*Micromachines* **2015**, *6*, 793-812; doi:10.3390/mi6060793

OPEN ACCESS

*micromachines*

**ISSN 2072-666X**

www.mdpi.com/journal/micromachines

*Article*

# PDR/INS/WiFi Integration Based on Handheld Devices for Indoor Pedestrian Navigation

**Yuan Zhuang [1,*], Haiyu Lan [1], You Li [1,2] and Naser El-Sheimy [1]**

[1] Department of Geomatics Engineering, University of Calgary, Calgary, AB T2N 1N4, Canada; E-Mails: hlan@ucalgary.ca (H.L.); liyou331@gmail.com (Y.L.); elsheimy@ucalgary.ca (N.E.)

[2] GNSS Research Center, Wuhan University, 129 Luoyu Road, Wuhan 430079, China

* Author to whom correspondence should be addressed; E-Mail: zhy.0908@gmail.com; Tel.: +1-403-210-7897.

Academic Editor: Stefano Mariani

*Received: 1 June 2015 / Accepted: 17 June 2015 / Published: 23 June 2015*

**Abstract:** Providing an accurate and practical navigation solution anywhere with portable devices, such as smartphones, is still a challenge, especially in environments where global navigation satellite systems (GNSS) signals are not available or are degraded. This paper proposes a new algorithm that integrates inertial navigation system (INS) and pedestrian dead reckoning (PDR) to combine the advantages of both mechanizations for micro-electro-mechanical systems (MEMS) sensors in pedestrian navigation applications. In this PDR/INS integration algorithm, a pseudo-velocity-vector, which is composed of the PDR-derived forward speed and zero lateral and vertical speeds from non-holonomic constraints (NHC), works as an update for the INS to limit the velocity errors. To further limit the drift of MEMS inertial sensors, trilateration-based WiFi positions with small variances are also selected as updates for the PDR/INS integrated system. The experiments illustrate that positioning error is decreased by 60%–75% by using the proposed PDR/INS integrated MEMS solution when compared with PDR. The positioning error is further decreased by 15%–55% if the proposed PDR/INS/WiFi integrated solution is implemented. The average accuracy of the proposed PDR/INS/WiFi integration algorithm achieves 4.5 m in indoor environments.

**Keywords:** PDR/INS/WiFi integration; PDR/INS integration; pseudo-velocity update; indoor pedestrian navigation; smartphone; motion constraints

## 1. Introduction

Global Navigation Satellite System (GNSS) positioning systems calculate the position and velocity of GNSS receivers by means of trilateration techniques [1,2]. GNSS provides an accurate positioning solution for pedestrian navigation in open-sky environments. However, GNSS faces problems such as signal blockage and multipath in urban areas and other GNSS-challenging environments. In these cases, GNSS/INS (inertial navigation system) integrated systems are usually used for navigation applications [3–5]. However, GNSS/INS integrated systems cannot work well for pedestrian navigation in deep indoor environments where GNSS is totally unavailable. On the other hand, the demand for indoor navigation is quickly increasing in various applications including: health care monitoring, logistics, Location Based Services (LBS), emergency services, tourism, and people management [6]. Several different techniques have been developed for indoor navigation, such as WiFi [7,8], Bluetooth [9], near field communication (NFC) [10,11], radio-frequency identification (RFID) [12], and micro-electro-mechanical systems (MEMS) sensors [13,14]. To date, self-contained navigation systems based on MEMS sensors can be applied to different indoor applications, including mobile robot navigation [15] and pedestrian navigation [16]. There are two types of mechanizations for sensor-based pedestrian navigation: INS [17,18] and pedestrian dead reckoning (PDR) [2]. The INS mechanization calculates the position, velocity, and attitude (PVA) by integrating raw data from accelerometers and gyroscopes. This mechanization can provide 3D PVA information, however, navigation errors increase rapidly with time due to the drift characteristics of MEMS sensors.

To improve the MEMS navigation performance for pedestrians, PDR is proposed to reduce the accumulated navigation errors. PDR has four critical procedures: step detection, step/stride length estimation, heading estimation, and 2D position calculation. PDR provides a more accurate position solution than INS, without other aiding sources, because it uses fewer integration calculations. However, there are two assumptions in the typical PDR algorithm: (1) the handheld device is leveled (roll and pitch are 0 degrees), and (2) the device's heading is kept the same as the user. Actually, the roll and pitch cannot always be 0 degrees. In this case, PDR-based heading, calculated by the direct integration of the data from the vertical gyroscope, is inaccurate. The heading estimation error will finally affect the positioning accuracy. Furthermore, the PDR navigation solution still drifts with time due to the MEMS sensor errors. Therefore, both INS and PDR require additional aiding sources, such as GNSS, WiFi and magnetometers, to reduce the navigation errors [19–21]. As previously discussed, GNSS is unavailable and unreliable in indoor environment. Iron materials and electronic devices indoors can easily disturb magnetometers. On the other hand, most public buildings such as universities, colleges, airports, shopping malls, and office buildings already have well-established WiFi infrastructure [7], making WiFi the main aiding resource for INS and PDR indoors.

Fingerprinting and trilateration are two main approaches for WiFi positioning [22]. Fingerprinting-based WiFi positioning usually has two operating phases: the pre-survey phase and the online positioning phase [23]. In the pre-survey phase, Received Signal Strength (RSS) values from available Access Points (APs) and position information are collected as fingerprints for creating the radio map database. In the online positioning phase, the phone's position is estimated by comparing observed RSS values with the fingerprints in the pre-built database. Trilateration-based WiFi positioning first calculates the ranges between the phone/device and APs through the wireless signal propagation model. Then, the

phone's position is estimated through the use of trilateration [24]. Fingerprinting usually provides more accurate position solutions at the cost of survey work in the pre-survey phase.

In this paper, we propose a new algorithm, which integrates INS and PDR for MEMS inertial-sensors-based pedestrian navigation. In this algorithm, step detection and step length estimation are kept the same as the traditional PDR algorithm. Then, the step length is used to calculate the forward speed; meanwhile, non-holonomic constraint (NHC) assumes that lateral and vertical speeds are zeroes based on the fact that a moving platform cannot skid or jump. After that, the forward, lateral, and vertical speeds are combined to pseudo-velocity, which works as the velocity update for the INS to limit the velocity errors. Additionally, Zero Velocity Update (ZUPT) applies zero-velocity as an update if the object is obviously at rest. Furthermore, Zero Angular Rate Update (ZARU) applies the constant heading as the heading update if the object is obviously at rest. In the proposed algorithm of MEMS inertial-sensors-based pedestrian navigation, the attitude information from the INS potentially improves the heading estimation when compared with PDR, because it also considers the effects of the roll and pitch. ZARU can enhance the accuracy of heading estimation of the proposed system when the object is static, and further improve the whole attitude estimation. The pseudo-velocity update and ZUPT also improve the accuracy of the attitude due to the coupling of velocity and attitude errors in the INS dynamic error model. However, the heading cannot be estimated very well in the proposed PDR/INS integrated algorithm because it is weakly coupled with the velocity update. Therefore, a WiFi solution is also used in the PDR/INS/WiFi integrated system to improve the position and attitude estimation when it is accurate. In this paper, trilateration is used for WiFi positioning because it can be implemented without pre-surveys. The approximate accuracy of WiFi positioning, which is derived from the position covariance matrix in the least squares (LSQ), is used as an indicator to determine whether the WiFi position is suitable for the PDR/INS/WiFi integration. The proposed algorithms are evaluated in typical indoor scenarios. The performances of PDR, INS, the PDR/INS integration, and the PDR/INS/WiFi integration are also compared.

## 2. Related Works

### *2.1. Recent MEMS-Based Navigation Technologies*

INS mechanization [17] is the typical technology used for MEMS-based navigation. In the past few years, INS mechanization has been applied to many navigation applications such as those in the studies of [4,25,26]. On the other hand, over the last decade, some algorithms based on PDR have been proposed for pedestrian navigation because it drifts slower when compared with INS as shown in [16,27,28]. Recently, PDR solutions have become practical in people's daily use [16] because many handheld devices are equipped with inertial sensors. The main advantage of INS and PDR solutions is that they do not require additional infrastructure. However, since the navigation is based on noisy inertial sensors, navigation solutions of both PDR and INS systems drift over time [29].

### *2.2. Integrated Technologies for WiFi and MEMS Sensors*

An indoor positioning system for pedestrians, combing WiFi fingerprinting with foot-mounted inertial and magnetometer sensors, is proposed in [30]. However, foot mounted systems are not as

convenient as handheld devices for pedestrians, and the requirement of the pre-survey makes WiFi fingerprinting impractical for a large area. Research work in [20] proposes an optimized adaptive version of the mixture particle filter for integrating WiFi fingerprinting with INS, and the system was tested in a four-wheel indoor vehicle. The system focuses on the applications of indoor vehicles, and does not adopt the handheld devices. The system also requires pre-survey for fingerprinting. An advanced integration of WiFi and INS, based on a particle filter, is proposed in [31]. Nevertheless, the particle filter is not suitable for handheld devices such as smartphones due to its heavy computational load. Furthermore, a PDR/WiFi/barometer integrated system by using the AKF (adaptive Kalman filter) is proposed in [32]. The system is also based on WiFi fingerprinting and foot-mounted sensors. A maximum likelihood-based fusion algorithm that integrates the PDR and WiFi fingerprinting is proposed in [29]. The algorithm was implemented in smartphones that made the system practical, other than the pre-survey for fingerprinting. Currently, few papers have discussed the PDR/INS/WiFi integrated technique for indoor pedestrian navigation applications; therefore, this paper proposed an approach to fill this gap. The goal of this paper is to provide a practical and reliable indoor pedestrian navigation solution, based on handheld devices, by using the PDR/INS/WiFi integration.

## 3. Methodology

### *3.1. INS Mechanization in the Navigation Frame*

The essential process in any inertial navigation algorithm is the INS mechanization. The INS mechanization equations integrate the specific forces and angular rates provided by accelerometers and gyroscopes to compute the PVA of an objective [33]. These equations can be presented as follows [34]:

$$\begin{bmatrix} \dot{r}^n \\ \dot{v}^n \\ \dot{C}_b^n \end{bmatrix} = \begin{bmatrix} D^{-1}v^n \\ C_b^n f^b - (2\Omega_{ie}^n + \Omega_{en}^n)v^n + g^n \\ C_b^n(\Omega_{ib}^b - \Omega_{in}^b) \end{bmatrix} \quad (1)$$

where $r^n = \begin{bmatrix} \varphi & \lambda & h \end{bmatrix}^T$ is the position vector (latitude, longitude, and height) in the navigation frame. $v^n = \begin{bmatrix} v_N & v_E & v_D \end{bmatrix}^T$ is the velocity vector. $C_b^n$ is the transformation matrix from the body frame to the navigation frame as a function of attitude components. $g^n$ is the gravity vector in the navigation frame, $\Omega_{ib}^b$ and $\Omega_{in}^b$ are the skew-symmetric matrices of angular velocity vectors $\omega_{ib}^b$ and $\omega_{in}^b$, respectively, and $D^{-1}$ is a 3 × 3 matrix whose non-zero elements are functions of φ and $h$ [33]. Refer to [33] for details of the solution and numerical implementation of the above differential equation. Both inertial sensor errors and integration errors will cause the INS solution to diverge quickly. Therefore, other sources are usually needed to aid the INS [33].

### *3.2. Extended Kalman Filter*

Due to the drift characteristics of MEMS sensors, the Extended Kalman Filter (EKF) is usually used to fuse other solutions such as the position and velocity from GNSS, motion constraints, and the position and velocity from PDR and WiFi. The EKF estimates the optimal state of a process by satisfying the criterion which minimizes the mean of the square errors (MMSE). When the EKF is used to fuse other sources with INS, the state vector is determined first as follows:

$$x = \left[\delta r_{1\times 3} \quad \delta v_{1\times 3} \quad \varepsilon_{1\times 3} \quad d_{1\times 3} \quad b_{1\times 3}\right]^T \tag{2}$$

where $\delta r$ are position errors; $\delta v$ are velocity errors; $\varepsilon$ are attitude errors; $d_{1\times 3}$ and $b_{1\times 3}$ are the bias vector for gyroscopes and accelerometers. The estimated biases will be fed back into the INS mechanization.

The dynamic error model is also an important part when using the EKF for improving the INS solution. In addition to nine navigation elements in Equation (1), states of the dynamic error model include the sensor errors (three accelerometer biases and three gyro drifts). These sensor errors are usually modeled as the first order Gauss-Markov process. Therefore, the dynamic error model used in the EKF for the navigation parameters (position error, velocity error, attitude error, gyro bias and accelerometer bias) can be determined through the linearization of the INS mechanization equations and by neglecting insignificant terms in the resultant linear model. A simplified form is then obtained as [33]:

$$\dot{x}(t) = \begin{pmatrix} \delta\dot{r}^n \\ \delta\dot{v}^n \\ \dot{\varepsilon}^n \\ \dot{d} \\ \dot{b} \end{pmatrix} = \begin{pmatrix} D^{-1}\delta v^n - D^{-1}D_r\delta v^n \\ -f^n \times \varepsilon^n - (2\Omega_{ie}^n + \Omega_{en}^n)\delta v^n + v^n \times (2\delta\omega_{ie}^n + \delta\omega_{en}^n) + \delta g^n + C_b^n b \\ -\Omega_{in}^n \varepsilon^n + \delta\omega_{in}^n + C_b^n d \\ -\alpha d + \omega_d \\ -\beta b + \omega_b \end{pmatrix} \tag{3}$$

where $\delta r^n$ is the position error state vector in the $n$-frame, $\delta v^n$ is the velocity error state vector in the $n$-frame, $\varepsilon^n$ is the attitude error state vector in the $n$-frame, $\delta g^n$ is the error in the computed gravity vector in the $n$-frame, $b$ and $d$ are accelerometer bias and gyro drift vectors, $\alpha$, $\beta$, $\omega_d$, $\omega_b$ are the parameters for the first order Gauss-Markov process and $D_r$ is a 3 × 3 matrix whose non-zero elements are functions of the user's latitude, $\varphi$, and ellipsoidal height, $h$.

By the discretization of dynamic error model and the observation model, the discrete-time system model and observation model can be expressed as:

$$\begin{cases} \delta x_{k+1} = \Phi_{k,k+1}\delta x_k + w_{k+1} \\ \delta z_{k+1} = H_{k+1}\delta x_{k+1} + v_{k+1} \end{cases} \tag{4}$$

where $\delta x_k$ is the state vector at epoch $k$, $\Phi_{k,k+1}$ is the state transition matrix from epoch $k$ to epoch $k + 1$, and $w_{k+1}$ is the process noise. $\delta z_{k+1}$ is the observation misclosure vector at epoch $k + 1$, $H_{k+1}$ is the design matrix at epoch $k + 1$, and $v_k$ is the observation noise. For more knowledge about EKF for integrated navigation system, refer to [35].

### *3.3. PDR Algorithm*

Dead Reckoning (DR) determines current vehicle position from the knowledge of the previous position and the measurements of motion direction and traveled distance [33]. DR algorithm is usually used for vehicle navigation. PDR applies the traditional DR algorithm for personal navigation applications. PDR algorithm is often applied to obtain the personal position information at each step of the personal movement. In this case, techniques are studied for step detection, and the traveled distance equals to step length. The PDR mechanization equation is given by:

$$E_k = E_{k-1} + \hat{s}_{(k-1,k)} \cdot \sin(\hat{H}_{k-1})$$
$$N_k = N_{k-1} + \hat{s}_{(k-1,k)} \cdot \cos(\hat{H}_{k-1}) \quad (5)$$

where ($E_{k-1}$, $N_{k-1}$) and ($E_k$, $N_k$) are previous and current positions in the local level frame, $\hat{s}_{(k-1,k)}$ is the estimated $k^{th}$ stride length, and $\hat{H}_{k-1}$ is the estimated $(k-1)^{th}$ heading. Figure 1 clearly illustrates the PDR concept. Besides the PDR mechanization, a typical PDR algorithm includes three other parts: step detection, step length estimation and heading estimation.

Usually, the cycle pattern of the acceleration norm is used to detect steps. There are several different approaches to detect steps, which include peak detection, zero crossing, auto/cross correlation and spectral analysis. In this paper, the peak detection is used for step detection due to its small computation and high successful rate. An example of peak detection is discussed in Section 4.1.

In the PDR algorithm, it is necessary to estimate the step length at each step as the forward moving distance of a person. Different practical models have been proposed for estimating the step length. In this paper, the empirical model proposed in [36] is used, which assumes the step length is proportional to the vertical movement of the human hip. The vertical movement is obtained from the largest acceleration difference in vertical direction at each step. Therefore, the equation used to estimate the step length is written as follows:

$$SL = \sqrt[4]{a_{z\max} - a_{z\min}} \cdot K \quad (6)$$

where $a_{z\max}$ is the maximum value of vertical acceleration $a_z$, $a_{z\min}$ is the minimum value of $a_z$, and $K$ is a calibrated constant parameter.

In the PDR algorithm, gyroscopes are usually used to estimate the moving direction of a person. As we discussed before, PDR assumes that the handheld device is level (roll and pitch are 0 degrees). Sometimes, roll and pitch effects cannot be ignored. In this case, PDR-based heading, calculated by the direct integration of the data from the vertical gyroscope, is inaccurate. Therefore, PDR-derived heading is not used in our proposed PDR/INS integrated algorithm.

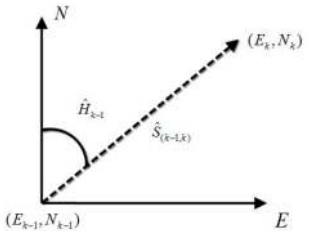



**Figure 1.** Illustration for the PDR concept.

### *3.4. Proposed Scheme*

An overview of the proposed PDR/INS/WiFi integration system on portable devices such as smartphones is illustrated in Figure 2. The part outside the dashed boxes is the proposed MEMS inertial solution based on the integration of PDR/INS and motion constraints (NHC, ZUPT, and ZARU). The part inside the dashed boxes is the WiFi positioning solution for the whole PDR/INS/WiFi integrated solution. In the traditional pedestrian navigation, raw data from MEMS inertial sensors are put into INS mechanization or PDR mechanization to estimate the user's position.

In the proposed algorithm, gyroscopes and accelerometers are used to set up the INS mechanization, detect the motion status, and set up the PDR mechanization. In this algorithm, INS, PDR, and motion constraints work together to generate a more accurate and robust pedestrian navigation solution. In Figure 2, if the step detection is successful, and the ZUPT detection is unsuccessful; the PDR step length estimation is executed to provide step length information. Next, the forward speed is derived from the estimated step length. Furthermore, NHC assumes that lateral and vertical speeds are zeroes based on the fact that a moving platform cannot skid or jump. The PDR-based forward speed and the NHC-based lateral and vertical speeds are combined to 3-axis pseudo-velocity which works as a velocity update for the INS to limit the velocity error. On the other hand, the step detection is unsuccessful, and the ZUPT detection is successful. If this is the case, ZUPT applies zero velocity as the velocity update for the INS to reduce the velocity error; ZARU is also used for the heading update for the INS to improve the attitude solution. Based on the proposed MEMS inertial-sensors-based pedestrian navigation solution, WiFi is used to further limit the drift of MEMS sensor. In the WiFi part of Figure 2, trilateration is used for estimating the WiFi positioning solution and its accuracy. The WiFi positions with good accuracy are selected as updates for the proposed sensor solution. The final proposed pedestrian navigation system is based on the integration of PDR, INS, and WiFi.

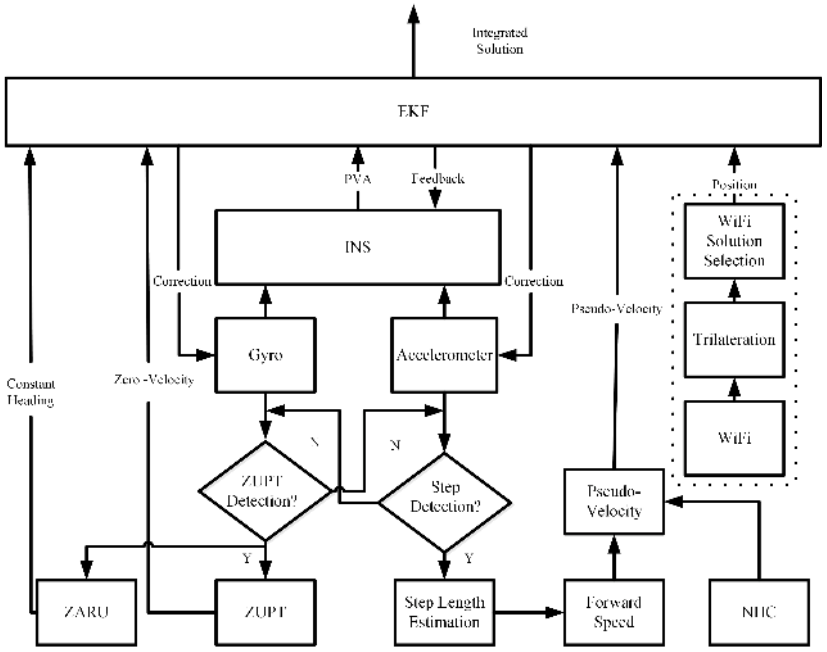



**Figure 2.** Block diagram of the proposed PDR/INS/WiFi integration system.

### *3.5. Proposed PDR/INS Integrated Pedestrian Navigation Solution*

As previously discussed, INS and PDR are two traditional approaches for MEMS inertial-sensors-based pedestrian navigation. INS-based navigation provides complete 3D PVA. However, its error increases rapidly with time due to the drift characteristics of MEMS inertial sensors and the integrations used in the INS mechanization. On the other hand, PDR provides more accurate navigation solution because it uses step detection and step length estimation which avoid using integrations for calculating distance. However, PDR derives the heading information from the integration of the vertical gyroscope, which is inaccurate if the roll and pitch effects cannot be neglected. In this paper, we propose a PDR/INS

integrated navigation solution to combine the advantages of the two schemes, and avoid the drawbacks. In the proposed PDR/INS integrated MEMS navigation solution, INS mechanization is used to process the MEMS inertial sensor data first. Next, the dynamic error model is used as the system model for the EKF estimation. At last, the PDR-based forward speed and the NHC-based lateral and vertical speeds are combined to the pseudo-velocity which works as the velocity update for the INS to limit the velocity error.

In the proposed algorithm, the gyroscopes and accelerometers measurements are used to detect the status of the pedestrian: moving or static. The status of the pedestrian is determined as "moving" if the following two conditions are satisfied: (1) the standard deviation of the angular rate norms during a certain time is larger than the threshold; and (2) steps are detected by using the approach in the section "PDR Algorithm". On the other hand, the status of the pedestrian is determined as "static" if the following two conditions are satisfied: (1) the standard deviation of the angular rate norms during a certain time is less than the threshold; and (2) no steps are detected.

For the "moving" case, the step length estimation has been discussed in the section "PDR Algorithm". To use the step length to provide information about the forward speed, we assume that a pedestrian's moving speed is constant for a short time. This assumption is correct for most moving cases of pedestrians. The forward speed can be derived from the step length as follows:

$$v_{\text{forward}} = SL / T_{\text{step}} \tag{7}$$

where $SL$ is the step length, and $T_{\text{step}}$ is the step time. NHC assumes that lateral and vertical speeds are zeroes based on the fact that a moving platform cannot skid or jump. For the pedestrian navigation, NHC can be adopted for most cases. Therefore, the pseudo-velocity-vector in the body frame can be expressed as:

$$v^b_{\text{pseudo}} = \begin{bmatrix} SL / T_{\text{step}} & 0 & 0 \end{bmatrix}^T \tag{8}$$

The pseudo-velocity-vector used for INS velocity update is the core of the PDR/INS integration. When using EKF for pseudo-velocity update, the misclosure of the velocity in the body frame is given by:

$$\delta z = v^b_{\text{INS}} - v^b_{\text{pseudo}} \tag{9}$$

where $v^b_{\text{INS}} = \left(C^n_b\right)^T \cdot v^n_{\text{INS}}$ represents the INS-based velocity in the body frame; $C^n_b$ represents the transformation matrix from the body frame to navigation frame; and $v^n_{\text{INS}}$ represents INS-based velocity in the navigation frame. Finally, the observation model for the pseudo-velocity-vector update is expressed in:

$$\delta v^b = H_{v^b} \delta x + v_{v^b} \tag{10}$$

where $v_{v^b}$ represents the measurement white noise; $H_{v^b}$ represents the corresponding design matrix which can be expressed as:

$$H_{v^b} = \begin{bmatrix} 0_{3\times3} & \left(C^n_b\right)^T & -\left(C^n_b\right)^T V^n & 0_{3\times6} \end{bmatrix} \tag{11}$$

where $V^n$ is the skew-symmetric matrix of $v^n$.

If "static" is detected, ZUPT and ZARU are used as the updates for the INS to limit the navigation error. The ZUPT-based zero velocity vector in the body frame is given by:

$$v_{\text{ZUPT}}^{b} = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^{T} \tag{12}$$

Similar to the pseudo-velocity vector, the ZUPT-based zero velocity vector is used as the velocity update for the INS. If the pedestrian is detected as "static", the pedestrian heading is unchanging based on ZARU. Therefore, misclosure of the heading for the INS heading update is given by:

$$\delta z = \psi_{\text{INS}} - \psi_{\text{pre-stored}} \tag{13}$$

where $\psi_{\text{INS}}$ is the INS-based heading; and $\psi_{\text{pre-stored}}$ is the pre-stored heading of the last epoch before the "static" is detected. Finally, the observation model for the heading update is expressed in:

$$H_{\psi} = \begin{bmatrix} 0_{1\times 6} & \partial\psi / \partial\varepsilon_{N} & \partial\psi / \partial\varepsilon_{E} & \partial\psi / \partial\varepsilon_{D} & 0_{1\times 6} \end{bmatrix} \tag{14}$$

For details of $\partial\psi / \partial\varepsilon_{N}$, $\partial\psi / \partial\varepsilon_{E}$, and $\partial\psi / \partial\varepsilon_{D}$, please refer to [37].

### *3.6. Proposed PDR/INS/WiFi Integrated Indoor Pedestrian Navigation Solution*

To further improve the navigation performance, WiFi is also used in the integrated system if it is available. Both trilateration-based and fingerprinting-based WiFi positioning solutions can be used as updates in the loosely-coupled integration. In this research, trilateration is selected for WiFi positioning for the following two reasons: (1) it requires less efforts to build the database; and (2) the memory cost for trilateration-based database is much less than fingerprinting-based database. However, a trilateration-based WiFi solution can be noisy due to the complex characteristics of an indoor environment. Therefore, in the integrated system, it is significant to use an approach to select accurate WiFi positions. It is fortunate that variances of WiFi positions are estimated in the state covariance matrix of the LSQ. Although these variances are not perfectly estimated, they still can be used as a rough indicator for selecting the WiFi positions for the integration. In this research, WiFi positions with variances less than 20 $m^2$ are chosen as the updates for the MEMS sensors. The misclosure of the WiFi-based position measurements is given by:

$$\delta z_{\text{WiFi}} = \left( \begin{bmatrix} \varphi \\ \lambda \\ h \end{bmatrix}_{\text{MEMS}} - \begin{bmatrix} \varphi \\ \lambda \\ h \end{bmatrix}_{\text{WiFi}} \right) \begin{bmatrix} M + h \\ (N + h)\cos\varphi \\ 1 \end{bmatrix}^{T} \tag{15}$$

where $\varphi_{\text{MEMS}}$, $\lambda_{\text{MEMS}}$, and $h_{\text{MEMS}}$ are MEMS navigation solution estimated latitude, longitude, and altitude; $\varphi_{\text{WiFi}}$, $\lambda_{\text{WiFi}}$, and $h_{\text{WiFi}}$ are WiFi-based latitude, longitude, and altitude. $M$ is the meridian radius of the earth's curvature; and $N$ is the prime vertical radius of the earth's curvature. The observation equation for the WiFi position measurements is formulated as:

$$\delta z_{\text{WiFi}} = H_{\text{WiFi}} \delta x + v_{\text{WiFi}} \tag{16}$$

where $v_{\text{WiFi}}$ is the measurement white noise of the WiFi positions; $H_{\text{WiFi}}$ is the corresponding design matrix, which can be expressed as:

$$H_{\text{WiFi}} = \begin{bmatrix} (M + h) & 0 & 0 & \\ 0 & (N + h)\cos\varphi & 0 & 0_{3\times 12} \\ 0 & 0 & 1 & \end{bmatrix} \tag{17}$$

The covariance matrix, $R_{\mathrm{WiFi}}$, for the WiFi-based position measurements is given by:

$$R_{\mathrm{WiFi}} = diag\left(\begin{bmatrix}\sigma_{lat}^2 & \sigma_{lon}^2 & \sigma_{alt}^2\end{bmatrix}\right) \tag{18}$$

where $\sigma_{lat}^2$, $\sigma_{lon}^2$ and $\sigma_{alt}^2$ represent the variances of $\begin{bmatrix}\varphi & \lambda & h\end{bmatrix}_{\mathrm{WiFi}}^T$ in meters. *diag* returns a square diagonal matrix with the elements of vector $\begin{bmatrix}\sigma_{lat}^2 & \sigma_{lon}^2 & \sigma_{alt}^2\end{bmatrix}$ on the main diagonal.

## 4. Results

To evaluate the performance of the proposed pedestrian navigation algorithms, several experiments were performed by three pedestrians. Three different smartphones that contain an accelerometer triad, a gyroscope triad and a WiFi transreceiver were used to collect the experimental data. Different pedestrians collected data with various smartphones on three experimental trajectories in building E (about 120 m × 40 m), as shown in Figure 3. Two tasks were carried out in the field tests. The first part validates the performance of the proposed PDR/INS integrated MEMS pedestrian navigation algorithm. This section also compares the proposed MEMS solution with traditional PDR and INS algorithms. The second part shows the performance of the proposed PDR/INS/WiFi integration algorithm.

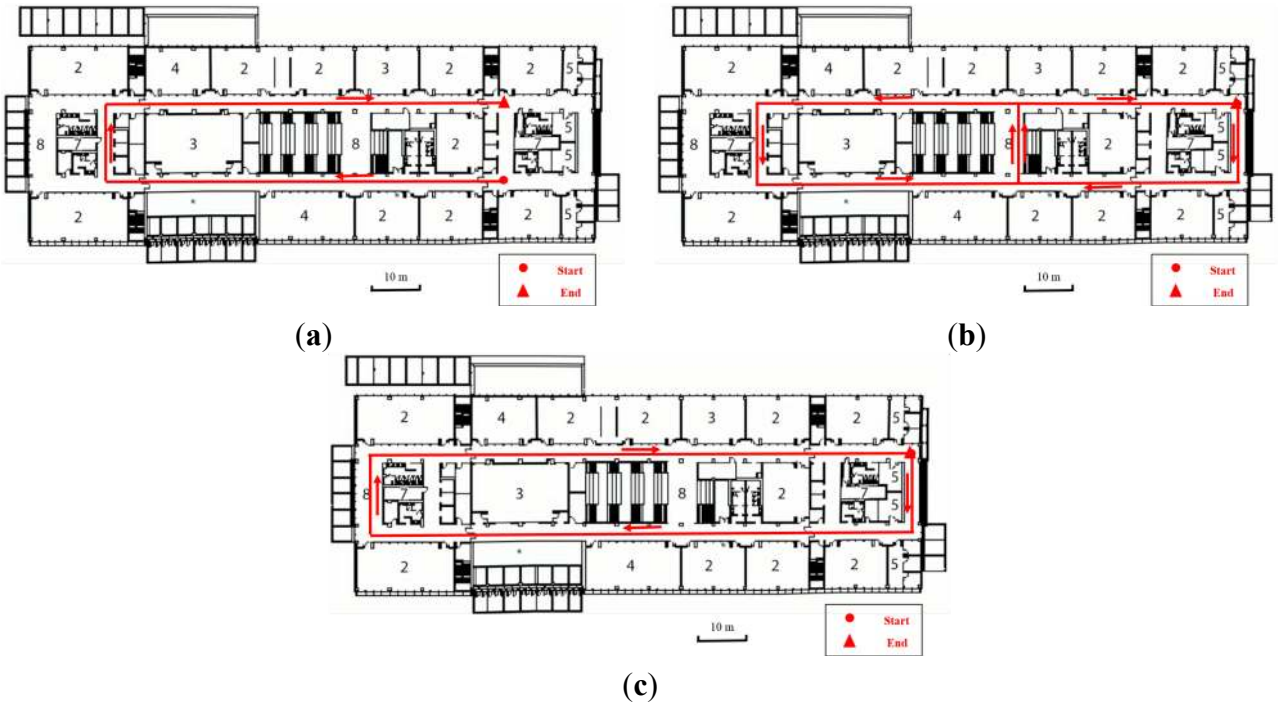



**Figure 3.** Three experimental trajectories in building E (about 120 m × 40 m). (**a**) Trajectory I; (**b**) Trajectory II; (**c**) Trajectory III.

### *4.1. PDR/INS Integrated MEMS Pedestrian Navigation Algorithm*

This section illustrates the performance of the proposed PDR/INS integrated MEMS pedestrian navigation algorithm by using Trajectory I. Two other trajectories were also used to test the performance of proposed MEMS navigation solution, which are summarized in the next section. The proposed PDR/INS integrated MEMS solution, PDR solution, and the reference trajectory in experimental Trajectory I are

shown in Figure 4. In Figure 4, the maximum navigation errors of the proposed method and traditional PDR are about 12 m and 34 m, respectively. This outcome shows that the proposed method is more accurate than the PDR. The average heading drift of the proposed method is also smaller than the PDR.

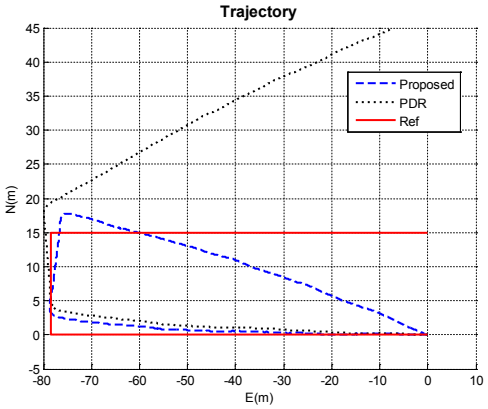



**Figure 4.** Trajectories of PDR, PDR/INS integrated MEMS solution, and the reference.

The velocity solution of the proposed algorithm is shown in Figure 5a. Figure 5a clearly shows the user's moving status: (a) keeps static (ZUPT), (b) walks west, (c) keeps static (ZUPT), (d) walks west, (e) walks north, (f) walks east, (g) keeps static (ZUPT), and (h) walks east. The moving trend fits the trajectory in Figure 4. The moving speed is in the typical range of a normal person. The pseudo-velocity update and ZUPT play important roles on accurately estimating a user's velocity. Without the pseudo-velocity update and ZUPT, the estimated velocity and position solution drifts quickly. The attitude solution of the proposed method is shown in Figure 5b. Roll and pitch angles are between −10 and 10 degrees in this trajectory. The estimated azimuth trend is (a) about −90 degrees, (b) about 0 degrees, and (c) about 100 degrees. The true azimuth trend is (a) −90 degrees, (b) 0 degrees, and (c) 90 degrees. The estimated azimuth from the proposed method is close to the true azimuth.

The results of the step detection, step length estimation, and pseudo-velocity are shown in Figure 6. In Figure 6a,b, acceleration norm and detected peaks are described by "−" and "+". Figure 6b zooms in on some parts of the trajectory which show that the step detection algorithm can successfully detect the peaks and steps. The step length estimation result is shown in Figure 6c. The user's step length is around 0.64 m as shown in this figure. The step-length-derived pseudo-velocity is shown in Figure 6d. The pseudo-velocity is calculated from the step length. The average velocity of this user is about 1.2 m/s.

To illustrate the performance of the proposed PDR/INS integrated MEMS navigation solution, results of the stand-alone PDR and INS are also shown in this paper. Pure PDR results are shown in Figures 4 and 7. As we discussed before, the maximum navigation error of PDR is about 32 m in Figure 4, which is worse than the proposed method. In Figure 7, the azimuth error, which is the main error for the whole trajectory, reaches 20 degrees by the end of the trajectory.

Results of the pure INS algorithm are shown in Figure 8. Figure 8a shows the horizontal trajectory by using pure INS algorithm. The maximum drift reaches up to 73 km at the end of the end of the trajectory. That's because MEMS sensors in smartphones are low cost, and their errors accumulate with time quickly. The pure INS algorithm cannot provide an accurate navigation solution for pedestrians by using handheld devices. The INS velocity solution is shown in Figure 8b. In this figure, it is clear that the velocity solution are not located in the typical range of pedestrian walking velocity.

The velocity drifts to several hundred meters per second by the end of the trajectory. Figure 8c shows the attitude solution from the INS algorithm. The estimated roll and pitch can reach about ± 50 degrees sometimes, which highly drift from the true roll and pitch range (*i.e.*, between −10 and 10 degrees). The estimated heading also drifts from the true value when compared with the reference trajectory in Figure 4. Figure 8 shows that estimated position, velocity, and attitude all highly drift from true values by using pure INS algorithm.

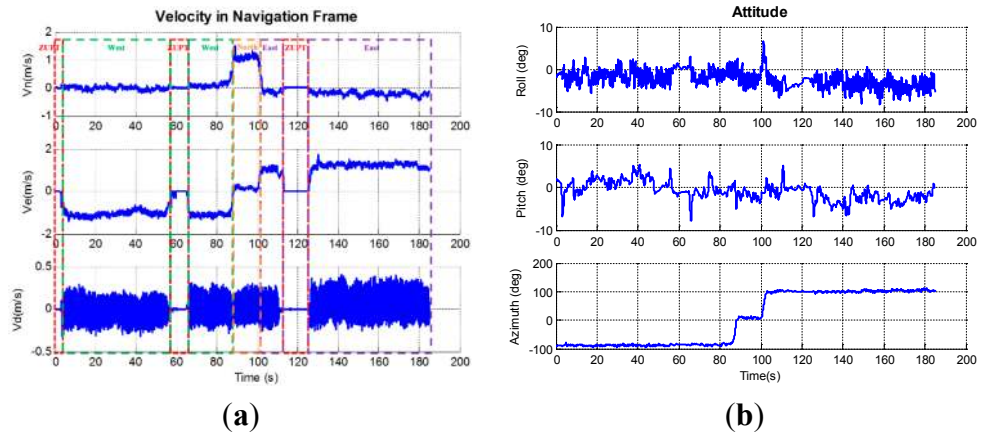



**Figure 5.** The solution of the proposed PDR/INS integrated MEMS solution. (**a**) Velocity; (**b**) Attitude.

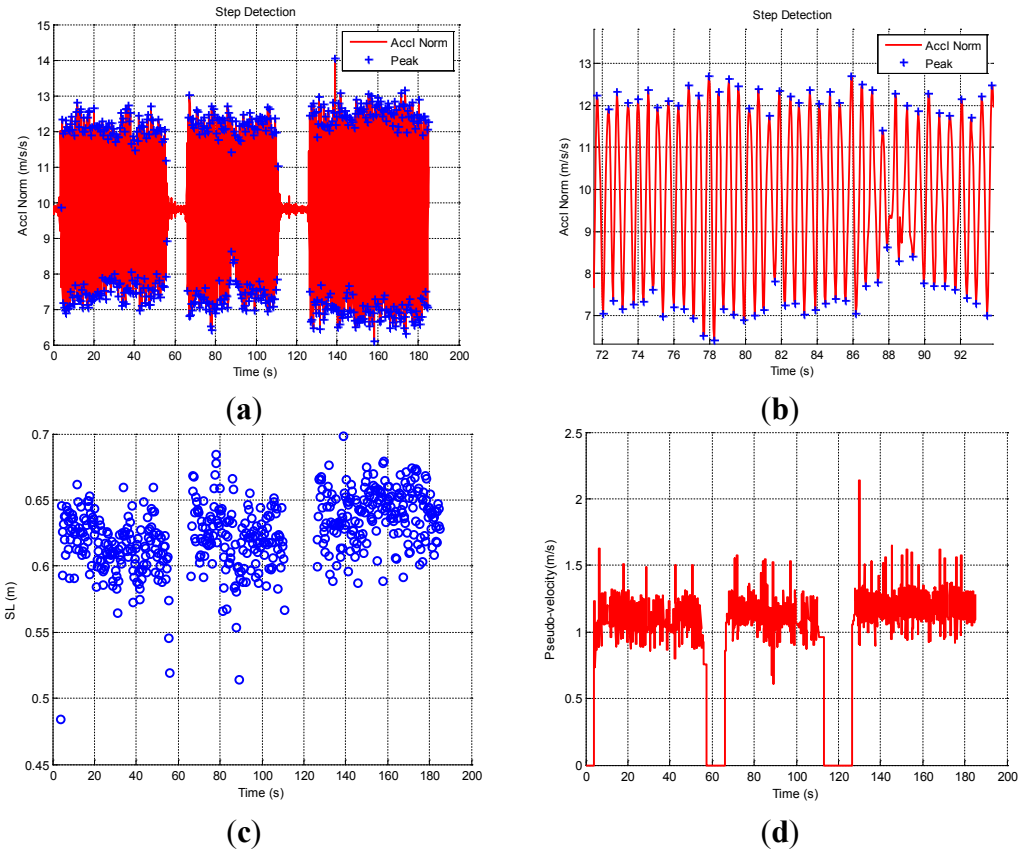



**Figure 6.** Results of the step detection and step length estimation. (**a**) Step detection for the whole trajectory; (**b**) Zoom-in of some parts of step detection for the trajectory; (**c**) Result of step length estimation; (**d**) Result of pseudo-velocity from step length.

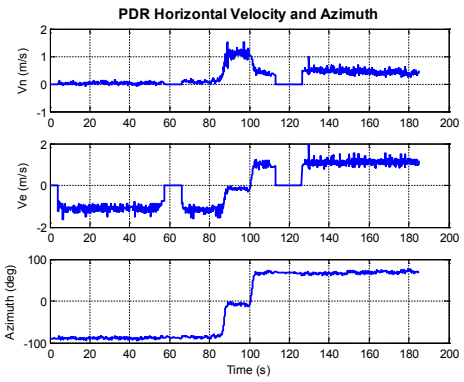
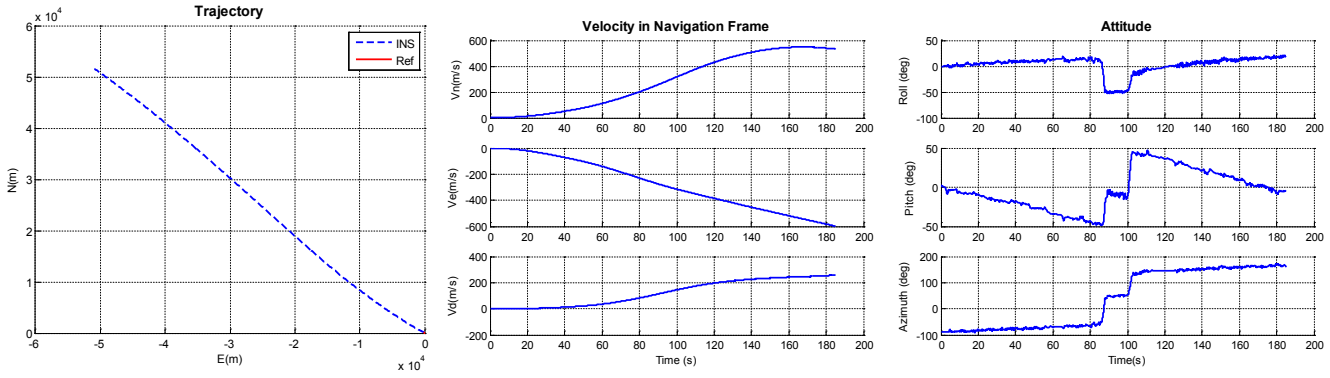



**Figure 7.** Results of PDR horizontal velocity and azimuth.




(**a**) (**b**) (**c**)

**Figure 8.** INS navigation solution. (**a**) Trajectory; (**b**) Velocity; (**c**) Attitude.

### *4.2. Proposed PDR/INS/WiFi Integrated Indoor Pedestrian Navigation Solution*

The results (position and variance) of trilateration-based WiFi positioning of Trajectory I are shown in Figure 9. The trilateration algorithm relies on the propagation model, which is affected by several factors such as multipath and fading characteristics. Therefore, it is usually not as accurate as the fingerprinting algorithm for WiFi positioning. However, trilateration does not consume much time and labour to build and maintain a database like fingerprinting. The results contain very large errors reaching 15 m at many points. To improve the performance, some WiFi positions with large variances are left out of the integration. In this research, WiFi positions with variances less than 20 $m^2$ are selected for integration as shown in Figure 9a. The threshold, 20 $m^2$, is obtained by field tests. In Figure 9a, "Sel WiFi" represents the selected WiFi positions with variances less than 20 $m^2$.

Navigation results of PDR, the proposed MEMS solution, and WiFi/MEMS integration (Trajectory I: Pedestrian 1, Smartphone A) are shown in Figure 10. The trajectory of the proposed integration of MEMS/WiFi (also called PDR/INS/WiFi) in Trajectory I is shown as the "blue dash line" in Figure 10a. This trajectory is taken by "Pedestrian 1" using "Smartphone A". The proposed MEMS solution, PDR, and the reference are shown as a "green dash dot line", "black dot line", and "red solid line", respectively. The proposed MEMS solution drifts less than the PDR solution. Selected WiFi positions are also shown as "circles" in Figure 10a. With the help of the selected WiFi positions, the integrated MEMS/WiFi solution drifts less than the proposed MEMS solution as shown in Figure 10a. The

integrated MEMS/WiFi solution is also better than the WiFi solution when one compares Figure 9a with Figure 10a. The cumulative error percentages of PDR, the proposed MEMS solution, and WiFi/MEMS integration in Trajectory I are shown in Figure 10b. In this figure, The "LC" represents the WiFi/MEMS integration. The positioning performance of different algorithms in Trajectory I are depicted in Table 1. As shown in Figure 10b and Table 1, WiFi/MEMS integration has the best navigation performance, and the PDR has the worst navigation performance.

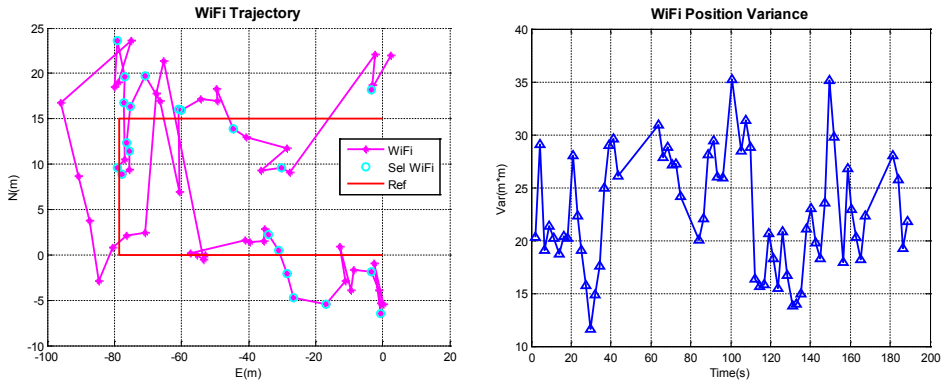



**Figure 9.** Trilateration-based WiFi positioning solution. (**a**) WiFi trajectory; (**b**) Variances of WiFi positions.

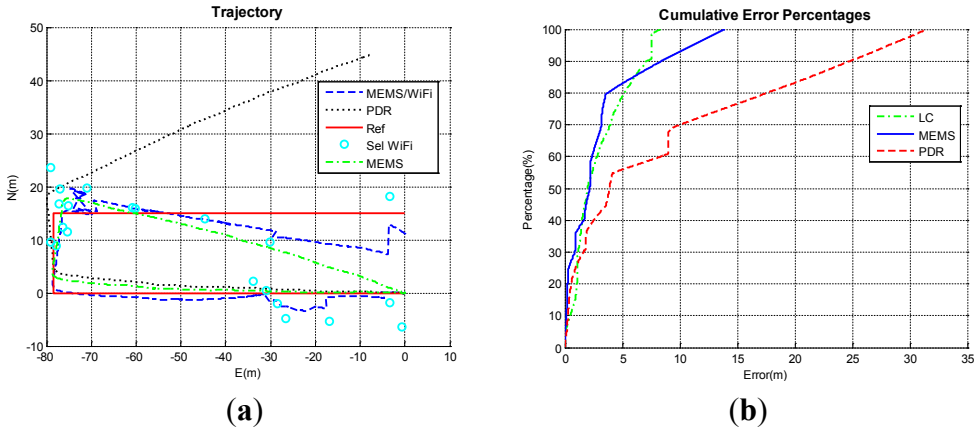



**Figure 10.** Navigation results of PDR, the proposed MEMS solution, and WiFi/MEMS integration (Trajectory I: Pedestrian 1, Smartphone A). (**a**) Trajectories; (**b**) Cumulative error percentages.

**Table 1.** Positioning performance of different algorithms in Trajectory I.

| Algorithm | Error (m) | | | |
|---|---|---|---|---|
| | **Maximum** | **Minimum** | **Mean** | **RMS** |
| **PDR** | 31.48 | 0.13 | 8.44 | 12.57 |
| **MEMS (PDR/INS)** | 13.93 | 0.25 | 2.91 | 4.40 |
| **MEMS/WiFi Integration** | 8.22 | 0.21 | 2.91 | 3.73 |

Navigation results of PDR, the proposed MEMS solution, and WiFi/MEMS integration (Trajectory II: Pedestrian 2, Smartphone B) are shown in Figure 11. The trajectory of the proposed MEMS/WiFi

integration is shown as the "blue dash line" in Figure 11a as taken by "Pedestrian 2" using "Smartphone B". Similar to Trajectory I, the proposed MEMS solution drifts less than the PDR solution, and the integrated MEMS/WiFi solution drifts less than the MEMS solution as shown in Figure 11a. The results of trilateration-based WiFi positioning in Trajectory II are shown in Figure 12. The integrated MEMS/WiFi solution also performs better than the WiFi solution when comparing Figure 11a with Figure 12. The cumulative error percentages of PDR, the proposed MEMS solution, and WiFi/MEMS integration in the second trajectory are shown in Figure 11b, and the positioning performance of different algorithms for Trajectory II are depicted in Table 2. In Trajectory II, the WiFi/MEMS integrated system improves about 3.5 m (RMS) when compared with the MEMS solution, which is better than in Trajectory I (0.7 m). This is probably because the MEMS solution drifts more in Trajectory II due to a longer walking period.

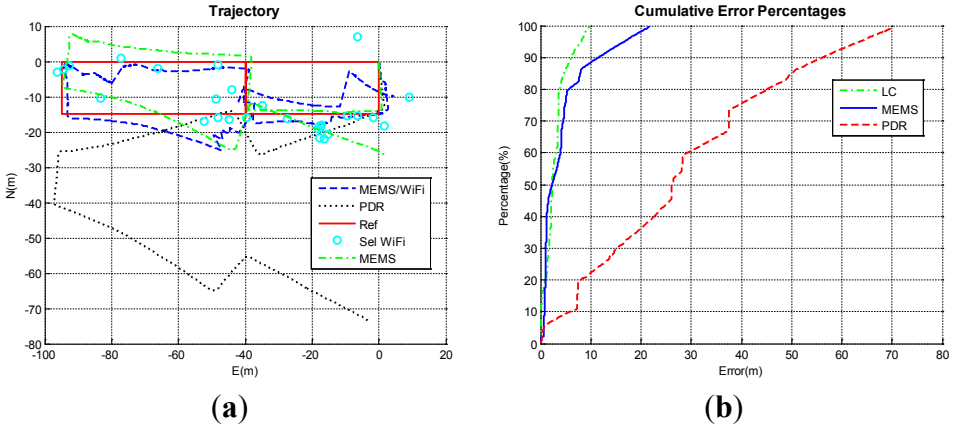



**Figure 11.** Navigation results of PDR, the proposed MEMS solution, and WiFi/MEMS integration (Trajectory II: Pedestrian 2, Smartphone B). (**a**) Trajectories; (**b**) Cumulative error percentages.

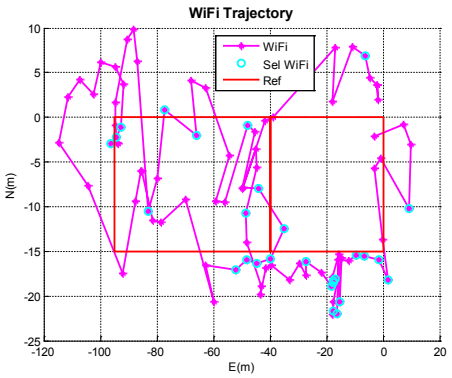



**Figure 12.** WiFi trajectory (Trajectory II).

**Table 2.** Positioning performance of different algorithms in Trajectory II.

| Algorithm | Error (m) | | | |
|---|---|---|---|---|
| | **Maximum** | **Minimum** | **Mean** | **RMS** |
| **PDR** | 75.03 | 0.14 | 27.92 | 34.00 |
| **MEMS (PDR/INS)** | 27.59 | 0.52 | 5.72 | 8.28 |
| **MEMS/WiFi Integration** | 12.21 | 0.48 | 4.17 | 4.87 |

Navigation results of PDR, the proposed MEMS solution, and WiFi/MEMS integration (Trajectory III: Pedestrian 3, Smartphone C) are shown in Figure 13. The third trajectory of the proposed MEMS/WiFi integration is shown as the "blue dash line" in Figure 13a as taken by "Pedestrian 3" using "Smartphone C". Similar to Trajectory I and Trajectory II, the proposed MEMS solution drifts less than the PDR solution, and the integrated MEMS/WiFi solution drifts even less than the MEMS solution as shown in Figure 13a. The results of trilateration-based WiFi positioning in Trajectory III are shown in Figure 14. The integrated MEMS/WiFi solution is also better than the WiFi-only solution when comparing Figure 13a with Figure 14. The cumulative error percentages of PDR, the proposed MEMS solution, and the WiFi/MEMS integration of the third trajectory are shown in Figure 13b, and the positioning performance of different algorithms for Trajectory III are depicted in Table 3.

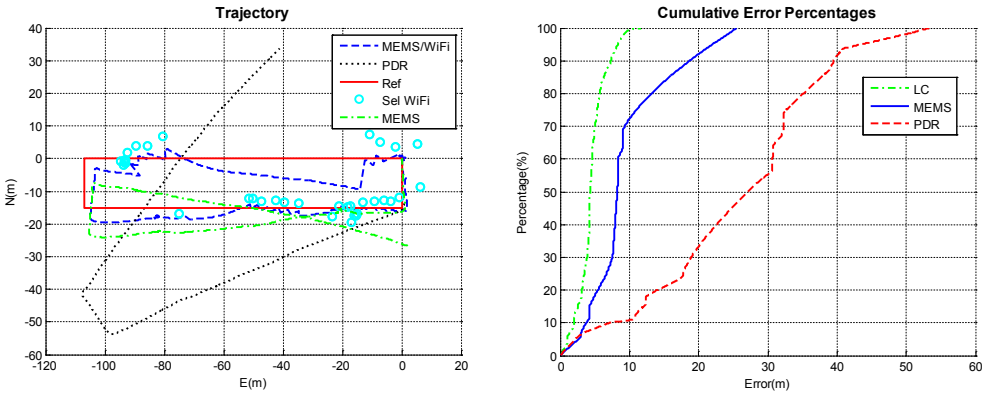



**Figure 13.** Navigation results of PDR, the proposed MEMS solution, and WiFi/MEMS integration (Trajectory III: Pedestrian 3, Smartphone C). (**a**) Trajectories; (**b**) Cumulative error percentages.

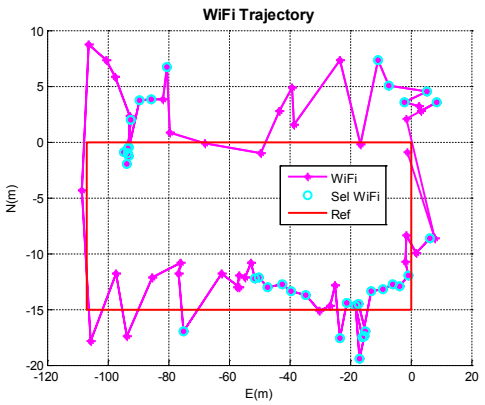



**Figure 14.** WiFi trajectory (Trajectory III).

**Table 3.** Positioning performance of different algorithms in Trajectory III.

| Algorithm | Error (m) | | | |
|---|---|---|---|---|
| | **Maximum** | **Minimum** | **Mean** | **RMS** |
| **PDR** | 53.49 | 0.05 | 25.35 | 28.08 |
| **MEMS (PDR/INS)** | 25.51 | 0.01 | 9.55 | 11.02 |
| **MEMS/WiFi Integration** | 11.66 | 0.01 | 4.49 | 4.90 |

The positioning performance in three trajectories is summarized in Table 4. In Table 4, the average position errors of the proposed PDR/INS/WiFi integration algorithm for all the three trajectories are less than 4.5 meters. Furthermore, the positioning error is decreased by 60%–75% by using the proposed PDR/INS integrated MEMS solution when compared with PDR. The positioning error is further decreased by 15%–55% if the proposed PDR/INS/WiFi integrated solution is implemented.

**Table 4.** Summary of positioning performance in three trajectories.

| Algorithm | Mean Error (m) | | | RMS Error (m) | | |
|---|---|---|---|---|---|---|
| | Trajectory I | Trajectory II | Trajectory III | Trajectory I | Trajectory II | Trajectory III |
| **PDR** | 8.44 | 27.92 | 25.35 | 12.57 | 34.00 | 28.08 |
| **MEMS (PDR/INS)** | 2.91 | 5.72 | 9.55 | 4.40 | 8.28 | 11.02 |
| **MEMS/WiFi Integration** | 2.91 | 4.17 | 4.49 | 3.73 | 4.87 | 4.90 |

## 5. Conclusions

A new low-cost MEMS navigation solution for pedestrians is presented in this paper. The proposed navigation solution utilizes traditional INS and PDR mechanizations to introduce a PDR/INS integrated navigation solution. In this algorithm, PDR-based forward speed and NHC-based lateral and vertical speeds are combined to the pseudo-velocity which worked as the velocity update for the INS. ZUPT and ZARU were also used as the updates in the EKF if they are available. These updates effectively reduced the accumulated position, velocity, and attitude errors when there are no other aiding sources. Furthermore, selected trilateration-based WiFi positions are used as updates for the PDR/INS integrated system to limit the drift of the proposed MEMS inertial-sensors-based pedestrian navigation solution, and finally to build the PDR/INS/WiFi integrated scheme for indoor pedestrian navigation.

The proposed system was tested in indoor pedestrian experiments to demonstrate its performance, accuracy, and ability in providing an effective solution. By testing the proposed system with different pedestrians and various smartphones, the performance of the proposed system does not rely on pedestrians or smartphones. The experiments demonstrate that positioning error of the proposed PDR/INS integrated MEMS solution decreases by 60%–75% when compared with PDR. The positioning error is further decreased by 15%–55% if using the proposed PDR/INS/WiFi integrated solution. The experimental results also show that the average accuracy of the proposed PDR/INS/WiFi integration algorithm achieves 4.5 m indoors.

## Acknowledgments

This work was supported in part by the research funding from Naser El-Sheimy. The first author (Yuan Zhuang) would also thank the Alberta Innovates Technology Futures (AITF) for the Graduate Student Scholarship.

## Author Contributions

Yuan Zhuang and Naser El-Sheimy conceived and designed the research; Yuan Zhuang, Haiyu Lan, and You Li performed the research; Yuan Zhuang, Haiyu Lan, and You Li analyzed the data; Naser El-Sheimy contributed materials; Yuan Zhuang and You Li wrote the paper.

## Conflicts of Interest

The authors declare no conflict of interest.

## References


1. Kaplan, E.D.; Hegarty, C.J. *Understanding GPS: Principles and Applications*; Artech House Publishers: Norwood, MA, USA, 2006.
2. He, Z.; Renaudin, V.; Petovello, M.G.; Lachapelle, G. Use of high sensitivity GNSS receiver doppler measurements for indoor pedestrian dead reckoning. *Sensors* **2013**, *13*, 4303–4326.
3. Nassar, S.; El-Sheimy, N. A combined algorithm of improving INS error modeling and sensor measurements for accurate INS/GPS navigation. *GPS Solut.* **2006**, *10*, 29–39.
4. Sadi, F.; Klukas, R. New jump trajectory determination method using low-cost mems sensor fusion and augmented observations for GPS/INS integration. *GPS Solut.* **2013**, *17*, 139–152.
5. Niu, X.; Li, Y.; Zhang, H.; Cheng, Y.; Shi, C. Observability analysis of non-holonomic constraints for land-vehicle navigation systems. *J. Glob. Position. Syst.* **2012**, *11*, 80–88.
6. Li, Y.; Georgy, J.; Niu, X.; Li, Q.; El-Sheimy, N. Autonomous calibration of MEMS gyros in consumer portable devices. *IEEE Sens. J.* **2015**, *15*, 4062–4072.
7. Zhuang, Y.; Syed, Z.; Georgy, J.; El-Sheimy, N. Autonomous smartphone-based WiFi positioning system by using access points localization and crowdsourcing. *Pervasive Mob. Comput.* **2015**, *18*, 118–136.
8. Zhuang, Y.; Wright, B.; Syed, Z.; Shen, Z.; El-Sheimy, N. Fast WiFi access point localization and autonomous crowdsourcing. In Proceedings of the Ubiquitous Positioning Indoor Navigation and Location Based Service (UPINLBS), Corpus Christ, TX, USA, 20–21 November 2014; pp. 272–280.
9. Zhao, X.; Xiao, Z.; Markham, A.; Trigoni, N.; Ren, Y. Does BTLE measure up against WiFi? A comparison of indoor location performance. In Proceedings of the 20th European Wireless Conference, Barcelona, Spain, 14–16 May 2014; pp. 1–6.
10. Ozdenizci, B.; Coskun, V.; Ok, K. NFC internal: An indoor navigation system. *Sensors* **2015**, *15*, 7571–7595.
11. Cassimon, D.; Engelen, P.-J.; Yordanov, V. Compound real option valuation with phase-specific volatility: A multi-phase mobile payments case study. *Technovation* **2011**, *31*, 240–255.
12. Ruiz, A.J.; Granja, F.S.; Prieto Honorato, J.C.; Rosas, J.I. Accurate pedestrian indoor navigation by tightly coupling foot-mounted IMU and RFID measurements. *Instrum. Meas. IEEE Trans.* **2012**, *61*, 178–189.
13. Yuan, X.; Yu, S.; Zhang, S.; Wang, G.; Liu, S. Quaternion-based unscented Kalman filter for accurate indoor heading estimation using wearable multi-sensor system. *Sensors* **2015**, *15*, 10872–10890.
14. Liu, C.; Zhang, S.; Yu, S.; Yuan, X.; Liu, S. Design and analysis of gyro-free inertial measurement units with different configurations. *Sens. Actuators A Phys.* **2014**, *214*, 175–186.
15. Lee, Y.-C.; Yu, W.; Cho, J.-I. Adaptive localization for mobile robots in urban environments using low-cost sensors and enhanced topological map. In Proceedings of the 15th International Conference on Advanced Robotics (ICAR), Tallinn, Estonia, 20–23 June 2011; pp. 569–575.

16. Zhuang, Y.; Chang, H.W.; El-Sheimy, N. A MEMS multi-sensors system for pedestrian navigation. In Proceedings of the China Satellite Navigation Conference (CSNC), Wuhan, China, 4 May 2013; pp. 651–660.
17. Titterton, D.H.; Weston, J.L. *Strapdown Inertial Navigation Technology*, 2nd ed.; Institution of Engineering and Technology: Stevenage, UK, 2004.
18. Li, Y.; Niu, X.; Zhang, Q.; Zhang, H.; Shi, C. An *in situ* hand calibration method using a pseudo-observation scheme for low-end inertial measurement units. *Meas. Sci. Technol.* **2012**, *23*, 105104.
19. Chiang, K.-W.; Chang, H.-W. Intelligent sensor positioning and orientation through constructive neural network-embedded INS/GPS integration algorithms. *Sensors* **2010**, *10*, 9252–9285.
20. Atia, M.; Noureldin, A.; Georgy, J.; Korenberg, M. Bayesian filtering based WiFi/INS integrated navigation solution for GPS-denied environments. *Navigation* **2011**, *58*, 111–125.
21. Lan, H.; El-sheimy, N. A state constraint Kalman filter for pedestrian navigation with low cost MEMS inertial sensors. In Proceedings of the 27th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+ 2014), Tampa, FL, USA, 8–12 September 2014; pp. 579–589.
22. Gezici, S. A survey on wireless position estimation. *Wirel. Pers. Commun.* **2008**, *44*, 263–282.
23. Bahl, P.; Padmanabhan, V.N. Radar: An in-building RF-based user location and tracking system. In Proceedings of the 19th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2000), Tel Aviv, Israel, 26–30 March 2000; Volume 2, pp. 775–784.
24. Zhao, X.; Goodall, C.; Syed, Z.; Wright, B.; El-Sheimy, N. Wi-Fi assisted multi-sensor personal navigation system for indoor environments. In Proceedings of the 2010 International Technical Meeting of the Institute of Navigation, San Diego, CA, USA, 25–27 January 2010; pp. 236–243.
25. Waegli, A.; Skaloud, J. Optimization of two GPS/MEMS-IMU integration strategies with application to sports. *GPS Solut.* **2009**, *13*, 315–326.
26. Nassar, S.; Shin, E.-H.; Niu, X.; El-Sheimy, N. Accurate INS/GPS positioning with different inertial systems using various algorithms for bridging GPS outages. In Proceedings of the 18th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2005), Long Beach, CA, USA, 13–16 September 2005; pp. 1401–1410.
27. 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.
28. Ladetto, Q.; Gabaglio, V.; Merminod, B. Combining gyroscopes, magnetic compass and GPS for pedestrian navigation. In Proceedings of the International Symposium on Kinematic Systems in Geodesy, Geomatics, and Navigation, Banff, Canada, 30 August–2 September 1994; pp. 205–213.
29. Chen, L.-H.; Wu, E.H.K.; Jin, M.-H.; Chen, G.-H. Intelligent fusion of Wi-Fi and inertial sensor-based positioning systems for indoor pedestrian navigation. *IEEE Sens. J.* **2014**, *14*, 4034–4042.
30. Frank, K.; Krach, B.; Catterall, N.; Robertson, P. Development and evaluation of a combined WLAN & inertial indoor pedestrian positioning system. In Proceedings of the 22nd International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS 2009), Savannah, GA, USA, 22–25 September 2009; pp. 538–546.
31. Evennou, F.; Marx, F. Advanced integration of WiFi and inertial navigation systems for indoor mobile positioning. *Eurasip J. Appl. Signal Process.* **2006**, *2006*, 164.

32. Chai, W.; Chen, C.; Edwan, E.; Zhang, J.; Loffeld, O. 2D/3D indoor navigation based on multi-sensor assisted pedestrian navigation in Wi-Fi environments. In Proceedings of the Ubiquitous Positioning, Indoor Navigation, and Location Based Service (UPINLBS), Helsinki, Finland, 3–4 October 2012; pp. 1–7.
33. El-Sheimy, N. *Inertial Techniques and INS/DGPS Integration*; Department of Geomatics Engineering, University of Calgary: Calgary, Canada, 2006.
34. Shin, E.-H. Estimation Techniques for Low-Cost Inertial Navigation. Ph.D. Thesis, University of Calgary, Calgary, Canada, 2005.
35. Gelb, A. *Applied Optimal Estimation*; MIT Press: Cambridge, MA, USA, 1999.
36. Weinberg, H. Using the ADXL202 in pedometer and personal navigation applications. In *Analog Devices AN-602 Application Note*; Analog Devices: Norwood, MA, USA, 2002.
37. Syed, Z.; Aggarwal, P.; Yang, Y.; El-Sheimy, N. Improved vehicle navigation using aiding with tightly coupled integration. In Proceedings of the Vehicular Technology Conference, 2008 (VTC 2008), Singapore, 11–14 May 2008; pp. 3077–3081.