sensors

Article

Wireless Fingerprinting Uncertainty Prediction Based on Machine Learning

You Li [1], Zhouzheng Gao [2,3,*], Zhe He [1], Yuan Zhuang [4], Ahmed Radi [1], Ruizhi Chen [4] and Naser El-Sheimy [1]

[1] Department of Geomatics Engineering, University of Calgary, 2500 University Dr NW, Calgary, AB T2N 1N4, Canada; liyou331@gmail.com (Y.L.); hezhe310@gmail.com (Z.H.); ahmed.elboraee@ucalgary.ca (A.R.); elsheimy@ucalgary.ca (N.E.-S.)

[2] School of Land Science and Technology, China University of Geosciences (Beijing), 29 Xueyuan Road, Beijing 100083, China

[3] German Research Centre for Geosciences (GFZ), Telegrafenberg, 14473 Potsdam, Germany

[4] State Key Laboratory of Surveying, Mapping and Remote Sensing, Wuhan University, 129 Luoyu Road, Wuhan 430079, China; zhy.0908@gmail.com (Y.Z.); ruizhi.chen@whu.edu.cn (R.C.)

* Correspondence: zhouzhenggao@126.com; Tel.: +86-186-2791-5172

Received: 14 December 2018; Accepted: 9 January 2019; Published: 15 January 2019

Abstract: Although wireless fingerprinting has been well researched and widely used for indoor localization, its performance is difficult to quantify. Therefore, when wireless fingerprinting solutions are used as location updates in multi-sensor integration, it is challenging to set their weight accurately. To alleviate this issue, this paper focuses on predicting wireless fingerprinting location uncertainty by given received signal strength (RSS) measurements through the use of machine learning (ML). Two ML methods are used, including an artificial neural network (ANN)-based approach and a Gaussian distribution (GD)-based method. The predicted location uncertainty is evaluated and further used to set the measurement noises in the dead-reckoning/wireless fingerprinting integrated localization extended Kalman filter (EKF). Indoor walking test results indicated the possibility of predicting the wireless fingerprinting uncertainty through ANN the effectiveness of setting measurement noises adaptively in the integrated localization EKF.

Keywords: indoor localization; fingerprinting; machine learning; neural network; received signal strength; Kalman filter; inertial navigation

1. Introduction

The location information has become essential for the location-based service (LBS) applications. Global navigation satellite systems (GNSS, e.g., Global positioning systems (GPS), BeiDou navigation satellite systems (BDS), GLONASS and Galileo) have achieved great success and commercialization during the past decades for outdoor applications. However, their performances may be degraded in indoor environments. To fill the gap for indoor mass-market localization, indoor localization with consumer portable devices has been researched.

Among various indoor localization techniques, wireless localization has been widely used due to the promotion of access points (APs, e.g., based stations, beacons, and gateways) for wireless technologies such as low-power wide-area network (LPWAN), cellular network, wireless local area network (WiFi), Bluetooth low energy (BLE), ZigBee, ultra-wide-band (UWB), and radio frequency identification (RFID). Among various wireless measurements (e.g., time-of-arrival (ToA) [1], angle-of-arrival (AoA) [2], and received signal strength (RSS) [3]), RSS does not require specific hardware for time or phase synchronization and thus has been supported by consumer electronics.

For large-scale mass-market indoor localization applications, specific devices or networks may not be affordable. Thus, the existing wireless APs in public areas are used. Additionally, to lower the requirement for the costly database training process, methods such as crowdsourcing [4] and simultaneous localization and mapping (SLAM) [5] have been investigated. Although there are numerous research that has improved the indoor wireless localization performance through advanced models or estimation techniques, the performance of indoor localization has been limited by the issues inherent to wireless measurements. Such issues include the requirement for signal geometry and the existence of RSS fluctuations and disturbances. Due to the complexity of indoor environments and the signal of opportunity (SoO) characteristic of wireless signals, it is difficult to assure wireless localization performance anywhere.

Therefore, two methods are used in this research to enhance indoor wireless localization:

- Method #1: Integration with inertial sensors. The dead-reckoning (DR) solutions from inertial sensors are used to enhance the continuity and robustness of localization.
- Method #2: Setting the wireless localization uncertainty adaptively in data fusion. To achieve this objective, machine learning (ML) methods are used to predict the wireless localization uncertainty, which is further used to set the weight of wireless position updates.

For Method #1, DR is self-contained to provide seamless outdoor/indoor localization solutions [6]. However, low-cost sensors suffer from significant sensor errors. The uncompensated sensor errors may accumulate and cause increasing position errors due to the integral computation in DR algorithm. Therefore, it is vital to correct for DR errors through the integration with external sensors such as wireless localization. In such integration, the DR data is used to build the system model and provide a short-term prediction, while wireless localization solutions are the updates. Kalman filter and particle filter [6] are widely used techniques for information fusion. Profile matching [7] is also used to provide a robuster solution.

For Method #2, there are methods that use adaptive Kalman filters based on variables such as residuals [8] and innovations [9]. These methods are effective in reducing the impact of unreliable wireless measurements when the DR solution is reliable. However, due to the dependency on the DR solution, their performance may be degraded when both wireless and DR data are not accurate. Therefore, one objective of this paper is to estimate wireless localization uncertainty without the need for DR solutions. The research [10] predicts the fingerprinting-based localization uncertainty from three levels (i.e., the signal, geometry, and database levels). In contrast, this paper predicted the localization uncertainty through the use of ML. Two ML methods are used, including the artificial neural network (ANN)-based method and the Gaussian distribution (GD) based method.

ANN has been introduced into the localization area over a decade ago [11] but has not been widely adopted until recent years. Table 1 illustrates part of papers that use ANN in localization within the year of 2018. The majority of works use wireless measurements as the input, while some use data from camera [12], LiDAR [13], inertial [14], and sound [15] sensors. For wireless measurements, RSS (e.g., RSS from WiFi [16], BLE [17], ZigBee [18], RFID [19], cellular [20], and photodiode [21]), RSS features (e.g., two-dimensional RSS map [22], differential RSS [23], and RSS statistics [24]), channel information (e.g., the channel state information (CSI) [25] and channel impulse response (CIR) [26]), and angle-of-arrival (AoA) [27] have been used. These measurements are used for various purposes (i.e., outputs) through ANN. The majority of works directly output two-dimensional or three dimensional locations, while the other works also output information such as attitude angles [28], floor identifications [24], room identifications [16], region identifications [15], AoA [29], distances [18], step lengths [14], and moving status [30]. Additionally, data quality and status indexes such as non-line-of-sight (NLoS) [25], similarity of fingerprints [31] and images [32], localization errors [13] and localization success rate [33] may be generated from ANN.

Table 1. Part of existing works that use ANN to improve localization (within the year of 2018).

Method	Input	Output	ANN Type/Algorithm	Hidden Layer
[34]	RSS, WiFi	Floor index and location	N/A	N/A
[35]	RSS, WiFi	Location	FF	1–3
[16]	RSS, WiFi	Room index and location	SCG and RBP	2–4
[36]	RSS, WiFi	Location	GAN	3
[37]	RSS, WiFi	Room index	N/A	3
[38]	RSS, WiFi	Location	N/A	1
[39]	RSS, WiFi	Region index	CPN	2
[17]	RSS, BLE	Location	RBF	1
[18]	RSS, ZigBee	Distance	ANFIS	3
[21]	RSS, photodiode	Cell index	CNN	2
[20]	RSS, cellular	Location	MLP	1
[19]	RSS, RFID	Location	FF	2
[31]	RSS	Fingerprint similarity	N/A	1
[40]	RSS	Location	N/A	1
[22]	RSS map	Room index and location	CNN	8
[41]	RSS map	Location	CNN	3
[23]	Differential RSS	Location	RBF	1
[24]	RSS statistics	Floor index	MLP	1
[42]	CSI, WiFi	Location	GCC	N/A
[25]	CSI, WiFi	NLoS identification	RNN	10
[43]	CIR	Location	CNN	3
[26]	CIR, UWB	NLoS identification	CNN	6
[27]	AoA	Location	CNN	8
[29]	GCC	AoA	GCC	2
[15]	Sound	Region index	CNN	10
[44]	Sound	AoA	TDNN	3
[13]	Laser data	Location error	RBF	1
[32]	RGB image	Image similarity	CNN	5
[12]	RGB image	Relation between images	CNN	2
[45]	RGB image	pose	CNN	8
[28]	RGB image	pose	CNN	3
[33]	RGB image, likelihood model, BM model	Localization success rate	CNN	9
[14]	Inertial sensor data	step length	N/A	2–4
[30]	Inertial sensor data	static detection	RNN	4

ANN types or algorithms: CNN-convolution neural network; FF-feed-forward neural network; RBF-radial basis function neural network; BP-back-propagation neural network; GAN-generative adversarial neural network; RNN-recurrent neural network; MLP-multi-layer perceptron; SCG-scaled conjugate gradient; RBP-resilient back propagation; CPN-counter-propagation neural network; ANFIS-adaptive neural fuzzy inference system; GCC-generalized cross-correlation; TDNN-time delay neural network; N/A-not provided.

When investigated on the types of ANN used, CNN [21] is the most widely used. Meanwhile, other types such as RNN [25], RBF [23], and MLP [20] have been used in multiple works. There are also ANN types or algorithms such as GAN [36], CPN [39], ANFIS [18], GCC [29], and TDNN [44]. The majority of works use one to three hidden layers.

Besides ANN, there are ML methods that have been used for indoor fingerprinting. These approaches include nearest neighbors [46], GD-based methods [47], Gauss Process model [48], random forest [49], and support vector machine [50]. When using such methods for wireless fingerprinting, RSS measurements are used as inputs while two-dimensional locations are the outputs. The principle for wireless fingerprinting is to evaluate the similarity between the measured RSS vector and those stored in the database by calculating the likelihood for each RP. As one of the most widely used methods, the GD-based method is selected as the second tool to predict wireless fingerprinting uncertainty in this paper.

Compared to the existing wireless fingerprinting works, the main contributions of this paper are

- Although wireless fingerprinting has been widely used for indoor localization, its performance is difficult to quantify. Thus, this paper predicts fingerprinting-based location uncertainty by given RSS measurements. Two ML methods, including an ANN-based method and the GD-based method, are applied.
- Compared to the existing ML works, this paper uses ML from a new perspective. Specifically, instead of directly estimating the location or navigation states, this paper uses ML to learn and predict the relation between RSS and localization uncertainty. The ML-predicted location uncertainty is further used to set the measurement noises in the dead-reckoning/wireless fingerprinting integrated localization extended Kalman filter (EKF).

This paper is organized as follows. Section 2 illustrates the methodology, including wireless fingerprinting, GD, ANN, and DR/wireless integrated localization. Section 3 describes the tests and results, and Section 4 draws the conclusions.

2. Methodology

As shown in Figure 1, an EKF is used to fuse the inertial sensor data and RSS. Data from inertial sensors is used to predict the states through the INS mechanization and construct the EKF system model, while wireless fingerprinting provides the measurement models. Compared to traditional DR/Wireless fingerprinting integration methods, this paper introduces the fingerprinting uncertainty prediction module, as illustrated by the blue boxes in Figure 1. The fingerprinting uncertainty prediction module is based on ML, which inputs RSS and outputs the predicted fingerprinting-based location uncertainty. The methods of wireless fingerprinting and GD, ANN, and DR/Wireless fingerprinting integrated EKF are described in Sections 2.1–2.3, respectively.

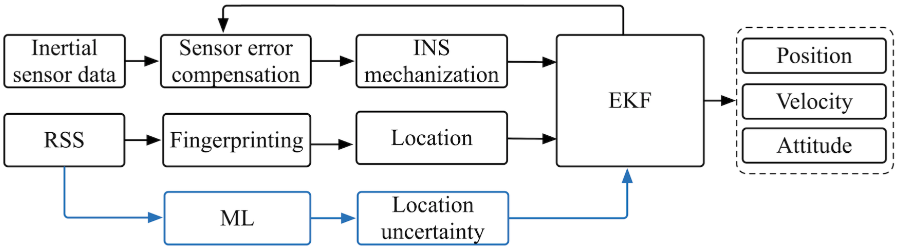

Figure 1. Algorithm flow chart.

2.1. Wireless Fingerprinting

Fingerprinting consists of the training and localization steps. The training step is conducted to generate a [location, RSS] database that consists of a set of reference points (RPs) with known coordinates and RSS from available APs, while the localization step finds the closest match between the measured RSS and those stored in the database. The reference fingerprint in the database can be written as

$$\lambda_i = \left[\mathbf{p}_i,\ \left(\xi_{i,1}, r_{i,1}\right),\ \left(\xi_{i,2}, r_{i,2}\right),\ \ldots,\ \left(\xi_{i,j}, r_{i,j}\right),\ \ldots,\ \left(\xi_{i,n_i}, r_{i,n_i}\right)\right] \tag{1}$$

where λ_i is the reference fingerprint at the ith RP in the database and $\mathbf{p}_i$ is the coordinate of RP i. $\xi_{i,j}$ and $r_{i,j}$ are respectively the index (e.g., media access control address) and the RSS with AP j at RP i. n_i is the number of available APs at RP i.

The objective for localization is to find the reference fingerprint λ that maximizes $\beta(\boldsymbol{\lambda}|\mathbf{x})$ by

$$\hat{\boldsymbol{\lambda}}_{MLE} = \arg\max_{\lambda}\left(\beta(\boldsymbol{\lambda}|\mathbf{l})\right) \tag{2}$$

where $\mathbf{l}$ is the measured RSS vector. The signs $\beta(\cdot)$ and $\arg\max$ represent the probability density function and the arguments of the maxima, respectively. $\beta(\boldsymbol{\lambda}|\mathbf{l})$ is the conditional probability of $\boldsymbol{\lambda}$ on $\mathbf{l}$.

Based on the Bayes' rule, the probability for the ith RP ($\beta(\boldsymbol{\lambda}_i|\mathbf{l})$) can be modeled by

$$\beta(\boldsymbol{\lambda}_i|\mathbf{l}) = \frac{\beta(\mathbf{l}|\boldsymbol{\lambda}_i)\beta(\boldsymbol{\lambda}_i)}{\beta(\mathbf{l})} \tag{3}$$

where $\beta(\mathbf{l}|\boldsymbol{\lambda}_i)$ is the likelihood, $\beta(\boldsymbol{\lambda})$ is the prior and $\beta(\mathbf{l})$ is a normalizing constant. A uniform $\beta(\mathbf{l})$ is used.

Denote $\mathbf{l} = [l_1 \; \cdots \; l_j \; \cdots \; l_n]$, where l_j is the measured RSS with AP j, and n is the number of APs in the RSS vector. Assume RSS from the APs are independent with one another, $\beta(\mathbf{l}|\mathbf{f}_i)$ can be simplified to

$$\beta(\mathbf{l}|\boldsymbol{\lambda}_i) = \beta(l_1 \; \cdots \; l_j \; \cdots \; l_n|\boldsymbol{\lambda}_i) = \prod_{j=1}^{n} \beta(l_j|r_{i,j}) \tag{4}$$

The likelihood $\beta(l_j|r_{i,j})$ may be computed by several approaches, such as the histogram [51], GD [47], and log-normal [52] based methods. When the GD-based method is used, $\beta(l_j|r_{i,j})$ is modeled as

$$\beta(l_j|r_{i,j}) = \frac{1}{\sigma_{i,j}\sqrt{2\pi}} \exp\left(-\frac{(l_j - \mu_{i,j})^2}{2\sigma_{i,j}^2}\right) \tag{5}$$

where $\mu_{i,j}$ and $\sigma_{i,j}^2$ are the mean and variance of RSS with AP j at RP i. These values are calculated from static RSS data at the training step.

When using the GD-based method to predict location uncertainty, a [location, RSS] database was additionally trained. Afterwards, the likelihood for each RP was computed by using Equation (3). The κ RPs that had the highest likelihood values are selected. The κ reference RSS vectors at the selected RPs were then fed into the [RSS, location uncertainty] database to obtain the corresponding location uncertainty values. Finally, the predicted location uncertainty was computed as

$$\hat{\epsilon} = \sum_{i=1}^{\kappa} \frac{\epsilon_i \beta_i}{\sum_{j=1}^{\kappa} \beta_j} \tag{6}$$

where $\hat{\epsilon}$ is the predicted position uncertainty value, ϵ_i is the location uncertainty for the ith selected RP and β_i is its likelihood.

2.2. Artificial Neural Network

ANN is actually the framework for using ML algorithms to process complex data inputs. ANN consists of an input layer, an output layer, and at least one hidden layer. Each layer consists of at least one neuron. In this paper, two hidden layers are used, as demonstrated in Figure 2. The neurons are connected via weights that are considered the bulk of the trained neural network in order to estimate the desired output.

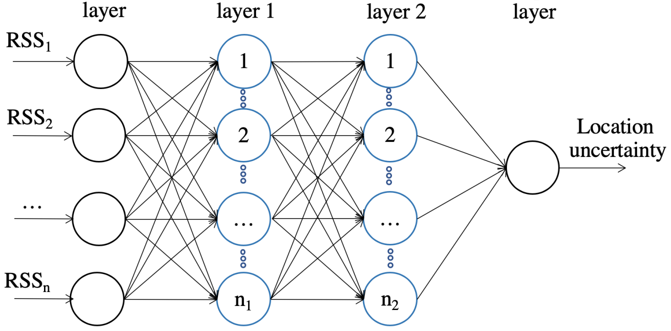

Figure 2. ANN structure.

This paper applies the multi-layer perceptron (MLP) [53], which is a supervised learning method by using an error back-propagation algorithm. The back-propagation algorithm optimizes the parameters by minimizing the sum of squared errors (i.e., the cost function c) of the neurons in the output layer as

$$c = \frac{1}{2m}\sum_{i=1}^{m} ||\tilde{\mathbf{y}}(\mathbf{x}_i) - \hat{\mathbf{y}}^{\iota}(\mathbf{x}_i)||^2 \tag{7}$$

where $\mathbf{x}_i$ represents the ith training example and $\tilde{\mathbf{y}}(\mathbf{x}_i)$ is the corresponding desired output; $\hat{\mathbf{y}}(\mathbf{x}_i)$ is the output from the ANN when $\mathbf{x}_i$ is input. ι denotes the number of layers in the ANN and m is the number of training examples.

An important factor in back-propagation is to find the optimum values for weights and biases of the ANN in order to achieve desirable outputs from given inputs. The weights and biases are updated using the four standard back-propagation equations in [54]. These equations calculate the error at the output layer according to (7) and then back-propagate this error to adjust the weights and biases based on a learning rate. Specifically, the back-propagation algorithm can be divided into five steps: input, feed-forward, output error computation, error back-propagation, and output. Refer to [55] for more details about the back-propagation algorithm.

2.3. Wireless/Dead-Reckoning Integrated Localization

As shown in Figure 1, an EKF is used to fuse the data from DR and wireless fingerprinting. The simplified psi-angle motion model in [56] is applied as the continuous system model as

$$\dot{\mathbf{x}} = \mathbf{F}\mathbf{x} + \mathbf{w} \tag{8}$$

$$\mathbf{x} = \begin{bmatrix} \delta\mathbf{p}^n & \delta\mathbf{v}^n & \boldsymbol{\psi} & \mathbf{b}_g & \mathbf{b}_a \end{bmatrix}^T \tag{9}$$

$$\mathbf{F} = \begin{bmatrix} -[\boldsymbol{\omega}_{en}^n\times] & \mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & -[(2\boldsymbol{\omega}_{ie}^n + \boldsymbol{\omega}_{en}^n)\times] & [\mathbf{f}^n\times] & \mathbf{0}_{3\times3} & \mathbf{C}_b^n \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & -[(\boldsymbol{\omega}_{ie}^n + \boldsymbol{\omega}_{en}^n)\times] & -\mathbf{C}_b^n & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & -(\frac{1}{\tau_{bg}})\mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & -(\frac{1}{\tau_{ba}})\mathbf{I}_{3\times3} \end{bmatrix} \tag{10}$$

$$\mathbf{w} = \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{C}_b^n\mathbf{n}_a & -\mathbf{C}_b^n\mathbf{n}_g & \mathbf{n}_{bg} & \mathbf{n}_{ba} \end{bmatrix}^T \tag{11}$$

where $\mathbf{x}$, $\mathbf{F}$, and $\mathbf{w}$ are the state vector, the dynamics matrix, and the system noise vector, respectively. The states $\delta\mathbf{p}^n$, $\delta\mathbf{v}^n$, $\boldsymbol{\psi}$, $\mathbf{b}_g$, and $\mathbf{b}_a$ are the vectors of position errors, velocity errors, attitude errors, gyro biases, and accelerometer biases, respectively; $\mathbf{C}_b^n$ is the direction cosine matrix, which is from the device body frame (i.e., b-frame) to the local level frame (i.e., n-frame), predicted by the inertial navigation system (INS) mechanization [56]; $\mathbf{f}^n$ is the specific force vector projected to the n-frame, and $\boldsymbol{\omega}_{ie}^n$ and $\boldsymbol{\omega}_{en}^n$ are the angular rate of the Earth and that of the n-frame with respect to the Earth frame (i.e., e-frame), respectively; $\mathbf{n}_g$ and $\mathbf{n}_a$ are noises in gyro and accelerometer readings, respectively; τ_{bg} and τ_{ba} denote for the correlation time of sensor biases; and $\mathbf{n}_{bg}$ and $\mathbf{n}_{ba}$ are the driving noises for $\mathbf{b}_g$ and $\mathbf{b}_a$. The sign $[\mathbf{v}\times]$ denotes the skew-symmetric matrix of $\mathbf{v}$. $\mathbf{0}_{3\times3}$ and $\mathbf{I}_{3\times3}$ represent the three-dimensional zero matrix and identify matrix, respectively.

To mitigate DR errors, the pedestrian velocity model [57] is used as a velocity constraint. The corresponding measurement model is

$$\hat{\mathbf{v}} - \tilde{\mathbf{v}} = \mathbf{H}_v\mathbf{x} + \boldsymbol{\zeta}_v \tag{12}$$

$$\mathbf{H}_v = \begin{bmatrix} \mathbf{0}_{3\times3} & (\mathbf{C}_b^n)^T & -(\mathbf{C}_b^n)^T[\mathbf{v}^n\times] & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \end{bmatrix} \tag{13}$$

where $\hat{\mathbf{v}}^n$ and $\tilde{\mathbf{v}}^n$ denote the velocity predicted by the system model and that obtained from the pedestrian motion, respectively; $\boldsymbol{\zeta}_v$ is the velocity measurement noise vector. The term $\tilde{\mathbf{v}}^n$ can be computed as

$$\tilde{\mathbf{v}} = [\frac{s_k}{t_k - t_{k-1}} \; 0 \; 0]^T \tag{14}$$

where s_k is the step length between time epochs t_{k-1} and t_k. Refer to [58,59] for details about step detection and step length estimation, respectively.

When Wireless fingerprinting position solutions are available, they are used as EKF location updates. The measurement model is

$$\hat{\mathbf{p}}^n - \tilde{\mathbf{p}}^n = \mathbf{H}_p\mathbf{x} + \boldsymbol{\zeta}_p \tag{15}$$

$$\mathbf{H}_p = \begin{bmatrix} \mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \end{bmatrix} \tag{16}$$

where $\hat{\mathbf{p}}^n$ and $\tilde{\mathbf{p}}^n$ denote the position predicted by the system model and that obtained from wireless localization, respectively; $\boldsymbol{\zeta}_p$ is the location measurement noise vector.

With the above system and measurement models, the EKF is applied to estimate the state vector in real time. The EKF works by predicting the process states and then obtaining feedback from noisy measurements. Refer to [56] for the details of EKF. In principle, EKF solutions are the weighted average of predictions and measurements. The weights of the predictions and measurements are reflected by the parameters in the system noise matrix and measurement noise matrix, respectively. Since the EKF system model is constructed by self-contained motion models, elements in the system noise matrix are set according to the stochastic sensor characteristics, which can be obtained from in-lab calibration. In contrast, the elements in the measurement noise matrix may vary according to navigation environment changes. Thus, the ML-predicted location uncertainty is further used to set the measurement noises in the EKF.

3. Tests and Results

3.1. Test Description

Indoor walking tests were conducted in the MacEwan Student Hall, at the University of Calgary. Five Android smart devices were used, including a Samsung Galaxy S4, a Galaxy S7, a Huawei P10, a Lenovo Phab 2 Pro smartphone, and a Nexus 9 tablet.All devices were equipped with three-axis inertial sensors and a WiFi receiver. The data rates were 20 Hz for for inertial sensors and 0.5 Hz for WiFi.

The test area was an indoor shopping mall environment, which had a size of approximately 160 m by 90 m. Figure 3 demonstrate the pictures in the test area. To train the wireless fingerprinting database, the walk-survey method was adopted. Two testers participated in the training data collection. Each tester held the five smartphones together and walked within the test area for five training trajectories, as illustrated in Figure 4. There were in total 12,867 WiFi fingerprints in the training data. Each WiFi fingerprint consisted of multiple available MAC address and RSS measurements. In the testing process, each tester held the smartphones horizontally and walked along two pre-designed testing trajectories. Each testing trajectory lasted for around 30 min. One test trajectory is shown in Figure 5, in which various colors represent different data segments with starting and ending time indicated in the legend. There were in total 7692 WiFi fingerprints in the testing data. All the training and testing trajectories started and ended at the same points (as demonstrated by the red dots in Figure 3), so as to provide a loop-closure for the whole trajectory and thus ensure the accuracy for the reference trajectory.

Figure 3. Test environment.

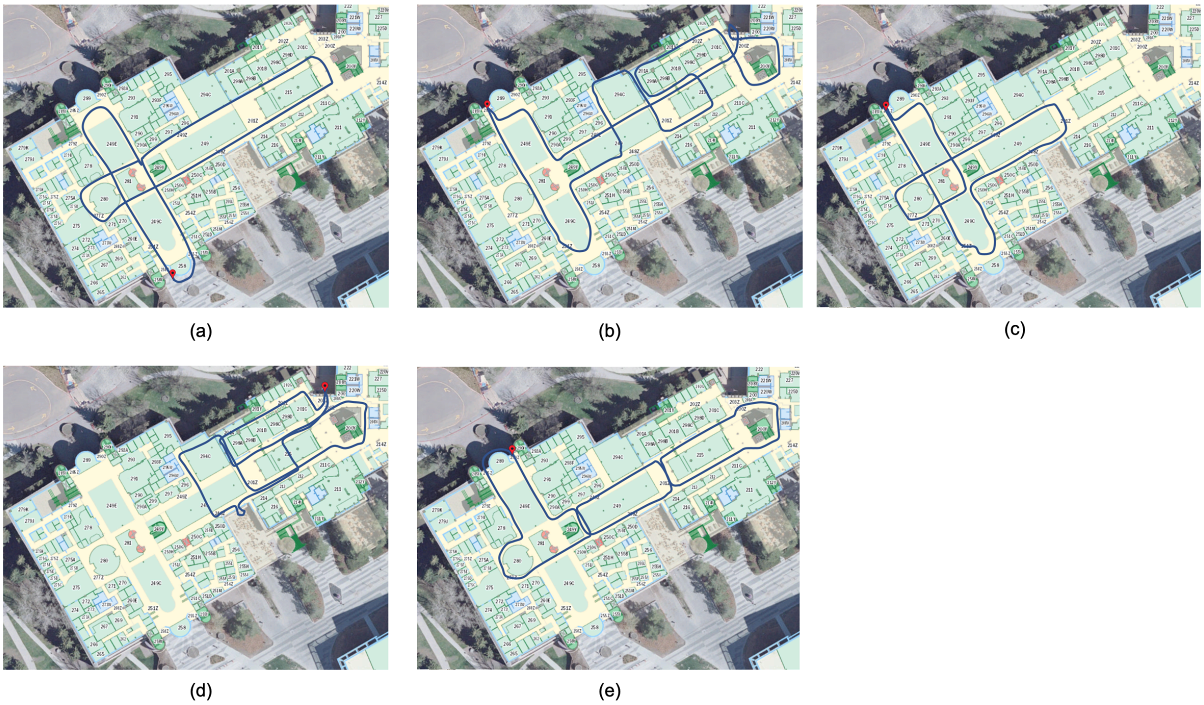

Figure 4. Training trajectories.

During both the training and testing periods, a Lenovo Phab 2 Pro smartphone was carried by each tester to collect red-green-blue-depth (RGB-D) images and compute SLAM solutions, so as to generate the reference trajectories. Refer to [60] for details about RGB-D-based SLAM. For all the SLAM-based reference solutions, a loop-closure had been detected between the start and end points. The SLAM-based reference location solutions were outputted with a data rate of 1 Hz. When a WiFi data epoch was obtained, the SLAM location solution that had the closest timestamp was used as the reference location.

Figure 6 shows the spatial distributions of RSS from the 10 APs used for localization. The AP-localization method in [61] can be used to determine the AP locations. The RSS heatmaps were generated by combining the reference locations from SLAM and the corresponding RSS values that had a similar timestamp. The colors from blue and red represent weak and strong RSS values, respectively. The average coverage area was over 1000 m^2 per AP, which was larger than the majority of the existing indoor localization works (e.g., approximately 60 m^2 per AP in [35] and 40 m^2 per AP in [38]. For practical engineering practice, an AP density of 100 to 400 m^2 per AP is commonly used for meter-level WiFi or BLE based localization).

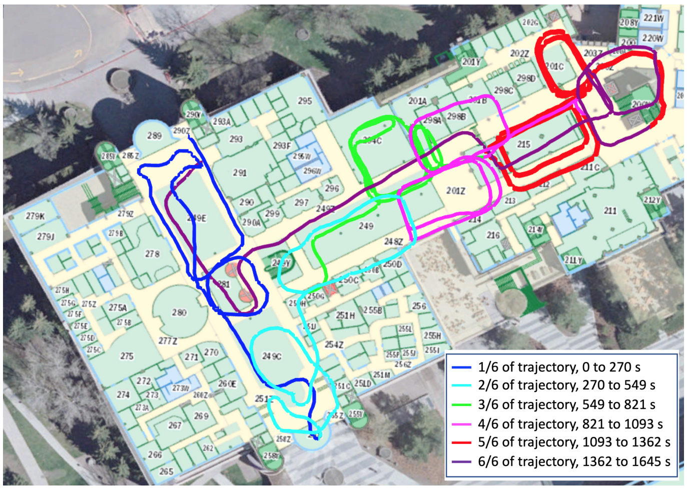
1/6 of trajectory, 0 to 270 s
2/6 of trajectory, 270 to 549 s
3/6 of trajectory, 549 to 821 s
4/6 of trajectory, 821 to 1093 s
5/6 of trajectory, 1093 to 1362 s
6/6 of trajectory, 1362 to 1645 s

Figure 5. One test trajectory.

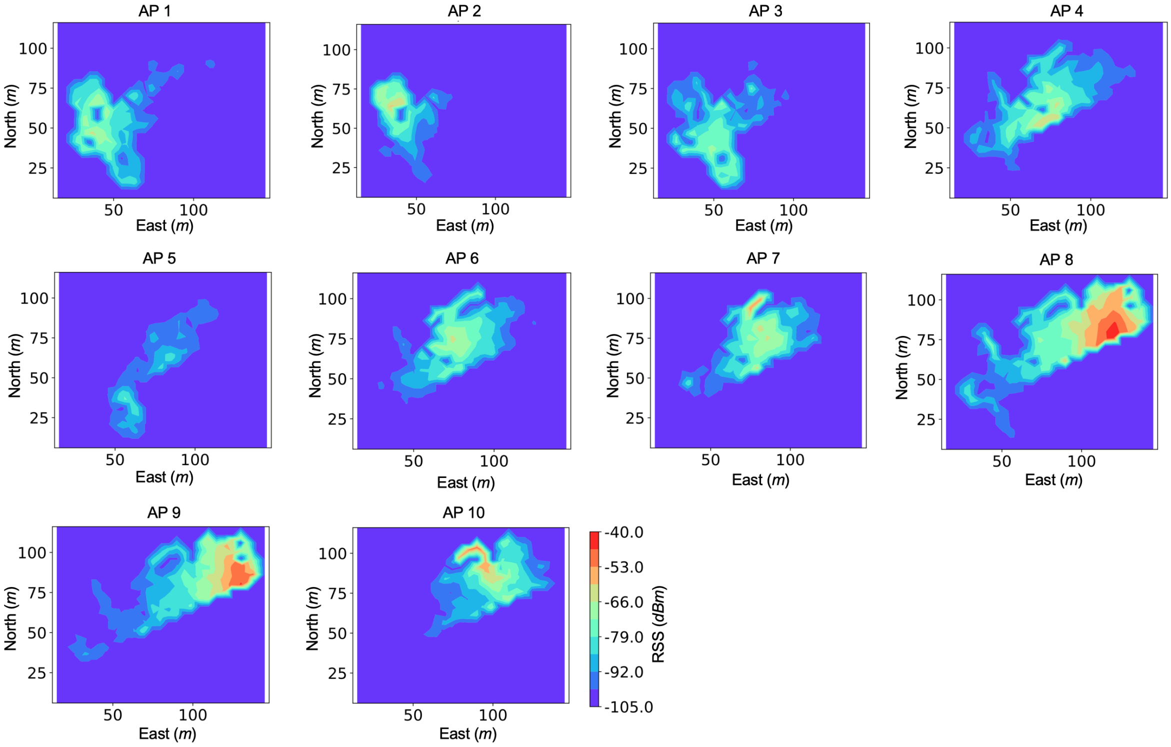

Figure 6. WiFi RSS distributions within test area.

Figure 7a shows the changes of RSS when the tester walked along the path in Figure 3, while Figure 7b illustrates the number of APs that had a RSS stronger than the threshold −95 dBm. The RSS values changed significantly when the user moved in the test area. During the time period 1200 to 1500 s (i.e., when the tester moved to the north-east corner), RSS from only two or three AP were received; however, the RSS values were strong (e.g., larger than −65 dBm) during this period.

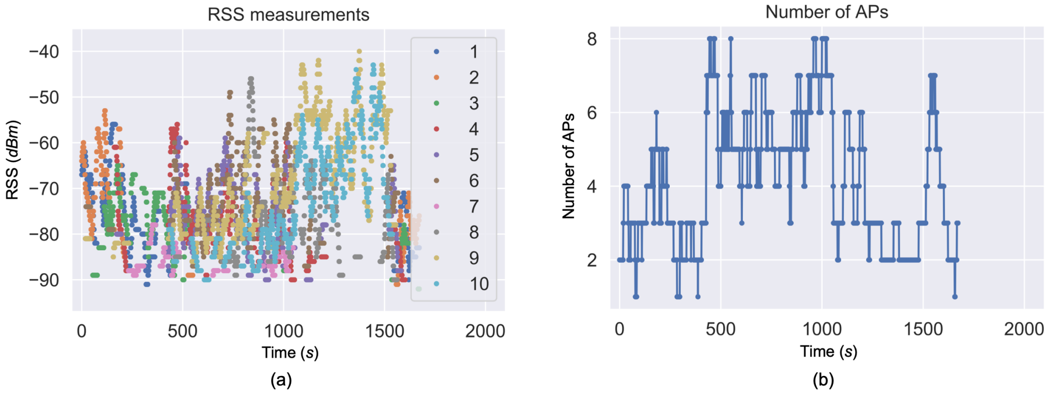

Figure 7. RSS time series (**a**) and number of available APs (**b**).

3.2. DR and WiFi Fingerprinting Solutions

Figure 8a,b illustrates one set of location solutions from DR and WiFi fingerprinting, respectively. The DR solution reflected the moving trajectory but drifted over time, while the WiFi fingerprinting solution fitted with the reference trajectory in the long term but suffered from large position errors (e.g., errors larger than 10 m) at several occasions. Thus, it is worthwhile to integrate DR and WiFi fingerprinting data to obtain a long-term accurate solution.

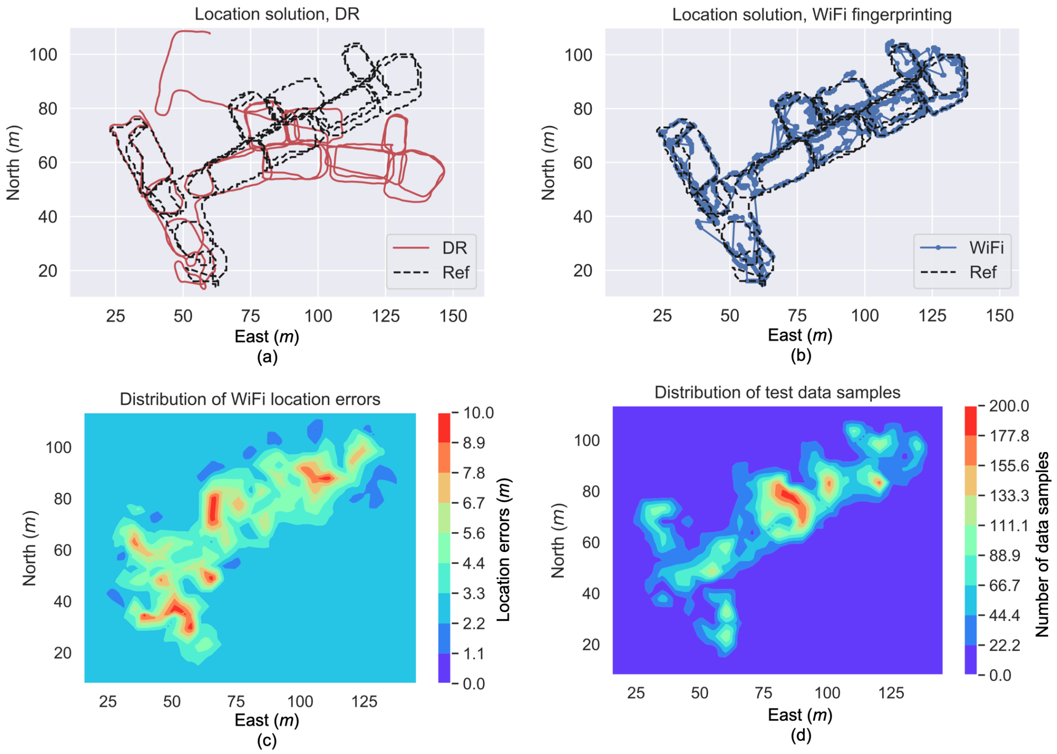

Figure 8. Location solutions from DR (**a**) and WiFi fingerprinting (**b**), heatmap of WiFi fingerprinting location errors (**c**), and heatmap of data samples used for computing the location errors (**d**).

Figure 8c visualizes the spatial distribution of location errors from WiFi fingerprinting, while Figure 8d illustrates the distribution of data samples used for computing the location errors in Figure 8c. It can be seen that the location uncertainty varied from within 2 m to over 17 m in the test area. This phenomenon indicates the necessity for adaptively setting of the wireless localization uncertainties in multi-sensor integration. The algorithm should be intelligent to reduce the weight of unreliable wireless position updates.

3.3. ANN Training and Prediction

As illustrated in Section 2.2, the RSS and wireless localization uncertainty data were combined to train the MLP. The RSS were used as the inputs, while the location uncertainties were used as the outputs. To determine the numbers of layers and neurons in each layer, Figure 9 shows the relationships between node number and training/testing computational time costs and prediction errors by using the experimental data. The computational time was computed in Python 3.6 by using a Macbook that had a 2.5 GHz Intel Core i7 processor. Figure 9 indicates that the MLP structures with two hidden layers generally provided more accurate prediction solutions than those with one hidden layer. Meanwhile, the MLP structures with around 30 neurons in each layer were capable to provide a similar prediction accuracy to those with more neurons. Thus, the MLP structure with two hidden layers, each with 30 neurons, was adopted to balance the computational cost and accuracy.

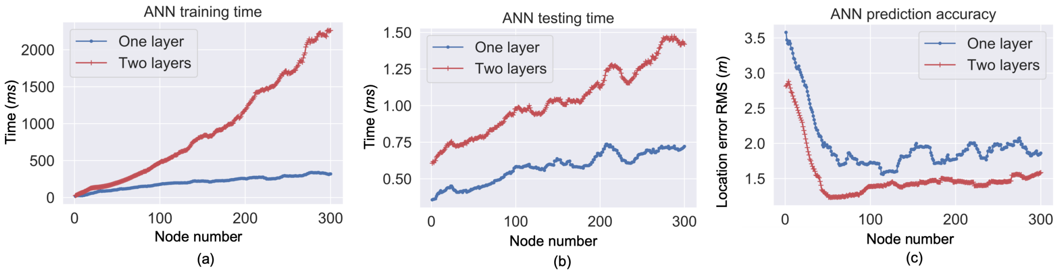

Figure 9. Processing time for ANN training (**a**) and testing (**b**), as well as RMS of differences between ANN-predicted location errors and actual ones (**c**).

To train the MLP, the limited-memory Broyden–Fletcher–Goldfarb–Shanno (L-BFGS) algorithm is applied. Refer to [62] for more details of the L-BFGS algorithm. Once the ANN was trained, it was used to predict the wireless fingerprinting uncertainty by given the RSS measurements. The GD method in Section 2.1 was also applied to predict the location uncertainty.

Figure 10a,b illustrate the time series of the actual fingerprinting-based location errors as well as the location uncertainty that was predicted by the GD and ANN methods, respectively. The prediction solutions from ANN had a lower frequency in variations compared to the GD solutions. This phenomenon is partly because the GD-based method is used on each RP; thus, it has a higher sensitivity than ANN, which is based on parametric models. Generally, the location uncertainty prediction solutions from both approaches generally reflected the varieties of actually location errors. This outcome indicates the feasibility of predicting fingerprinting location uncertainty by using RSS measurements.

Figure 11 illustrates the distributions of the location uncertainties that were predicted by the ANN and GD-based methods, respectively. Compared to Figure 8c, both approaches had reflected the high location errors in regions such as those around the points [80, 110] m (in [east, north]) and [40, 50] m, and the low location errors in regions such as those at around the point [100, 110] m.

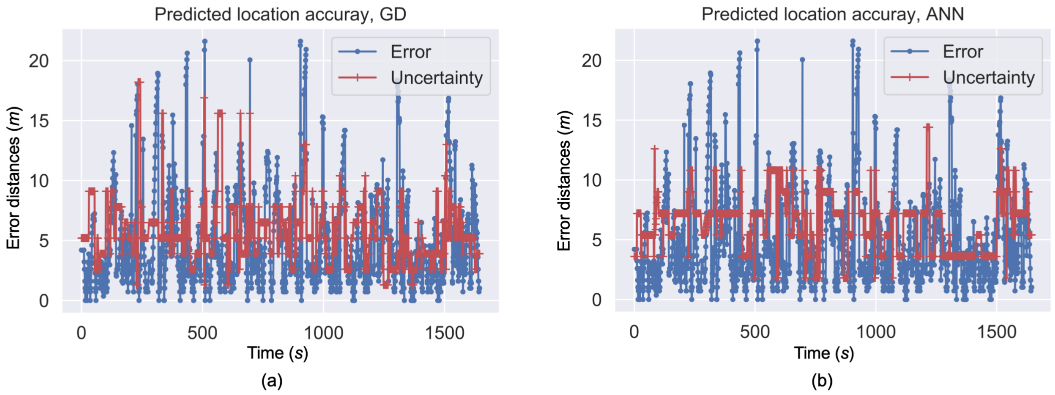

Figure 10. Predicted location uncertainty by GD (**a**) and ANN (**b**).

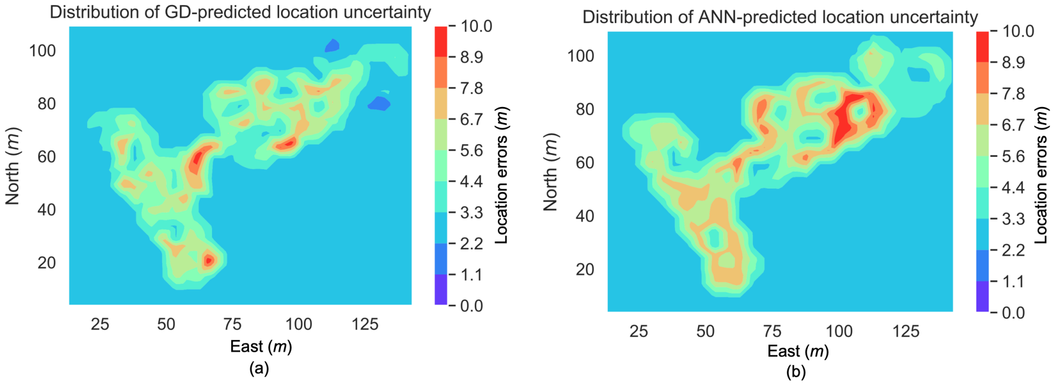

Figure 11. Distribution of location uncertainties predicted by GD (**a**) and ANN (**b**).

3.4. Localization with ML-Predicted Location Uncertainty

The predicted location uncertainties were further used to set the wireless location measurement noises in the EKF. Figure 12a demonstrates the location solutions from DR/WiFi integration that uses three strategies for setting measurement noises. These strategies include DR/WiFi-CN (i.e., setting the measurement noises at a constant value, which was 5 m according to preliminary results), DR/WiFi-GD (i.e., using location uncertainty values predicted by the GD method to set the measurement noises), and DR/WiFi-ANN (i.e., using location uncertainty values predicted by the ANN method to set the measurement noises). Figure 12b is the zoomed-in figures of Figure 12a. It can be seen that the DR/WiFi-CN solution suffered from large position errors at several occasions. A possible reason for such large errors is that the DR/WiFi-CN algorithm was not intelligent enough to increase the measurement noise values when there were large fingerprinting-based location errors. Through the use of the ML-predicted location uncertainties, either the DR/WiFi-GD or DR/WiFi-ANN strategy had reduced these large position errors.

Figure 12c demonstrates the time series of location errors in the test, and Figure 12d illustrates the cumulative distribution function (CDF) of location errors. Table 2 illustrates the corresponding location error statistics, including the standard deviation (STD), mean, root mean squares (RMS) errors, as well as the error within which the probability is 80% (i.e., the 80% error), the error within which the probability is 95% (i.e., the 95% error), and the maximum error. The DR/WiFi-GD and DR/WiFi-ANN strategies provided similar localization performance in general. The DR/WiFi-GD solution had a smaller error RMS, while the DR/WiFi-ANN solution had a smaller maximum error.

Compared to the DR/WiFi-CN strategy, the DR/WiFi-GD and DR/WiFi-ANN strategies had reduced the error RMS values by 27.9% and 23.3%, respectively, and had reduced the 95% errors by 32.3% and 28.7%. Thus, both the DR/WiFi-GD and DR/WiFi-ANN strategies are effective in reducing the location errors for DR/WiFi integration. This phenomenon indicates the effectiveness of adaptively setting location measurement noises in the EKF.

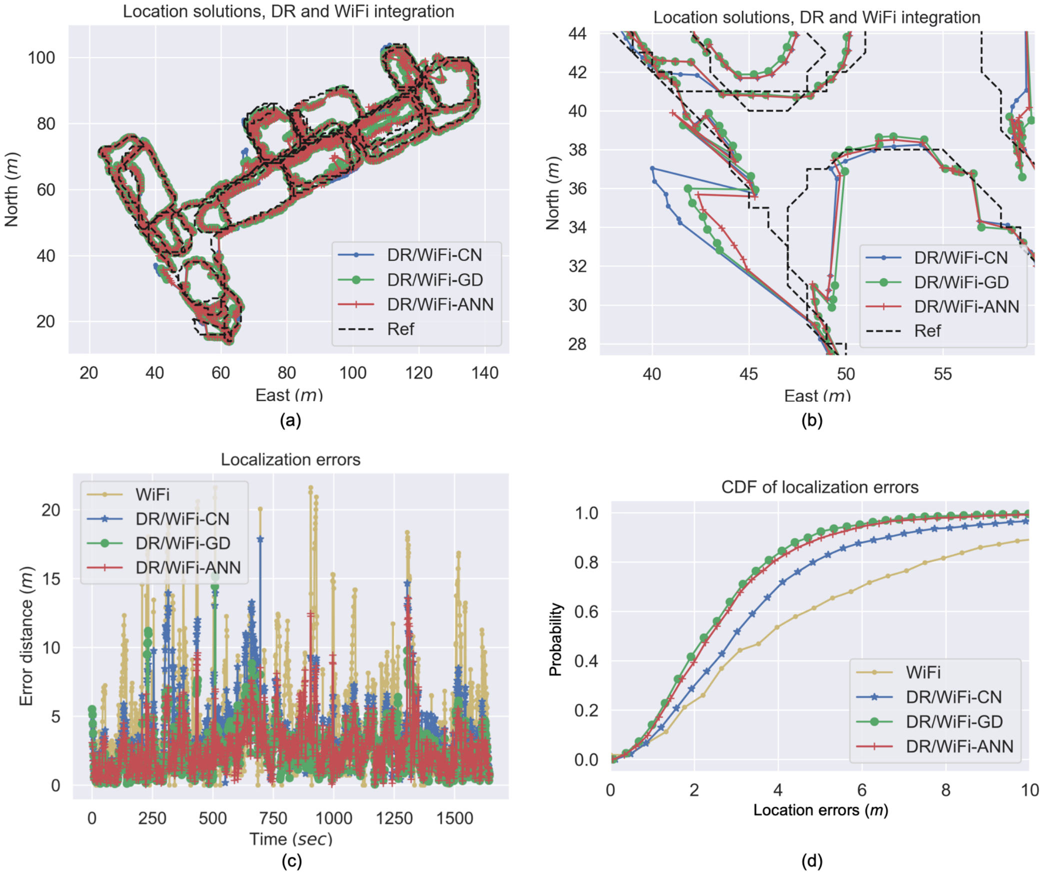

Figure 12. DR/WiFi integrated solutions (**a**), zoomed-in plots (**b**), location errors (**c**) and their CDF (**d**).

Table 2. Statistics of location errors (unit: m).

Strategy	**STD**	**Mean**	**RMS**	**80%**	**95%**	**Max**
WiFi	3.4	4.9	6.4	7.5	13.7	21.6
DR/WiFi-CN	2.5	3.5	4.3	4.8	8.7	17.9
DR/WiFi-GD	1.7	2.6	3.1	3.7	5.9	15.2
DR/WiFi-ANN	1.9	2.7	3.3	3.9	6.2	13.6

Finally, to visualize the improvement by using DR/WiFi fingerprinting integration and using the ML-predicted location uncertainty in the EKF, Figure 13 shows the distributions of location errors by using WiFi fingerprinting, DR/WiFi-CN, DR/WiFi-GD, and DR/WiFi-ANN. There were several regions that suffered from location errors that were larger than 8 m. When integrated with DR, the majority of these large errors had been mitigated. Furthermore, by introducing the ML-predicted location uncertainty into the EKF, more of such large errors had been mitigated and the majority of location errors were within 4 m.

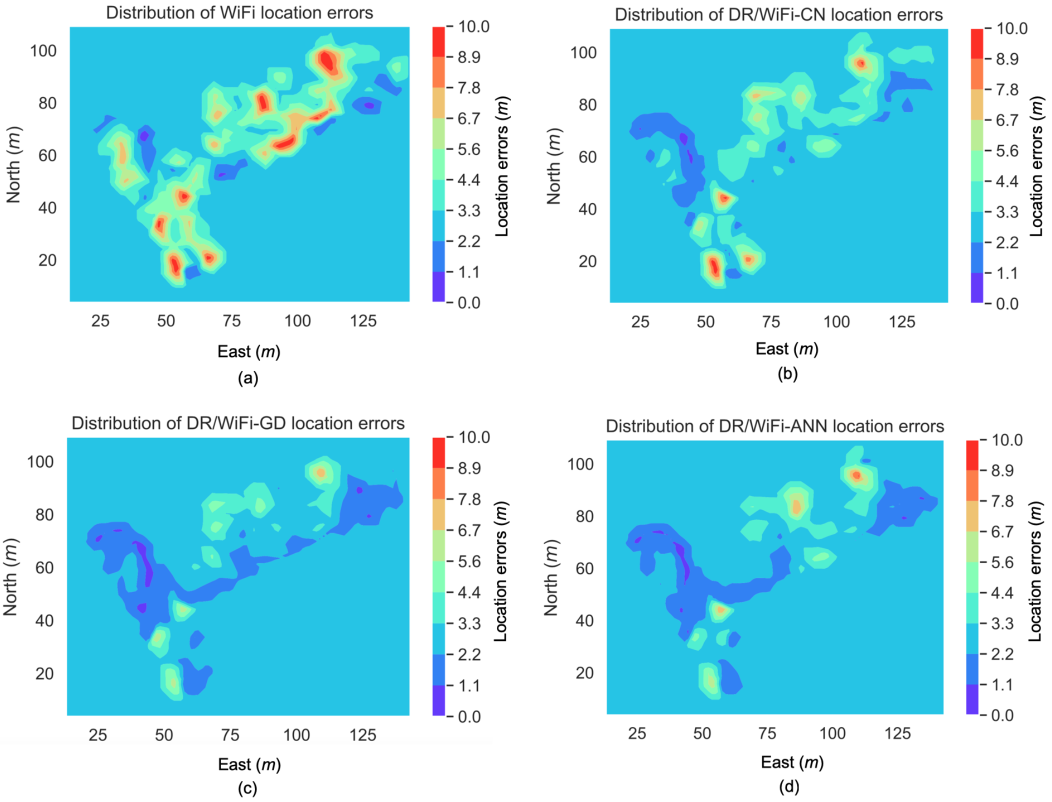

Figure 13. Distribution of location errors from WiFi (**a**), DR/WiFi-CN (**b**), DR/WiFi-GD (**c**), and DR/WiFi-ANN (**d**).

4. Conclusions

This paper has verified the possibility for predicting wireless fingerprinting uncertainty through machine learning. Compared to the traditional DR/WiFi fingerprinting integrated method that uses a constant measurement noise setting for the wireless fingerprinting-based location update, the proposed method, which sets the measurement noise adaptively by using machine learning-based approaches, had reduced the indoor localization errors by 23.3 % to 32.3 %. Therefore, it is suggested to investigate not only the location solution, but also the corresponding location uncertainty from fingerprinting, so as to achieve more reliable indoor localization solutions.

Author Contributions: Conceptualization, Y.L. and Z.G.; methodology, Y.L. and Z.G.; software, Y.L. and A.R.; validation, Y.Z.; formal analysis, Z.H.; investigation, Z.G.; data curation, A.R. and Y.Z.; writing—original draft preparation, Y.L.; writing—review and editing, Z.H. and Y.Z; visualization, A.R.; supervision, R.C. and N.E.; project administration, R.C. and N.E.; funding acquisition, N.E. and Z.G.

Funding: This paper is partly supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) CREATE grants, NSERC Discovery grants, the Alberta Innovates Technology Future (AITF) grants, and the National Natural Science Foundation of China (NSFC) for Young Scientists (Grant No. 41804027, 61771135).

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

Abbreviations

The following abbreviations are used in this manuscript:

ANFIS	adaptive neural fuzzy inference system
ANN	artificial neural network
AoA	angle-of-arrival
AP	access point
BDS	BeiDou navigation satellite system
BLE	bluetooth low energy

BP	back-propagation
CNN	convolution neural network
CPN	counter propagation neural network
DOP	dilution of precision
DR	dead-reckoning
EKF	extended Kalman filter
FF	feed-forward neural network
GAN	generative adversarial neural network
GCC	generalized cross correlation
GD	gaussian distribution
GNSS	global navigation satellite systems
GPS	global positioning services
INS	inertial navigation system
L-BFGS	limited-memory Broyden–Fletcher–Goldfarb–Shanno
LPWAN	low-power wide-area network
ML	machine learning
MLP	multi-layer perceptron
N/A	not provided
NLoS	non-line-of-sight
RBF	radial basis function neural network
RBP	resilient back propagation
RFID	radio frequency identification
RGB-D	red-green-blue-depth
RMS	root mean squares
RNN	recurrent neural network
RP	reference point
RSS	received signal strength
SCG	scaled conjugate gradient
SLAM	simultaneous localization and mapping
SoO	signal of opportunity
STD	standard deviation
TDNN	time delay neural network
ToA	time-of-arrival
UTC	coordinated universal time
UWB	ultra-wide-band
WiFi	wireless local area network

References

1. Ferreira, A.G.; Fernandes, D.; Catarino, A.P.; Monteiro, J.L. Performance Analysis of ToA-Based Positioning Algorithms for Static and Dynamic Targets with Low Ranging Measurements. *Sensors* **2017**, *17*, 1915. [CrossRef]
2. Tomic, S.; Beko, M.; Dinis, R.; Bernardo, L. On Target Localization Using Combined RSS and AoA Measurements. *Sensors* **2018**, *18*, 1266. [CrossRef] [PubMed]
3. Kanaris, L.; Kokkinis, A.; Liotta, A.; Stavrou, S. Fusing Bluetooth Beacon Data with Wi-Fi Radiomaps for Improved Indoor Localization. *Sensors* **2017**, *17*, 812. [CrossRef] [PubMed]
4. Lashkari, B.; Rezazadeh, J.; Farahbakhsh, R.; Sandrasegaran, K. Crowdsourcing and Sensing for Indoor Localization in IoT—A Review. *IEEE Sens. J.* **2018**. [CrossRef]
5. Gao, C.; Harle, R. Semi-Automated Signal Surveying Using Smartphones and Floorplans. *IEEE Trans. Mob. Comput.* **2018**, *17*, 1952–1965. [CrossRef]
6. Pei, L.; Liu, D.; Zou, D.; Choy, R.L.F.; Chen, Y.; He, Z. Optimal Heading Estimation Based Multidimensional Particle Filter for Pedestrian Indoor Positioning. *IEEE Access* **2018**, *6*, 49705–49720. [CrossRef]
7. Li, Y.; Gao, Z.; He, Z.; Zhang, P.; Chen, R.; El-Sheimy, N. Multi-Sensor Multi-Floor 3D Localization With Robust Floor Detection. *IEEE Access* **2018**, *6*, 76689–76699. [CrossRef]

8. Yu, M. INS/GPS Integration System using Adaptive Filter for Estimating Measurement Noise Variance. *IEEE Trans. Aerosp. Electron. Syst.* **2012**, *48*, 1786–1792. [CrossRef]
9. Aghili, F.; Su, C. Robust Relative Navigation by Integration of ICP and Adaptive Kalman Filter Using Laser Scanner and IMU. *IEEE/ASME Trans. Mechatron.* **2016**, *21*, 2015–2026. [CrossRef]
10. Li, Y.; He, Z.; Gao, Z.; Zhuang, Y.; Shi, C.; El-Sheimy, N. Towards Robust Crowdsourcing-Based Localization: A Fingerprinting Accuracy Indicator Enhanced Wireless/Magnetic/Inertial Integration Approach. *IEEE Internet Things J.* **2018**. [CrossRef]
11. Chiang, K.W.; Noureldin, A.; El-Sheimy, N. A new weight updating method for INS/GPS integration architectures based on neural networks. *Meas.Sci. Technol.* **2004**, *15*, 2053–2061. [CrossRef]
12. Patel, M.; Emery, B.; Chen, Y. ContextualNet: Exploiting Contextual Information Using LSTMs to Improve Image-Based Localization. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, QLD, Australia, 21–26 May 2018; pp. 1–7.
13. Guan, X.; Cai, C. A new integrated navigation system for the indoor unmanned aerial vehicles (UAVs) based on the neural network predictive compensation. In Proceedings of the 2018 33rd Youth Academic Annual Conference of Chinese Association of Automation (YAC), Nanjing, China, 18–20 May 2018; pp. 575–580.
14. Gu, F.; Khoshelham, K.; Yu, C.; Shang, J. Accurate Step Length Estimation for Pedestrian Dead Reckoning Localization Using Stacked Autoencoders. *IEEE Trans. Instrum. Meas.* **2018**. [CrossRef]
15. Zhang, X.; Sun, H.; Wang, S.; Xu, J. A New Regional Localization Method for Indoor Sound Source Based on Convolutional Neural Networks. *IEEE Access* **2018**, *6*, 72073–72082. [CrossRef]
16. Akram, B.A.; Akbar, A.H.; Shafiq, O. HybLoc: Hybrid Indoor Wi-Fi Localization Using Soft Clustering-Based Random Decision Forest Ensembles. *IEEE Access* **2018**, *6*, 38251–38272. [CrossRef]
17. Li, J.; Wei, Y.; Wang, M.; Luo, J.; Hu, Y. Two indoor location algorithms based on sparse fingerprint library. In Proceedings of the 2018 Chinese Control And Decision Conference (CCDC), Shenyang, China, 9–11 June 2018; pp. 6753–6758.
18. Gharghan, S.K.; Nordin, R.; Jawad, A.M.; Jawad, H.M.; Ismail, M. Adaptive Neural Fuzzy Inference System for Accurate Localization of Wireless Sensor Network in Outdoor and Indoor Cycling Applications. *IEEE Access* **2018**, *6*, 38475–38489. [CrossRef]
19. Berz, E.L.; Tesch, D.A.; Hessel, F.P. Machine-learning-based system for multi-sensor 3D localisation of stationary objects. *IET Cyber-Phys. Syst. Theory Appl.* **2018**, *3*, 81–88. [CrossRef]
20. Abdallah, A.A.; Saab, S.S.; Kassas, Z.M. A machine learning approach for localization in cellular environments. In Proceedings of the 2018 IEEE/ION Position, Location and Navigation Symposium (PLANS), Monterey, CA, USA, 23–26 April 2018; pp. 1223–1227.
21. Konings, D.; Parr, B.; Alam, F.; Lai, E.M. Falcon: Fused Application of Light Based Positioning Coupled With Onboard Network Localization. *IEEE Access* **2018**, *6*, 36155–36167. [CrossRef]
22. Jang, J.; Hong, S. Indoor Localization with WiFi Fingerprinting Using Convolutional Neural Network. In Proceedings of the 2018 Tenth International Conference on Ubiquitous and Future Networks (ICUFN), Prague, Czech Republic, 3–6 July 2018; pp. 53–758.
23. Li, Z.; He, G.; Li, M.; Ma, L.; Chen, Q.; Huang, J.; Cao, J.; Feng, S.; Gao, H.; Wang, S. RBF neural network based RFID indoor localization method using artificial immune system. In Proceedings of the 2018 Chinese Control And Decision Conference (CCDC), Shenyang, China, 9–11 June 2018; pp. 2837–2842.
24. Elbakly, R.; Aly, H.; Youssef, M. TrueStory: Accurate and Robust RF-Based Floor Estimation for Challenging Indoor Environments. *IEEE Sens. J.* **2018**, *18*, 10115–10124. [CrossRef]
25. Choi, J.; Lee, W.; Lee, J.; Lee, J.; Kim, S. Deep Learning Based NLOS Identification With Commodity WLAN Devices. *IEEE Trans. Veh. Technol.* **2018**, *67*, 3295–3303. [CrossRef]
26. Bregar, K.; Mohorcic, M. Improving Indoor Localization Using Convolutional Neural Networks on Computationally Restricted Devices. *IEEE Access* **2018**, *6*, 17429–17441. [CrossRef]
27. Wang, X.; Wang, X.; Mao, S. Deep Convolutional Neural Networks for Indoor Localization with CSI Images. *IEEE Trans. Netw. Sci. Eng.* **2018**. [CrossRef]
28. Valada, A.; Radwan, N.; Burgard, W. Deep Auxiliary Learning for Visual Localization and Odometry. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, QLD, Australia, 21–25 May 2018; pp. 6939–6946.
29. Sun, Y.; Chen, J.; Yuen, C.; Rahardja, S. Indoor Sound Source Localization With Probabilistic Neural Network. *IEEE Trans. Ind. Electron.* **2018**, *65*, 6403–6413. [CrossRef]

30. Wagstaff, B.; Kelly, J. LSTM-Based Zero-Velocity Detection for Robust Inertial Navigation. In Proceedings of the 2018 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Nantes, France, 24–27 September 2018; pp. 1–8.
31. Dong, B.; Burgess, T.; Neuner, H.; Fercher, S. Neural Network Based Radio Fingerprint Similarity Measure. In Proceedings of the 2018 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Nantes, France, 24–27 September 2018; pp. 1–8.
32. Luo, R.C.; Shih, W. Autonomous Mobile Robot Intrinsic Navigation Based on Visual Topological Map. In Proceedings of the 2018 IEEE 27th International Symposium on Industrial Electronics (ISIE), Cairns, QLD, Australia, 13–15 June 2018; pp. 541–546.
33. Akail, N.; Moralesl, L.Y.; Murase, H. Reliability Estimation of Vehicle Localization Result. In Proceedings of the 2018 IEEE Intelligent Vehicles Symposium (IV), Changshu, China, 26–30 June 2018; pp. 740–747.
34. Park, C.U.; Shin, H.; Choi, Y. A Parallel Artificial Neural Network Learning Scheme Based on Radio Wave Fingerprint for Indoor Localization. In Proceedings of the 2018 Tenth International Conference on Ubiquitous and Future Networks (ICUFN), Prague, Czech Republic, 3-6 July 2018; pp. 794–797.
35. Adege, A.B.; Yen, L.; Lin, H.P.; Yayeh, Y.; Li, Y.R.; Jeng, S.S.; Berie, G. Applying Deep Neural Network (DNN) for large-scale indoor localization using feed-forward neural network (FFNN) algorithm. In Proceedings of the 2018 IEEE International Conference on Applied System Invention (ICASI), Chiba, Japan, 13–17 April 2018; pp. 814–817.
36. Zhu, C.; Xu, L.; Liu, X.; Qian, F. Tensor-Generative Adversarial Network with Two-Dimensional Sparse Coding: Application to Real-Time Indoor Localization. In Proceedings of the 2018 IEEE International Conference on Communications (ICC), Kansas City, MO, USA, 20–24 May 2018; pp. 1–6.
37. Sabanci, K.; Yigit, E.; Ustun, D.; Toktas, A.; Aslan, M.F. WiFi Based Indoor Localization: Application and Comparison of Machine Learning Algorithms. In Proceedings of the 2018 XXIIIrd International Seminar/Workshop on Direct and Inverse Problems of Electromagnetic and Acoustic Wave Theory (DIPED), Tbilisi, Georgia, 24–27 September 2018; pp. 246–251.
38. Aikawa, S.; Yamamoto, S.; Morimoto, M. WLAN Finger Print Localization using Deep Learning. In Proceedings of the 2018 IEEE Asia-Pacific Conference on Antennas and Propagation (APCAP), Auckland, New Zealand, 5–8 August 2018; pp. 541–542.
39. Anzum, N.; Afroze, S.F.; Rahman, A. Zone-Based Indoor Localization Using Neural Networks: A View from a Real Testbed. In Proceedings of the 2018 IEEE International Conference on Communications (ICC), Kansas City, MO, USA, 20-24 May 2018; pp. 1–7.
40. Jeevarathnam, N.G.; Uysal, I. Grid-Based RFID Localization Using Tag Read Count And Received Signal Strength. In Proceedings of the 2018 International Joint Conference on Neural Networks (IJCNN), Rio de Janeiro, Brazil, 8–13 July 2018; pp. 1–8.
41. Ibrahim, M.; Torki, M.; ElNainay, M. CNN based Indoor Localization using RSS Time-Series. In Proceedings of the 2018 IEEE Symposium on Computers and Communications (ISCC), Natal, Brazil, 25–28 June 2018; pp. 01044–01049.
42. Wu, G.; Tseng, P. A Deep Neural Network-Based Indoor Positioning Method using Channel State Information. In Proceedings of the 2018 International Conference on Computing, Networking and Communications (ICNC), Maui, HI, USA, 5–8 March 2018; pp. 290–294.
43. Niitsoo, A.; Edelhauβer, T.; Mutschler, C. Convolutional Neural Networks for Position Estimation in TDoA-Based Locating Systems. In Proceedings of the 2018 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Nantes, France, 24–27 September 2018; pp. 1–8.
44. Huang, Z.; Xu, J.; Pan, J. A Regression Approach to Speech Source Localization Exploiting Deep Neural Network. In Proceedings of the 2018 IEEE Fourth International Conference on Multimedia Big Data (BigMM), Xi'an, China, 13–16 September 2018; pp. 1–6.
45. Sinha, H.; Patrikar, J.; Dhekane, E.G.; Pandey, G.; Kothari, M. Convolutional Neural Network Based Sensors for Mobile Robot Relocalization. In Proceedings of the 2018 23rd International Conference on Methods & Models in Automation & Robotics (MMAR), Miedzyzdroje, Poland, 27–30 August 2018; pp. 774–779.
46. Fu, Y.; Chen, P.; Yang, S.; Tang, J. An Indoor Localization Algorithm Based on Continuous Feature Scaling and Outlier Deleting. *IEEE Internet Things J.* **2018**, *5*, 1108–1115. [CrossRef]

47. Haeberlen, A.; Flannery, E.; Ladd, A.M.; Rudys, A.; Wallach, D.S.; Kavraki, L.E. Practical robust localization over large-scale 802.11 wireless networks. In Proceedings of the MobiCom'04, Philadelphia, PA, USA, 26 September–1 October 2004.
48. Yoo, J.; Johansson, K.H.; Kim, H.J. Indoor Localization Without a Prior Map by Trajectory Learning From Crowdsourced Measurements. *IEEE Trans. Instrum. Meas.* **2017**, *66*, 2825–2835. [CrossRef]
49. Guo, X.; Ansari, N.; Li, L.; Li, H. Indoor Localization by Fusing a Group of Fingerprints Based on Random Forests. *IEEE Internet Things J.* **2018**. [CrossRef]
50. Leonardo, R.; Barandas, M.; Gamboa, H. A Framework for Infrastructure-Free Indoor Localization Based on Pervasive Sound Analysis. *IEEE Sens. J.* **2018**, *18*, 4136–4144. [CrossRef]
51. Roos, T.; Myllymaki, P.; Tirri, H.; Misikangas, P.; Sievanen, J. A Probabilistic Approach to WLAN User Location Estimation. *Int. J. Wirel. Inf. Netw.* **2002**, *9*, 155–164. [CrossRef]
52. Kaemarungsi, K. Distribution of WLAN received signal strength indication for indoor location determination. In Proceedings of the 2006 1st International Symposium on Wireless Pervasive Computing, Phuket, Thailand, 16-18 January 2006; pp. 1–6.
53. Liao, J.K.; Tsai, G.J. The Development of an Artificial Neural Networks Aided Image Localization Scheme for Indoor Navigation Applications with Floor Plans Built by Multi-platform Mobile Mapping Systems. In Proceedings of the International Technical Meeting of the Satellite Division of the Institute of Navigation, Portland, Oregon, 16–19 September 2017; pp. 3010–3027.
54. Hagan, M.T.; Menhaj, M.B. Training feedforward networks with the Marquardt algorithm. *IEEE Trans. Neural Netw.* **1994**, *5*, 989–993. [CrossRef]
55. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016.
56. Shin, E. Estimation Techniques for Low-Cost Inertial Navigation. Ph.D. Thesis, Department of Geomatics Engineering, University of Calgary, Calgary, AL, Canada, 2005.
57. Syed, Z.; Aggarwal, P.; Yang, Y.; El-Sheimy, N. Improved Vehicle Navigation Using Aiding with Tightly Coupled Integration. In Proceedings of the VTC Spring 2008—IEEE Vehicular Technology Conference, Singapore, 11–14 May 2008; pp. 3077–3081.
58. Zhang, H.; Yuan, W.; Shen, Q.; Li, T.; Chang, H. A Handheld Inertial Pedestrian Navigation System With Accurate Step Modes and Device Poses Recognition. *IEEE Sens. J.* **2015**, *15*, 1421–1429. [CrossRef]
59. Wang, L.; Sun, Y.; Li, Q.; Liu, T. Estimation of Step Length and Gait Asymmetry Using Wearable Inertial Sensors. *IEEE Sens. J.* **2018**, *18*, 3844–3851. [CrossRef]
60. Labbe, M.; Michaud, F. Appearance-Based Loop Closure Detection for Online Large-Scale and Long-Term Operation. *IEEE Trans. Robot.* **2013**, *29*, 734–745. [CrossRef]
61. Zhuang, Y.; Li, Y.; Lan, H.; Syed, Z.; El-Sheimy, N. Smartphone-based WiFi access point localisation and propagation parameter estimation using crowdsourcing. *Electron. Lett.* **2015**, *51*, 1380–1382. [CrossRef]
62. Morales, J.; Nocedal, J. Remark on "algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound constrained optimization". *J. ACM Trans. Math. Softw.* **2011**, *38*, 1–4. [CrossRef]