

Article

RadarPDR: Radar-Assisted Indoor Pedestrian Dead Reckoning

Jianbiao He ¹, Wei Xiang ², Qing Zhang ³ and Bang Wang ^{2,*} ¹ Shenzhen Polytechnic (SZPT), Liuxian Avenue 7098, Shenzhen 518055, China² School of Electronic Information and Communications, Huazhong University of Science and Technology (HUST), Luoyu Road 1037, Wuhan 430074, China³ Hubei International Trade Supply Chain Management Co., Ltd., Wuhan 430000, China

* Correspondence: wangbang@hust.edu.cn

Abstract: Pedestrian dead reckoning (PDR) is the critical component in indoor pedestrian tracking and navigation services. While most of the recent PDR solutions exploit in-built inertial sensors in smartphones for next step estimation, due to measurement errors and sensing drift, the accuracy of walking direction, step detection, and step length estimation cannot be guaranteed, leading to large accumulative tracking errors. In this paper, we propose a radar-assisted PDR scheme, called RadarPDR, which integrates a frequency-modulation continuous-wave (FMCW) radar to assist the inertial sensors-based PDR. We first establish a segmented wall distance calibration model to deal with the radar ranging noise caused by irregular indoor building layouts and fuse wall distance estimation with acceleration and azimuth signals measured by the inertial sensors of a smartphone. We also propose a hierarchical *particle filter* (PF) together with an extended Kalman filter for position and trajectory adjustment. Experiments have been conducted in practical indoor scenarios. Results demonstrate that the proposed RadarPDR is efficient and stable and outperforms the widely used inertial sensors-based PDR scheme.

Keywords: trajectory correction; radar ranging and correction; pedestrian dead reckoning; indoor tracking



Citation: He, J.; Xiang, W.; Zhang, Q.; Wang, B. RadarPDR: Radar-Assisted Indoor Pedestrian Dead Reckoning. *Sensors* **2023**, *23*, 2782. <https://doi.org/10.3390/s23052782>

Academic Editor: Guenther Retscher

Received: 21 January 2023

Revised: 20 February 2023

Accepted: 26 February 2023

Published: 3 March 2023



Copyright: © 2023 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 (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Pedestrian dead reckoning (PDR) is a widely-used pedestrian tracking solution in satellite signal-masking environments where global navigation satellite systems such as the global positioning system (GPS) cannot work correctly [1–4]. The cumulative error problem is the main obstacle for achieving high tracking accuracy in both the Inertial Navigation Systems (INSs) and the Step-and-Heading Systems (SHSs) of PDR [5–7]. For example, SHSs adopt the step-and-heading strategy to calculate the step vectors (*step length, heading*) from the inertial sensor data and then evaluate the pedestrian trajectory according to the step vectors in a step-by-step way. However, such recursive estimation accumulates large errors originated from sensing drift with the increase of walking distance. Researchers have devoted a lot of effort towards solving the problem. Their approaches can be generally divided into two categories: (1) mining latent information to correct sensor errors; (2) introducing auxiliary information to adjust deviated trajectory.

Latent information assisted methods utilized prior knowledge or general inference as an extra supplement to fuse inertia step vector estimates for improving pedestrian trajectory accuracy. In foot-mounted INSs, the zero-velocity update (ZUPT) algorithm is a workable solution to limit long-term accumulation of errors [8–11]. The ZUPT algorithm is based on the fact that the speed of the sole should be zero during the stance phase period, when the sole is completely in contact with the ground. As such, the acceleration integral could be avoided [12]. The building layout is another source of latent information closely related to indoor positioning services, which can affect pedestrian motion characteristics. For instance, pedestrians tend to walk in straight lines in straight corridors [13], or in one of

several dominant directions of the buildings [14–16]. Pedestrian trajectory and heading direction can be inferred with the help of such layout characteristics and geometries in buildings. However, those inferences are not general enough in many indoor scenarios. For example, a pedestrian could have walked not according to layout dominant directions for a long time [17].

Auxiliary information assisted methods adopt location-specific information inherent in the environment or created by additional technologies to adjust the PDR positioning result via a fusion algorithm. The auxiliary techniques or information include Wi-Fi [18–23], Bluetooth [24,25], Ultra-Wideband (UWB) [26–28], visible light [17,29], and indoor map [30–35]. Due to the azimuth effect, the procedure of information fusion and trajectory calibration is a nonlinear model that can be solved by particle filtering [20,28,33,34] and extended Kalman filter (EKF) [36–38]. The auxiliary information-assisted methods are more applicable in practice, since they do not depend on a specific building layout. However, most of the auxiliaries require additional hardware infrastructure, which often has expensive deployment costs.

With miniaturization, lightweight, and low power consumption of sensors, more sensors can be integrated into mobile terminals and wearable devices. For such considerations, we have initialized experiments of using a lightweight and portable 24 GHz *frequency-modulation continuous-wave radar* (FMCW radar, simplified as radar hereafter) to measure the distance between a radar and a wall. Figure 1 presents our experiment device and experiment scenario, where a student used a stand to connect an external yet small radar with a smartphone. On the one hand, we recognized that such wall distance estimations can be exploited to improve the accuracy of inertial sensor-based PDR. On the other hand, although a radar can estimate the distance to a wall, the estimation could fluctuate severely during pedestrian walking due to the irregular indoor layout, such as doors, windows, corners, obstructions, etc. Therefore, we need to first calibrate wall distance estimations from a radar and next fuse such estimations to adjust step positions.

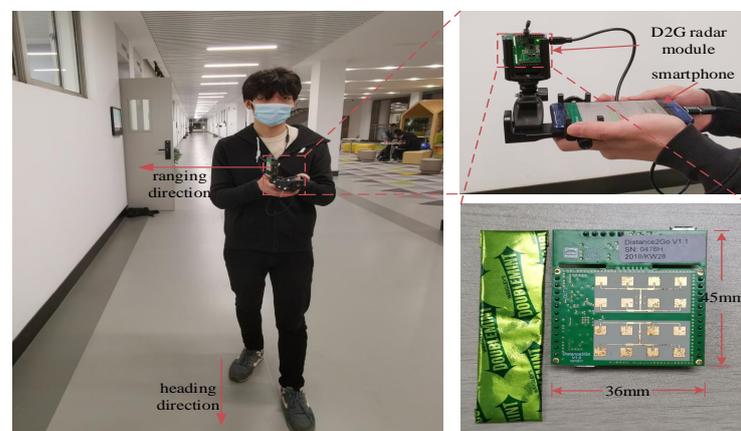


Figure 1. The FMCW radar and experiment scenario.

In this paper, we propose a *Radar-assisted Pedestrian Dead Reckoning* scheme, called RadarPDR, for indoor tracking, which employs an external radar for trajectory adjustment. To the best of our knowledge, we are the first to use and experiment with an FMCW radar for indoor pedestrian tracking. In RadarPDR, we first propose a segmented wall distance calibration model, which exploits the space-time correlation of neighbor points to deal with the radar ranging noise caused by irregular indoor building layouts. We next design a hierarchical particle filter together with an extended Kalman filter to fuse calibrated wall distances with acceleration, angular velocity, and azimuth information to adjust pedestrian step positions. We conduct field experiments in typical indoor scenarios. Results validate the effectiveness of our RadarPDR in terms of lower tracking errors compared with the

widely used inertial sensor-based PDR scheme. Our main contributions can be summarized as follows:

- Employ an FMCW ranging radar to assist inertial PDR for indoor pedestrian tracking;
- Correct distance measurement errors caused by concave/convex wall structure in actual indoor environments;
- Fuse corrected wall distance, pedestrian step length, and head direction for pedestrian trajectory adjustment.

2. Related Works

Radio-frequency (RF) is the most common auxiliary technology, such as Wi-Fi and Bluetooth, to assist PDR indoor tracking. Some research has proposed to integrate fingerprinting-based localization with PDR [18–21,39]. For example, Shi et al. [19] use a weighted K-Nearest Neighbor algorithm to obtain Wi-Fi RSS (received signal strength) fingerprinting locations, which are fused with PDR locations via a simple weighted average. Some researchers propose to fuse the lidar data and inertial sensor data to solve the accumulative error problem in localization [23,39–41]. Liu et al. [39] develop a radio receiver to estimate the Doppler speed and range information from digital terrestrial multimedia broadcasting (DTMB) signals and carry out an extended Kalman filter (EKF) algorithm to fuse the information for the PDR. Li et al. [20] propose to construct a radio map from pedestrian trajectories for RSS fingerprinting, while adjusting trajectory with the aid of Wi-Fi RSS fingerprinting results via selective particle filtering. These methods need to construct a radio map for fingerprinting either by site survey or crowdsourced trajectories.

Some research fuses the trilateral localization method with PDR [22,33,39,40,42–44]. Treating iBeacon as an auxiliary, Xia et al. [33] calculate distances between pedestrian and bluetooth anchor nodes based on an RSS path loss model to estimate pedestrian locations by the trilateration method. These locations are fused with PDR results via a particle filter. Tiwari and kumar Jain [22] propose a Heron-bilateration-based position determination (HBPD) technique to locate a mobile device with a Wi-Fi *access point* (AP). They also propose a hybrid indoor localization system (HILS) to integrate the HBPD technique with the PDR method. These methods require enough anchor points with known locations. Xu et al. [34] propose an enhanced particle filter that fuses smartphone built-in sensors' outputs and the distances in between smartphones. Tao et al. [42] continuously collect light intensity values from smartphones, which are used to detect light source location for PDR correction. De Cock et al. [43] present a smartphone-based multi-floor indoor localization system and Asraf et al. [45] propose a deep-learning PDR framework based on smartphones' inertial sensors. Tian et al. [28] propose a PF-based fusion algorithm to correct position error due to sensor drift in a PDR-only system. Compared to the path loss model, these methods have higher distance estimation accuracy. However, almost all RF-based auxiliary technologies require pre-deployed hardware infrastructures.

Indoor layout is another piece of frequently used auxiliary information, mainly working in two ways: deterministic trajectory matching [30–32] and probabilistic map model [33–35]. Lan and Shih [31] correct the drift error of the gyroscope by comparing the geometric similarity between a PDR trajectory and an indoor map. Nguyen-Huu et al. [32] detect a potential landmark based on turning behaviors in corners or door areas and then update the current position with the landmark position for meeting some physical constraints. Zhou et al. [40] use the conditional random field (CRF) model to estimate the optimal landmark sequence passed by pedestrians with the help of a pseudo indoor plan for PDR position correction. On the other hand, probabilistic map models usually occur in PF-based fusion schemes [33,34]. In these methods, an indoor environment is firstly divided into reachable and unreachable areas, which serve as particle weights in the particle filtering process. For example, Koroglu et al. [35] use a continuous probability distribution function to convert a floor plan into a likelihood heat map for weighting particles. However, these methods require a prior knowledge of indoor layouts.

For ranging-assisted indoor PDR, many research studies calculate distances by communicating with surrounding Wi-Fi access points and then integrate the distance measurements with inertial data to improve positioning accuracy [34,46–49]. For example, Sun et al. [46] perform ranging based on Wi-Fi fine timing measurement technology and further calculate position through their improved trilateral positioning algorithm. The ranging-based positions are eventually fused with inertial PDR results under an EKF framework. Other research attempts to integrate active ranging devices. Sadruddin et al. [50] integrate inertial PDR as an odometry into a body-mounted laser scanner platform for simultaneous localization and mapping, which also enables more accurate pedestrian localization. Mao et al. [51] use a smartphone to measure distances to a surrounding wall. Wu et al. [48] recognize the steady-heading and non-steady-heading human activities to optimize the PDR without any prior knowledge and Zhang et al. [47] use the Wavelet-CNN deep learning network to train the smartphones' built-in MEMS sensor data for human activity recognition.

This paper utilizes an external FMCW radar to assist PDR tracking. Since such a radar is entirely a self-transmitting and self-receiving device, our RadarPDR scheme can be applied to environments without prior indoor layouts and does not need additional expensive hardware infrastructure.

3. Problem Description

We investigate how to use a ranging radar to adjust pedestrian trajectory for indoor tracking. As illustrated in Figure 2, a user holds a smartphone with an FMCW radar, and a sequence of steps represents his trajectory. Let $s_n = (h_n, x_n, y_n)$ denote the pedestrian pose at the n -th step, where h_n is the heading orientation and (x_n, y_n) the step coordinate. The radar is used to measure the distance between the user and a wall. The sampling rate of radar is 10 Hz, which is higher than the pedestrian walking speed. Thus, between two steps, we can have M wall distance measurements, denoted by $d_{n,1}, \dots, d_{n,M}$. Each observation $d_{n,m}$ links an observation point and a wall point. The observation point is the pedestrian pose when this observation occurs. We assume that M observation points are evenly distributed within one step. The wall point is a virtual point on the wall surface, and the distance between the two points is $d_{n,m}$.

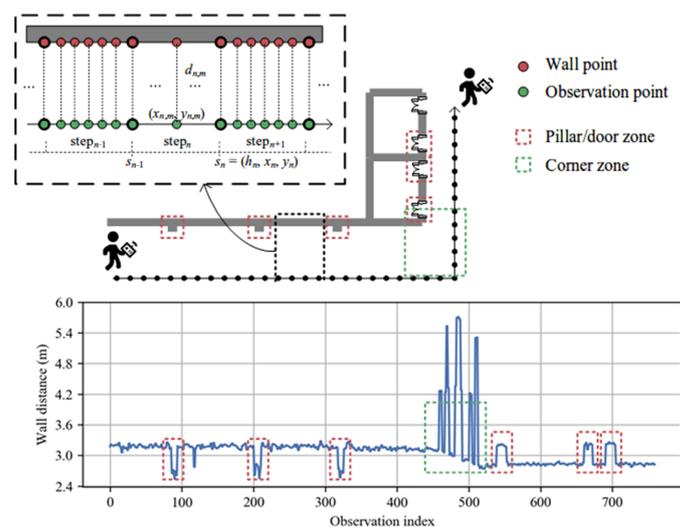


Figure 2. Problem description. The user trajectory is represented by a sequence of pedestrian poses. Each pose s_n is denoted as a tuple (h_n, x_n, y_n) with h_n the heading orientation and (x_n, y_n) the step coordinate. The wall distance between adjacent steps establishes the positional relationship between pedestrian pose and the surrounding walls. Each wall distance $d_{n,m}$ links an observation point on the trajectory and a wall point on the wall surface. The wall distance observations suffer from large fluctuations due to corner and barrier (e.g., pillar or door) zones on the wall.

The proposed RadarPDR adopts a particle filtering framework for trajectory adjustment. Although the principle is straightforward, its implementation faces many practical challenges: From our field experiences, if a user walks strictly in parallel with a flat wall, distance measurements of an FMCW radar have high accuracy, with ranging errors often smaller than 0.2 m. However, such ideal cases are not often in practical indoor environments. On the one hand, a user may not always walk along with a strict line, and some turning steps are not unexpected. Sometimes, other people may pass across the user and a wall, leading to interferences in distance measurements. On the other hand, it is not easy to find a long flat wall in many buildings, as indoor walls often have some concave and convex structures, such as recessing doorways or protruding pillars. Furthermore, when passing by a corner, wall distance measurements have even larger fluctuations, which often leads to poor positional information for failed trajectory adjustment.

Figure 2 plots the distance measurements of a trajectory in our field experiments. On the one hand, it can be seen that irregular wall structures can cause some distance pulses, and wall corners can cause large distance variations. On the other hand, it can be seen that, besides such distance pulses and large variations, some distance measurements can remain relatively stable in a short walking interval. These observations inspire us to design an adaptive scheme to enforce trajectory adjustment: That is, the pedestrian pose s_n is subject to adjustment only when the wall distance $d_{n,m}$ contains sufficient positional information. Otherwise, we accept the pose estimation from the underlying dead reckoning system. Now, the research questions can be summarized as follows:

- When to enforce pedestrian pose adjustment?
- How to correct measurement for wall distance $d_{n,m}$?
- How to adjust a pedestrian pose s_n based on $d_{n,m}$?

We next present our solution RadarPDR to these questions.

4. The Proposed RadarPDR Scheme

4.1. Overview

Figure 3 presents the system diagram of our RadarPDR scheme, which consists of four main modules: (i) data sensing and preprocessing; (ii) inertial PDR, (iii) wall model calibration; (iv) multi-source information fusion.

As sources, accelerometer, gyroscope, and radar continuously yield raw observation data according to their acquisition frequency. The data preprocessing module cuts these data streams into step-wise pieces implemented by peak detection [52] based on acceleration observations. Once a new step is completed, these sorts of observations will be distributed into the following processing modules.

Following standard procedures, the inertial PDR module works to estimate step length and heading increment using inertial sensor observations. Step length is estimated by the Weinberg algorithm [53] based on the acceleration series; heading increment is computed based on the azimuth observations from the gyroscope-based orientation sensor embedded in our Android smartphone. The wall model calibration module aims to tackle the corner and barrier problems per our discussion in the previous section. In contrast, the multi-source information fusion module tries to integrate all available information and finally estimates an accurate pedestrian trajectory.

4.2. Wall Model Calibration

This module focuses on two realistic challenges in the wall model: the corner problem and the barrier problem. We assign an operation indicator for each step based on corner detection, which works to tackle the adjustment failure in the corner zone by an adaptive operation. Moreover, a wall distance correction algorithm is developed to handle the outlier issue due to walls' barrier structures.

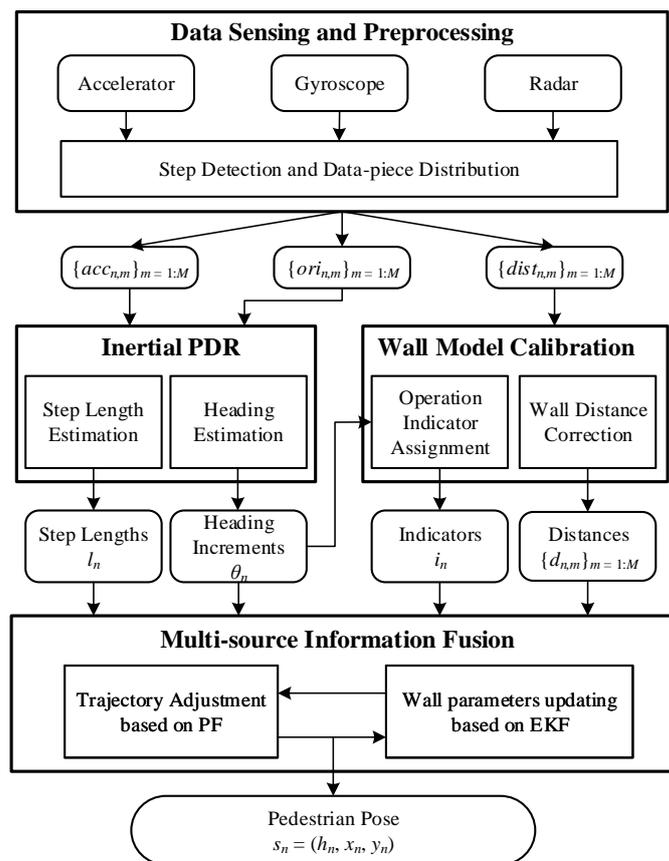


Figure 3. System diagram.

4.2.1. Operation Indicator Assignment

The proposed scheme is based on a flat-wall observation that the wall is approximately flat during several adjacent steps. A wall parameter tuple (α, β) with slope α the slope and β the intercept can represent the wall model. Long trajectories can be related to segmented flat walls with several corners. For example, Figure 2 depicts a trajectory beside a wall consisting of two flat-segments and one corner. In the non-corner zone, each flat-segment needs independent parameter initialization and an updating procedure. In addition, wall distances obtained in the corner zone show significant fluctuation. In short, a corner and non-corner zone need different operations in the following modules. Therefore, we generate an operation indicator for every step based on corner detection. In RadarPDR, each step belongs to one of three states: operate, suspend, and reset. The operation of each state is as follows: (i) trajectory adjustment and wall parameter updating; (ii) dead reckoning; (iii) dead reckoning and wall parameter initialization.

The indicator assignment principle is as follows: Firstly, if the heading increment θ_n is more significant than an angle threshold T_θ , then n -step is judged to be in a corner zone, and its corresponding indicator i_n is assigned to *suspend*; otherwise we think that n -step is in a flat-segment zone, and it needs further judgment. Then, if n -step belongs to the beginning N_i steps of the flat-segment, i_n is assigned to *reset*; otherwise, i_n is set to *operate*. Figure 4 presents wall distance observations, heading orientations, and corresponding indicators of several steps around a corner, where $N_i = 1$ and $T_\theta = 25^\circ$. The T_θ is chosen according to our experiences and has been proven practicable in our experiments to detect whether a pedestrian has cornered.

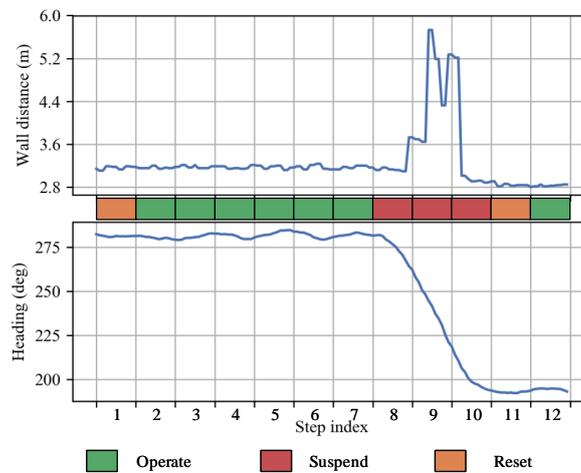


Figure 4. Illustration of the operation indicator assignment strategy. The plots show wall distance observations (above) and heading orientations (below) of several steps around a corner in a typical trajectory. *Suspend* is assigned to corner steps where heading increment $\theta_n >$ threshold T_θ . *Reset* is assigned to the beginning N_i steps of each non-corner segment. Other steps are indicated as *Operate* state.

4.2.2. Wall Distance Correction

In this paper, barrier refers to the convex and concave structures embedded in a flat wall, such as recessing doorways or protruding pillars. These unexpected barriers violate the flat-wall model, yielding noises in wall observations. Figure 5 depicts a series of noisy wall distance data measured in a field experiment. We classify these noises into two categories: (i) random noise; (ii) impulse noise. The random noise comes from inherent radar ranging error, just like many other sensors. It has a relatively small amplitude; the impulse noises come from internal signal fluctuation or wall barrier structures, presenting a much larger amplitude than random noise. Our distance correction algorithm discussed in this subsection aims to handle both random and impulse noises to ensure that the flat-wall model is well satisfied.

Firstly, we discuss the random noise, which can be approximated to Gaussian white noise. To simplify the description, we assume the number of single-step observations $M = 1$ and simplify the notation of the single observation $d_{n,1}$ at the n -th step to d_n . The general situation ($M > 1$) will be explained later. Let v_n and q_n denote the change rate of d_n and the process noise. Let $dist_n$ and r_n denote the wall distance observation and the random noise following a Gaussian distribution with zero mean and R_n variance, respectively. The random noise can be corrected by a Kalman filter (KF) with the following state-space model:

Prediction:

$$d_n = d_{n-1} + v_n \quad (1)$$

$$v_n = v_{n-1} + q_n \quad (2)$$

Observation:

$$dist_n = d_n + r_n \quad (3)$$

The prediction functions describe the state-transition procedure of wall distance and its change rate, and they are designed by the fact that pedestrians tend to walk in a straight line or change their directions slowly in most cases. When a pedestrian walks in a straight line, the wall distance change rate v_n remains unchanged, while the process noise $q_n \sim N(0, Q_n)$ gives a fuzzy range for the direction changing case.

Considering the impulse noise denoted by I_n , the above observation function is rewritten as

$$dist_n = d_n + r_n + I_n \quad (4)$$

Due to the lack of precise statistics of I_n , the impulse noise cannot be eliminated easily by general Kalman filtering as the random noise is. Thus, we design a two-stage correction strategy. That is, for each observation $dist_n$, we judge whether it is an impulse noise firstly, and then select the correction method correspondingly.

The judgment stage is based on a threshold method imposed on the wall distance change rate of v_n . On the one hand, a normal walking trajectory can be regarded as a continuous curve with a relatively large curvature radius, usually a few meters, except in the corner zone, which leads to relatively small v_n in inlier observations. On the other hand, a much larger amplitude of impulse noise or outlier observation makes the wall distance change rate estimation $\hat{v}_{n|n}$ severely deviate its normal range. In short, there is a huge margin between the change rate of inlier wall distance and that of the outlier one, as shown in Figure 6. Therefore, we define the outlier detection criterion as

$$\hat{v}_{n|n} \geq T_v \quad (5)$$

where T_v is the upper limit of the change rate of the inlier wall distance selected in the margin.

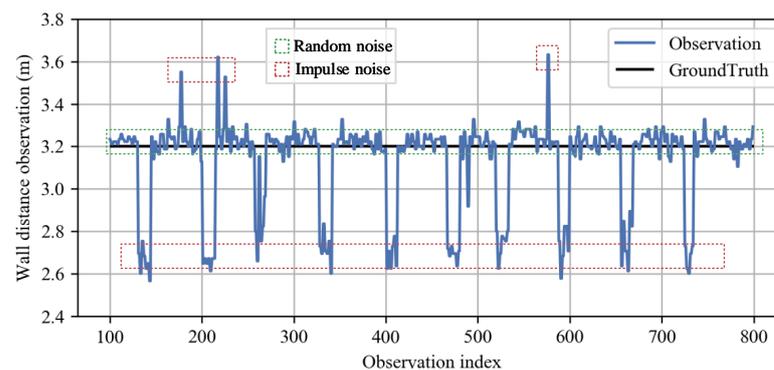


Figure 5. Classification of wall distance observation noise. The plot shows wall distance observations in a straight walking test along a parallel wall. The random noise with small amplitude comes from inherent radar ranging error, while the impulse noise with larger amplitude comes from signal fluctuation or wall barrier structures.

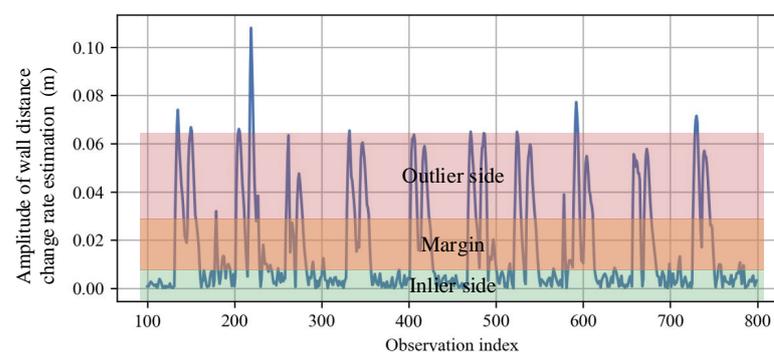


Figure 6. Judgment margin. The plot shows the amplitude of wall distance change rate $\hat{v}_{n|n}$ estimated on the wall distance observations in Figure 5. The amplitudes of outlier observations are much greater than the inlier ones, thus the judgment threshold T_v is easily selected from the margin area.

The correction stage adopts the KF solution with the state-space model given by Equations (1)–(3), if there is no outlier. Otherwise, it will switch to a modified KF version. In our problem, a KF's observation is much less reliable than its prediction when an outlier occurs. The prediction is close to the ground truth in a short time due to the slow-change

feature of wall distance change rate. Thus, we drop the noisy observation directly and use modified prediction functions to obtain wall distance correction. The modification lies in the fact that the trend instead of the single estimation of v_n is used here. The final correction is summarized as

$$\bar{d}_n = \begin{cases} \hat{d}_{n|n}, & \text{if } \hat{v}_{n|n} < T_v \text{ (inlier case),} \\ \hat{d}_{n-1|n-1} + \tilde{v}_n, & \text{if } \hat{v}_{n|n} \geq T_v \text{ (outlier case),} \end{cases} \quad (6)$$

where \tilde{v}_n denotes the trend of a few recent distances at the n -th step, calculated by an exponential average of wall distance change rate corrections.

The proposed wall distance correction algorithm in a general case is summarized in Algorithm 1. Note that the judgment-correction procedure is just simply repeated M times according to the order of occurring, when we have M observations in one step.

Algorithm 1 Wall distance correction algorithm

Input: $\{dist_{n,m}\}_{n=1:N, m=1:M}$: wall distance observations

Output: $\{\bar{d}_{n,m}\}_{n=1:N, m=1:M}$: wall distance corrections

for each step $n \in 1, 2, \dots, N$ **do**

for each observation $m \in 1, 2, \dots, M$ **do**

 1. Get state estimations by standard Kalman filtering:

$$\hat{d}_{n,m|m}, \hat{v}_{n,m|m} = \text{KF}(dist_{n,m})$$

 2. Detect outlier and compute correction:

if $\hat{v}_{n,m|m} < T_v$ **then**

$$\bar{d}_{n,m} = \hat{d}_{n,m|m}, \bar{v}_{n,m} = \hat{v}_{n,m|m}$$

else

$$\bar{d}_{n,m} = \hat{d}_{n,m-1|m-1} + \tilde{v}_{n,m}, \bar{v}_{n,m} = \tilde{v}_{n,m}$$

end if

 3. Update trend:

$$\tilde{v}_{n,m+1} = \gamma \bar{v}_{n,m} + (1 - \gamma) \bar{v}_{n,m}$$

end for

end for

4.3. Multi-Source Information Fusion

After processing by the upper modules, for each step, we now have information including step length l_n , heading increment θ_n and wall distance $d_{n,m}$. We next fuse them to estimate pedestrian pose s_n . Based on the Bayes filtering framework, we define a motion model and an observation model to output their relation. The motion model presents the recurrence relation of pedestrian pose s_n with step length l_n and heading increment θ_n as follows:

$$s_n = f(s_{n-1}, l_n, \theta_n) = \begin{bmatrix} h_{n-1} + \theta_n \\ x_{n-1} + l_n \sin h_n \\ y_{n-1} + l_n \cos h_n \end{bmatrix}. \quad (7)$$

As shown in Figure 7, given the pedestrian pose s_n , the ranging direction r_n and the wall distance d_n , the wall point (x_n^w, y_n^w) is represented as

$$x_n^w = x_n + d_n \sin r_n, \quad y_n^w = y_n + d_n \cos r_n. \quad (8)$$

Combining the wall model $y = \alpha_n x + \beta_n$ and the wall point (x_n^w, y_n^w) , d_n is represented as

$$y_n + d_n \cos r_n = \alpha_n (x_n + d_n \sin r_n) + \beta_n, \quad (9)$$

$$d_n = \frac{\alpha_n x_n + \beta_n - y_n}{\cos r_n - \alpha_n \sin r_n}. \quad (10)$$

Note that the ranging direction is always perpendicular to the pedestrian heading in our device setup. To replace r_n with the heading direction h_n under the relationship $h_n = r_n - \frac{\pi}{2}$, d_n is then represented as

$$d_n = \frac{\alpha_n x_n + \beta_n - y_n}{\cos(h_n + \frac{\pi}{2}) - \alpha_n \sin(h_n + \frac{\pi}{2})} = \frac{y_n - \alpha_n x_n - \beta_n}{\sin h_n + \alpha_n \cos h_n}. \quad (11)$$

In short, the observation model gives wall distance $d_{n,m}$ at observation point $s_{n,m}$ with wall parameters o_n as follows:

$$d_{n,m} = g(s_{n,m}, o_n) = \frac{y_{n,m} - \alpha_n x_{n,m} - \beta_n}{\sin h_n + \alpha_n \cos h_n}. \quad (12)$$

Now, it can be summarized that $l_n, \theta_n, d_{n,m}$ are the given information inherited from the upper modules, while s_n or $s_{n,m}, o_n$ are the unknown states to be estimated in this module (note that $s_n = s_{n,1}$). Furthermore, in the observation model, if $s_{n,m}$ serves as the state to be estimated, then o_n needs to be the known parameter and vice versa. Therefore, we adopt a PF-EKF architecture to solve this dual-estimation problem, where PF takes charge of trajectory adjustment for s_n , while EKF is in charge of wall parameter estimation for o_n .

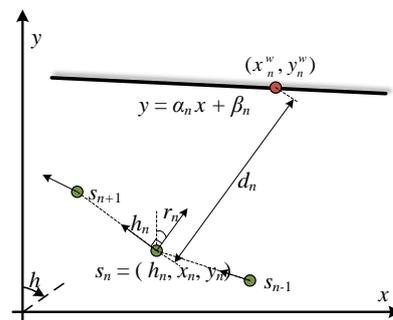


Figure 7. Illustration of the observation model. The wall distance d_n is the straight-line distance from the observation point $s_n = (h_n, x_n, y_n)$ to the wall point (x_n^w, y_n^w) along ranging direction r_n .

4.3.1. Trajectory Adjustment

The basic idea of PF-based trajectory adjustment [54] is as follows: During each round of iteration, PF makes a group of guesses, known as particles, around the unreliable state estimated by the motion model. This is the particle sampling procedure. Some of the particles are positive adjustments, closer to the ground truth, while others are negative since they have large deviations. We propose a judgment procedure to weight the particles. Finally, another sampling procedure is implemented on the weighted particles. Those with high weights are more likely to be selected, leading to an improvement in pedestrians' pose estimation. We next explain these three steps in details below.

The sampling procedure is performed on step length l_n and heading increment θ_n from which the errors of the pose originate. We assume that the errors of l_n and θ_n follow Gaussian distribution with zero mean and σ_l, σ_θ standard deviation, respectively:

$$\delta l_n^{[k]} \sim N(0, \sigma_l^2), \quad \delta \theta_n^{[k]} \sim N(0, \sigma_\theta^2) \quad (13)$$

Then, the corrections are represented by

$$l_n^{[k]} = l_n - \delta l_n^{[k]}, \quad \theta_n^{[k]} = \theta_n - \delta \theta_n^{[k]} \quad (14)$$

where n is the step index and k is the particle index. According to the motion model, particle pose is reckoned as

$$s_n^{[k]} = [h_n^{[k]}, x_n^{[k]}, y_n^{[k]}]^T = f(s_{n-1}, l_n^{[k]}, \theta_n^{[k]}). \quad (15)$$

K particles are generated within one step by repeating the above procedure.

The essence of weight calculation is to judge the reliability of particles. Each particle can obtain its wall distance prediction $\hat{d}_{n,m}^{[k]}$ by feeding its observation point $s_{n,m}^{[k]}$ and wall parameters to the observation model. At the same time, a distance observation $d_{n,m}$ obtained by radar is available. By calculating the proximity between $\hat{d}_{n,m}^{[k]}$ and $d_{n,m}$, we can obtain the particle weight $w_{n,m}^{[k]}$. This procedure is formulated as

$$\hat{d}_{n,m}^{[k]} = g(s_{n,m}^{[k]}, o_n) \quad (16)$$

$$w_{n,m}^{[k]} = \frac{1}{\sqrt{2\pi S_n}} \exp \frac{-(d_{n,m} - \hat{d}_{n,m}^{[k]})^2}{2S_n} \quad (17)$$

where S_n is the variance of distance observation residual $|d_{n,m} - \hat{d}_{n,m}^{[k]}|$ at the n -step and o_n is the wall parameter, and both of them are calculated in the EKF part. The weight $w_{n,m}^{[k]}$ considers the distance observation residual following a Gaussian distribution with zero mean and S_n variance, a larger residual meaning a lower weight. When we have M observations in the n -th step, we will obtain M weights for one particle. Then, the weight of the k -th particle at the n -th step is calculated by an average of these M weights:

$$w_n^{[k]} = \frac{1}{M} \sum_{m=1}^M w_{n,m}^{[k]} \quad (18)$$

Resampling is a two-sided procedure which aims to ensure that higher weights get resampled more while maintaining particle diversity. We adopt the systematic resampling algorithm introduced in [55]. After the resampling procedure, the final estimated pose state is obtained by averaging the resampled particles.

4.3.2. Wall Parameters Estimation

Due to the absence of a prior map information, the wall parameters necessary for particle weighting are unknown to us initially. The proposed wall parameter estimation algorithm consists of two phases: the initialization phase and the updating phase. The initialization phase happens at the beginning N_i steps of each flat-segment, while the updating phase happens at the remaining steps of that flat-segment.

We first introduce the initialization phase. Although the EKF algorithm is insensitive to initial values, wall parameters from random initialization may deviate from the ground truth a lot, which could distort the PF weighting procedure. To deal with such issues, we design an initialization strategy based on the flat-wall discovery as follows: Given an observation point $(h_n, x_{n,m}, y_{n,m})$ and wall distance observation $d_{n,m}$, its corresponding wall point $(x_{n,m}^w, y_{n,m}^w)$ can be calculated by the following geometric relationship:

$$x_{n,m}^w = x_{n,m} + d_{n,m} \cos(h_n) \quad (19)$$

$$y_{n,m}^w = y_{n,m} - d_{n,m} \sin(h_n). \quad (20)$$

The wall parameters are estimated by using the least-square method with all wall points within this phase. In this phase, pedestrian poses and observation points are estimated by dead reckoning via the motion model, since the PF-EKF architecture does not work without wall parameters.

We realize a step-by-step online estimation for wall parameters via extended Kalman filtering with pedestrian pose and wall distance observations during the updating phase. Let $o_n = [\alpha_n, \beta_n]$ denote the wall parameters at the n -th step. According to the flat-wall model, o_n changes little in several neighboring steps. We thus have the following state transition function:

$$o_n = F_n o_{n-1} + w_n, \quad (21)$$

where F_n is a two-dimensional identity matrix. The observation function is exactly the observation model given by Equation (12), rewritten as

$$d_n = g(s_n; o_n) + v_n. \quad (22)$$

Note that here s_n is considered as a known parameter from the PF. w_n and v_n are independent Gaussian noise processes with zero mean and W_n and V_n covariance matrices, respectively.

The extended KF utilizes only the estimated state $\hat{o}_{n-1|n-1}$ of the previous step and the current observation d_n to estimate the current state $\hat{o}_{n|n}$, and the update procedures at the n -th step are formulated below:

$$\hat{o}_{n|n-1} = F_n \hat{o}_{n-1|n-1} \quad (23)$$

$$P_{n|n-1} = F_n P_{n-1|n-1} F_n^T + W_n \quad (24)$$

$$S_n = G_n P_{n|n-1} G_n^T + V_n \quad (25)$$

$$K_n = P_{n|n-1} G_n^T S_n^{-1} \quad (26)$$

$$\hat{o}_{n|n} = \hat{o}_{n|n-1} + K_n (d_n - g(s_n; \hat{o}_{n|n-1})) \quad (27)$$

$$P_{n|n} = (I - K_n G_n) P_{n|n-1} \quad (28)$$

where S_n is the distance observation residual. G_n is the observation matrix that is equal to the Jacobi matrix of $g(s_n; o_n)|_{o_n=\hat{o}_{n|n-1}}$,

$$G_n = \left[\frac{(\beta_n - y_n) \cos h_n - x_n \sin h_n}{D^2} \quad \frac{1}{D} \right] \Big|_{o_n=\hat{o}_{n|n-1}} \quad (29)$$

where $D = \sin h_n + \alpha_n \cos h_n$. I is an identity matrix. Just like the processing style in wall distance correction, we repeat the update procedure M times according to the occurring order of these M wall distance observations at one step.

In the PF part, wall parameters must ensure pedestrian poses, while this is the exact opposite in the EKF part. Consequently, these two parts are concatenated to boost each other, as illustrated in Figure 8. At the iteration of n -step, the step length, heading increment, wall distance observations from previous modules and wall parameters, and residual covariance from the EKF are fed to the PF, which outputs pedestrian pose estimation after the three procedures of new particles' sampling, weighting, and resampling. Then, the pose estimated by PF and wall distance observations are fed into the EKF, which outputs updated wall parameters. The complete trajectory adjustment algorithm is summarized in Algorithm 2.

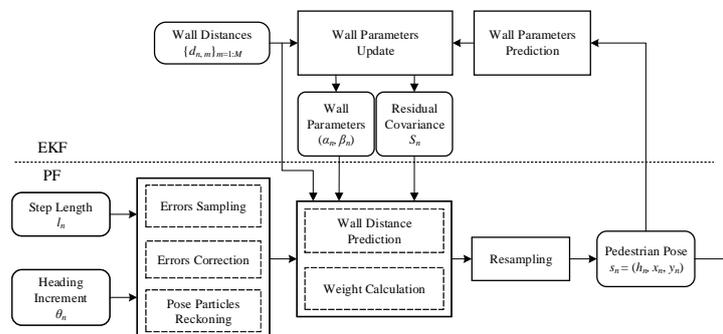


Figure 8. Flow chart of the PF-EKF algorithm.

Algorithm 2 Trajectory adjustment algorithm

Input: $\{l_n\}_{n=1}^N$: step lengths,
 $\{\theta_n\}_{n=1}^N$: heading increments,
 $\{d_{n,m}\}_{n=1:N,m=1:M}$: wall distance,
 $\{i_n\}_{n=1}^N$: indicators

Output: $\{s_n = (h_n, x_n, y_n)\}_{n=1:N}$: pose estimations

- 1: **for** each step $n \in 1, 2, \dots, N$ **do**
- 2: **if** the indicator i_n is *operate* **then**
- 3: **for** each particle $k \in 1, 2, \dots, K$ **do**
- 4: Get its pose $s_n^{[k]}$ according to Equations (13)–(15)
- 5: Get its weight $w_n^{[k]}$ according to Equations (16)–(18)
- 6: **end for**
- 7: Perform Resampling
- 8: Get pose estimation s_n by average of resampled particles
- 9: Update wall parameters o_n according to Equations (21)–(29)
- 10: **end if**
- 11: **if** the indicator i_n is *suspend* **then**
- 12: Get pose estimation s_n by dead reckoning motion model
- 13: **end if**
- 14: **if** the indicator i_n is *reset* **then**
- 15: Get pose estimation s_n by dead reckoning motion model
- 16: Initialize wall parameters
- 17: **end if**
- 18: **end for**

5. Experiments

5.1. Experiment Setup

We used Distance2Go (D2G), an FMCW radar module by Infineon, to acquire wall distances. The radar is powered by a highly integrated transceiver BGT24MTR11, operating at 24–24.2 GHz. The module is connected to a smartphone via a USB cable, ensuring data exchange and power supply. We developed our data acquisition app running on a smartphone to ensure that the radar module and the other two embedded sensors are operated in a synchronized way when collecting measurements. The sampling rate for wall distance is 10 Hz. The inertial data are obtained from an accelerometer and gyroscope embedded in our Honor 8 smartphone with a sampling rate at 20 Hz.

As shown in Figure 1, we adopted a supporting stand to fix the D2G module on a smartphone, keeping a constant 90° angle between heading and ranging direction. The experiments were conducted on the fifth floor of a classroom building at our university. We designed two different routes, as shown in Figure 9, to experiment our scheme. The L-shape and Z-shape route are combinations of straight lines and right angles, and their lengths are 192 m and 136.8 m, respectively. A student holding the device repeats each route 20 times.

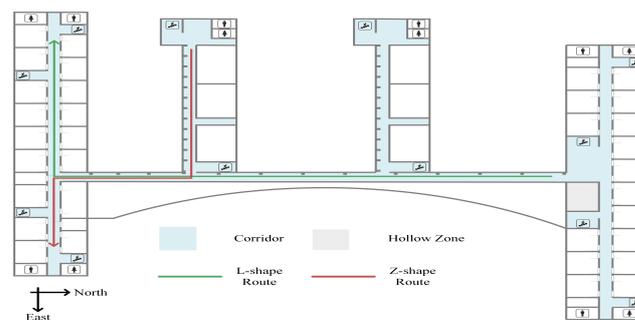


Figure 9. Experiment environment.

To verify the effectiveness of RadarPDR scheme and its modules, we design the following comparison schemes:

- PurePDR: The traditional inertial pedestrian dead-reckoning, using the motion model to estimate pedestrian trajectory;
- RadarPDR w/o DC: The RadarPDR scheme without the wall distance correction (DC) module;
- RadarPDR w/o PU: The RadarPDR scheme without the wall parameter updating (PU) module.

5.2. Wall Distance Results

In order to test the effectiveness of the wall distance correction module, we apply two different algorithms to correct wall distance observations on 20 trajectories of each route, including DC in RadarPDR and KF in RadarPDR w/o DC. Specifically, DC is the completed method summarized in Algorithm 1 while KF is the standard Kalman filter given by Equations (1)–(3) without the proposed outlier correction stage. Their parameters are set as: process noise variance $Q_n = 0.01^2$, observation noise variance $R_n = 0.2^2$, wall distance change rate limit $T_v = 0.01$, and smoothing factor $\gamma = 0.9$. For KF, the process noise variance Q_n is the same as DC, and the observation noise variance $R_n = 2^2$, which compensates for its weakness in handling outliers.

Since the PDR tracking is a recursive process, a small number of wall distance outliers can have a big impact on the whole trajectory position estimates. Therefore, the error range is more representative of an error correction algorithm's contribution to the whole system. Figure 10 presents wall distance error range by different algorithms. Although the observation noise setting of KF is much larger than DC, its correction effect is still limited, and the noise range by our scheme has been reduced by 30.7% and 40.7%, in the L-shape and Z-shape route, respectively. DC's correction performance is even better, reducing the noise by 93.2% and 96.6% in the two routes.

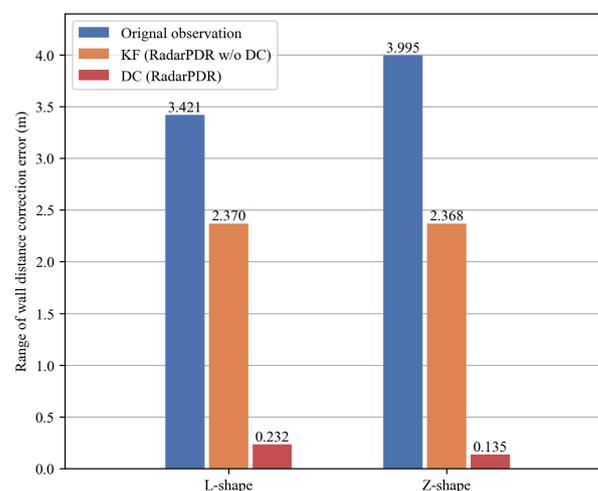


Figure 10. Wall distance error range of all the testing trajectories on the L-shape and the Z-shape routes.

5.3. Wall Parameters' Results

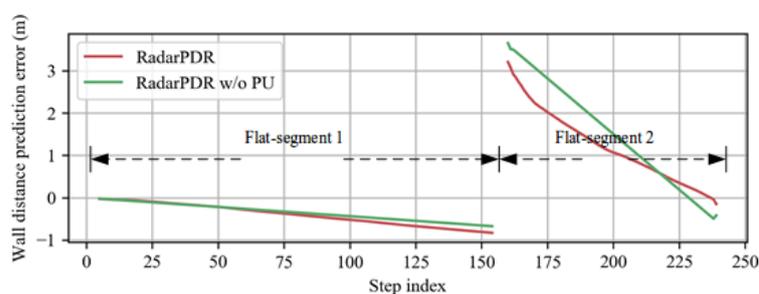
To demonstrate the effectiveness of the wall parameter estimation module, we compare the parameter estimation results of the RadarPDR and RadarPDR w/o PU scheme. The RadarPDR has an initialization phase and an updating phase, whose parameters are set as: the number of initialization steps $N_i = 4$, the process noise of wall estimation $W_n = [[0.01, 0], [0, 0.1]]^T$, and the observation noise $V_n = 2.4$. The RadarPDR w/o PU has only the initialization phase with the same parameter setting as RadarPDR. Since the range of wall parameter variables is infinite, the estimation error is not workable as an evaluation

criterion here. We define the *wall distance prediction error* (WDPE) to evaluate the estimation performance by

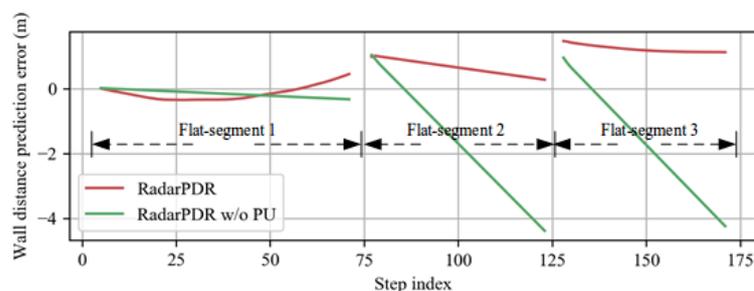
$$WDPE = \sum_{m=1}^M g(s_{n,m}^g, o_n) - d_{n,m}^g \quad (30)$$

where o_n is the wall parameter estimation at the n -th step; $s_{n,m}^g, d_{n,m}^g$ are the ground truth of the m -th observation point and wall distance at the n -th step.

Figure 11 shows the variation trend of the WDPE as the pedestrian moves. According to the observation model, the wall parameter estimation performance depends on wall distance observation and pedestrian pose estimation. After the wall distance correction, its noise level on the entire trajectory is almost the same, while the pedestrian pose estimations have a cumulative error effect. Therefore, the variation trend of WDPE along with the trajectory mainly depends on the cumulative error effect.



(a)



(b)

Figure 11. Wall distance prediction error. (a) average WDPE of all the testing trajectories on the L-shape route; (b) average WDPE of all the testing trajectories on the Z-shape route.

Firstly, we analyze the distribution characteristics of the initial WDPE of each flat-segment in a trajectory. The wall parameters from the initialization phase calculate the initial WDPEs. Thus, they are almost the same in the RadarPDR and RadarPDR w/o PU scheme. The initial WDPE in the first flat-segment is relatively small, while quite larger in the next flat-segments. This is because the pedestrian pose errors are small at the beginning of the trajectory and gradually increase as the walking distance increases, resulting in a large wall parameter re-initialization error after the corner zone. Then, we compare the WDPE of different schemes in each segment. In the first flat-segment of the L-shape and Z-shape route, the amplitude of WDPE in RadarPDR is slightly larger than that in RadarPDR w/o PU, which can be attributed to the cumulative error effect of pedestrian pose used in EKF updating. In the next flat-segments, however, the WDPE converges to a small amplitude in the RadarPDR but can only change linearly with a flawed initial direction in the RadarPDR w/o PU. This is because RadarPDR considers more observations for wall parameter updating, which is able to reduce the error in the initialization phase as only a

few initial observations are used. Furthermore, the initial direction in the RadarPDR w/o PU can easily deviate from ground truth according to Figure 11.

5.4. Trajectory Adjustment Results

In this experiment, we compare the performance of four different trajectory estimation/correction schemes. The PurePDR scheme only uses the motion model for trajectory estimation. Moreover, the rest are the proposed RadarPDR and its variants. They have the same parameters setting in the PF part: the number of particles $K = 128$, the number of initialization steps $N_i = 4$, the error scale of step length $\sigma_l = 0.4$ and heading increment $\sigma_\theta = 0.2$.

Figure 12 plots the adjusted trajectory results and their corresponding CDFs for L-shape and Z-shape routes. Compared with the PurePDR scheme, the RadarPDR w/o DC produces a comparable average localization error (ALE) in the L-shape route and even performs much worse in the Z-shape trajectory, which can be attributed to the large wall distance error deactivating the trajectory adjustment procedure. This observation also verifies the importance of the DC module. On the other hand, the RadarPDR w/o PU can achieve smaller ALE on both the L-shape and Z-shape routes. Moreover, the complete RadarPDR has the best trajectory adjustment performance, reducing its ALE by 48.1% in the L-shape route and 41.1% in the Z-shape route. It is also worth mentioning that the PurePDR overtakes the RadarPDR at the point (13.5 m, 0.98) in Figure 12d, resulting from one failed trajectory adjustment in the 20 tests on the Z-shape route. Combined with the analysis of WDPE in the previous section, this failure should be attributed to large initial wall parameter error, which makes the EKF estimation process divergent and leads to the wrong trajectory correction direction.

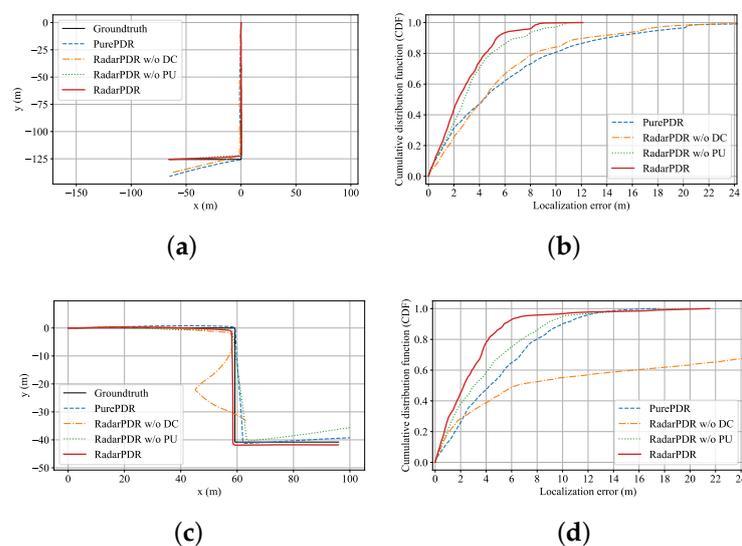


Figure 12. Trajectory and its CDF of ALE. (a) average trajectory of all the tests on the L-shape route; (b) the localization error CDF of all the testing trajectories on the L-shape route; (c) average trajectory of all the tests on the Z-shape route; (d) the localization error CDF of all the testing trajectories on the Z-shape route.

6. Conclusions

In this paper, we have reported our initial attempts of applying an external FMCW radar to assist PDR-based indoor pedestrian tracking. Although such a ranging radar can measure wall distances, its measurements during walking could come with large errors due to complicated indoor layouts and diverse trajectory types. The proposed RadarPDR deploys two algorithms addressing wall distance correction and trajectory adjustment. The trajectory adjustment algorithm fuses step lengths and heading increments from accelerometers and gyroscopes with wall distance from the radar, where the EKF takes

change of wall parameter estimation based on wall distance observations and pedestrian locations. Simultaneously, the PF utilizes outputs of the EKF and distance observations to adjust pedestrian trajectory. Our field experiments have validated the effectiveness of the proposed RadarPDR in terms of higher tracking accuracy.

Author Contributions: Conceptualization, J.H. and B.W.; Methodology, J.H., W.X., Q.Z. and B.W.; Validation, J.H., W.X. and Q.Z.; Writing—original draft, J.H.; Writing—review and editing, B.W.; Visualization, W.X.; Supervision, B.W. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the Shenzhen key technology research project: Key Technology Research and Development of High-Speed and High-Precision Vertical Five-Axis Machining Center (2019N048).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. Davidson, P.; Piché, R. A survey of selected indoor positioning methods for smartphones. *IEEE Commun. Surv. Tutor.* **2016**, *19*, 1347–1370. [[CrossRef](#)]
2. Basiri, A.; Lohan, E.S.; Moore, T.; Winstanley, A.; Peltola, P.; Hill, C.; Amirian, P.; e Silva, P.F. Indoor location based services challenges, requirements and usability of current solutions. *Comput. Sci. Rev.* **2017**, *24*, 1–12. [[CrossRef](#)]
3. Díez, L.E.; Bahillo, A.; Otegui, J.; Otím, T. Step length estimation methods based on inertial sensors: A review. *IEEE Sens. J.* **2018**, *18*, 6908–6926. [[CrossRef](#)]
4. Hou, X.; Bergmann, J. Pedestrian Dead Reckoning with Wearable Sensors: A Systematic Review. *IEEE Sens. J.* **2020**, *21*, 143–152. [[CrossRef](#)]
5. Harle, R. A survey of indoor inertial positioning systems for pedestrians. *IEEE Commun. Surv. Tutor.* **2013**, *15*, 1281–1293. [[CrossRef](#)]
6. Groves, P.D. Navigation using inertial sensors [Tutorial]. *IEEE Aerosp. Electron. Syst. Mag.* **2015**, *30*, 42–69. [[CrossRef](#)]
7. Wu, Y.; Zhu, H.B.; Du, Q.X.; Tang, S.M. A survey of the research status of pedestrian dead reckoning systems based on inertial sensors. *Int. J. Autom. Comput.* **2019**, *16*, 65–83. [[CrossRef](#)]
8. Foxlin, E. Pedestrian tracking with shoe-mounted inertial sensors. *IEEE Comput. Graph. Appl.* **2005**, *25*, 38–46. [[CrossRef](#)]
9. Park, S.K.; Suh, Y.S. A zero velocity detection algorithm using inertial sensors for pedestrian navigation systems. *Sensors* **2010**, *10*, 9163–9178. [[CrossRef](#)]
10. Zhang, W.; Li, X.; Wei, D.; Ji, X.; Yuan, H. A foot-mounted PDR system based on IMU/EKF+ HMM+ ZUPT+ ZARU+ HDR+ compass algorithm. In Proceedings of the 2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Sapporo, Japan, 18–21 September 2017; pp. 1–5.
11. Zhang, W.; Wei, D.; Yuan, H.; Yang, G. Cooperative positioning method of dual foot-mounted inertial pedestrian dead reckoning systems. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 8502114. [[CrossRef](#)]
12. Tong, X.; Su, Y.; Li, Z.; Si, C.; Han, G.; Ning, J.; Yang, F. A double-step unscented Kalman filter and HMM-based zero-velocity update for pedestrian dead reckoning using MEMS sensors. *IEEE Trans. Ind. Electron.* **2019**, *67*, 581–591. [[CrossRef](#)]
13. Borenstein, J.; Ojeda, L.; Kwanmuang, S. Heuristic Reduction of Gyro Drift in IMU-based Personnel Tracking Systems. *J. Navig.* **2009**, *62*, 41–58. [[CrossRef](#)]
14. Borenstein, J.; Ojeda, L. Heuristic drift elimination for personnel tracking systems. *J. Navig.* **2010**, *63*, 591. [[CrossRef](#)]
15. Jiménez, A.R.; Seco, F.; Zampella, F.; Prieto, J.C.; Guevara, J. Improved Heuristic Drift Elimination (iHDE) for pedestrian navigation in complex buildings. In Proceedings of the 2011 International Conference on Indoor Positioning and Indoor Navigation, Guimarães, Portugal, 21–23 September 2011; pp. 1–8.
16. Ju, H.; Lee, M.S.; Park, C.G.; Lee, S.; Park, S. Advanced heuristic drift elimination for indoor pedestrian navigation. In Proceedings of the 2014 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Busan, Republic of Korea, 27–30 October 2014; pp. 729–732.
17. Ju, H.; Park, S.Y.; Park, C.G. A smartphone-based pedestrian dead reckoning system with multiple virtual tracking for indoor navigation. *IEEE Sens. J.* **2018**, *18*, 6756–6764. [[CrossRef](#)]
18. Radu, V.; Marina, M.K. HiMLoc: Indoor smartphone localization via activity aware pedestrian dead reckoning with selective crowdsourced WiFi fingerprinting. In Proceedings of the International Conference on Indoor Positioning and Indoor Navigation, Montbeliard, France, 28–31 October 2013; pp. 1–10.

19. Shi, L.F.; Wang, Y.; Liu, G.X.; Chen, S.; Zhao, Y.L.; Shi, Y.F. A Fusion Algorithm of Indoor Positioning Based on PDR and RSS Fingerprint. *IEEE Sens. J.* **2018**, *18*, 9691–9698. [[CrossRef](#)]
20. Li, W.; Wang, B.; Yang, L.T.; Zhou, M. RMapTAFa: Radio map construction based on trajectory adjustment and fingerprint amendment. *IEEE Access* **2019**, *7*, 14488–14500. [[CrossRef](#)]
21. Zou, H.; Chen, Z.; Jiang, H.; Xie, L.; Spanos, C. Accurate indoor localization and tracking using mobile phone inertial sensors, WiFi and iBeacon. In Proceedings of the 2017 IEEE International Symposium on Inertial Sensors and Systems (INERTIAL), Kauai, HI, USA, 27–30 March 2017; pp. 1–4.
22. Tiwari, S.; Kumar Jain, V. HILS: Hybrid indoor localisation system using Wi-Fi received signal strength and inertial sensor's measurements of smart-phone. *IET Commun.* **2019**, *13*, 1595–1606. [[CrossRef](#)]
23. Li, H.; Guo, H.; Qi, Y.; Deng, L.; Yu, M. Research on multi-sensor pedestrian dead reckoning method with UKF algorithm. *Measurement* **2021**, *169*, 108524. [[CrossRef](#)]
24. Li, X.; Wang, J.; Liu, C. A Bluetooth/PDR integration algorithm for an indoor positioning system. *Sensors* **2015**, *15*, 24862–24885. [[CrossRef](#)]
25. Yu, N.; Zhan, X.; Zhao, S.; Wu, Y.; Feng, R. A precise dead reckoning algorithm based on Bluetooth and multiple sensors. *IEEE Internet Things J.* **2017**, *5*, 336–351. [[CrossRef](#)]
26. Chen, P.; Kuang, Y.; Chen, X. A UWB/improved PDR integration algorithm applied to dynamic indoor positioning for pedestrians. *Sensors* **2017**, *17*, 2065. [[CrossRef](#)] [[PubMed](#)]
27. Guo, S.; Zhang, Y.; Gui, X.; Han, L. An Improved PDR/UWB Integrated System for Indoor Navigation Applications. *IEEE Sens. J.* **2020**, *20*, 8046–8061. [[CrossRef](#)]
28. Tian, Q.; Kevin, I.; Wang, K.; Salcic, Z. A low-cost INS and UWB fusion pedestrian tracking system. *IEEE Sens. J.* **2019**, *19*, 3733–3740. [[CrossRef](#)]
29. Wang, Y.; Zhao, H. Improved smartphone-based indoor pedestrian dead reckoning assisted by visible light positioning. *IEEE Sens. J.* **2018**, *19*, 2902–2908. [[CrossRef](#)]
30. Abdelbar, M.; Buehrer, R.M. Improving cellular positioning indoors through trajectory matching. In Proceedings of the 2016 IEEE/ION Position, Location and Navigation Symposium (PLANS), Savannah, GA, USA, 11–14 April 2016; pp. 219–224.
31. Lan, K.C.; Shih, W.Y. On calibrating the sensor errors of a PDR-based indoor localization system. *Sensors* **2013**, *13*, 4781–4810. [[CrossRef](#)]
32. Nguyen-Huu, K.; Lee, K.; Lee, S.W. An indoor positioning system using pedestrian dead reckoning with WiFi and map-matching aided. In Proceedings of the 2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Sapporo, Japan, 18–21 September 2017; pp. 1–8.
33. Xia, H.; Zuo, J.; Liu, S.; Qiao, Y. Indoor localization on smartphones using built-in sensors and map constraints. *IEEE Trans. Instrum. Meas.* **2018**, *68*, 1189–1198. [[CrossRef](#)]
34. Xu, S.; Chen, R.; Yu, Y.; Guo, G.; Huang, L. Locating smartphones indoors using built-in sensors and Wi-Fi ranging with an enhanced particle filter. *IEEE Access* **2019**, *7*, 95140–95153. [[CrossRef](#)]
35. Koroglu, M.T.; Korkmaz, M.; Yilmaz, A.; Durdu, A. Multiple Hypothesis Testing Approach to Pedestrian INS with Map-Matching. In Proceedings of the 2019 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Pisa, Italy, 30 September–3 October 2019; pp. 1–8.
36. Qiu, S.; Wang, Z.; Zhao, H.; Qin, K.; Li, Z.; Hu, H. Inertial/magnetic sensors based pedestrian dead reckoning by means of multi-sensor fusion. *Inf. Fusion* **2018**, *39*, 108–119. [[CrossRef](#)]
37. Wang, X.; Jiang, M.; Guo, Z.; Hu, N.; Sun, Z.; Liu, J. An indoor positioning method for smartphones using landmarks and PDR. *Sensors* **2016**, *16*, 2135. [[CrossRef](#)]
38. Zhao, W.; Han, S.; Hu, R.Q.; Meng, W.; Jia, Z. Crowdsourcing and multisource fusion-based fingerprint sensing in smartphone localization. *IEEE Sens. J.* **2018**, *18*, 3236–3247. [[CrossRef](#)]
39. Liu, X.; Jiao, Z.; Chen, L.; Pan, Y.; Lu, X.; Ruan, Y. An Enhanced Pedestrian Dead Reckoning Aided With DTMB Signals. *IEEE Trans. Broadcast.* **2022**, *68*, 407–413. [[CrossRef](#)]
40. Zhou, Z.; Feng, W.; Li, P.; Liu, Z.; Xu, X.; Yao, Y. A fusion method of pedestrian dead reckoning and pseudo indoor plan based on conditional random field. *Measurement* **2023**, *207*, 112417. [[CrossRef](#)]
41. Yamagishi, S.; Jing, L. Pedestrian Dead Reckoning with Low-Cost Foot-Mounted IMU Sensor. *Micromachines* **2022**, *13*, 610. [[CrossRef](#)] [[PubMed](#)]
42. Tao, X.; Shi, T.; Ma, X.; Zhang, H.; Pei, Z. An improved indoor pedestrian dead reckoning algorithm using ambient light and sensors. *Multimed. Tools Appl.* **2022**, *81*, 32581–32592. [[CrossRef](#)]
43. De Cock, C.; Joseph, W.; Martens, L.; Trogh, J.; Plets, D. Multi-floor indoor pedestrian dead reckoning with a backtracking particle filter and viterbi-based floor number detection. *Sensors* **2021**, *21*, 4565. [[CrossRef](#)] [[PubMed](#)]
44. Tu, P.; Li, J.; Wang, H.; Wang, K.; Yuan, Y. Epidemic contact tracing with campus WiFi network and smartphone-based pedestrian dead reckoning. *IEEE Sens. J.* **2021**, *21*, 19255–19267. [[CrossRef](#)]
45. Asraf, O.; Shama, F.; Klein, I. PDRNet: A deep-learning pedestrian dead reckoning framework. *IEEE Sens. J.* **2021**, *22*, 4932–4939. [[CrossRef](#)]
46. Sun, M.; Wang, Y.; Xu, S.; Qi, H.; Hu, X. Indoor positioning tightly coupled Wi-Fi FTM ranging and PDR based on the extended Kalman filter for smartphones. *IEEE Access* **2020**, *8*, 49671–49684. [[CrossRef](#)]

47. Zhang, L.; Huang, L.; Yi, Q.; Wang, X.; Zhang, D.; Zhang, G. Positioning Method of Pedestrian Dead Reckoning Based on Human Activity Recognition Assistance. In Proceedings of the 2022 IEEE 12th International Conference on Indoor Positioning and Indoor Navigation (IPIN), Beijing, China, 5–8 September 2022; pp. 1–8.
48. Wu, B.; Ma, C.; Poslad, S.; Selviah, D.R. An adaptive human activity-aided hand-held smartphone-based pedestrian dead reckoning positioning system. *Remote. Sens.* **2021**, *13*, 2137. [[CrossRef](#)]
49. Yan, D.; Shi, C.; Li, T.; Li, Y. FlexPDR: Fully flexible pedestrian dead reckoning using online multimode recognition and time-series decomposition. *IEEE Internet Things J.* **2022**, *9*, 15240–15254. [[CrossRef](#)]
50. Sadruddin, H.; Mahmoud, A.; Atia, M.M. Enhancing Body-Mounted LiDAR SLAM using an IMU-based pedestrian dead reckoning (PDR) Model. In Proceedings of the 2020 IEEE 63rd International Midwest Symposium on Circuits and Systems (MWSCAS), Springfield, MA, USA, 9–12 August 2020; pp. 901–904.
51. Mao, P.; Mei, W.; Xiyu, S.; Liyan, L.; Kun, R. Smartphone-based indoor localization by fusion of acoustic ranging and PDR. *Chin. J. Internet Things* **2018**, *2*, 58.
52. Kim, J.W.; Jang, H.J.; Hwang, D.H.; Park, C. A step, stride and heading determination for the pedestrian navigation system. *J. Glob. Position. Syst.* **2004**, *3*, 273–279. [[CrossRef](#)]
53. Weinberg, H. Using the ADXL202 in pedometer and personal navigation applications. *Analog. Devices AN-602 Appl. Note* **2002**, *2*, 1–6.
54. Gustafsson, F. Particle filter theory and practice with positioning applications. *IEEE Aerosp. Electron. Syst. Mag.* **2010**, *25*, 53–82. [[CrossRef](#)]
55. Kitagawa, G. Monte Carlo filter and smoother for non-Gaussian nonlinear state space models. *J. Comput. Graph. Stat.* **1996**, *5*, 1–25. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.