{-1} + \mathbf{P}_{b,k}^{-1})^{-1} \quad (13)$$

where subscripts ' $f$ ', ' $b$ ', and ' $sm$ ' denote the forward, backward, and smoothed solutions, respectively.  $\hat{\mathbf{x}}$  is the estimated state vector, and  $\mathbf{P}$  represents the covariance matrix of  $\hat{\mathbf{x}}$ . In this research,  $\hat{\mathbf{x}} = [\varphi \quad \lambda \quad \psi \quad s \quad b]$ .  $\mathbf{P}_f$  and  $\mathbf{P}_b$  are predicted in the forward and backward process by [58]

$$\mathbf{P}_{k+1} = \Phi_k \mathbf{P}_k \Phi_k^T + \mathbf{Q}_k \quad (14)$$

where  $\mathbf{Q}$  is the covariance matrices of system noise sequence vector.

When other information such as GNSS or WiFi is available, the matrix  $\mathbf{P}$  can be updated through

$$\mathbf{P}_{k+1}^+ = (\mathbf{I} - \mathbf{K}_{k+1} \mathbf{H}_k) \mathbf{P}_{k+1}^{-1} \quad (15)$$

with

$$\mathbf{K}_{k+1} = \mathbf{P}_{k+1} \mathbf{H}_k^T (\mathbf{H}_k \mathbf{P}_{k+1} \mathbf{H}_k^T + \mathbf{R}_{k+1})^{-1} \quad (16)$$

where  $\mathbf{K}$  is the Kalman gain matrix,  $\mathbf{H}$  is the design matrix for measurements, and  $\mathbf{R}$  is the covariance matrices of measurement error vector.

### 3. WiFi Fingerprinting

WiFi fingerprinting is the widely used positioning approach based on WiFi RSS. The main advantage is that it does not rely on known WiFi AP position and signal propagation environment. WiFi fingerprinting consists of two phases: offline training phase and online positioning phase. The procedure of WiFi fingerprinting is shown in Figure 2 [59].

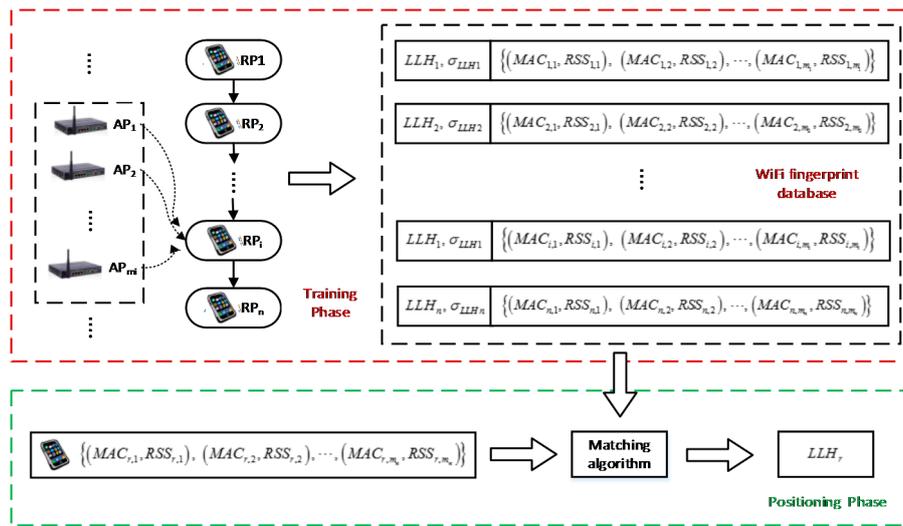


Figure 2. Procedure of WiFi fingerprinting.

#### 3.1. Training Phase

The purpose of training phase is to build or update a <location, RSS> DB that consists of a set of RPs with known coordinates and RSS from available WiFi APs. To generate DB, enough RPs should be selected to cover the whole area of interest. Then, the RSS from available APs are collected at every RP. The procedure for training is illustrated in the red dashed box in Figure 2. The fingerprint information at the  $i$ -th RP is recorded as

$$F_i = \left\{ LLH_i, \sigma_{LLH_i}, (MAC_{i,1}, RSS_{i,1}), (MAC_{i,2}, RSS_{i,2}), \dots, (MAC_{i,m_i}, RSS_{i,m_i}) \right\} \quad (17)$$

where  $F_i$  is fingerprint information at  $RP_i$ ,  $LLH_i$  and  $\sigma_{LLH_i}$  are the coordinate of  $RP_i$  and its accuracy,  $MAC_{i,j}$  and  $RSS_{i,j}$  are the MAC address and RSS of the  $j$ -th AP received at  $RP_i$ , and  $m_i$  is the number of available APs at  $RP_i$ .

In this paper, we use the continuous sensor-based navigation solution from the method described in Section 2 to obtain the position of RPs. Therefore, it is possible to train DB with daily-life navigation data from users, instead of performing a separate training phase. This is similar as [34,39,40].

### 3.2. Positioning Phase

In the positioning phase, the user location is estimated by comparing the RSS information with that stored in the DB. The procedure for positioning is illustrated in the green dashed box in Figure 2. Several approaches have been proposed for estimating the user location using RSS measurements, such as the nearest neighbor (NN) approach and the probabilistic estimation method [60]. The NN method selects the RP which has the minimal signal strength distance as the user's estimated position which is calculated as follows [31]:

$$d_i = \sqrt{\sum_{j=1}^{n_i} |SS_{rec,l_u}^j - SS_{DB,i}^j|}, \quad i \in I_{RP} \quad (18)$$

where,  $d_i$  is the signal strength distance at  $RP_i$  in the DB,  $SS_{rec,l_u}$  is the measured RSS vector at  $l_u$ ,  $SS_{DB,i}$  is the RSS vector at  $RP_i$ ,  $n_i$  is the number of WiFi signals received at  $RP_i$ , and  $I_{RP}$  is the location index set of RPs in the DB. Then the coordinates of  $RP_i$  which satisfies the condition  $d_{i^*} = \min(d_i | i \in I_{RP})$  is determined as the position estimation of  $l_u$ .

We use a multi-level quality control mechanism to optimize WiFi positioning. The first is the measurement level, where a threshold  $Th_{RSS}$  is set to filtering out APs with weak RSS. The second level is based on the minimal signal strength distance. If the minimal signal strength distance at a certain epoch is larger than the given threshold  $Th_d$ , we will not use the fingerprinting results at this epoch because probably the current user location has not been stored as a RP in the DB.

Furthermore, to mitigate the impact of blunder and get a more reliable positioning result, the  $k$ -NN estimation technique is considered [59], which estimates the position according to the  $k$  RPs that have the smallest distances. The position estimate is obtained by a weighed sum of the positions of the nearest RPs by

$$\hat{\mathbf{r}} = \sum_{i=1}^k \frac{c_i}{C} \mathbf{r}_i \quad (19)$$

where  $c_i = 1/d_i$ ,  $C = \sum_{i=1}^k c_i$ ,  $\mathbf{r}_i$  is the position of the  $i$ -th nearest RP,  $\hat{\mathbf{r}}$  is the estimated position.

## 4. Tests and Results

Walking tests were conducted in two indoor environments: the Energy, Environment, and Experiential learning (EEEL) building, in which the average weighted AP number (as defined in Section 4.1.2) at one point was over 15; and the Engineer building (ENB), in which the average weighted AP number was nearly seven. The tests were performed with a Samsung Galaxy 4 smartphone. A designed software was used to simultaneously receive the data from sensors, WiFi, and GNSS. The sample rate of sensors, WiFi, and GNSS data were set as 20 Hz, 1 Hz, and 1 Hz, respectively. Even though our attitude estimation algorithm can dealing with different scenarios such as handheld, ear, dangling, pocket, and backpack [61], we conducted the tests in this paper with only handheld mode to focus on investigating WiFi positioning.

#### 4.1. Tests at EEEL

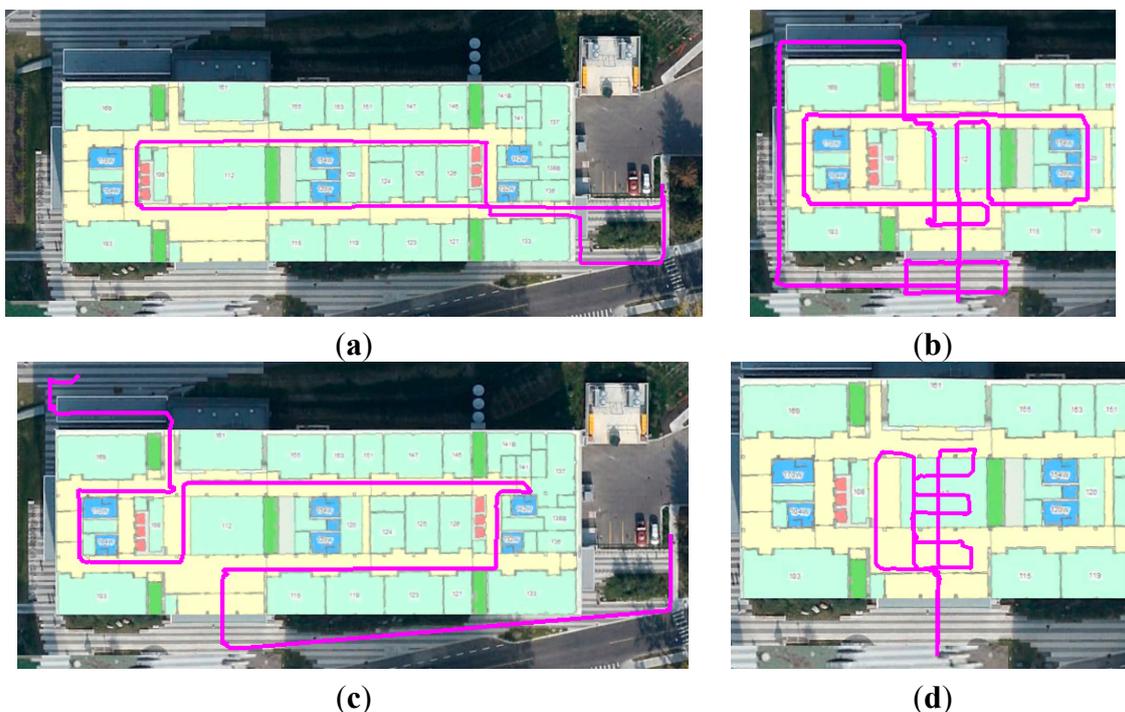
EEEL is a relatively new building with well-equipped infrastructures. There are metallic infrastructures inside the building, which may influence the propagation of WiFi signals. Figure 3 shows the indoor test environment. This building has a main corridor which is 3 m wide, and a lobby which is about  $30 \times 30 \text{ m}^2$ . The test area at EEEL was approximately  $120 \times 40 \text{ m}^2$ .



**Figure 3.** Indoor test environment at EEEL.

##### 4.1.1. Trajectories for Building DB

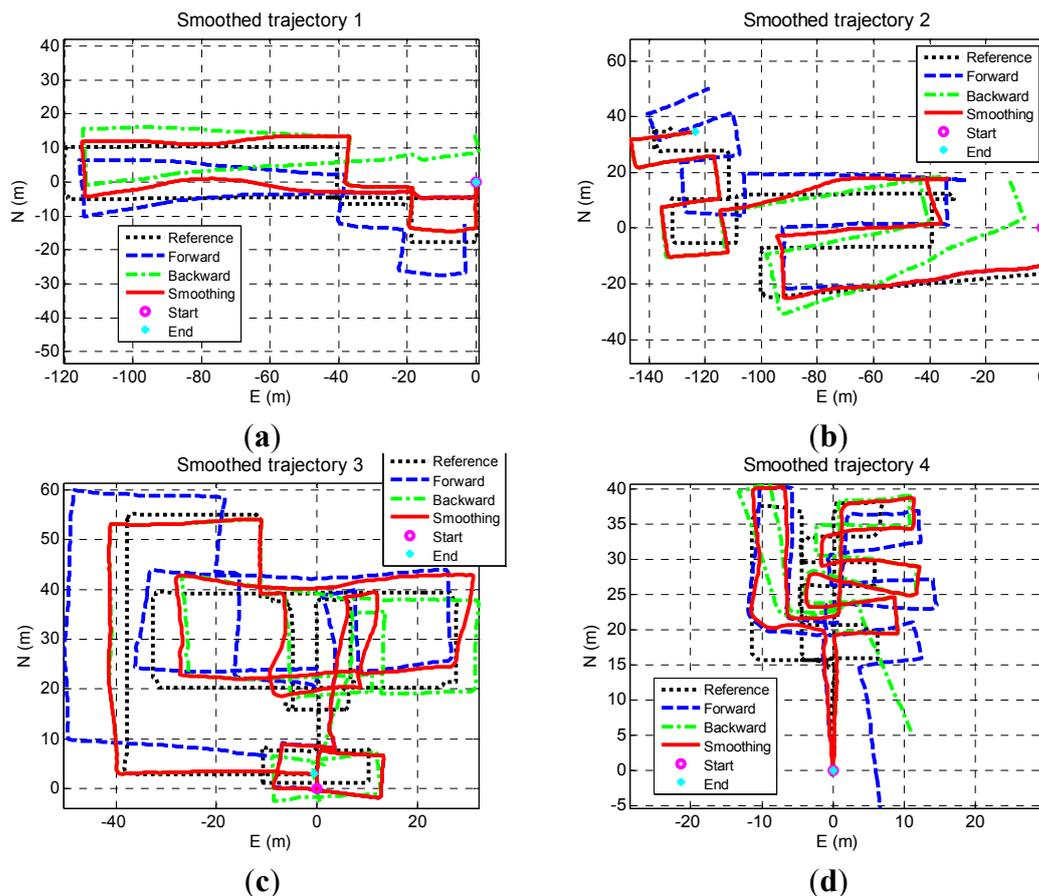
In this test, we generated the WiFi fingerprint DB inside the EEEL building using four different sensors stand-alone navigation trajectories. The true trajectories are shown in Figure 4. Each trajectory lasted for 5–10 min. Both the starting and ending points of each trajectory were in the outdoor environment, where the initial position was provided by GNSS.



**Figure 4.** Different trajectories used to generate WiFi DB (a) Trajectory 1; (b) Trajectory 2; (c) Trajectory 3; (d) Trajectory 4.

#### 4.1.2. Building WiFi DB Using Sensor-Based Navigation Solutions

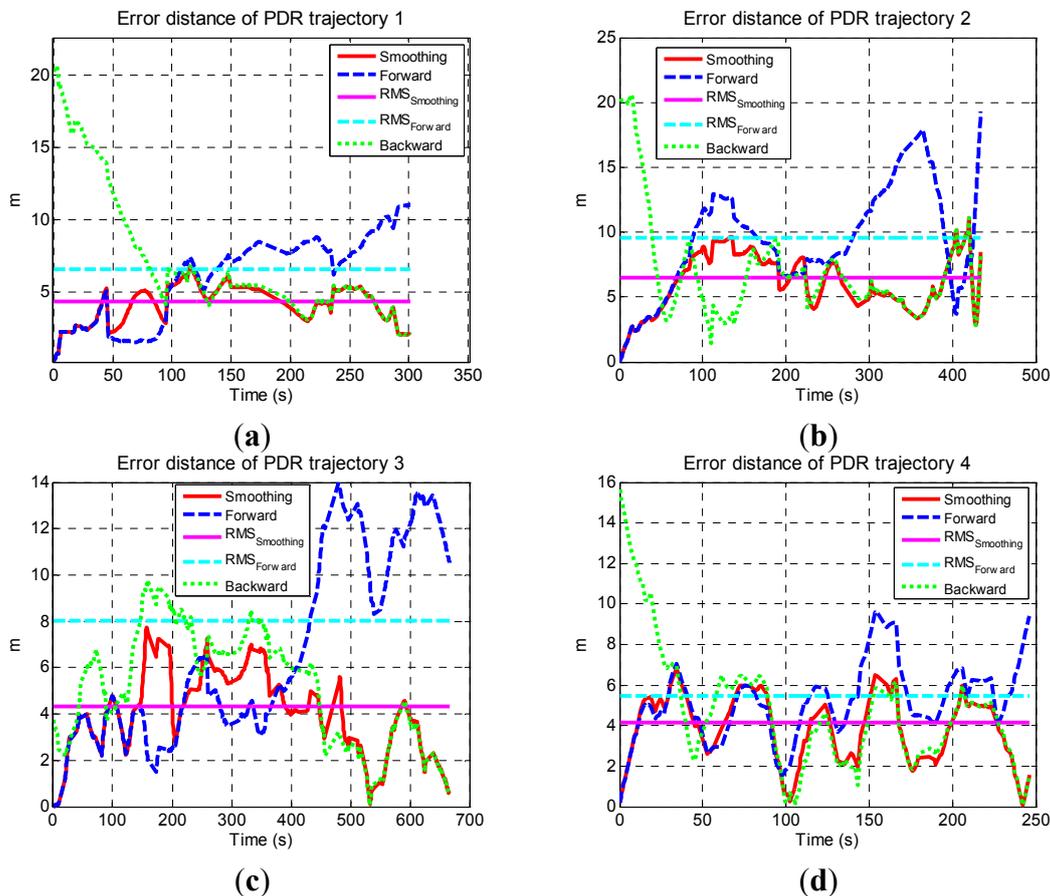
Figure 5a–d shows the sensor-based navigation solutions in a local geographic frame. In each figure, the blue dash line, green dashed line, red solid line, and black dotted line are the forward, backward, smoothed result, and the reference. The reference is provided by correcting the PDR solution aided by floor plan (*i.e.*, the true position of corners and intersections, and the true orientation of corridors). The floor plan information was obtained from Google Earth. The start and end points indicates the starting points of forward and backward PDR, respectively.



**Figure 5.** Sensor-based navigation solutions (a) Trajectory 1; (b) Trajectory 2; (c) Trajectory 3; (d) Trajectory 4.

Figure 5 shows that both forward and backward PDR trajectories had a similar shape with the reference and are accurate at the beginning, but suffered from long-term drifts. The drifts were caused by both heading and step length errors, which are the issues inherent in the sensors-only navigation algorithm. The smoothed trajectory significantly got closed to the reference at beginning and ending periods since forward and backward PDR solutions were accurate and had high weight at beginning and ending periods, respectively. To make the errors clear, the error distances (*i.e.*, the distance between estimated user position and the corresponding true position) of these solutions are illustrated in Figure 6a–d. In each figure, the blue dashed line, green dotted line, and red solid line represent the error distances of forward, backward, and smoothing result, respectively. The magenta solid line and cyan dashed line indicate the RMS values of forward and smoothed results.

The smoothed results were accurate at the beginning and ending parts of each trajectory, and had more drifts in the middle, which meets the characteristics of smoothing algorithm. By comprehensive using the data during the whole navigation process, the smoothed results were more accurate than the forward. The max and RMS values of the errors in forward and smoothed results are shown in Table 1. Each element in the final row is the RMS of the values of four trajectories in the corresponding column.



**Figure 6.** Error distances of sensor-based navigation solutions (a) Trajectory 1; (b) Trajectory 2; (c) Trajectory 3; (d) Trajectory 4.

**Table 1.** Statistical values of sensor-based navigation errors.

Trajectory	Errors in Forward Results		Errors in Smoothed Results		RMS Reduction
	Max	RMS	Max	RMS	
1	11.0	6.6	6.7	4.3	34.9%
2	19.3	9.5	11.1	6.5	31.6%
3	14.0	8.0	7.7	4.3	46.3%
4	9.6	5.5	7.0	4.1	25.5%
RMS	16.1	8.7	9.6	5.7	34.5%

The max and RMS values of error distance were generally reduced from 16.1 m and 8.7 m to 9.6 m and 5.7 m after smoothing, with a reduction of 34.5%. Then, the smoothed navigation solutions were used to build the WiFi fingerprint DB. The DB built through the proposed method (which will be denoted as “the proposed DB” for short) is shown in Figure 7a. To make a comparison, a reference DB was built

by using the conventional floor plan aided training approach (which will be denoted as “the reference DB” for short) and shown in Figure 7b. It is clear that the proposed DB has some shifts when comparing with the true path, while the reference DB fits the true path. We will evaluate the effect of such DB shift on WiFi positioning errors by WiFi positioning tests in the next subsection.

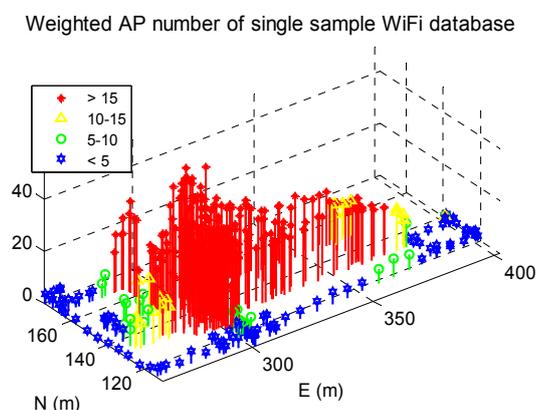


**Figure 7.** The DB built through proposed method (a), and the reference DB built by using the conventional floor plan aided training approach (b).

In addition, WiFi signal distribution in the reference DB was shown in Figure 8. The  $x$ - and  $y$ -axis indicate the length in the west-east and south-north directions, and the  $z$ -axis show the weighted AP number, which is calculated by

$$WAP_i = \sum_{j=1}^{n_i} a_{i,j}, \quad i \in I_{RP} \quad (20)$$

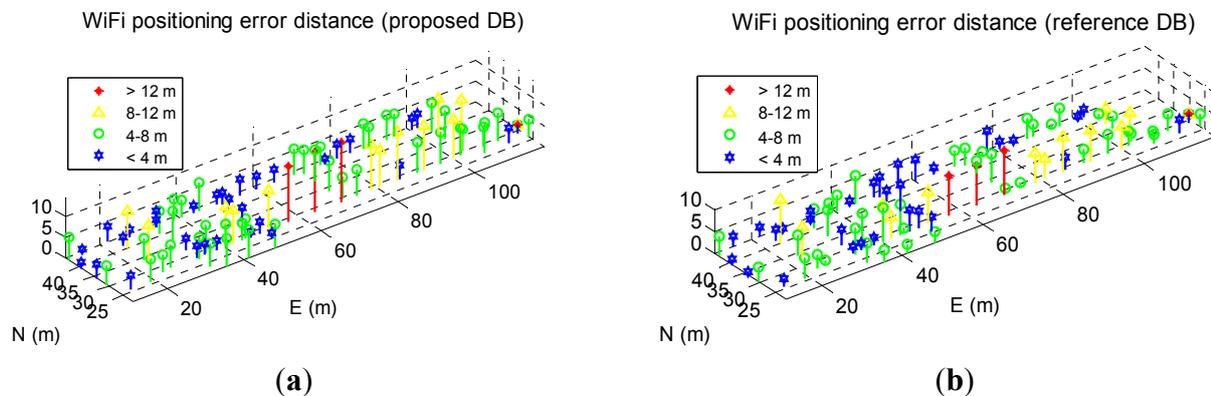
where  $WAP_i$  is the weighted AP number at  $RP_i$  in the DB,  $n_i$  is the number of WiFi signals received at  $RP_i$ ,  $I_{RP}$  is the location index set of RPs in the DB. The value of  $a_{i,j}$  is determined according to  $RSS_{i,j}$  (i.e., the RSS of  $AP_j$  at  $RP_i$ ) by the following rule: if  $RSS_{i,j} > -60$  dBm,  $a_{i,j} = 1$ ; else if  $-70$  dBm  $< RSS_{i,j} < -60$  dBm,  $a_{i,j} = 0.75$ ; else if  $-85$  dBm  $< RSS_{i,j} < -70$  dBm,  $a_{i,j} = 0.25$ ; else if  $RSS_{i,j} < -85$  dBm,  $a_{i,j} = 0$ . Compared with Figure 7b, the available WiFi signals were abundant in the middle area of EEEL, lesser but still enough in the marginal indoor areas, and even less in outdoor areas.



**Figure 8.** WiFi signal distribution in reference DB.



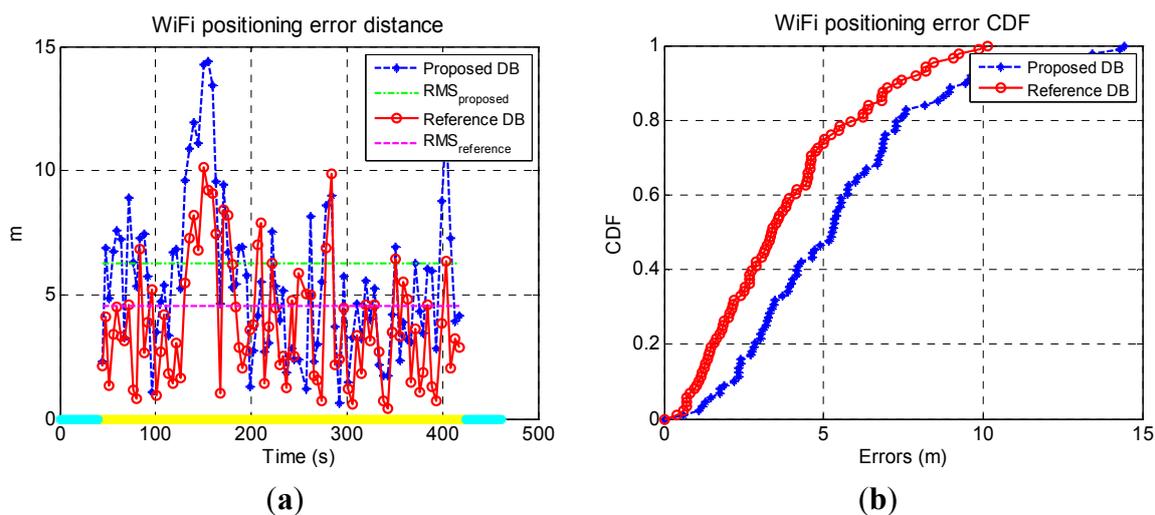
Figure 11a,b show the WiFi positioning errors on the navigation trajectory, where the  $x$ - and  $y$ -axis indicate the length of the trajectory in the west-east and south-north directions, and the  $z$ -axis is the positioning error distances.



**Figure 11.** WiFi fingerprinting error distribution when using proposed DB (a) and using reference DB (b).

The largest WiFi positioning errors existed in the middle area with abundant WiFi signals, instead of in the marginal areas with fewer WiFi signals, which does not meet our intuition. A possible reason for this is that the middle area has strongest signals, which causes ambiguity issue around this area. Integrating PDR with WiFi may probably mitigate the ambiguity issue, which is beyond the scope of this article.

To further evaluate the WiFi positioning errors, the time series of the errors are illustrated as blue dots and lines in Figure 12a. The RMS values of the errors are also calculated and shown as magenta lines. The yellow and blue lines on the  $x$ -axis indicate the indoor and outdoor environment, respectively. The statistical error CDF of both solutions are shown in Figure 12b.



**Figure 12.** Time series of WiFi fingerprinting errors (a) and corresponding CDF (b).

With a relatively accurate DB (the reference DB), the max error and RMS error of WiFi positioning were 10.1 m and 4.6 m. These values increased to 14.2 m and 6.1 m when using the proposed DB. Therefore, using the proposed DB which had shifts with a RMS of 5.7 m, the final fingerprinting errors

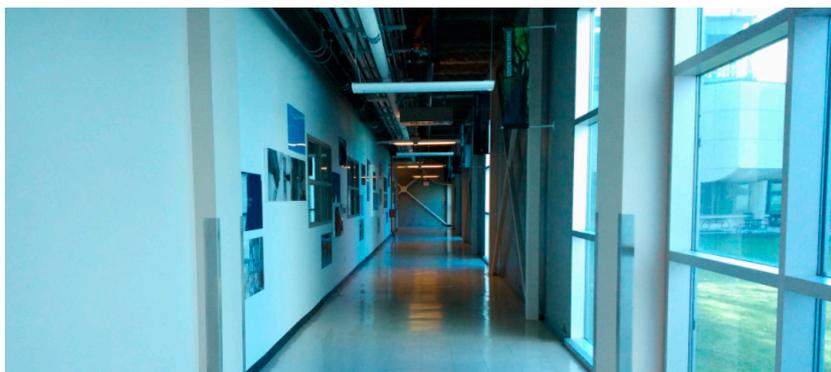
increased for 4.1 m in max value and 1.5 m in RMS. This indicated that there were other error sources in smartphone-based WiFi positioning which were more significant than such level of DB shifts. We can also see that the WiFi positioning results with both the proposed DB and the reference DB has large errors during 100 ~ 200 s, which indicates that such large errors may be introduced by the error sources in the WiFi positioning process, instead of the DB errors.

There is approximately 80% fewer errors than 7.5 m when using the proposed DB, while around 80% fewer errors than 6.0 m when using the reference DB. Therefore, the sample WiFi positioning results using the proposed DB were also 1.5 m worse than those using the conventional floor plan aided DB.

## 4.2. Tests at ENB

### 4.2.1. Trajectory for Building DB

ENB is mainly used for walking. Therefore, the environment at ENB is different from that at EEEL. Compared with EEEL, there are fewer WiFi APs and fewer metallic infrastructures at ENB. Figure 13 shows the indoor test environment. The test area at ENB was approximately  $140 \times 60 \text{ m}^2$ .



**Figure 13.** Indoor test environment at ENB.

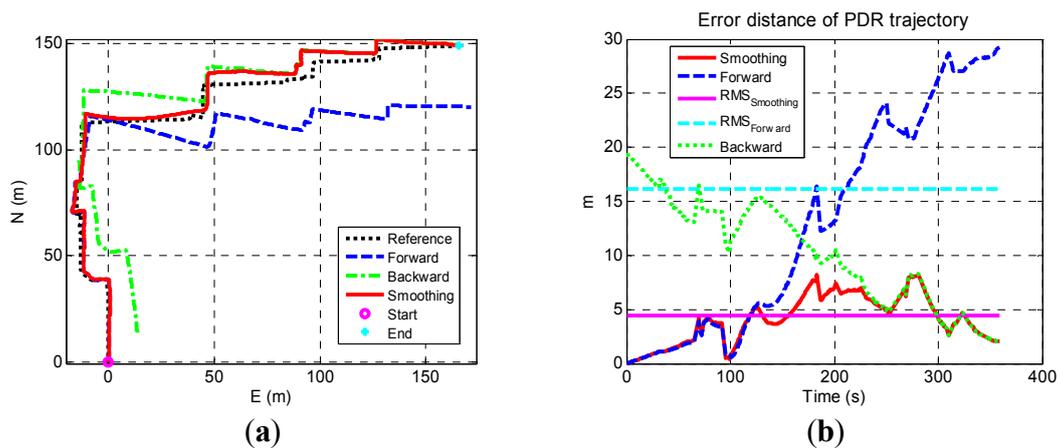
The corridors inside ENB is long and narrow, which is different from those in EEEL. The test trajectory is shown in Figure 14.



**Figure 14.** The trajectories used to generate WiFi DB in ENB.

#### 4.2.2. Building WiFi DB Using Sensor-Based Navigation Solution

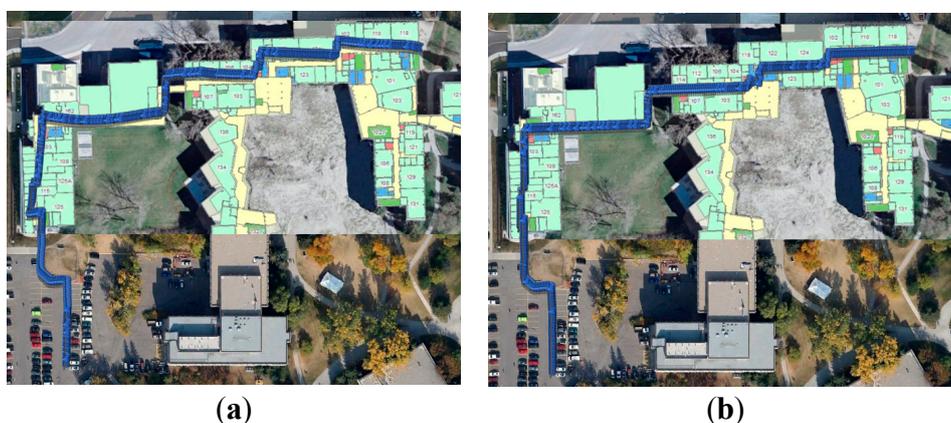
The sensor-based navigation solutions are shown in Figure 15a. The blue dashed line, green dashed line, red solid line, and black dotted line are the results of forward, backward, smoothed solution, and the reference, respectively. The start and end points indicates the starting points of forward and backward PDR. The error distances of these solutions are illustrated in Figure 15b. The blue dashed line, green dotted line, and red solid line represent the error distances of forward, backward, and smoothed results, respectively. The magenta solid line and cyan dashed line indicate the RMS values of the forward the smoothed results.



**Figure 15.** Sensor-based navigation solutions (a) and corresponding error distances (b).

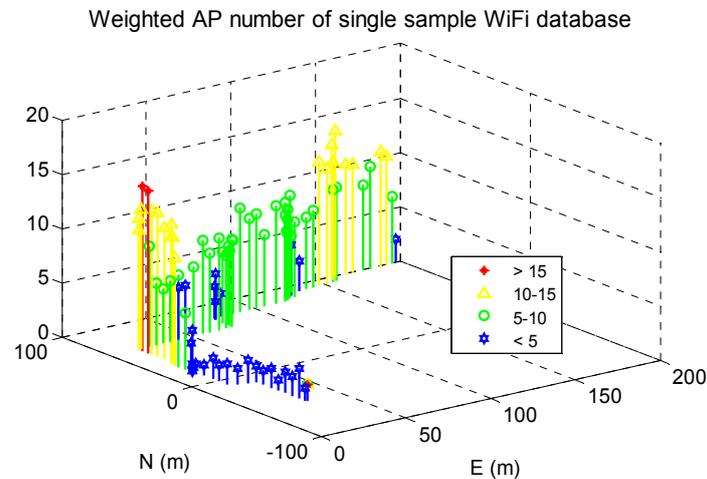
Due to the straightforward and long shape of the ENB trajectory, the navigation error at the end of the forward and backward PDR reached 25 m and 20 m, which is larger than those in the EEEL tests. However, the max error and RMS error were reduced from 29.2 m and 15.9 m in forward results to 8.1 m and 3.9 m in smoothed results.

The smoothed navigation solutions were used to build the WiFi fingerprint DB, as shown in Figure 16a. The reference DB generated through the conventional floor plan aided method is shown in Figure 16b. It is clear that the proposed DB has some shifts in the middle part even after smoothing when comparing with the true floor plan.



**Figure 16.** The DB built through proposed method (a), and the reference DB built by using the conventional floor plan aided training approach (b).

The WiFi signal distribution in the DB was shown in Figure 17. There are generally many fewer WiFi signals on this trajectory than those in EEEL because the tests were conducted on the 0-th floor, which is mainly used for walking instead of working.



**Figure 17.** WiFi signal distribution in the reference DB.

#### 4.2.3. WiFi Fingerprinting Using Generated DB

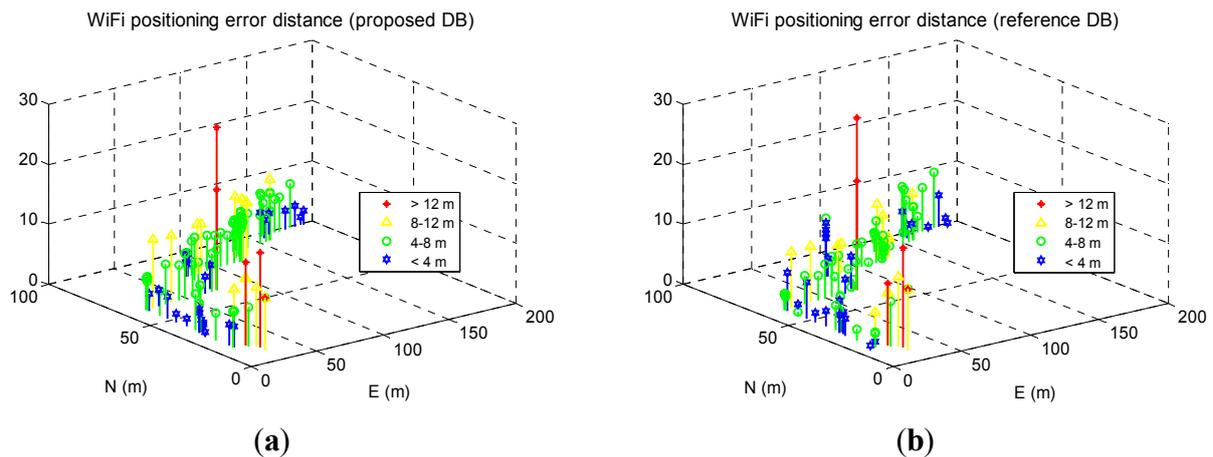
In this WiFi positioning test, we use the same trajectory as the DB training tests, but walked in the opposite direction. The WiFi fingerprinting results with the proposed DB and the conventional DB were shown in Figure 18a,b respectively.



**Figure 18.** WiFi fingerprinting result using proposed DB (a) and that using reference DB (b).

Some part of the WiFi positioning results shifted to the north in Figure 18a because of the shifts in the proposed DB. Figure 19 a,b show the WiFi positioning errors on the navigation trajectory, where the  $x$ - and  $y$ -axis indicate the length of the trajectory in the west-east and south-north directions, and the  $z$ -axis is the positioning error distances.

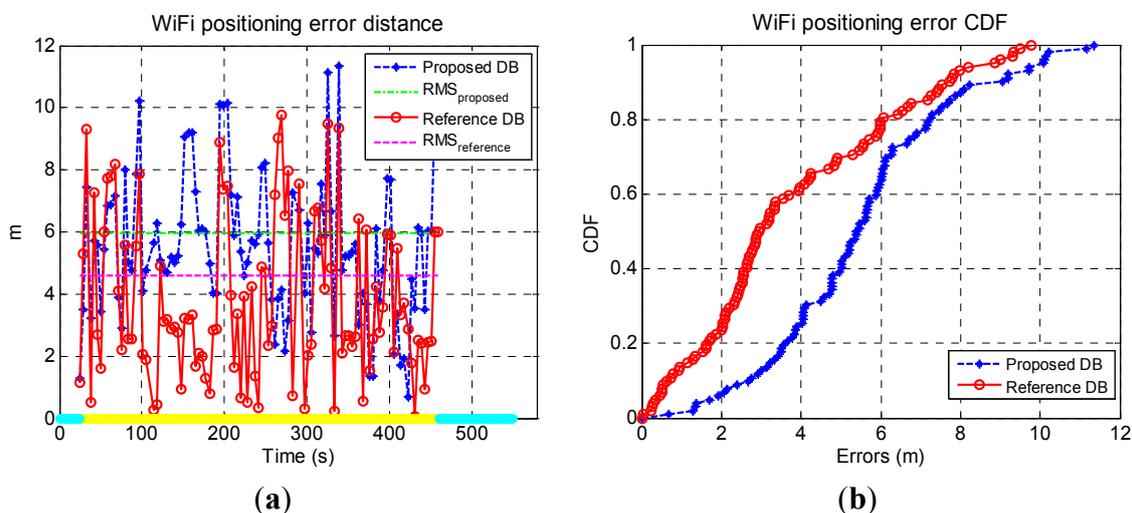
Compared with Figure 17, the largest WiFi positioning errors existed in the areas with fewer WiFi signals. To further evaluate the WiFi positioning errors, Figure 20a illustrates the time series of the WiFi positioning errors and their RMS value are illustrated, and Figure 20b shows the statistical error CDF of both solutions.



**Figure 19.** WiFi fingerprinting result using proposed DB (a) and that using reference DB (b).

The RMS of WiFi positioning errors decreased from 6.0 when using the proposed DB to 5.1 m when using the reference DB, while the max error increased from 11.4 m to 12.5 m. Therefore, even using the proposed DB which had shifts with a RMS of 3.9 m, the final WiFi fingerprinting results increased for only 0.9 m, which further verified that there were other error sources in smartphone-based WiFi positioning which is more significant than such level of DB shifts.

There are approximate 80% fewer errors than 7.1 m when using the proposed DB, while around 80% fewer errors than 6.4 m when using the conventional DB. Therefore, the sample WiFi positioning results using the proposed DB were 0.7 m worse than those using the conventional floor plan aided DB.



**Figure 20.** Time series of WiFi fingerprinting errors (a) and corresponding CDF (b).

## 5. Conclusions

This paper presents a method which trains the WiFi fingerprint database using sensor-based navigation solutions. Since MEMS sensors provide only short-term accuracy but suffer from the accuracy degradation with time, we use a strategy that combine different navigation trajectories that move in and out of a building into the database, and restrict the time length of available indoor navigation trajectories to a certain time threshold (e.g., 10 min). This simple strategy can help control the sensor-based navigation errors. In addition, we conduct post-processing (smoothing) to improve the sensor-based navigation

solution, since there are accurate GNSS position updates on the starting and ending points of indoor trajectories. Results show that the errors (RMS) of pedestrian navigation along different trajectories were reduced to 5.7 m during 5–10 min indoor walking.

Furthermore, we evaluated the effect of DB shifts on WiFi fingerprinting using the proposed DB (*i.e.*, the database generated using the proposed method). Results show that when the error of the proposed DB was 5.7 m, the WiFi positioning error was 6.1 m, which is 1.5 m larger than that with the reference DB (*i.e.*, the database built through the floor-plan aided method). In the tests inside another building, when the error of the proposed DB was 3.9 m, the WiFi positioning error was 6.0 m, which is 0.9 m larger than that with the reference DB. Therefore, the difference between the WiFi fingerprinting results with the proposed DB and the reference DB was not significant when comparing with the WiFi fingerprinting errors with smartphones. Therefore, although constructing a high-quality DB is vital as fingerprinting accuracy highly depend on DB quality, the smartphone-based fingerprinting errors will not increase linearly according to DB errors because there are other error sources.

Future works will focus on researching WiFi positioning based on crowd-sourcing, where the database accuracy can be further improved with mass data of sensor-based navigation.

## Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (Grant No. 41174028, 41231174, 41325015, 41204029 and 41375041), the National “863 Program” of China (Grant No. 2014AA123101), and the CSC Scholarship (Grant No. 201306270139).

## Author Contributions

Q.Z., X.N., and J.L. conceived and designed the research; P.Z. and Y.L. performed the research; P.Z., Y.Z., and Q.Z. analyzed the data; Y.Z. contributed materials; Q.Z, Y.L. and P.Z. wrote the paper.

## Conflicts of Interest

The authors declare no conflict of interest.

## References

1. Berg insight. LBS Research Series: Mobile Location-Based Services. Available online: <http://www.berginsight.com/ReportPDF/ProductSheet/bi-lbs8-ps.pdf> (accessed on 7 July 2014).
2. Mokbel, M.F.; Levandoski, J.J. Toward context and preference-aware location-based services. In Proceedings of the Eighth ACM International Workshop on Data Engineering for Wireless and Mobile Access, Providence, RI, USA, 29 July 2009; pp. 25–32.
3. He, Z.; Petovello, M.; Lachapelle, G. Indoor doppler error characterization for high sensitivity GNSS receivers. *IEEE Trans. Aerosp. Electron. Syst.* **2014**, *50*, doi:10.1109/TAES.2014.130111.
4. Oppermann, F.J.; Boano, C.A.; Römer, K. A decade of wireless sensing applications: Survey and taxonomy. In *The Art of Wireless Sensor Networks*; Springer: Berlin/Heidelberg, Germany, 2014; pp. 11–50.

5. Retscher, G. Pedestrian navigation systems and location-based services. In Proceedings of the Fifth IEE International Conference on 3G Mobile Communication Technologies, London, UK, 18–20 October 2004.
6. Mirowski, P.; Ho, T.K.; Yi, S.; MacDonald, M. SignalSLAM: Simultaneous localization and mapping with mixed WiFi, Bluetooth, LTE and magnetic signals. In Proceedings of the 2013 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Montbéliard, France, 28–31 October 2013.
7. Ruiz, A.R.J.; Granja, F.S.; Prieto Honorato, J.C.; Rosas, J.I.G. Accurate pedestrian indoor navigation by tightly coupling foot-mounted IMU and RFID measurements. *IEEE Trans. Instrum. Meas.* **2012**, *61*, 178–189.
8. Goncalo, G.; Helena, S. Indoor location system using ZigBee technology. In Proceedings of the Third IEEE International Conference on Sensor Technologies and Applications (SENSORCOMM'09), Athens, Greece, 18–23 June 2009.
9. Zampella, F.; de Angelis, A.; Skog, I.; Zachariah, D.; Jimenez, A. A constraint approach for UWB and PDR fusion. In Proceedings of 2012 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Sydney, Australia, 13–15 November 2012.
10. He, Z.; Renaudin, V.; Petovello, M.G.; Lachapelle, G. Use of high sensitivity GNSS receiver doppler measurements for indoor pedestrian dead reckoning. *Sensors* **2013**, *13*, 4303–4326.
11. Barnes, J.; Rizos, C.; Pahwa, A.; Politi, N.; van Cranenbroeck, J. The potential of locata technology for structural monitoring applications. *Positioning* **2007**, *1*, 166–172.
12. Fallah, N.; Apostolopoulos, I.; Bekris, K.; Folmer, E. Indoor human navigation systems: A survey. *Interact. Comput.* **2013**, *25*, 21–33.
13. Uri Schatzberg, L.B.; Banin, L.; Amizur, Y. Intel-Enhanced WiFi ToF indoor positioning system with MEMS-based INS and pedometeric information. In Proceedings of ION/IEEE PLANS, Monterey, CA, USA, 5–8 May 2014.
14. Panyov, A.A.; Golovan, A.A.; Smirnov, A.S. Indoor positioning using WiFi fingerprinting pedestrian dead reckoning and aided INS. In Proceedings of 2014 International Symposium on Inertial Sensors and Systems (ISISS), Laguna Beach, CA, USA, 25–26 February 2014.
15. Husen, M.N.; Lee, S. Indoor human localization with orientation using WiFi fingerprinting. In Proceedings of the Eighth International Conference on Ubiquitous Information Management and Communication, Siem Reap, Cambodia, 9–11 January 2014.
16. Laoudias, C.; Larkou, G.; Zeinalipour-Yazti, D.; Panayiotou, C.G. Airplace: Indoor geolocation on smartphones through WiFi fingerprinting. *Mob. Comput.* **2013**, *93*, 37.
17. Lin, P.; Li, Q.; Fan, Q.; Gao, X.; Hu, S. A real-time location-based services system using WiFi fingerprinting algorithm for safety risk assessment of workers in tunnels. *Math. Probl. Eng.* **2014**, *2014*, 1–10.
18. Farshad, A.; Li, J.; Marina, M.K.; Garcia, F.J. A microscopic look at WiFi fingerprinting for indoor mobile phone localization in diverse environments. In Proceedings of 2013 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Montbéliard, France, 28–31 October 2013.

19. Kodippili, N.; Dias, D. Integration of fingerprinting and trilateration techniques for improved indoor localization. In Proceedings of the Seventh International Conference on Wireless and Optical Communications Networks, Colombo, Sri Lanka, 6–8 September 2010; pp. 1–6.
20. Bahl, P.; Padmanabhan, V.N. RADAR: An in-building RF-based user location and tracking system. In Proceedings of Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies, Tel Aviv, Israel, 26–30 March 2000.
21. Torres-Solis, J.; Falk, T.H.; Chau, T. A review of indoor localization technologies: Towards navigational assistance for topographical disorientation. *Ambient Intell.* **2010**, *2010*, 51–84.
22. Bose, A.; Foh, C.H. A practical path loss model for indoor WiFi positioning enhancement. In Proceedings of the Sixth International Conference on Information, Communications & Signal Processing, Singapore, Singapore, 10–13 December 2007.
23. Akyildiz, I.F.; Sun, Z.; Vuran, M.C. Signal propagation techniques for wireless underground communication networks. *Phys. Commun.* **2009**, *2*, 167–183.
24. Nguyen, L.T.; Zhang, J. Wi-Fi fingerprinting through active learning using smartphones. In Proceedings of the 2013 ACM Conference on Pervasive and Ubiquitous Computing Adjunct Publication, Zurich, Switzerland, 8–12 September 2013.
25. Kannan, B.; Meneguzzi, F.; Bernardine Dias, M.; Sycara, K. Predictive indoor navigation using commercial smart-phones. In Proceedings of the 28th Annual ACM Symposium on Applied Computing, Coimbra, Portugal, 18–22 March 2013.
26. Kushki, A.; Plataniotis, K.N.; Venetsanopoulos, A.N. Kernel-Based positioning in wireless local area networks. *IEEE Trans. Mob. Comput.* **2007**, *6*, 689–705.
27. Ni, L.M.; Liu, Y.; Lau, Y.C.; Patil, A.P. LANDMARC: Indoor location sensing using active RFID. *Wirel. Netw.* **2004**, *10*, 701–710.
28. Fang, S.-H.; Lin, T.-N. Accurate indoor location estimation by incorporating the importance of access points in wireless local area networks. In Proceedings of IEEE Global Telecommunications Conference (GLOBECOM 2010), Miami, FL, USA, 6–10 December 2010.
29. Cheng, Y.-C.; Chawathe, Y.; LaMarca, A.; Krumm, J. Accuracy characterization for metropolitan-scale WiFi localization. In Proceedings of the Third International Conference on Mobile Systems, Applications and Services, Seattle, WA, USA, 6–8 June 2005.
30. Skyhook. Available online: <http://www.skyhookwireless.com/products/#optimized> (accessed on 7 July 2014).
31. Cheng, J.; Yang, L.; Li, Y.; Zhang, W. Seamless outdoor/indoor navigation with WIFI/GPS aided low cost Inertial Navigation System. *Phys. Commun.* **2014**, *13*, 31–43.
32. Xiang, Z.; Song, S.; Chen, J.; Wang, H.; Huang, J.; Gao, X. A wireless LAN-based indoor positioning technology. *IBM J. Res. Dev.* **2004**, *48*, 617–626.
33. Lin, H.; Zhang, Y.; Griss, M.; Landa, I. *Mobile Entity Localization and Tracking in GPS-Less Environments*; Springer: Berlin/Heidelberg, Germany, 2009; pp. 183–196.
34. Bolliger, P. Redpin-adaptive, zero-configuration indoor localization through user collaboration. In Proceedings of the First ACM International Workshop on Mobile Entity Localization and Tracking in GPS-Less Environments, San Francisco, CA, USA, 14–19 September 2008.
35. Li, Y.; Zhuang, Y.; Lan, H.; Zhang, P.; Niu, X.; El-Sheimy, N. WiFi-aided magnetic matching for indoor navigation with consumer portable devices. *Micromachines* **2015**, *6*, 747–764.

36. Ferris, B.; Fox, D.; Lawrence, N.D. WiFi-SLAM using Gaussian process latent variable models. In Proceedings of IJCAI-07, Hyderabad, India, 6–12 January 2007.
37. The SLAM problem: A survey. Available online: <http://eia.udg.es/~qsalvi/papers/2008-CCIAa.pdf> (accessed on 14 July 2015).
38. Woodman, O.; Harle, R. Rf-based initialisation for inertial pedestrian tracking. In *Pervasive Computing*; Springer: Berlin/Heidelberg, Germany, 2009; pp. 238–255.
39. Kim, Y.; Chon, Y.; Cha, H. Smartphone-based collaborative and autonomous radio fingerprinting. *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.* **2012**, *42*, 112–122.
40. Zhuang, Y.; Shen, Z.; Syed, Z.; Georgy, J.; Syed, H.; El-Sheimy, N. Autonomous WLAN heading and position for smartphones. In Proceedings of 2014 IEEE/ION Position, Location and Navigation Symposium-PLANS, Monterey, CA, USA, 5–8 May 2014.
41. Chintalapudi, K.; Padmanabha Iyer, A.; Padmanabhan, V.N. Indoor localization without the pain. In Proceedings of the Sixteenth Annual International Conference on Mobile Computing and Networking, Chicago, IL, USA, 20–24 September 2010.
42. Shaeffer, D.K. MEMS inertial sensors: A tutorial overview. *IEEE Commun. Mag.* **2013**, *51*, 100–109.
43. Zhuang, Y.; Syed, Z.; Georgy, J.; El-Sheimy, N. Autonomous smartphone based WiFi positioning system by using access points localization and crowdsourcing. *Pervasive Mob. Comput.* **2015**, *18*, 118–136.
44. Niu, X.; Li, Y.; Zhang, H.; Wang, Q.; Ban, Y. Fast thermal calibration of low-grade inertial sensors and inertial measurement units. *Sensors* **2013**, *13*, 12192–12217.
45. Titterton, D.H.; Weston, J.L. *Strapdown Inertial Navigation Technology*, 2nd ed.; The American Institute of Aeronautics and Astronautics: Reston, VA, USA, 2004.
46. Niu, X.; Li, Y.; Zhang, Q.; Cheng, Y.; Shi, C. Observability analysis of non-holonomic constraints for land-vehicle navigation systems. *J. Glob. Position. Syst.* **2012**, *11*, 80–88.
47. Afzal, M.H.; Renaudin, V.; Lachapelle, G. Use of earth's magnetic field for mitigating gyroscope errors regardless of magnetic perturbation. *Sensors* **2011**, *11*, 11390–11414.
48. Estimation techniques for low-cost inertial navigation. Available online: [http://www.ucalgary.ca/engo\\_webdocs/NES/05.20219.EHShin.pdf](http://www.ucalgary.ca/engo_webdocs/NES/05.20219.EHShin.pdf) (accessed on 14 July 2015).
49. Liu, H.; Nassar, S.; El-Sheimy, N. Two-filter smoothing for accurate INS/GPS land-vehicle navigation in urban centers. *IEEE Trans. Veh. Technol.* **2010**, *59*, 4256–4267.
50. Simón Colomar, D.; Nilsson, J.-O.; Händel, P. Smoothing for ZUPT-Aided INSs. In Proceedings of 2012 International Conference on Indoor Positioning and Indoor Navigation, IPIN, Sydney, Australia, 13–15 November 2012.
51. Li, Y.; Niu, X.; Zhang, Q.; Zhang, H.; Shi, C. An *in situ* hand calibration method using a pseudo-observation scheme for low-end inertial measurement units. *Meas. Sci. Technol.* **2012**, *23*, 105104.
52. Li, W.; Wang, J. Effective adaptive Kalman filter for MEMS-IMU/magnetometers integrated attitude and heading reference systems. *J. Navig.* **2013**, *66*, 99–113.
53. Han, S.; Wang, J. A novel method to integrate IMU and magnetometers in attitude and heading reference systems. *J. Navig.* **2011**, *64*, 727–738.
54. Saxena, A.; Gupta, G.; Gerasimov, V.; Ourselin, S. *Knowledge-Based Intelligent Information and Engineering Systems*; Springer-Verlag: Berlin/Heidelberg, Germany, 2005; pp. 595–601.

55. Zhuang, Y.; Lan, H.; Li, Y.; El-Sheimy, N. PDR/INS/WiFi integration based on handheld devices for indoor pedestrian navigation. *Micromachines* **2015**, *6*, 793–812.
56. Renaudin, V.; Susi, M.; Lachapelle, G. Step length estimation using handheld inertial sensors. *Sensors* **2012**, *12*, 8507–8525.
57. Zhao, X.; Syed, Z.; Wright, D.B.; El-Sheimy, N. An economical and effective multi-sensor integration for portable navigation system. In Proceedings of the 22nd International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS 2009), Savannah, GA, USA, 22–25 September 2009.
58. Welch, G.; G. Bishop. *An Introduction to the Kalman Filter*; University of North Carolina: Chapel Hill, CA, USA, 1995.
59. Li, B.; Salter, J.; Dempster, A.G.; Rizos, C. Indoor positioning techniques based on wireless LAN. In Proceedings of the First IEEE International Conference on Wireless Broadband and Ultra Wideband Communications, Sydney, Australia, 27–30 August 2006.
60. Lassabe, F.; Canalda, P.; Chatonnay, P.; Spies, F. Indoor Wi-Fi positioning: techniques and systems. *Ann. Telecommun.* **2009**, *64*, 651–664.
61. Li, Y.; Georgy, J.; Niu, X.; Li, Q.; El-Sheimy, N. Autonomous calibration of MEMS gyros in consumer portable devices. *IEEE Sens. J.* **2015**, *15*, 4062–4072.

© 2015 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/4.0/>).