

Article

# A Novel Pedestrian Dead Reckoning Algorithm for Multi-Mode Recognition Based on Smartphones

Limin Xu <sup>1,2,\*</sup>, Zhi Xiong <sup>1,2,\*</sup>, Jianye Liu <sup>1,2</sup>, Zhengchun Wang <sup>1,2</sup> and Yiming Ding <sup>1,2</sup>

<sup>1</sup> College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Jiangning, Nanjing 211106, China; LJYAC@nuaa.edu.cn (J.L.); wangzc@nuaa.edu.cn (Z.W.); DingYiming@nuaa.edu.cn (Y.D.)

<sup>2</sup> Navigation Research Center, Nanjing University of Aeronautics and Astronautics, Jiangning, Nanjing 211106, China

\* Correspondence: xlmin@nuaa.edu.cn (L.X.); xiongzhi@nuaa.edu.cn (Z.X.); Tel.: +86-158-5065-0308 (L.X.); +86-138-1380-8576 (Z.X.)

Received: 27 November 2018; Accepted: 29 January 2019; Published: 1 February 2019



**Abstract:** With the rapid development of smartphone technology, pedestrian navigation based on built-in inertial sensors in smartphones shows great application prospects. Currently, most smartphone-based pedestrian dead reckoning (PDR) algorithms normally require a user to hold the phone in a fixed mode and, thus, need to correct the gyroscope heading with inputs from other sensors, which restricts the viability of pedestrian navigation significantly. In this paper, in order to improve the accuracy of the traditional step detection and step length estimation method for different users, a state transition-based step detection method and a step length estimation method using a neural network are proposed. In order to decrease the heading errors and inertial sensor errors in multi-mode system, a multi-mode intelligent recognition method based on a neural network was constructed. On this basis, we propose a heading correction method based on zero angular velocity and an overall correction method based on lateral velocity limitation (LV). Experimental results show that the maximum positioning errors obtained by the proposed algorithm are about 0.9% of the total path length. The proposed novel PDR algorithm dramatically enhances the user experience and, thus, has high value in real applications.

**Keywords:** pedestrian dead reckoning; smartphone-based navigation; neural network; multi-mode recognition; heading correction

## 1. Introduction

With the development of society, location-based service became part of people's lives. A pedestrian navigation and positioning service system mainly depends on a global positioning system (GPS) in an outdoor environment [1,2]. For indoor applications in, e.g., airports, train stations, underground garages, and shopping malls, GPS often becomes unavailable because of signal occlusion, which would hinder the application of location-based services in these areas. Therefore, pedestrian indoor navigation technology is critical to ensure the success of location-based services.

Currently, pedestrian indoor navigation and positioning can be achieved using two types of methods. The first type of method is based on wireless technologies, e.g., WiFi [3–5], ultra-wideband (UWB) [6,7], visual sensors [8], radio frequency identification (RFID) [9], iBeacon [10], Bluetooth, and/or ZigBee, with a multi-source information fusion technique [11] to obtain pedestrian location information. For the first type of method, its location errors do not accumulate over time. However, these methods need a significant cost to deploy wireless devices as beacons before navigation and are infeasible for the unknown environment.

The second type of method is based on a pedestrian dead reckoning (PDR) algorithm using a micro-electro-mechanical system (MEMS) [12]. In these methods, an inertial sensor is fixed on the foot [13,14], waist [15,16], leg, or shoulder of a user for data acquisition, in order to determine the position of pedestrian using the number and the size of walking steps and the heading angle [17]. These methods belong to autonomous positioning technologies and have a high positioning accuracy in a short-time period. However, their positioning performance is drastically affected by the accumulative errors of iterative calculation. Therefore, many researchers are committed to solving the accumulative errors in the traditional PDR algorithm [18,19]. Zhou et al. [19] proposed a PDR algorithm with low computational complexity using a foot-mounted inertial measurement unit (IMU) and estimating orientation of the foot based on extended Kalman filter (EKF), which can correct the accumulative errors of PDR. Nonetheless, it can only be applied for foot-mounted navigation and, therefore, has poor viability for different users.

The MEMS-based algorithm brings inconvenience to practical application due to the extra-installation of inertial sensors. In allusion to the inconvenience of pedestrian positioning technology based on fixed inertial sensors, smartphone-based pedestrian navigation gradually became a research hotspot for experts. Bilke et al. [20] proposed a locating system based on recognizing geomagnetic field disturbances and ambient light. Li [21] combined Bluetooth and an inertial navigation system to implement a set of navigation systems based on a smartphone. Kang and Han [22] designed a smartphone-based algorithm, named smartPDR, but users had to reduce the sway of the smartphone in experiments. Wang et al. [23] presented a PDR approach based on motion mode recognition using a smartphone. The motion mode recognition was achieved using a support vector machine (SVM) and a decision tree (DT), which thereby increased the complexity of the algorithm. Zeng et al. [24] proposed an autonomous inertial heading correction algorithm based on the Kalman filter and the different usage modes of the smartphone, e.g., normal mode, landscape mode, or call mode. The usage mode could be determined based on the gravity-assisted (GA) method. However, in References [22–24], step detection methods were both based on peak detection, which would introduce detection errors due to the step detection errors. Moreover, the step length estimation method was based on a nonlinear model or constant which would cause position errors for different users. Furthermore, mode recognition methods do not have universality, as the accuracy of these methods is affected by the configurations of the threshold.

The heading error is the one of the main error sources in PDR. In References [20–22], smartphone-based navigation algorithms required users to hold a smartphone in a fixed mode as long as possible. However, it is difficult for users to keep a fixed mode during the movement process, which leads to significant heading errors. Wang et al. [23] proposed a principal component analysis (PCA)-based method with global accelerations (PCA-GA) to infer pedestrian headings. However, one set of experiments represented only one phone pose, which is not applicable to multi-mode changes of a smartphone. Although Reference [24] proposed using a Kalman filter to correct the heading errors in a multi-mode application, heading correction values are hard to estimate precisely due to weak constraints and observation errors.

In this paper, we propose a novel pedestrian dead reckoning algorithm for multi-mode recognition based on a smartphone. Firstly, different from the peak detection in References [22–24], state transition was carried out to improve the accuracy of step detection for different phone modes. Secondly, we adopted the neural network to obtain the step length and the smartphone mode without setting thresholds manually, which increased the universality of step length estimation and mode recognition methods. The main contributions of the paper are that we design the heading correction method based on zero angular velocity (ZA) and an overall correction method based on lateral velocity limitation (LV) to correct the error of heading estimation and inertial sensors in a smartphone-based PDR. Experimental results show that the algorithm proposed in this paper is effective.

The rest of the paper is organized as follows: Section 2 presents our proposed algorithm in detail, including the comparison between the traditional PDR algorithm and the proposed novel PDR

algorithm, along with the step detection, step length estimation, multi-mode intelligent recognition, and heading correction methods. The experimental results are systematically analyzed in Section 3. Sections 4 and 5 discuss the system and draw conclusions.

## 2. Materials and Methods

### 2.1. Proposed System Scheme

As shown in Figure 1, using an accelerometer and a gyroscope, the traditional scheme of the PDR algorithm estimates the position, velocity, and heading using the results of step detection, step length estimation, and heading estimation.

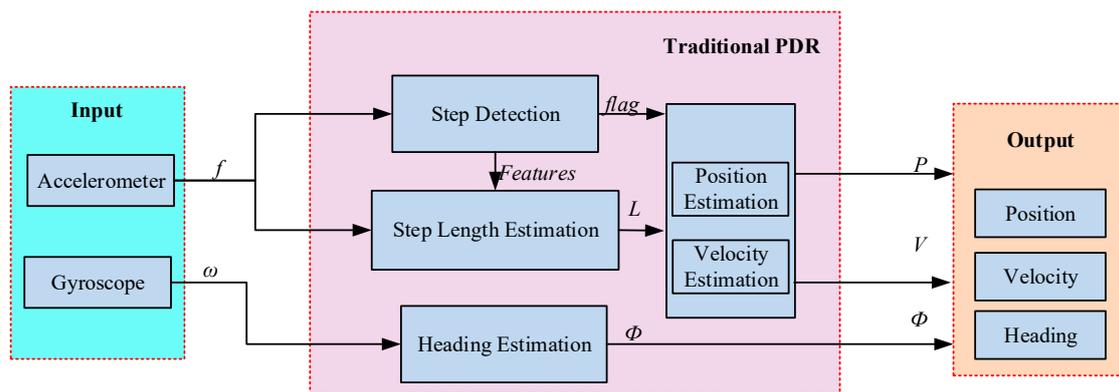


Figure 1. Scheme of traditional pedestrian dead reckoning (PDR) algorithm.

In Figure 1,  $f$  and  $\omega$  are the outputs of the accelerometer and gyroscope, respectively,  $flag$  indicates the result of step detection,  $L$  is the step length estimation, and  $\phi$  is the heading estimation of the pedestrian. Pedestrian navigation involves a body coordinate system and a navigation coordinate system. The  $x$ -axis of the body coordinate system corresponds to the lateral direction of the smartphone, the  $y$ -axis corresponds to the forward direction of the smartphone, and the  $z$ -axis satisfies the right-hand rule. The navigation coordinate system involves east, north, up (ENU) coordinates.

The traditional heading estimation method uses a  $z$ -axis gyroscope  $\omega_{zk}$ , denoted by

$$\varphi_k = \varphi_{k-1} - T\omega_{zk}, \quad (1)$$

where  $\varphi_k$  and  $\varphi_{k-1}$  are the headings at time instants  $kT$  and  $(k-1)T$ , respectively. The initial heading is set manually.

The traditional position estimation method is resolved using the step length  $L_i$  and the heading of the  $i$ -th step, denoted by

$$\begin{cases} x_i = x_{i-1} + L_i \sin \varphi_i \\ y_i = y_{i-1} + L_i \cos \varphi_i \end{cases}. \quad (2)$$

The velocity estimation method is calculated by

$$\begin{cases} f_{e,k} = f_{x,k} \cos \varphi_k + f_{y,k} \sin \varphi_k \\ f_{n,k} = -f_{x,k} \sin \varphi_k + f_{y,k} \cos \varphi_k \end{cases}, \quad (3)$$

where  $T$  represents sampling period,  $f_{x,k}$  and  $f_{y,k}$  are the lateral acceleration and forward acceleration on the horizontal plane at time instants  $kT$ , and  $f_{e,k}$  and  $f_{n,k}$  are the eastward acceleration and northward acceleration at time instants  $kT$ . Then, the eastward velocity and northward velocity are calculated by

$$v_{e,k} = v_{e,k-1} + Tf_{e,k}, v_{n,k} = v_{n,k-1} + Tf_{n,k}. \quad (4)$$

The accurate estimations of step, step length, and heading are crucial for a pedestrian navigation algorithm. Since the accuracy of built-in inertial sensors in a smartphone is at a lower level, the traditional PDR algorithm would have relatively large navigation errors; in particular, the drift of heading can result in a drastic increase in errors. In addition, considering the user experience, this paper studies a multi-mode smartphone-based navigation algorithm. In order to solve the defects of the traditional PDR algorithm, we propose a novel PDR algorithm, which includes step detection based on state transition, a step length estimation method based on neural network and differential GPS, and multi-mode intelligent recognition, a heading correction method based on zero angular velocity (ZA), and an overall correction method based on lateral velocity limitation (LV), as shown in Figure 2. In the figure, the purple dotted boxes represent the neural networks which are outlined by different features. We estimate the results directly using these neural networks in a PDR algorithm. Specific algorithm theories are shown in Sections 2.2–2.6.

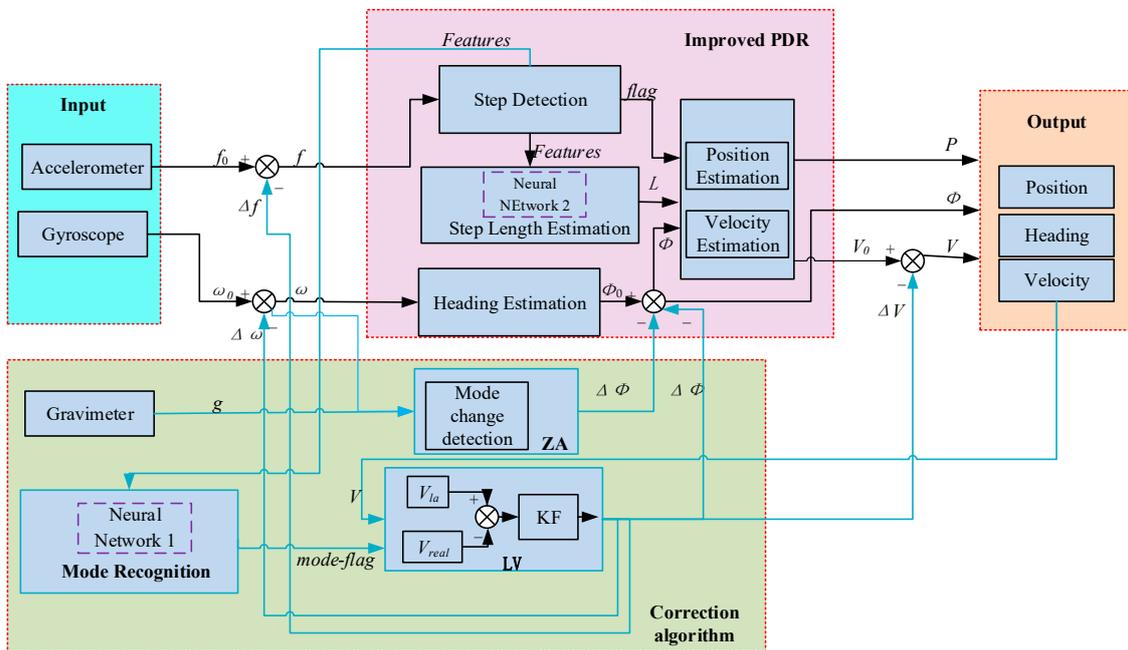


Figure 2. The scheme of the proposed novel PDR algorithm.

### 2.2. Step Detection Based On State Transition

During the walking process of the human handheld smartphone, the lifting and undulating of the foot cause the lifting and undulating of the human body. This cyclical fluctuation is reflected in the acceleration of the smartphone, as shown in Figure 3a. The delt-accNorm in Figure 3a represents the differences between the acceleration modulus and the modulus of the pedestrian’s initial stationary stage, denoted by Equation (5). The purpose of the differences is to reduce the difference of the threshold of step detection between different pedestrians. In view of the noise of the accelerometer in the course of motion, the curve burr as shown in Figure 3a will reduce the accuracy of step detection. Thus, we smooth out the interference of acceleration using a moving average filter shown in Equation (6). The smoothed result is shown in Figure 3b.

$$\Delta f_i = \sqrt{f_{ix}^2 + f_{iy}^2 + f_{iz}^2} - \sqrt{f_{0x}^2 + f_{0y}^2 + f_{0z}^2}, \tag{5}$$

$$f_k = \frac{1}{2n + 1} \sum_{i=k-n}^{i=k+n} \Delta f_i, \tag{6}$$

where  $f_{ix}$ ,  $f_{iy}$ , and  $f_{iz}$  represent three-axis acceleration at time instants  $iT$ , while  $f_{0x}$ ,  $f_{0y}$ , and  $f_{0z}$  represent the mean value of the three-axis acceleration at the initial stationary stage.  $f_k$  represents the delt-accNorm at time instant  $kT$ ,  $n$  is the size of the sliding window, and  $T$  is the sampling period.

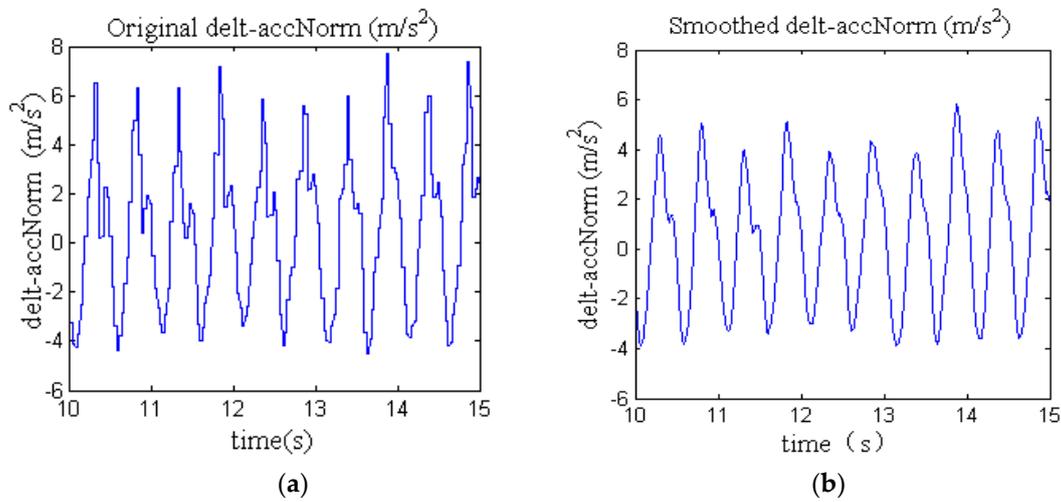


Figure 3. Original and smoothed delt-accNorm.

The valley value  $f_{k, valley}$  and the peak value  $f_{k, peak}$  of delt-accNorm are taken as the conditions of step detection. The specific models are as follows:

- (1) Peak value detection: if  $f_{k, peak} > TH_{peak}$ , then the mode-flag is set to 1.
- (2) Valley value detection: if  $f_{k, valley} > TH_{valley}$ , and the time difference between the last peak and the valley value  $\Delta T_{k, peak, valley}$  satisfies  $\Delta T_{k, peak, valley} > TH_{peak, valley}$ , then the mode-flag is set to 2.
- (3) On the basis of (2), the next peak detection is carried out, and if the time difference between the last valley and this peak value  $\Delta T_{k, valley, peak}$  satisfies  $\Delta T_{k, valley, peak} > TH_{valley, peak}$ , then the mode-flag is set to 3.
- (4) When the mode-flag is 3, it represents that we detected one step successfully. Then, the mode-flag is set to 1 and the cycle is repeated.

$TH_{peak}$ ,  $TH_{valley}$ ,  $TH_{peak, valley}$ , and  $TH_{valley, peak}$  are the detection thresholds of peak value, valley value, the time difference between the last peak and the current valley value, and the time difference between the last valley and the current peak value, respectively. The process of step detection is shown in Figure 4.

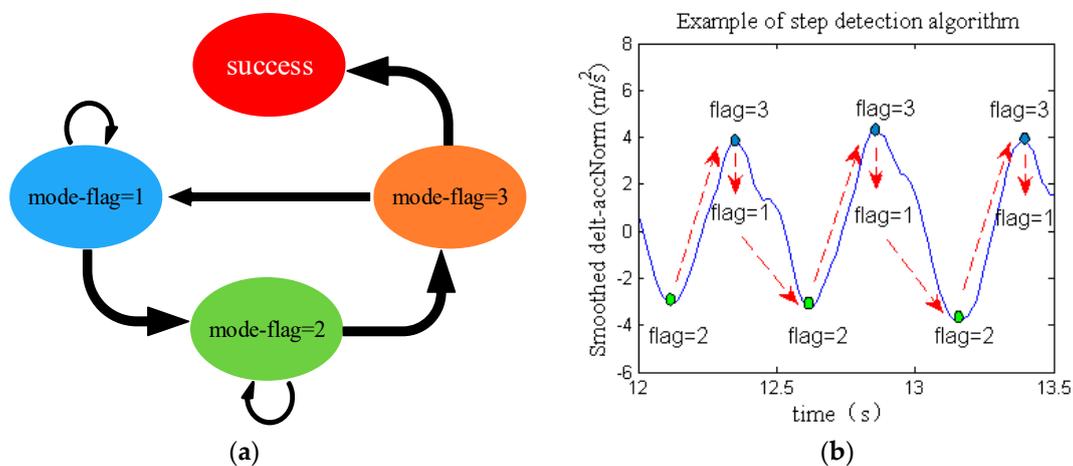


Figure 4. (a) Flowchart of step detection. (b) Example of step detection algorithm.

The difference between the handheld and the foot-fixed smartphone-based navigation algorithm is that the latter has an obvious zero velocity interval [13], while the former has only very short zero velocity moments. The zero velocity moment of the handheld smartphone-based navigation is the first sampling period which is less than the modulus value of the initial stationary stage after the peak appears. Meanwhile, the zero velocity flag is set to 1. Moreover, in order to reduce the cumulative errors of velocity, the velocity of that moment is set to zero.

In order to verify the accuracy of the step detection method, we carried out three experiments, and the experimental routes included a straight line and a rectangular path in an indoor corridor. In Table 1, “actual” means the actual steps and “algorithm” represents the results of the proposed algorithm. In summary, the average accuracy of the step detection method could reach 99%.

**Table 1.** Results of the step detection.

No.	Actual	Algorithm	Accuracy
1	67	66	98.5%
2	206	205	99.5%
3	310	307	99.0%

### 2.3. Step Length Estimation Method Based on Neural Network and Differential GPS

The step length of the pedestrian is affected by factors such as height, step frequency, and environment. Based on the human motion model and the acceleration data collected, it was found that peak values, valley values, and the time differences between contiguous peak values of  $\text{delt-accNorm}$  are closely related to the step length estimation. In this paper, we constructed a three-layer neural network with four nodes in the input layer, one node in the output layer, and 12 nodes in the hidden layer. The inputs of the model were  $f_{k,peak}$ ,  $f_{k,valley}$ ,  $\Delta T_{k,peak,valley}$ , and  $\Delta T_{k,valley,peak}$ . The output of the model was the step length, calculated using the position of a high-precision differential GPS. To solve the problem whereby the estimation accuracy of traditional step length estimation methods differs from that of different users with different velocities, we collected training data from different users walking fast and normally to ensure the diversity of the training data and to improve the applicability of the method. Furthermore, the transfer functions of the hidden layer and the output layer were a tangent sigmoid function (tansig) and a pure linear function (purelin). The backprop network training function was the Levenberg–Marquardt backpropagation (BP) function (trainlm), and the backprop weight/bias learning function involved BP learning rules with a momentum term (learngdm). The weight initialization strategy of the network satisfied a random distribution between  $-1$  and  $1$ . In addition, the target error, the learning of neural network, and the time of training were set as  $0.001$ ,  $0.1$ , and  $5000$ , respectively.

In the training stage, we firstly normalized input data and output data, i.e., the acceleration characteristic information and the step length, respectively. Then, the backpropagation (BP) neural network was trained.

In the test stage, we input real-time acceleration characteristic information and calculated the step length using the trained neural network.

To highlight the advantages of the proposed step length estimation method, we compared it with the traditional non-linear method in Reference [3]. The experimental route was a  $54.3\text{-m}$  straight line, and four experiments were carried out. The results are shown in Table 2. We can see that the error of the proposed method was obviously smaller than that of the nonlinear method.

**Table 2.** Errors of step length estimation.

No.	Nonlinear Method (m)	Proposed Method (m)
1	2.13	0.51
2	1.59	0.32
3	1.24	0.43
4	1.03	0.16

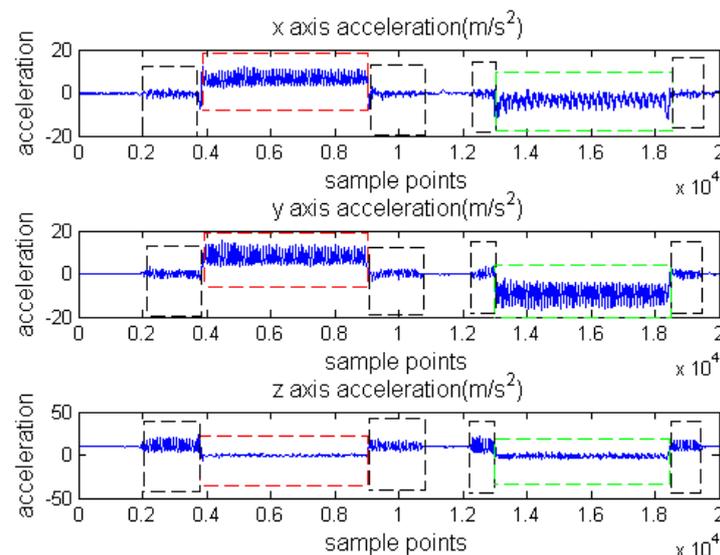
#### 2.4. Multi-Mode Intelligent Recognition

In addition to the normal mode of a handheld smartphone, the phone mode and the swinging-hand mode were also studied, as shown in Figure 5.



**Figure 5.** Different smartphone modes: (a) normal mode; (b) phone mode; (c) swinging-hand mode.

In consideration of the different induction characteristics of sensors corresponding to different modes, we analyzed the differences of triaxial sensors using the three modes. The acceleration data are shown in Figure 6, where the mode in the black dotted rectangle is the normal mode, the mode in the red dotted rectangle is the phone mode, and the mode in the green dotted rectangle is the swinging-hand mode.



**Figure 6.** Acceleration of different modes.

Traditional mode recognition methods mainly include a decision tree and SVM. We need to accurately analyze the data characteristics of different modes and set accurate thresholds when using the decision tree method. When using the SVM method, we need to determine the appropriate kernel function and provide a large amount of storage space. The above two methods have certain limitations; thus, we propose a mode recognition method based on the BP neural network. This paper not only takes  $f_{k,peak}$ ,  $f_{k,valley}$ ,  $\Delta T_{k,peak,valley}$ , and  $\Delta T_{k,valley,peak}$  as characteristic variables, but also extracts the

$x$ -axis acceleration variation between the current step and the previous step, the  $x$ -axis acceleration mean value, and the standard deviation of the current step to form a seven-dimensional input vector. The neural network had three layers. There were seven nodes in the input layer, ten nodes in the hidden layer, and one node representing the phone-flag in the output layer. The transfer function, backprop network training function, and backprop weight/bias learning function of the network were the same as those in Section 2.3. The weight initialization strategy of the network satisfied a random distribution between  $-1$  and  $1$ . In addition, the target error, the learning of the neural network, and the time of training were set as  $0.01$ ,  $0.1$ , and  $1000$ , respectively.

In the training stage, the feature values of each step were extracted as inputs, and the phone-flag was set as the output, i.e., the corresponding phone-flags for normal mode, phone mode, and swinging-hand mode were set as  $1$ ,  $2$ , and  $3$ , respectively. Then, we normalized the training data and obtained a trained neural network.

In the test stage, we input real-time feature data, and the trained neural network was utilized to obtain the phone-flag.

In order to prove the accuracy of this method, we carried out four smartphone mode changing experiments in the indoor corridor. The actual phone-flag order was  $1-2-1-3-1-2-1-3-1$ . The accuracy of the multi-mode recognition is shown in Table 3, with an average accuracy of  $99.58\%$ .

**Table 3.** The accuracy of mode recognition.

No.	1	2	3	4
Accuracy	99.38%	99.34%	100%	99.63%

### 2.5. Heading Correction Method Based on Zero Angular Velocity

The traditional method of heading estimation is shown in Equation (1), which is only applicable to the state of a smartphone plane parallel to the horizontal plane. In view of the large attitude angle errors caused by different smartphone modes when users walk, if not corrected, the heading errors will increase. Therefore, the first step of the heading correction method based on zero angular (ZA) velocity (ZA) algorithm was horizontal alignment. In this paper, the attitude conversion matrix of horizontal alignment was calculated using gravimeter data.

The pitch angle  $\theta$  and the roll angle  $\gamma$  were calculated based on the data of three-axis gravitational acceleration  $[g_x, g_y, g_z]$ , where  $g$  is the gravitational constant, denoted by

$$\theta = \arcsin\left(\frac{g_x}{g}\right), \quad \gamma = -\arctan\left(\frac{g_x}{g_z}\right). \quad (7)$$

The acceleration and the angular velocity were projected to the horizontal coordinate by pitch angle and roll angle, denoted by

$$f_{hor} = (C_1 C_2)^{-1} f_b, \quad \omega_{hor} = (C_1 C_2)^{-1} \omega_b, \quad (8)$$

where  $b$  represents the body coordinate system and  $hor$  represents the horizontal coordinate system obtained by the projection.  $C_1$  and  $C_2$  are the attitude conversion matrices, denoted by:

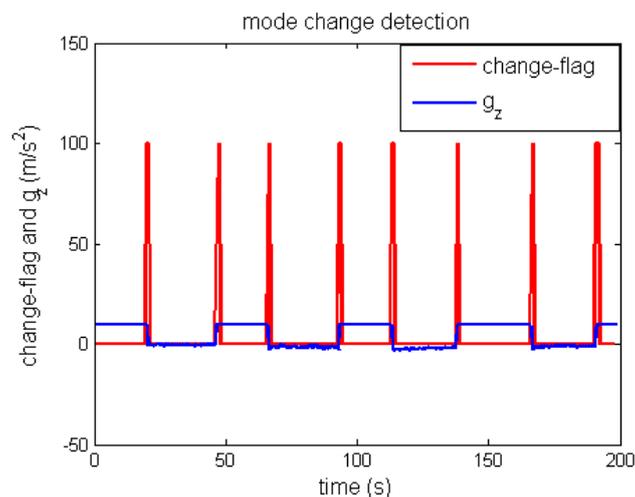
$$C_1 = \begin{bmatrix} \cos \gamma & 0 & -\sin \gamma \\ 0 & 1 & 0 \\ \sin \gamma & 0 & \cos \gamma \end{bmatrix}, \quad C_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & \sin \theta \\ 0 & -\sin \theta & \cos \theta \end{bmatrix}. \quad (9)$$

During pedestrian movement, there are two situations that cause a large heading change. One is the pedestrian turning process, and the other is the mode switching process. If the  $z$ -axis horizontal angular velocity recurs directly, a heading error caused by the mode switching will be introduced.

Therefore, the second step of the ZA algorithm was to detect the mode switching state and correct the z-axis horizontal angular velocity.

Because of the misjudgment of the mode recognition algorithm, it is limited to judging the smartphone mode switching state only through the results of Section 2.4. In order to accurately detect the mode switching state, we analyzed the z-axis gravity of the body coordinate system, as shown by the blue line in Figure 7. We can see that the z-axis gravity changes significantly when the smartphone mode switches, which is due to the axis inconsistency of induced gravity acceleration at different smartphone modes. However, the z-axis gravity will not change dramatically when the user makes a turn. This is because the handheld smartphone mode is in a fixed mode. We took the z-axis gravity variation as the characteristic quantity of mode switching. The mode change detection algorithm is shown in Equation (10), where  $\Delta g_{i,z}$  represents the z-axis gravity variation between the current step and the previous step, and  $th_{gz}$  represents the threshold of gravity variation.

$$\begin{cases} \Delta g_{i,z} = g_{i,z} - g_{i-1,z} > th_{gz}, change\_flag = 100 \\ \text{else, } change\_flag = 0 \end{cases} \quad (10)$$



**Figure 7.** The mode recognition results.

In Figure 4, the red line represents the change-flag of mode switching. It is known that the detection algorithm is accurate and there is no misclassification. When mode switching is detected, the z-axis angular velocity is set to zero to reduce the heading error caused by mode switching.

In addition, when the heading angle changes significantly, it must be a turning state based on the fixed mode, as shown in Equation (11), where  $\Delta\varphi_i$  means the heading variation between the current step and the previous step, and  $th_\varphi$  represents the threshold of heading variation, which we set to 20.

$$\begin{cases} (\Delta\varphi_i = \varphi_i - \varphi_{i-1} > th_\varphi) \&\&(change\_flag = 0), turn\_flag = 1 \\ \text{else, } turn\_flag = 0 \end{cases} \quad (11)$$

## 2.6. Overall Correction Method Based on Lateral Velocity Limitation

The ZA algorithm only corrects the heading error caused by the mode switching, but not the error caused by a slight sway of the smartphone in the fixed mode. Especially in the phone mode and the swinging-hand mode, the heading errors caused by the sway of a handheld phone are more obvious. Since there is no lateral motion when the user walks straight, a virtual measurement of zero lateral velocity was constituted, and an overall correction algorithm based on lateral velocity constraint was established. Furthermore, on the basis of step detection in Section 2.2, when it is not the zero velocity

moment, we propose to use the overall correction method based on lateral velocity limitation to correct the navigation errors.

The seven-dimensional kalman filter state vector was constructed as follows:

$$X = [\delta\varphi \quad \delta V_e \quad \delta V_n \quad \varepsilon_{bz} \quad \varepsilon_{rz} \quad \nabla_{rx} \quad \nabla_{ry}]^T, \tag{12}$$

where  $\delta\varphi$  is the heading error,  $[\delta V_e; \delta V_n]$  are the eastward velocity error and the northward velocity error, respectively,  $[\varepsilon_{bz}; \varepsilon_{rz}]$  are the zero bias and the random error of the first-order Markov process of the z-axis gyroscope, and  $[\nabla_{rx}; \nabla_{ry}]$  are the random errors of the first-order Markov process of the x-axis and the y-axis accelerometers.

The state equation is denoted by

$$\dot{X} = AX + GW, \tag{13}$$

where  $A$  is the system matrix,  $G$  is the system noise covariance matrix, and  $W$  is the system noise. These three matrices are calculated using a recursive model based on PDR as

$$A = \begin{bmatrix} 0 & 0 & 0 & -1 & -1 & 0 & 0 \\ -f_x \sin \varphi + f_y \cos \varphi & 0 & 0 & 0 & 0 & \cos \varphi & \sin \varphi \\ f_x \cos \varphi + f_y \sin \varphi & 0 & 0 & 0 & 0 & -\sin \varphi & \cos \varphi \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\frac{1}{T_\varepsilon} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -\frac{1}{T_a} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{T_a} \end{bmatrix},$$

$$G = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \text{ and}$$

$$W = \begin{bmatrix} w_{\varepsilon,bz} & w_{\varepsilon,rz} & w_{\varepsilon,rx} & w_{\varepsilon,ry} \end{bmatrix}.$$

$T_\varepsilon$  and  $T_a$  are the time variables of the first-order Markov of gyroscope and accelerometer, respectively.

Lateral velocity  $v_{la}$  is a measurement, which is calculated from eastward velocity and northward velocity as

$$v_{la} = (v_e + \delta v_e) \cos(\varphi + \delta\varphi) - (v_n + \delta v_n) \sin(\varphi + \delta\varphi). \tag{14}$$

Equation (15) is obtained from a linearization of Equation (14).

$$\begin{aligned} v_{la} &= v_{real} + \delta v_e \cos \varphi - \delta v_n \sin \varphi \\ &\quad - v_e \sin \varphi \delta\varphi - v_n \cos \varphi \delta\varphi \end{aligned} \tag{15}$$

The measurement equation is denoted by

$$Z = v_{la} - (v_{real} - \delta v_{real}) = HX + V, \tag{16}$$

$$H = \begin{bmatrix} -v_e \sin \varphi - v_n \cos \varphi & \cos \varphi & -\sin \varphi & 0 & 0 & 0 & 0 \end{bmatrix}, V = [\delta v_{real}],$$

where  $v_{real}$  is the actual lateral velocity, i.e.,  $v_{real}$  is zero. The measurement matrix  $H$  is calculated using Equation (14). The measurement noise  $\delta v_{real}$  is the additional lateral velocity error caused by the sway of the smartphone and is considered white noise. For different smartphone modes, the additional

lateral velocity is different. Thus, we set the measurement noise of the normal mode to 0.4, and the measurement noise of the phone mode and the swinging-hand mode to 0.9 in order to improve the filtering accuracy.

### 3. Results

#### 3.1. Performance Analysis of Smartphone Sensors

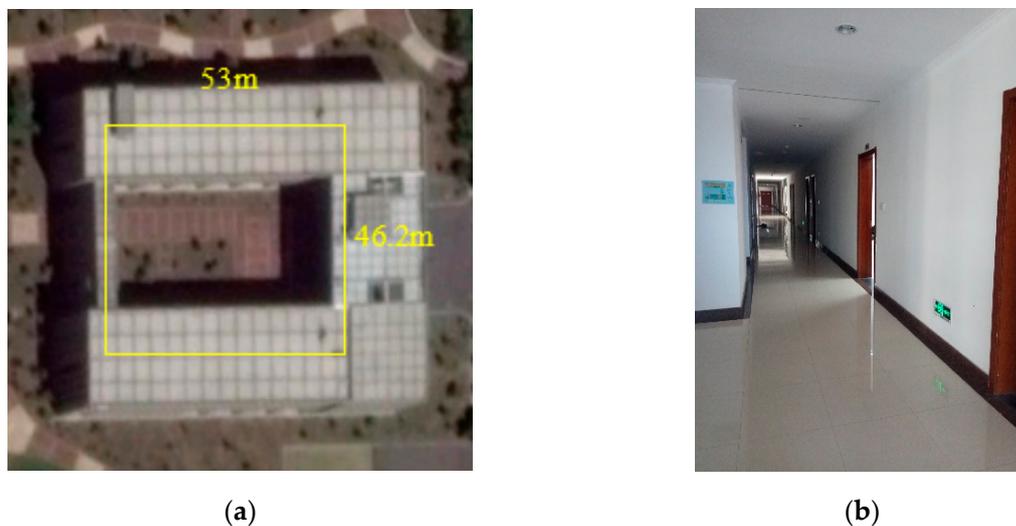
We analyzed the static data of smartphone sensors to estimate the sensors bias accurately. The mobile phone was placed on the horizontal turntable of the laboratory, and the static data were obtained for about an hour. The sampling frequency was 200 Hz. According to the static data analysis, the bias of the gyroscope and accelerometer are shown in Table 4. The results show that the accuracy performance was at a low level. This is because the smartphone sensors used were commercial products with relatively low accuracy and, hence, made the PDR test results underperform.

**Table 4.** Performance parameters.

Bias of Sensors	<i>x</i> -axis	<i>y</i> -axis	<i>z</i> -axis
Gyroscope ( $^{\circ}/s$ )	0.45943	0.04161	−0.1461
Accelerometer ( $m/s^2$ )	0.1662	−0.00019	9.8611

#### 3.2. Experiment Comparison of Different Algorithms

The experiments were carried out in indoor scenarios. We conducted the walking experiments in the buildings of the Navigation Research Center (NRC) of Nanjing University of Aeronautics and Astronautics, shown in Figure 8.



**Figure 8.** The environment of the walking experiments: (a) display of experiment area on Baidu map; (b) a corridor of the experiment area.

##### 3.2.1. Experiment Based on Smartphone Normal Mode

In order to verify the overall correction method based on the lateral velocity limitation algorithm proposed in Section 2.6, two different experiments based on the smartphone normal mode were performed. The first involved walking along a straight line of 54 m. The second one involved walking along an L shape for 91.2 m. The distance of the experiment was constrained by the experimental area. The relative distance was measured using a high-precision laser rangefinder. The positions were all initialized as (0, 0) in an *xy*-coordinate system with north at the top.

In Figures 9 and 10, the solid black line represents the actual trajectory of the experiment, and the blue line represents the result of the improved PDR (IM\_PDR), which includes the step detection based on state transition, the step length estimation method based on a neural network, and the traditional heading estimation method. The red line represents the result of the LV algorithm based on the improved PDR (IM\_PDR+LV). The heading errors are calculated as the absolute values.

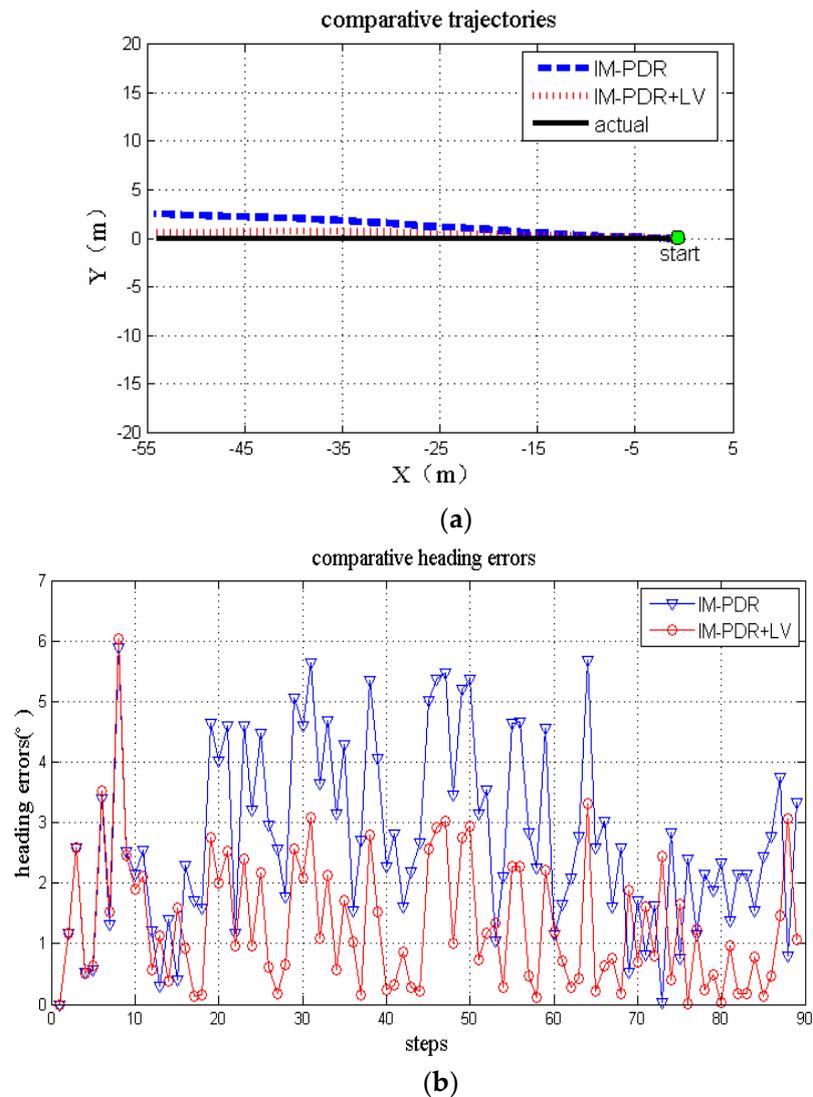
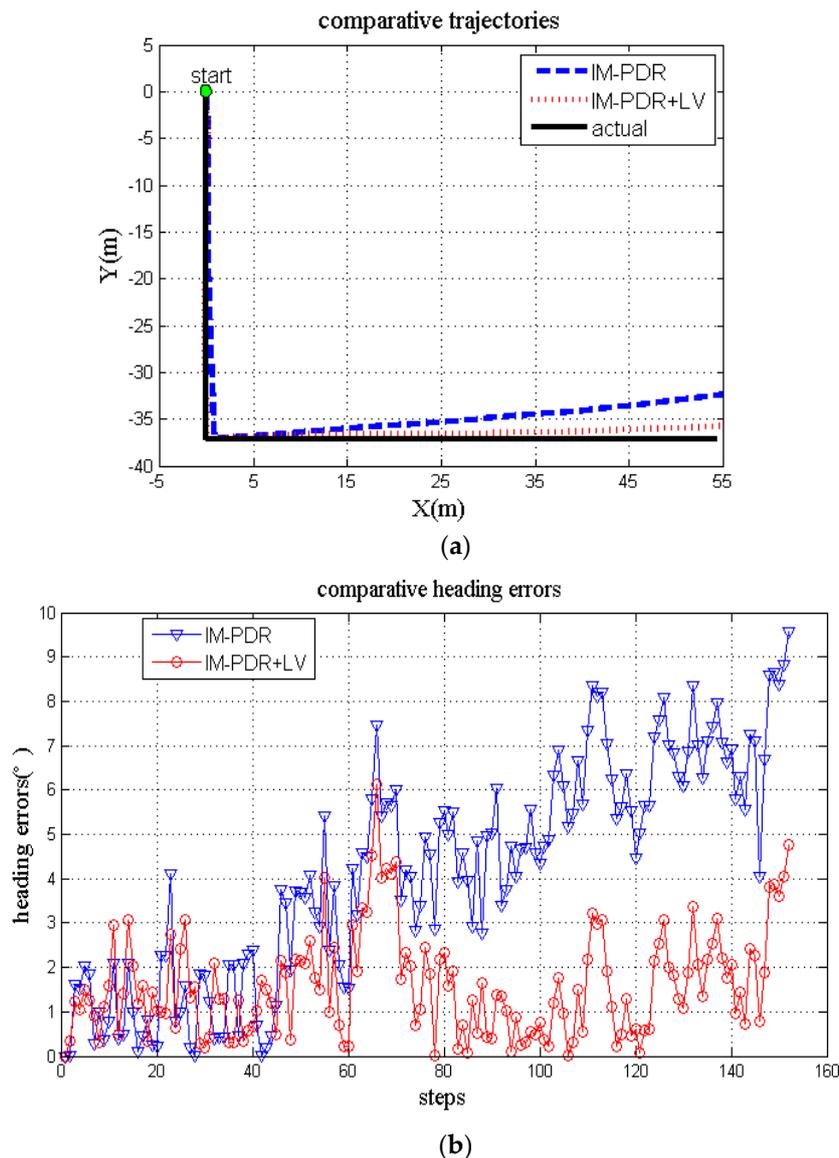


Figure 9. Results of straight line test: (a) comparative trajectories; (b) comparative heading errors.



**Figure 10.** Results of L-shape test: (a) comparative trajectories; (b) comparative heading errors.

As shown in Figures 9 and 10, for Route 1, the average heading error of the LV algorithm was  $1.06^\circ$ , which was 68% higher than that of the improved PDR ( $3.33^\circ$ ), and the maximal positioning error of the LV algorithm was 0.52 m, which was 80% less than that of the improved PDR (2.6 m). For Route 2, the average heading error of the LV algorithm proposed in this paper was  $4.75^\circ$ , which was 51.3% less than the average heading error of the improved PDR ( $9.57^\circ$ ) and the maximal positioning error of the LV algorithm was 1.53 m, which was 68.3% less than that of the improved PDR (1.82 m). In Tables 5 and 6, the results show that the LV algorithm can significantly improve the accuracy of the improved PDR algorithm.

**Table 5.** Average heading errors of improved PDR and LV algorithm based on the improved PDR.

No.	IM_PDR (°)	IM_PDR+LV (°)
1	3.33	1.06
2	9.57	4.75

**Table 6.** Maximal positioning errors of improved PDR and LV algorithm based on the improved PDR.

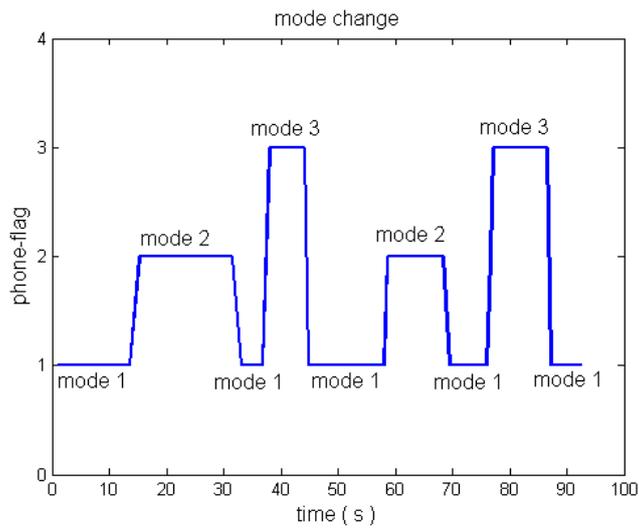
No.	IM_PDR (m)	IM_PDR+LV (m)
1	2.60	0.52
2	4.82	1.53

### 3.2.2. Experiments Based on Smartphone Multi-Mode

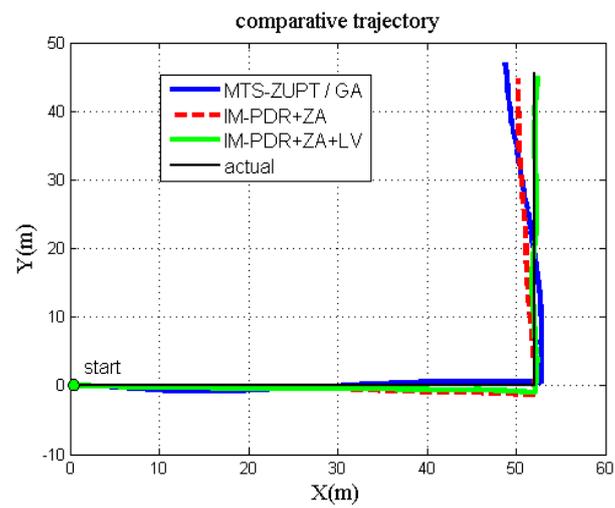
For multi-mode smartphone-based navigation, the heading errors introduced by the mode switching cause obvious positioning errors. Thus, the overall correction method based on the lateral velocity limitation cannot improve the navigation accuracy, and the results of IM\_PDR and IM\_PDR+LV become impractical. However, combining the ZA and LV methods to correct the errors of PDR can improve the accuracy of pedestrian navigation. Thus, in order to verify the proposed novel PDR algorithms (IM\_PDR+ZA+LV), three methods were used as control groups as follows:

- (1) MTS-ZUPT/GA: PDR algorithm based on Middle Time Simulated-Zero Velocity Update and Gravity Assisted methods proposed in Reference [24].
- (2) IM\_PDR+ZA: improved PDR combined with ZA algorithm.
- (3) IM\_PDR+ZA+LV: improved PDR combined with ZA + LV algorithm.

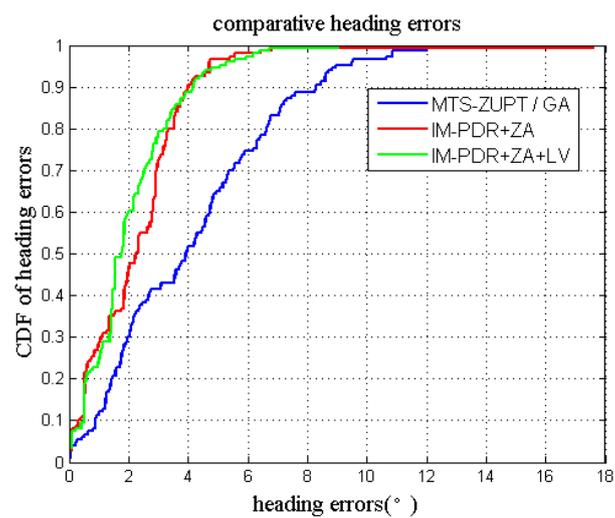
We performed two different experiments. The first involved walking along an L shape for 97.6 m. There were eight smartphone mode changes during the entire journey, as shown in Figure 11a. The second involved walking along a rectangular path with a total length of 199.4m. There were also eight smartphone mode changes in the entire trajectory, as shown in Figure 12a. The distance of the experiment was limited by the experimental area. The positions were all initialized as (0, 0) in an  $xy$ -coordinate system with north at the top. As shown in Figures 11b and 12b, the black line represents the actual trajectory in the experiment, and the blue line represents the result of the PDR algorithm proposed in Reference [24]. The red line represents the result of the improved PDR combined with the ZA algorithm, while the green line represents the result of the improved PDR combined with the ZA + LV algorithm. In Figures 11c and 12c, the CDF of heading errors means the cumulative distribution of the absolute heading errors.



(a)

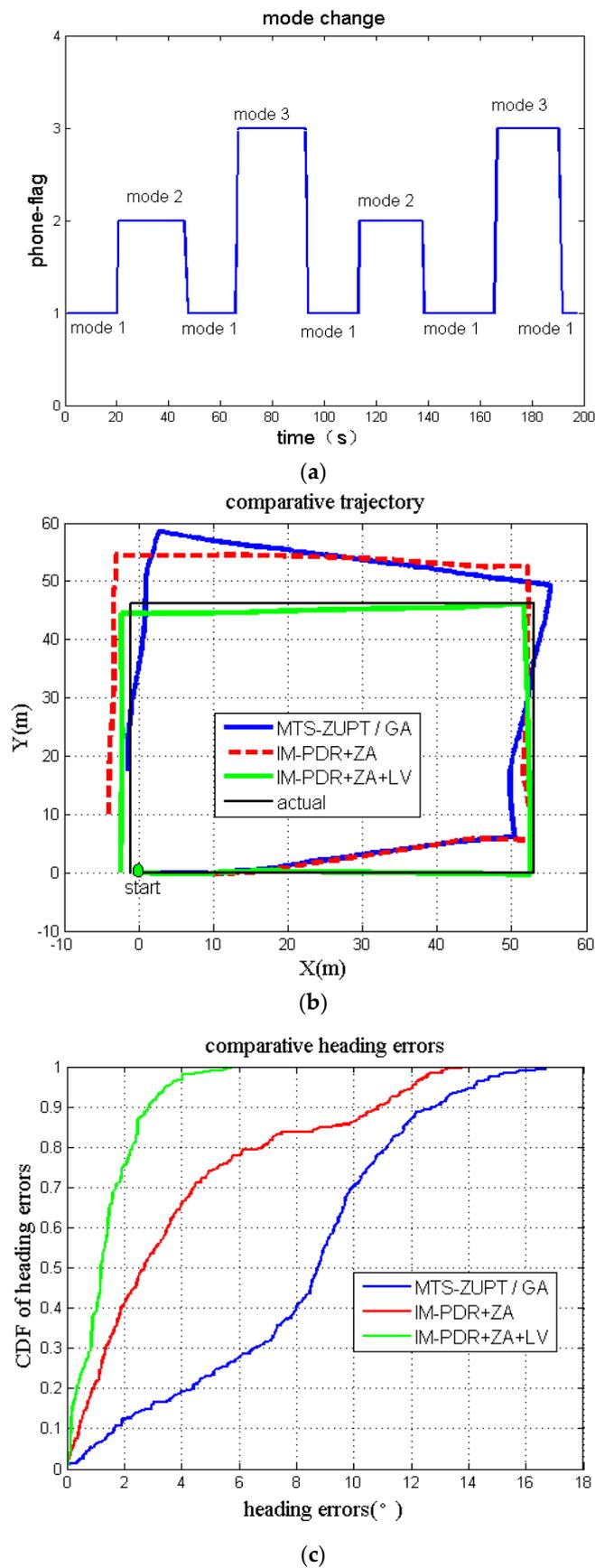


(b)



(c)

**Figure 11.** Results of L-shape taste: (a) mode change results; (b) comparative trajectories; (c) comparative heading errors.



**Figure 12.** Results of rectangular test: (a) mode change results; (b) comparative trajectories; (c) comparative heading errors.

The result of the MTS-ZUPT/GA algorithm proposed in Reference [24] introduced significant errors due to the inaccurate step detection, constant step length, and weak constraints of heading correction. The maximum positioning errors of Route 1 and Route 2 were 3.52 m and 17.6 m, respectively, and the average heading errors of Route 1 and Route 2 were 4.15° and 8.02°, respectively. When we adopted the ZA algorithm into the improved PDR, the problem of the large heading error caused by the mode switching could be resolved. However, the heading error introduced by a slight shaking of the handheld smartphone was not corrected. The maximum positioning errors of Route 1 and Route 2 were 1.95 m and 10.05 m, respectively, and the average heading errors of Route 1 and Route 2 were 2.29° and 3.93°, respectively. When the LV + ZA algorithm based on the improved PDR was adopted, the above problems could be effectively mitigated. The walking trajectory was very close to the actual route. The maximum errors of Route 1 and Route 2 were 0.64 m and 2.44 m, which account for 0.6% and 1.2% of the total path length, respectively, and the average heading errors of Route 1 and Route 2 were 2.06° and 1.40°, respectively, as shown in Tables 7 and 8. In summary, the novel algorithm proposed in this paper is superior to the MTS-ZUPT/GA algorithm proposed in Reference [24].

**Table 7.** Average heading errors of different approaches.

No.	MTS-ZUPT/GA (°)	IM_PDR+ZA (°)	IM_PDR+ZA+LV (°)
1	4.15	2.29	2.06
2	8.02	3.93	1.40

**Table 8.** Maximal positioning errors of different approaches.

No.	MTS-ZUPT/GA (m)	IM_PDR+ZA (m)	IM_PDR+ZA+LV (m)
1	3.52	1.95	0.64
2	17.6	10.5	2.44

#### 4. Discussion

The proposed pedestrian dead reckoning algorithm based on smartphone mode recognition uses only the built-in sensor of a smartphone. In comparison to Wi-Fi/UWB-based wireless indoor positioning technologies and the MEMS-based PDR algorithms, this algorithm possesses advantages of low-cost deployment and easy data acquisition. The algorithm can be applied in areas where wireless positioning signals are blocked or not available. The step detection method proposed in this paper has capabilities to not only detect the user's step frequency, but also to locate the short zero velocity moment in the pedestrian navigation process accurately, which underpins the velocity correction. The proposed step length estimation method has high viability. The step length can be accurately estimated for different users, and the error is obviously smaller than the commonly used nonlinear step length estimation method in Reference [23]. Aiming at smartphone applications in multiple modes, a multi-mode intelligent recognition method based on a neural network was studied, which can accurately recognize three commonly used smartphone modes, including the normal mode, the phone mode, and the swinging-hand mode. To demonstrate the effectiveness of the heading correction method based on zero angular velocity and the overall correction method based on lateral velocity limitation, we performed two types of experiments. Firstly, when the smartphone mode was fixed, the LV algorithm was added to the improved PDR algorithm in order to correct the heading angle errors caused by the shaking of the handheld smartphone with an average heading error 65.65% less than that of the improved PDR algorithm. Secondly, in the case of the multi-mode environment, the novel pedestrian dead reckoning algorithm proposed was adopted. Its maximum errors accounted for 0.6–1.2% of the total path length, and the average heading errors were less than the results of the PDR algorithm proposed in Reference [24].

However, the proposed algorithm also has a few limitations. Firstly, since there is no external sensor assistance, the initial heading angle can only be configured manually. The magnetic information could be considered as the initial heading estimation. Secondly, three commonly used smartphone modes were analyzed, i.e., the normal mode, the phone mode, and the swinging-hand mode. More smartphone modes, such as those involving the phone in pockets and bags, need to be further studied using similar methodologies. Finally, this algorithm focuses on the normal walking state. Other pedestrian motion states such as standing, running, and climbing up and down stairs can be studied in the future to construct a more viable pedestrian positioning and navigation algorithm.

## 5. Conclusions

In this paper, we proposed a novel pedestrian dead reckoning algorithm with the built-in inertial sensors for multi-mode smartphone applications. Concerning the inaccuracy of the traditional nonlinear method, a novel step length estimation method assisted by neural network and differential GPS was proposed. In addition, in order to improve the viability of the smartphone-based navigation, we proposed a multi-mode intelligent recognition algorithm using a tailored neural network to effectively distinguish different motion modes of the smartphone, achieving an average multi-mode recognition accuracy of 99.58%. Heading errors caused by either the mode switching or a slight shaking of the handheld smartphone could be well corrected using the proposed heading correction method based on zero angular velocity and the overall correction method based on lateral velocity limitation. By combining the ZA and LV methods in the improved PDR, the maximum positioning errors could be improved to be only 0.6–1.2% of the total path length, which is remarkably better than the positioning accuracy of the PDR algorithm studied in Reference [24]. The algorithm proposed here can be applied to smartphones in the normal mode, the phone mode, and the swinging-hand mode, which dramatically enhances the user experience and, thus, has high value in real applications in the future.

**Author Contributions:** Conceptualization, Z.X. and L.X.; methodology, L.X. and Z.X.; validation, L.X., Z.W., and Y.D.; software, L.X.; writing—original draft, L.X. and Z.X.; writing—review and editing, J.L., Z.W., and Y.D.

**Funding:** This work was partially supported by the National Natural Science Foundation of China (Grant No. 61673208, 61703208, 61873125, 61533008, 61533009), the Postgraduate Research & Practice Innovation Program of Jiangsu Province (KYCX18\_0302), the advanced research project of the equipment development (30102080101), the “333 project” in Jiangsu Province (Grant No. BRA2016405), the Scientific Research Foundation for the Selected Returned Overseas Chinese Scholars (Grant No. 2016), the Foundation Research Project of Jiangsu Province (The Natural Science Fund of Jiangsu Province, Grant No. BK20181291, BK20170815, BK20170767), the Aeronautic Science Foundation of China (Grant No. 20165552043, 20165852052), the Fundamental Research Funds for the Central Universities (Grant No. NP2018108, NZ2018002, NJ20170005, NP2017209, NZ2016104), the Foundation of Jiangsu Key Laboratory “Internet of Things and Control Technologies”, and the Priority Academic Program Development of Jiangsu Higher Education Institutions, Science, and Technology on Avionics Integration Laboratory.

**Acknowledgments:** The authors would like to acknowledge the College of Automation Engineering, Nanjing University of Aeronautics and Astronautics and the Navigation Research Center of NUAA for the continuous support in our research activity.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References

1. Chen, L.; Hu, H. IMU/GPS based pedestrian localization. In Proceedings of the 2012 4th Computer Science and Electronic Engineering Conference, Colchester, UK, 12–13 September 2012; pp. 23–28.
2. Liu, Y.; Cai, T.; Yang, H.; Liu, C.; Song, J.; Yu, M. The Pedestrian Integrated Navigation System with micro IMU/GPS/magnetometer/barometric altimeter. *Gyroscope Navig.* **2016**, *7*, 29–38. [[CrossRef](#)]
3. Zhang, P.; Zhao, Q.; Li, Y.; Niu, X.J.; Liu, J. PDR/WiFi fingerprinting/magnetic matching-based indoor navigation method for smartphones. *J. Geomat.* **2016**, *41*, 29–32.
4. Nan, Z.; Zhao, H.; Feng, W.; Wang, Z. A novel particle filter approach for indoor positioning by fusing WiFi and inertial sensors. *Chin. J. Aeronaut.* **2015**, *28*, 1725–1734.

5. Chen, J.; Ou, G.; Peng, A.; Zheng, L.; Shi, J. An INS/WiFi indoor localization system based on the Weighted Least Squares. *Sensors* **2018**, *18*, 1458. [[CrossRef](#)] [[PubMed](#)]
6. Fan, Q.; Sun, B.; Sun, Y. Performance Enhancement of MEMS-Based INS/UWB integration for indoor navigation applications. *IEEE Sens.* **2017**, *17*, 3116–3130. [[CrossRef](#)]
7. Wang, Y.; Li, X. The IMU/uwb fusion positioning algorithm based on a particle filter. *Int. J. Geo-Inf.* **2017**, *6*, 235.
8. Zhao, Q.; Zhang, B.X.; Lyu, S.C.; Zhang, H.; Sun, D.; Li, G.; Feng, W. A CNN-SIFT hybrid pedestrian navigation method based on first-person vision. *Remote Sens.* **2018**, *10*, 1229. [[CrossRef](#)]
9. Xu, H.; Wu, M.; Li, P.; Zhu, F.; Wang, R. An RFID indoor positioning algorithm based on support vector regression. *Sensors* **2018**, *18*, 1504. [[CrossRef](#)] [[PubMed](#)]
10. Wu, X.; Shen, R.; Fu, L.; Tian, X.; Liu, P.; Wang, X. Ibill: Using ibeacon and inertial sensors for accurate indoor localization in large open areas. *IEEE Access* **2017**, *5*, 14589–14599. [[CrossRef](#)]
11. Xiong, Z.; Chen, J.H.; Wang, R.; Liu, J.Y. A new dynamic vector formed information sharing algorithm in federated filter. *Aerosp. Sci. Technol.* **2013**, *29*, 37–46. [[CrossRef](#)]
12. Zhang, L.; Liu, J.Y.; Lai, J.Z.; Xiong, Z. Performance analysis of adaptive neuro fuzzy inference system control for MEMS navigation system. *Math. Probl. Eng.* **2014**, *2014*, 1–7. [[CrossRef](#)]
13. Foxlin, E. Pedestrian tracking with shoe-mounted inertial sensors. *IEEE Comput. Graph. Appl.* **2005**, *25*, 38–46. [[CrossRef](#)] [[PubMed](#)]
14. Zwiener, J.; Lorenz, A.; Jager, R. Algorithm and system development for seamless out-/indoor navigation with lowcost GNSS and foot-mounted MEMS sensor. In Proceedings of the International Scientific Congress and Exhibition Interexpo Geo-Siberia, Novosibirsk, Russia, 19–21 April 2014; pp. 83–97.
15. Nilsson, J.; Zachariah, D.; Skog, I.; Händel, P. Cooperative localization by dual foot-mounted inertial sensors and inter-agent ranging. *EURASIP J. Adv. Signal Process.* **2013**, *2013*, 164. [[CrossRef](#)]
16. Kourogi, M.; Ishikawa, T.; Kurata, T. A method of pedestrian dead reckoning using action recognition. In Proceedings of the 2010 IEEE/ION Position Location & Navigation Symposium, Indian Wells, CA, USA, 4–6 May 2010; IEEE: Indian Wells, CA, USA, 2010.
17. Kuang, J.; Niu, X.; Chen, X. robust pedestrian dead reckoning based on MEMS-IMU for smartphones. *Sensors* **2018**, *18*, 1391. [[CrossRef](#)] [[PubMed](#)]
18. Ren, M.; Pan, K.; Liu, Y.; Guo, H.; Zhang, X.; Wang, P. A novel pedestrian navigation algorithm for a foot-mounted inertial-sensor-based system. *Sensors* **2016**, *16*, 139. [[CrossRef](#)] [[PubMed](#)]
19. Zhou, X.C.; Chen, J.X.; Dong, Y.; Lu, X.R.; Cui, J.W.; Zheng, B.Y. Pedestrian navigation with foot-mounted inertial sensors in wearable body area networks. In Proceedings of the Signal & Information Processing Association Summit & Conference, Hong Kong, China, 16–19 December 2015; IEEE: Siem Reap, Cambodia, 2015.
20. Bilke, A.; Sieck, J. Using the magnetic field for indoor localisation on a mobile phone. In *Progress in Location-Based Services; Lecture Notes in Geoinformation and Cartography*; Krisp, J.M., Ed.; Springer: Berlin/Heidelberg, Germany, 2013; pp. 195–208.
21. Li, Y. *Research and Implementation of Indoor Navigation Based on Smart Phone*; Beijing University of Posts and Telecommunications: Beijing, China, 2017.
22. Kang, W.; Han, Y. SmartPDR: Smartphone-Based Pedestrian Dead Reckoning for Indoor Localization. *IEEE Sens. J.* **2015**, *15*, 2906–2916. [[CrossRef](#)]
23. Wang, B.; Liu, X.; Yu, B.; Jia, R.; Gan, X. Pedestrian Dead Reckoning Based on Motion Mode Recognition Using a Smartphone. *Sensors* **2018**, *18*, 1811. [[CrossRef](#)] [[PubMed](#)]
24. Zeng, Q.; Zeng, S.; Liu, J.; Meng, Q.; Chen, R.; Huang, H. Smartphone Heading Correction Based on Gravity Assisted and Middle Time Simulated-Zero Velocity Update Method. *Sensors* **2018**, *18*, 3349. [[CrossRef](#)] [[PubMed](#)]

