technologies

MDPI

Article

Enhanced Pedestrian Navigation with Wearable IMU: Forward–Backward Navigation and RTS Smoothing Techniques

Yilei Shen [1,2], Yiqing Yao [1,2,*], Chenxi Yang [1] and Xiang Xu [3]

[1] School of Instrument Science and Engineering, Southeast University, Nanjing 210096, China; tzsylsyl@163.com (Y.S.); chenxiyang@seu.edu.cn (C.Y.)
[2] The Key Laboratory of Micro-Inertial Instrument and Advanced Navigation Technology, Ministry of Education, Nanjing 210096, China
[3] School of Automation, Nanjing University of Science and Technology, Nanjing 210094, China; hsianghsu@163.com
* Correspondence: yucia@sina.com

Abstract

Accurate and reliable pedestrian positioning service is essential for providing Indoor Location-Based Services (ILBSs). Zero-Velocity Update (ZUPT)-aided Strapdown Inertial Navigation System (SINS) based on foot-mounted wearable Inertial Measurement Units (IMUs) has shown great performance in pedestrian navigation systems. Though the velocity errors will be corrected once zero-velocity measurement is available, the navigation system errors accumulated during measurement outages are yet to be further optimized by utilizing historical data during both stance and swing phases of pedestrian gait. Thus, in this paper, a novel Forward–Backward navigation and Rauch–Tung–Striebel smoothing (FB-RTS) navigation scheme is proposed. First, to efficiently re-estimate past system state and reduce accumulated navigation error once zero-velocity measurement is available, both the forward and backward integration method and the corresponding error equations are constructed. Second, to further improve navigation accuracy and reliability by exploiting historical observation information, both backward and forward RTS algorithms are established, where the system model and observation model are built under the output correction mode. Finally, both navigation results are combined to achieve the final estimation of attitude and velocity, where the position is recalculated by the optimized data. Through simulation experiments and two sets of field tests, the FB-RTS algorithm demonstrated superior performance in reducing navigation errors and smoothing pedestrian trajectories compared to traditional ZUPT method and both the FB and the RTS method, whose advantage becomes more pronounced over longer navigation periods than the traditional methods, offering a robust solution for positioning applications in smart buildings, indoor wayfinding, and emergency response operations.

Keywords: pedestrian navigation; wearable IMU; forward–backward (FB) smoother; Rauch–Tung–Striebel (RTS) smoother; zero-velocity update

check for updates

Academic Editor: Alessandro Tognetti

Received: 17 April 2025
Revised: 19 June 2025
Accepted: 4 July 2025
Published: 9 July 2025

Citation: Shen, Y.; Yao, Y.; Yang, C.; Xu, X. Enhanced Pedestrian Navigation with Wearable IMU: Forward–Backward Navigation and RTS Smoothing Techniques. *Technologies* **2025**, *13*, 296. https://doi.org/10.3390/technologies13070296

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

The growing demand for Indoor Location-Based Services (ILBS) has catalyzed a quest for accurate and reliable indoor positioning systems. In fact, the global indoor navigation and positioning market is projected to grow from $11.9 billion in 2024 and reach $31.4 billion by 2029 [1]. The unique challenges posed by indoor environments, where Global Navigation

Satellite System (GNSS) signals are often obstructed or unavailable, have driven scholars and industry experts to delve deeply into the field of pedestrian navigation.

The pursuit of pedestrian navigation has been to harness the potential of wearable sensors, notably Micro Electro-Mechanical Systems Inertial Measurement Units (MEMS IMUs), to ascertain the precise location and trajectory of the pedestrian [2]. In inertial-sensor-based dead reckoning (DR) systems, wearable MEMS IMUs are typically affixed to the pedestrian's feet, and numerical integration based on Strapdown Inertial Navigation System (SINS) mechanization is performed to locate the pedestrian's position [3–6]. Such an approach offers a navigation system that is self-contained, robust against external interference, and unencumbered by the need for external infrastructure, making it suitable for applications such as indoor wayfinding, search and rescue missions, and emergency response scenarios.

Despite its high precision in short time spans, the performance of DR-based pedestrian navigation systems is often degraded by the rapid accumulation of errors due to sensor noise and biases inherent in MEMS IMUs [7,8]. The Zero-Velocity Update (ZUPT) algorithm was proposed by Foxlin to mitigate this problem in foot-mounted pedestrian navigation systems [9] and has been widely adopted. ZUPT leverages the periodic nature of human gait, specifically the stance phase when the foot is in contact with the ground, to provide discrete corrections to the navigation system. This approach assumes that the wearable IMU's velocity is zero during the stance phase, allowing for the calibration of the system and a reduction in the accumulated error.

However, the reliance on ZUPT introduces its own set of limitations. The traditional ZUPT is only effective during the stance phase of the gait cycle, leaving the system in a pure inertial state during the swing phase, where errors can propagate rapidly. Moreover, the discontinuous nature of ZUPT corrections can lead to abrupt changes in speed estimation.

In response to these challenges, various studies have proposed alternative solutions. By exploiting the symmetrical and complementary nature of human bipedal gait, where one lower limb is in the swing phase while the other is in the stance phase, a dual IMU solution is proposed to improve measurement integrity and navigation accuracy [10]. Building upon this concept, the capability of the dual IMU system was extended by incorporating an ultrasonic ranging sensor between both feet, offering a distance constraint between IMUs [11]. Due to the prevalence of straight, parallel, or orthogonally arranged corridors in most indoor structures, these corridors largely dictate the primary building axes. When pedestrians traverse linear paths aligned with these dominant directions, the Heuristic Drift Elimination (HDE) algorithm can be utilized to estimate gyroscope bias [12]. To address scenarios where pedestrians follow straight trajectories that deviate from the building's primary axes, the improved HDE (iHDE) algorithm was proposed [13]. While these approaches offer improvements over the traditional ZUPT method, they are primarily carried out by introducing additional sensors or relying on special assumptions, either increasing the cost and complexity of the pedestrian navigation system, or limiting the generalizability and robustness of these systems, making them suitable only for particular scenarios.

In recent years, AI-based methods have gained significant attention in the field of pedestrian navigation, offering promising solutions to improve the accuracy and robustness of inertial navigation systems. These methods can be broadly categorized into two main approaches: AI-driven Zero-Velocity Detectors and AI-driven End-to-End Navigation Algorithms.

AI-driven Zero-Velocity Detectors leverage machine learning and deep learning to adaptively identify stance phases during gait cycles, addressing challenges such as varying pedestrian speeds and environmental noise. Wagstaff et al. [14] uses a long short-term memory (LSTM) recurrent neural network to classify zero-velocity events from raw inertial data, achieving higher accuracies than existing detectors for walking, running, and stair-

climbing motions. Chen et al. [15] proposed a novel detector based on a contrastive neural network, which is trained by comparing with the anchor data which consists of the known static inertial data from the period of initial alignment. This approach can detect the zero-velocity event adaptively and accurately and has better robustness, particularly in complex motion scenarios.

AI-driven End-to-End Navigation Algorithms aim to directly map raw IMU data to pedestrian trajectories, bypassing traditional SINS mechanization, while capturing long-range dependencies in IMU sequences. Chen et al. [16] formulated the inertial odometry task as an optimization problem, and explored the use of deep recurrent neural networks for estimating the displacement of a pedestrian over a specified time window. By training the proposed inertial odometry neural network (IONet) using inertial measurements and ground truth displacement data, both motion characteristics and systematic error drift can be learned. However, drift still occurs in estimating a pose containing both a position and an orientation because the integration of pose changes is also required. Thus, Kim et al. [17] proposed the Extended IONet that combines a 9-Axis IONet and PoseTuningNet to improve the accuracy of trajectory tracking by compensating for the drift problem of the 6-Axis IONet. Experiments conducted using the Oxford Inertial Odometry Dataset have shown superior performance of the proposed approach.

Despite these advances, AI-driven methods in its current form still face several challenges. Current AI models often struggle with generalization across different hardware, and some with limited accuracy. Certain methods still employ SINS algorithm as an integral part of the navigation approach [18]. Additionally, traditional methods provide transparent error models, which are essential for safety-critical applications. Real-time deployment remains challenging due to the high computational cost of deep learning architectures. Therefore, enhancing the accuracy of traditional navigation methods remains essential.

Besides the aforementioned solutions, optimizing the dead reckoning process of the pedestrian navigation system is a more direct and efficient way to enhance the positioning accuracy. When pedestrian gait transitions from swing phase to stance phase, the ZUPT algorithm only corrects the navigation system state during the stance phase, failing to re-estimate the system state, especially velocity errors from the preceding swing phase using the observations made in current stance phase. This presents an opportunity to leverage smoothing techniques to refine past estimates of attitude and velocity, thereby reducing accumulated position errors. Such smoothing techniques are frequently discussed in high-precision SINS/GNSS integrated navigation systems [19,20], where the Forward–Backward (FB) smoother and Rauch–Tung–Striebel (RTS) smoother are dominant algorithms [21,22]. The RTS smoother has been employed in pedestrian navigation systems based on low-precision SINS, either as a post-processing method [23] or in a step-wise manner [24]. However, the smoothing mechanisms and fusion framework have yet to be fully established, and the segmentation methodology remains to be discussed.

Thus, in this paper, a novel Forward–Backward navigation and RTS smoothing (FB-RTS) navigation scheme is proposed, designed to enhance the navigation accuracy of MIMU-based pedestrian navigation systems without the need for additional sensors or specialized conditions. First, both forward and backward integration method and the corresponding error equations are constructed. Second, both a backward and forward RTS algorithm is established, where the system model and observation model are built under the output correction mode. Finally, both navigation results are combined to achieve the final estimation of attitude and velocity, where the position is recalculated by the optimized data. This novel approach utilizes the historical navigation and filtering data stored during the swing phase to re-estimate and optimize the navigation results, harnessing the full potential of the gait cycle for improved accuracy and reliability.

The structure of this paper is organized as follows: Section 2 provides an overview of the traditional pedestrian navigation scheme. Section 3 details the proposed FB-RTS navigation scheme, including its theoretical foundations and implementation details. Section 4 presents extensive experimental validations that demonstrate the superior performance of the proposed algorithm, while the conclusions are drawn in Section 5.

2. Traditional Pedestrian Navigation Scheme

In the field of pedestrian navigation, the quest for precision has led to the development of many algorithms that harness the natural rhythm of human movement. Traditional algorithms based on the Strapdown Inertial Navigation System (SINS), the Kalman Filter and the Zero-Velocity Update (ZUPT) algorithm, have been fundamental in this pursuit, offering a solid foundation upon which more advanced methods, such as the FB-RTS algorithm, can be developed and refined.

2.1. SINS Mechanical Arrangement

The SINS mechanical arrangement is the foundation of inertial navigation, designed to operate independently of external references, making it an ideal choice for environments where GNSS signals may be unreliable or unavailable.

The attitude, velocity, and position update equations for the SINS are as follows:

$$\dot{\boldsymbol{C}}_b^n = \boldsymbol{C}_b^n \left[\boldsymbol{\omega}_{nb}^b \times\right] \tag{1}$$

$$\dot{\boldsymbol{V}}^n = \boldsymbol{C}_b^n \boldsymbol{f}^b + \boldsymbol{g}^n \tag{2}$$

$$\dot{L} = V_N / (R_M + h) \tag{3}$$

$$\dot{\lambda} = V_E / [(R_N + h) cosL] \tag{4}$$

$$\dot{h} = V_U \tag{5}$$

where $[\cdot\times]$ is the skew-symmetric matrix notation. Based on the aforementioned SINS attitude, velocity, and position update equations, the following SINS error equations can be derived:

$$\dot{\boldsymbol{\phi}}^n = \boldsymbol{\phi}^n \times \boldsymbol{\omega}_{in}^n - \delta\boldsymbol{\omega}_{in}^n - \boldsymbol{\varepsilon}^n \tag{6}$$

$$\delta\dot{\boldsymbol{V}}^n = -\boldsymbol{\phi}^n \times \boldsymbol{C}_b^n \boldsymbol{f}^b + \nabla^n \tag{7}$$

$$\delta\dot{L} = \frac{\delta V_N}{R_M + h} - \frac{\delta h \cdot V_N}{(R_M + h)^2} \tag{8}$$

$$\delta\dot{\lambda} = \frac{\delta V_E}{R_N + h} secL + \frac{\delta L \cdot V_E}{R_N + h} tanLsecL - \frac{\delta h \cdot V_E}{(R_N + h)^2} secL \tag{9}$$

$$\delta\dot{h} = \delta V_U \tag{10}$$

where $\boldsymbol{\phi}^n$ and $\delta\boldsymbol{V}^n$ represents the misalignment angle and the velocity error in the navigation frame, respectively, and $\boldsymbol{\varepsilon}^n$ and ∇^n are the bias errors from the gyroscopes and accelerometers in the navigation frame, respectively.

2.2. Kalman Filter and ZUPT

When using low-precision low-cost MEMS IMUs as inertial sensors, SINS suffers from rapid error accumulation. By constructing proper Kalman Filter and applying the ZUPT algorithm during stance phases, the usability of the navigation results of MEMS IMUs has been greatly improved.

Based on the SINS error equations described in Section 2.1, the following Error-State Kalman Filter (ESKF) can be constructed:

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{\phi}^n & \delta \boldsymbol{V}^n & \delta \boldsymbol{P} & \boldsymbol{\varepsilon}^b & \nabla^b \end{bmatrix}^T \tag{11}$$

$$\boldsymbol{X}_k = \boldsymbol{\Phi}_{k,k-1}\boldsymbol{X}_{k-1} + \boldsymbol{\Gamma}_{k-1}\boldsymbol{W}_{k-1} \tag{12}$$

$$\boldsymbol{Z}_k = \boldsymbol{H}_k\boldsymbol{X}_k + \boldsymbol{V}_k \tag{13}$$

In order to reliably detect the zero-velocity state of the foot-mounted IMU, the Stance Hypothesis Optimal Detection (SHOE) algorithm is employed. The algorithm considers both acceleration and angular velocity information, making it more accurate in detecting the zero-velocity state during pedestrian walking [25].

After obtaining zero-velocity state information, the ZUPT algorithm can be carried out; the observation vector and observation matrix would be as follows:

$$\boldsymbol{Z}_k = \boldsymbol{v}_k - \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T = \boldsymbol{v}_k \tag{14}$$

$$\boldsymbol{H}_k = \begin{bmatrix} \boldsymbol{0}_{3\times3} & \boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \end{bmatrix} \tag{15}$$

3. FB-RTS Navigation Scheme

As position errors accumulated during the swing phase can hardly be corrected by subsequent zero-velocity measurements, it is crucial to apply smoothing techniques to enhance the navigation results of foot-mounted pedestrian navigation systems. While FB and RTS have demonstrated their effectiveness in high-precision SINS/GNSS integrated navigation systems, the smoothing mechanisms have not yet been fully established and verified in pedestrian navigation systems based on low-precision SINS, and several distinct differences must be taken into account. First, the periodic nature of pedestrian gait ensures that the system experiences short, intermittent, and relatively regular periods without zero-velocity measurements, rather than prolonged and random ones due to GNSS signal obstruction. Second, the widespread use of low-precision MEMS IMUs in foot-mounted pedestrian navigation systems leads to rapid error divergence. In output correction mode where error estimates of Kalman Filter are directly subtracted from the output of SINS, shortly after the navigation process begins, the system errors accumulate to a degree that no longer meets the linearization conditions required for the Kalman Filter, causing deviations in the corrected navigation results.

Thus, it is necessary to execute the algorithm in segments to reduce the time intervals of output correction, thus avoiding the aforementioned issue. A similar data segmentation technique is outlined in [19], where the velocity error covariance serves as a threshold. Additionally, a fixed time threshold is implemented to prevent the smoothing algorithm from exhibiting incorrect behavior. Although effective, these static thresholds necessitate prior knowledge and manual adjustment, which may not be applicable across various gait patterns and walking speeds. Instead, this paper employs the midpoint of the stance phase as the boundary to divide the IMU data series into small segments, and perform the proposed FB-RTS algorithm in each segment. The obtained final values of attitude, velocity, and position then serve as the initial values for navigation in the next segment. The stance phase is identified by filtering the SHOE results detailed in Section 2.2 to further avoid false segmentation. The midpoint is deliberately selected to maximize the duration available for both the forward and backward Kalman Filters to achieve convergence, ensuring optimal performance. This approach has the added benefit that the position update frequency of the

algorithm matches the pedestrian's stride frequency, which is identical with PDR algorithm, meeting the real-time requirements for most applications.

3.1. Backward Navigation

ZUPT traditionally corrects navigation system errors only during the stance phase of pedestrian gait, where the foot is in contact with the ground. During the swing phase, when the foot is off the ground, the system reverts to a pure inertial navigation state, leading to rapid error accumulation. The Forward–Backward (FB) navigation algorithm capitalizes on the raw navigation data throughout the entire swing phase, aiming to reduce velocity errors and improve the accuracy of the navigation system in the absence of zero-velocity measurements.

By storing IMU data and going through iterative processes in both forward and reverse directions, errors can be estimated more accurately, therefore improving the system's navigation accuracy. The forward navigation process has been detailed in Section 2.1, and the attitude, velocity, and position update equations for the SINS backward navigation are as follows:

$$\dot{\overleftarrow{\boldsymbol{C}}}_b^n = \overleftarrow{\boldsymbol{C}}_b^n \left[\overleftarrow{\boldsymbol{\omega}}_{nb}^b \times \right] \tag{16}$$

$$\dot{\overleftarrow{\boldsymbol{V}}}^n = \overleftarrow{\boldsymbol{C}}_b^n \overleftarrow{\boldsymbol{f}}^b + \overleftarrow{\boldsymbol{g}}^n \tag{17}$$

$$\dot{\overleftarrow{L}} = \overleftarrow{V}_N / (R_M + \overleftarrow{h}) \tag{18}$$

$$\dot{\overleftarrow{\lambda}} = \overleftarrow{V}_E / [(R_N + \overleftarrow{h}) cos \overleftarrow{L}] \tag{19}$$

$$\dot{\overleftarrow{h}} = \overleftarrow{V}_U \tag{20}$$

where '←' denotes variables used in the backward navigation process.

The update equations for forward and backward navigation are identical in form, yet there are some critical differences in the values of the variables. Ideally, the attitudes, velocities, and positions derived from both forward and backward navigation should be identical. However, since position is integrated from velocity, for the positions to be the same, i.e., $\overleftarrow{L} = L, \overleftarrow{\lambda} = \lambda, \overleftarrow{h} = h$, the velocities must be opposite, i.e., $\overleftarrow{\boldsymbol{V}}^n(t_k) = -\boldsymbol{V}^n(t_k)$. Similarly, for the attitudes to be the same, i.e., $\overleftarrow{\boldsymbol{C}}_b^n(t_k) = \boldsymbol{C}_b^n(t_k)$, the angular velocities must all be opposite, including both the Earth's rotational angular velocity and the angular velocities measured by gyroscopes, i.e., $\overleftarrow{\boldsymbol{\omega}}_{ib}^b(t_k) = -\boldsymbol{\omega}_{ib}^b(t_k)$, $\overleftarrow{\boldsymbol{\omega}}_{ie}^n(t_k) = -\boldsymbol{\omega}_{ie}^n(t_k)$, $\overleftarrow{\boldsymbol{\omega}}_{en}^n(t_k) = -\boldsymbol{\omega}_{en}^n(t_k)$. Since velocity is integrated from acceleration, and the velocity is already opposite to the forward process, the specific forces should be the same as in the forward process, i.e., $\overleftarrow{\boldsymbol{f}}^b(t_k) = \boldsymbol{f}^b(t_k)$.

Given that the update equations for forward and backward navigation are identical in form, it can be deduced that the error equations for backward navigation are also formally consistent with those of forward navigation:

$$\dot{\overleftarrow{\boldsymbol{\phi}}}^n = \overleftarrow{\boldsymbol{\phi}}^n \times \overleftarrow{\boldsymbol{\omega}}_{in}^n - \delta \overleftarrow{\boldsymbol{\omega}}_{in}^n - \overleftarrow{\boldsymbol{\varepsilon}}^n \tag{21}$$

$$\delta \dot{\overleftarrow{\boldsymbol{V}}}^n = - \overleftarrow{\boldsymbol{\phi}}^n \times \overleftarrow{\boldsymbol{C}}_b^n \overleftarrow{\boldsymbol{f}}^b + \overleftarrow{\nabla}^n \tag{22}$$

$$\delta \dot{\overleftarrow{L}} = \frac{\delta \overleftarrow{V}_N}{R_M + \overleftarrow{h}} - \frac{\delta h \cdot \overleftarrow{V}_N}{(R_M + \overleftarrow{h})^2} \tag{23}$$

$$\dot{\overleftarrow{\delta\lambda}} = \frac{\overleftarrow{\delta V_E}}{\overleftarrow{R_N + h}} \overleftarrow{sec L} + \frac{\overleftarrow{\delta L \cdot V_E}}{\overleftarrow{R_N + h}} \overleftarrow{tan L} \overleftarrow{sec L} - \frac{\overleftarrow{\delta h \cdot V_E}}{\overleftarrow{(R_N + h)^2}} \overleftarrow{sec L} \tag{24}$$

$$\dot{\overleftarrow{\delta h}} = \overleftarrow{\delta V_U} \tag{25}$$

3.2. RTS Smoothing

The intermittent corrections provided by ZUPT can cause abrupt changes in navigation information as the system transits from the swing phase to the stance phase. The RTS (Rauch–Tung–Striebel) optimal fixed-interval smoothing algorithm promises to mitigate such problem [26]. RTS smoothing can reduce estimation error effectively, and the error curve obtained by RTS smoothing is smoother than that obtained by Kalman Filtering, as shown in Figure 1.

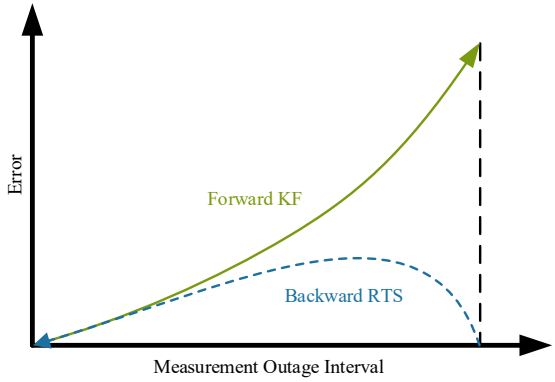


Figure 1. Computation procedure of RTS smoothing.

The forward Kalman Filter process and the backward RTS process are taken here as an example. After the forward filtering is completed, RTS smoothing is carried out from time N to time 0 with initial values of $\boldsymbol{X}^B_{N/N} = X^F_N$, $\boldsymbol{P}^B_{N/N} = P^F_N$, and the update process is as follows:

$$\boldsymbol{K}^B_{k/N} = \boldsymbol{P}^F_k \boldsymbol{\Phi}^T_{k+1,k} \left(\boldsymbol{P}^F_{k+1,k}\right)^{-1} \tag{26}$$

$$\boldsymbol{X}^B_{k/N} = \boldsymbol{X}^F_k + \boldsymbol{K}^B_{k/N} \left(\boldsymbol{X}^B_{k+1/N} - \boldsymbol{X}^F_{k+1,k}\right) \tag{27}$$

$$\boldsymbol{P}^B_{k/N} = \boldsymbol{P}^F_k + \boldsymbol{K}^B_{k/N} \left(\boldsymbol{P}^B_{k+1/N} - \boldsymbol{P}^F_{k+1,k}\right) \left(\boldsymbol{K}^B_{k/N}\right)^T \tag{28}$$

where the superscript F represents the variables of the forward filtering process, and the superscript B indicates the variables of the backward RTS process.

3.3. Forward–Backward Fusion

The Forward–Backward fusion framework utilizes the stored forward and backward navigation results and fuses them based on the covariance matrix of the Kalman Filter or the RTS smoothing algorithm. The idea is that the errors of both the forward and backward navigation accumulate over time, but the error accumulation processes of the two are opposite in time. Thus, at the same point in time, the forward and backward navigation errors exhibit different characteristics, as shown in Figure 2. Therefore, fusing the forward and backward navigation would yield better results.

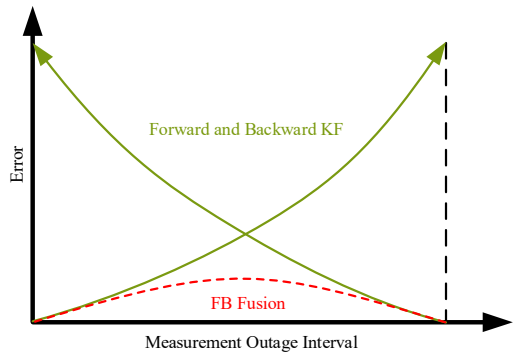


Figure 2. Computation procedure of FB navigation and fusion.

Let the state vectors obtained from the forward and backward navigation processes be $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$, with corresponding covariance matrices $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$. Then, the fused state estimate of the Forward–Backward navigation would be the following linear combination of the state vectors:

$$\boldsymbol{P}_g = \left(\boldsymbol{P}_1^{-1} + \boldsymbol{P}_2^{-1}\right)^{-1} \tag{29}$$

$$\boldsymbol{X}_g = \boldsymbol{P}_g\left(\boldsymbol{P}_1^{-1}\boldsymbol{X}_1 + \boldsymbol{P}_2^{-1}\boldsymbol{X}_2\right) \tag{30}$$

This fused state estimate can then be used to correct the navigation results. The corrected results are then used as reference values in the re-forward navigation process, yielding the final navigation results.

The proposed FB-RTS algorithm leverages the historical data already stored to optimize the navigation system. The architecture of the proposed FB-RTS algorithm is described in Figure 3. The two RTS modules operate in parallel rather than in a cascaded manner, ensuring its optimality by maintaining independence between forward and backward filtering processes. Song et al. demonstrated that when local estimators satisfy independence assumptions (e.g., uncorrelated process and measurement noises), their weighted fusion can achieve global optimality [27]. With the continuous advancements in computer technology, the processing power and storage capacity of computers and microcontrollers have seen significant improvements. This progress has made the implementation of the FB-RTS algorithm not only feasible but also cost-effective, with minimal additional expenditure.

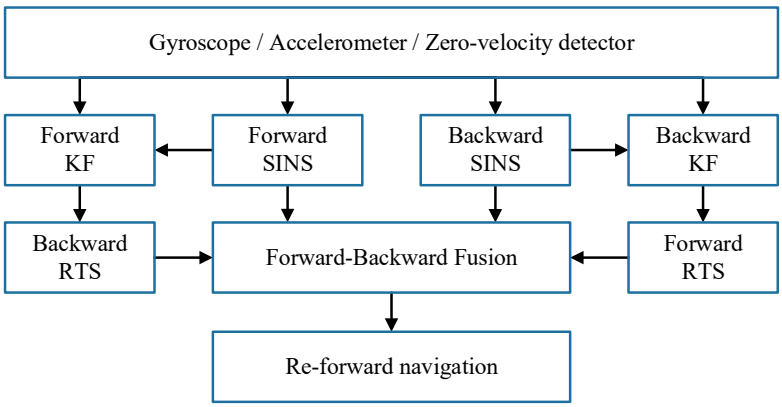


Figure 3. The architecture of the proposed FB-RTS algorithm.

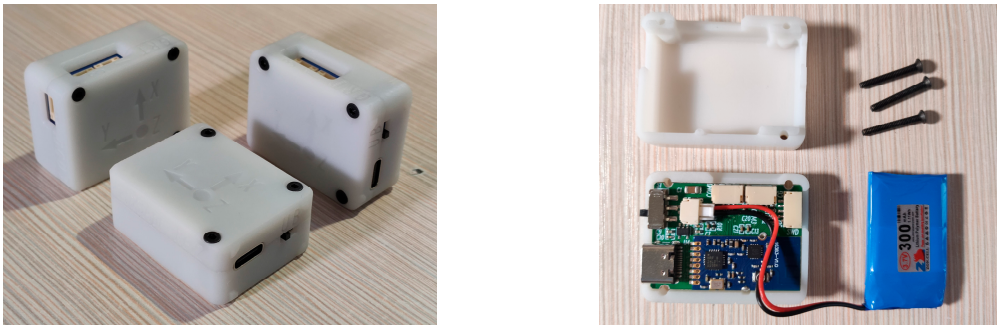
4. Experimental Validation

To thoroughly evaluate the effectiveness of the proposed FB-RTS algorithm, this study employs a comprehensive validation approach encompassing both simulation experiments and two sets of field tests, both indoor and outdoor. The simulation experiments are

designed to assess the algorithm's capability to enhance navigation accuracy and mitigate heading drift during periods of motion, specifically in the absence of zero-velocity updates. Detailed error curves, especially velocity error curve, can then be examined, which is not easily obtainable in field tests. The indoor and outdoor field tests are designed to represent real-world complexities and evaluate the algorithm's performance in terms of pedestrian localization accuracy and system reliability over extended periods and along comprehensive trajectories.

4.1. Experimental Overview

In this study, both field tests were conducted using self-developed wireless IMU sensor nodes, which are shown in Figure 4. Due to their relatively small size and low weight, they would not obstruct the normal gait of the subject. The IMU sensor employed in the sensor node was the ICM20948, which has been characterized through rigorous testing to provide the parameters shown in Table 1:



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

Figure 4. Self-developed wireless IMU sensor nodes. (**a**) Node exterior; (**b**) node interior.

Table 1. IMU error characteristics of the IMU sensor nodes used in field tests.

Sensor Type	White Noise	Bias Instability	Sampling Rate
Gyroscope	$0.5°/\sqrt{\text{h}}$	13°/h	125 Hz
Accelerometer	$168\mu g/\sqrt{\text{Hz}}$	162 μg	125 Hz

These parameters ensure that the IMU sensor node delivers data that is representative of typical low-cost MIMU devices, suitable for validating the real-life potential of the proposed navigation algorithm.

During field tests, a single IMU node is attached to the subjects' feet using Velcro strap, specifically at the dorsal foot position. The IMU is oriented such that the y-axis roughly aligns with the direction of tiptoe, and the z-axis points approximately towards upward direction, as shown in Figure 5. The IMU is worn on the dorsal foot due to the known fact that it can mitigate the vibrations and acceleration peaks caused by heel strikes. The initial yaw angle of the SINS algorithm is set by minimizing the RMSE between the trajectory segment generated by the SINS algorithm (from the starting point to a point 3 m away) and the corresponding reference trajectory [28].

The FB-RTS algorithm operates in segments bounded by stance-phase midpoints, allowing the algorithm to operate in real time on a Lenovo laptop with AMD Ryzen 6800H processor, sourced in Nanjing, China. While optimized results for a segment incur ~13.7 ms latency post-segment, real-time forward navigation results are always available with ~0.05 ms latency using prior optimized states. For typical 160-sample segments (1.28 s), high-accuracy outputs are available within ~1.29 s.



Figure 5. Wearing a node on subject's dorsal foot.

To closely simulate real-world conditions, the IMU error characteristics in the simulation experiments are modeled to match the parameters shown in Table 1, along with a 10 mg accelerometer constant bias, allowing for a realistic simulation environment that can effectively mimic the behavior of the IMU sensor.

Alongside the proposed FB-RTS algorithm, the following three different algorithms are also carried out and compared. The ZUPT method is the baseline, representing the traditional solution detailed in Section 2, where no smoothing or segmentation is involved. The RTS method represents the proposed forward RTS smoothing solution implemented in a segmented fashion. The FB method represents the proposed Forward–Backward navigation solution implemented in a segmented fashion.

The effectiveness of the proposed FB-RTS algorithm will be preliminarily analyzed qualitatively using attitude, velocity, and position error curves. Quantitative analysis will be performed first by calculating the mean position error, maximum position error, and Root Mean Square Error (RMSE) of position for each algorithm. The RMSE of position is calculated using the following formula:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \hat{x}_i)^2} \tag{31}$$

where x_i is the true position, and $\hat{x}_i$ is the estimated position at time i, with N being the total number of data points. Then, to provide a visual comparison of the algorithm's performance and highlight the superiority of the proposed method, Cumulative Distribution Function (CDF) plots and distance percentage (D%) plots are introduced. The CDF curve illustrates the cumulative probability distribution of the position errors, while the D% curve represents the position error normalized by the distance traveled, providing a direct measure of the algorithm's overall accuracy as the subject traverses along their trajectory.

4.2. Simulation Experiments

To effectively simulate the periodic zero-velocity characteristics of pedestrian walking, a simulated walking process is designed to replicate the cyclical process of "acceleration → deceleration → zero-velocity". The simulated body moves along a square path in a clockwise direction, completing 8 laps, adding up to 783.36 m of total distance. Each phase of "acceleration → deceleration" and the "zero-velocity" phase lasts for 1 s, providing the algorithms with regular zero-velocity intervals. The ground truth trajectory and velocity for the simulation are detailed in Figures 6 and 7. Note that V_E and V_U are consistently zero in Figure 7, thus the overlapping traces. Gyroscope constant bias was not added in simulation experiments, while bias instability was retained.

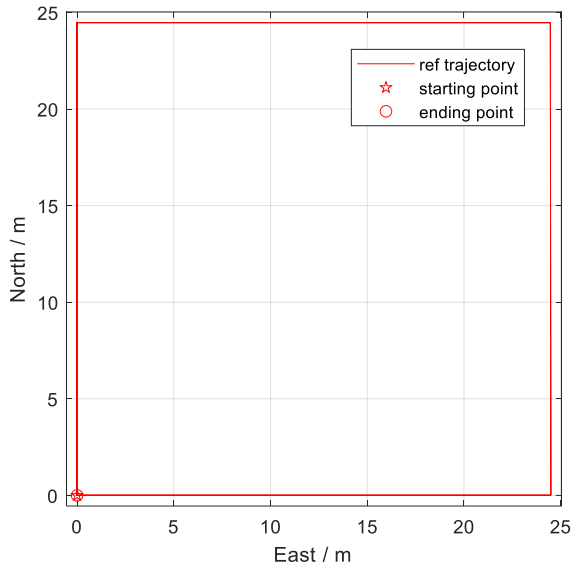


Figure 6. Trajectory for simulation experiments.

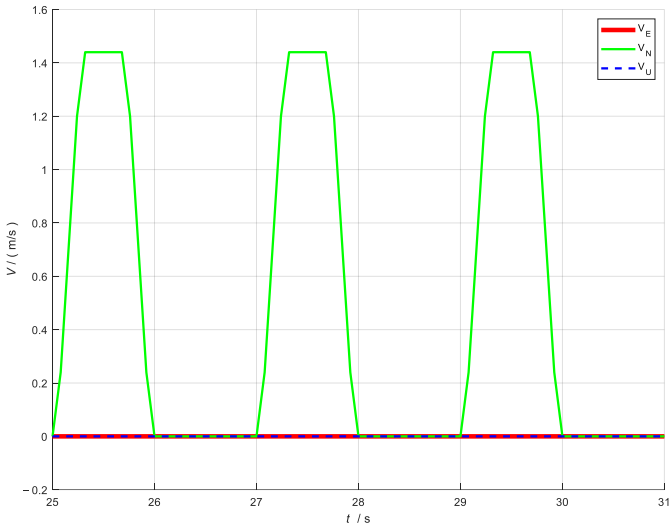


Figure 7. Zoomed-in view of velocity curve.

The trajectories resulting from the four algorithms are depicted in Figure 8, while Figures 9–12 illustrates the attitude, velocity, and position error curves. As depicted in Figure 9, the heading divergence of the FB-RTS algorithm is significantly reduced compared to other algorithms, while the errors in pitch and roll remain the same. It is important to note that the error curves for pitch and roll are mainly attributed to the biases in the accelerometers, which are in fact exaggerated to mimic a worst-case scenario. Upon examining Figure 11, it is evident that the RTS algorithm provides a noticeable smoothing effect over the ZUPT algorithm; however, it does not significantly enhance absolute accuracy due to the fact that it uses only the forward filtering information. On the other hand, the FB algorithm, leveraging historical velocity and positional information, significantly reduces velocity error compared to ZUPT, thereby improving positional accuracy; however, its error curves still exhibit some randomness. The FB-RTS algorithm combines the advantages of both, with the smallest fluctuations and the smoothest error curves, clearly demonstrating the superiority of the proposed FB-RTS algorithm, which is further elaborated quantitatively in Table 2.

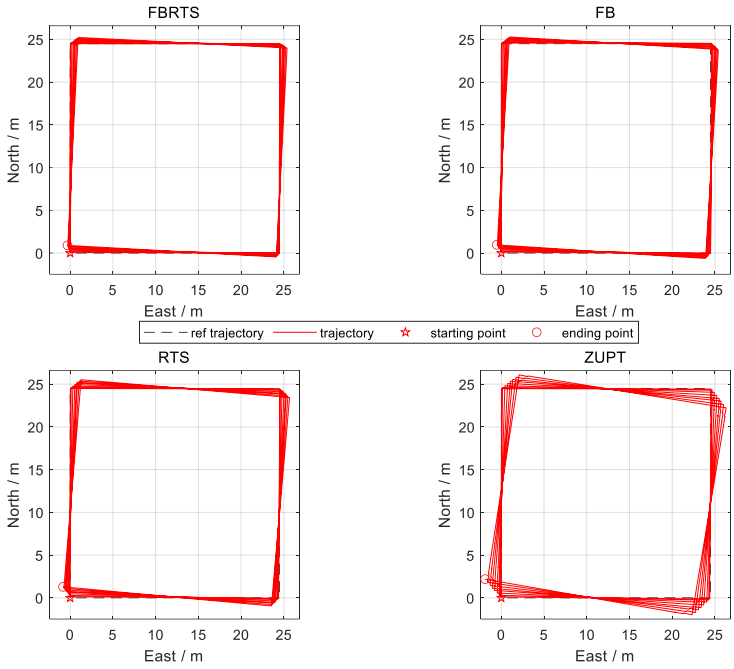


Figure 8. Trajectories of different algorithms.

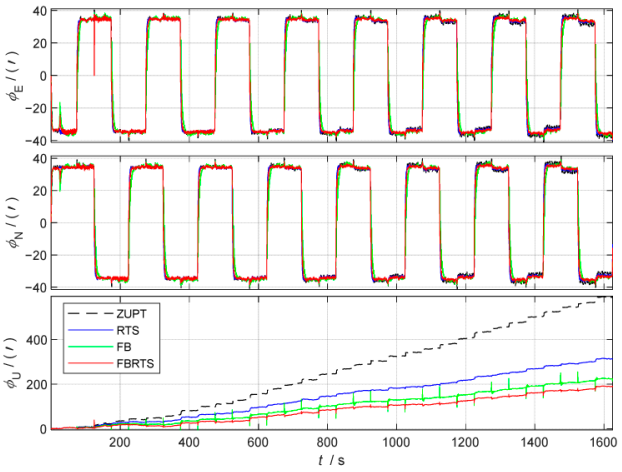


Figure 9. Attitude error curves of different algorithms.

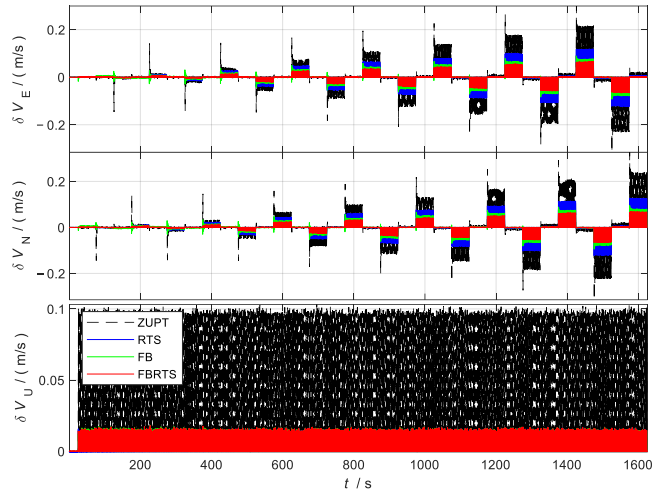


Figure 10. Velocity error curves of different algorithms.

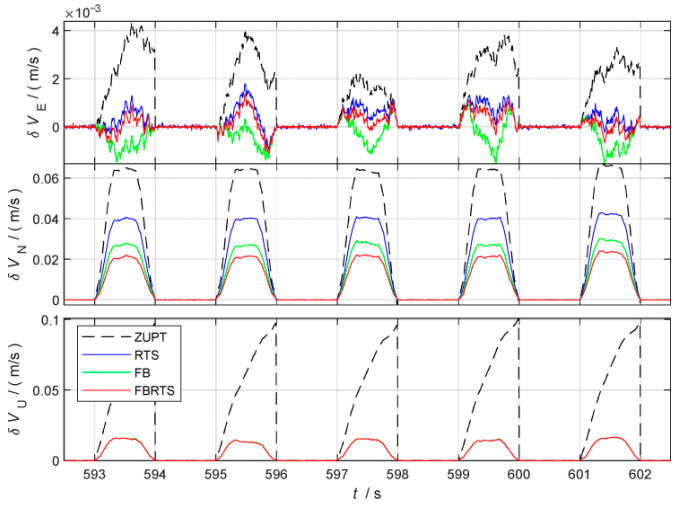


Figure 11. Zoomed-in view of velocity error curves of different algorithms.

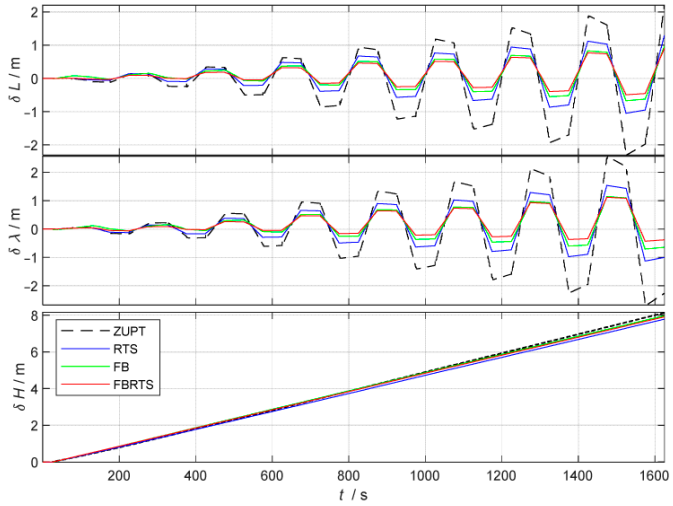


Figure 12. Position error curves of different algorithms.

Table 2. Error results of different algorithms in simulation experiments.

Algorithm	Mean Error/*m*	RMSE/*m*	Maximum Error/*m*
ZUPT	1.07	1.32	2.94
RTS	0.596	0.723	1.64
FB	0.426	0.520	1.23
FB-RTS	0.362	0.460	1.19

The CDF and D% plots for the four algorithms, as depicted in Figures 13 and 14, provide a visual comparison of the positional accuracy across the tested methods. The error data and corresponding curves clearly demonstrate a hierarchy in precision, with FB-RTS outperforming FB, RTS, and at last ZUPT. It should be pointed out that the initial 150 m of the D% plot reveal that the performance of the four methods may not strictly adhere to the aforementioned ranking, particularly during the early stages of the walk. This observation suggests that while the proposed FB-RTS algorithm can mitigate the error accumulation process, its significant advantage over traditional methods becomes more pronounced over longer navigation periods. This characteristic aligns well with the practical demands and trends for pedestrian navigation, where long-duration and high-availability systems are increasingly sought after.

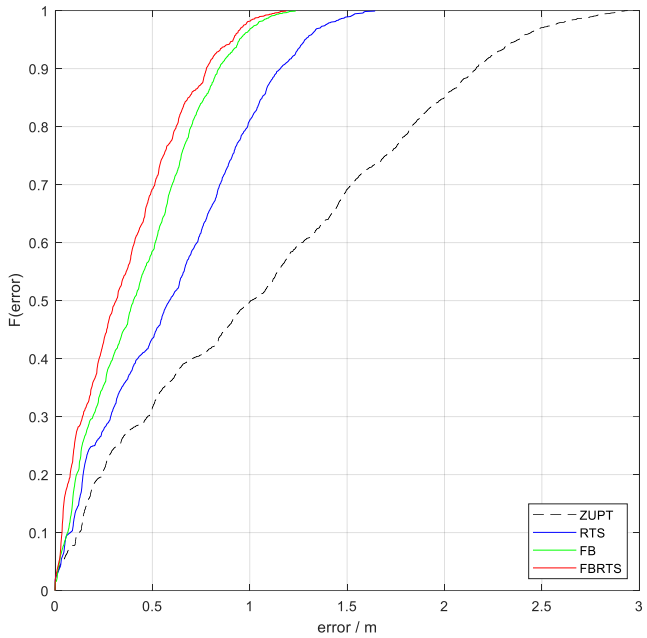


Figure 13. CDF plot for simulation experiments.

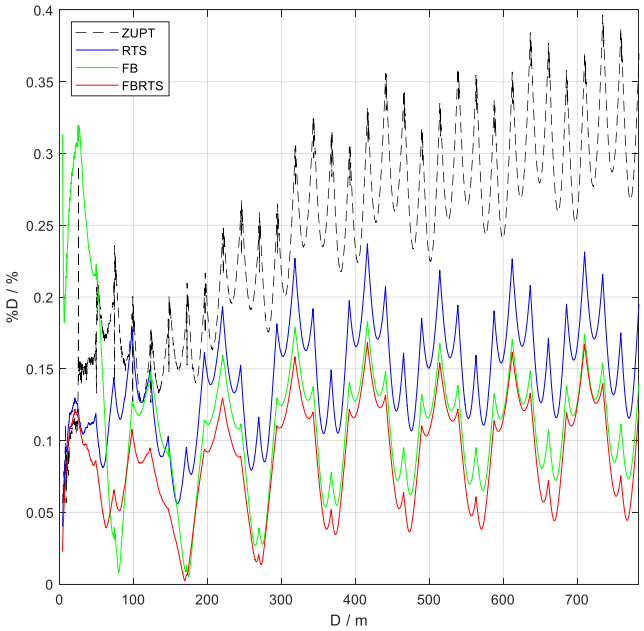


Figure 14. D% plot for simulation experiments.

The simulation experiments have provided a controlled and systematic framework to evaluate the FB-RTS algorithm against existing methods. The results substantiate the FB-RTS algorithm's enhanced performance in terms of error reduction and navigation accuracy. These findings pave the way for the following validation through real-world field tests.

4.3. Indoor Field Tests

Indoor field tests were conducted at the laboratory located in the Sipailou Campus of Southeast University, as shown in Figure 15. To obtain reliable ground truths for location reference, ten landmark points were marked at the intersection of the corridors by adhering paper markers to the ground, and the distances between these landmarks were calibrated

using a laser rangefinder with an error parameter of ±3 mm. The resulting indoor map and the positions of the landmark points are depicted in Figure 16.

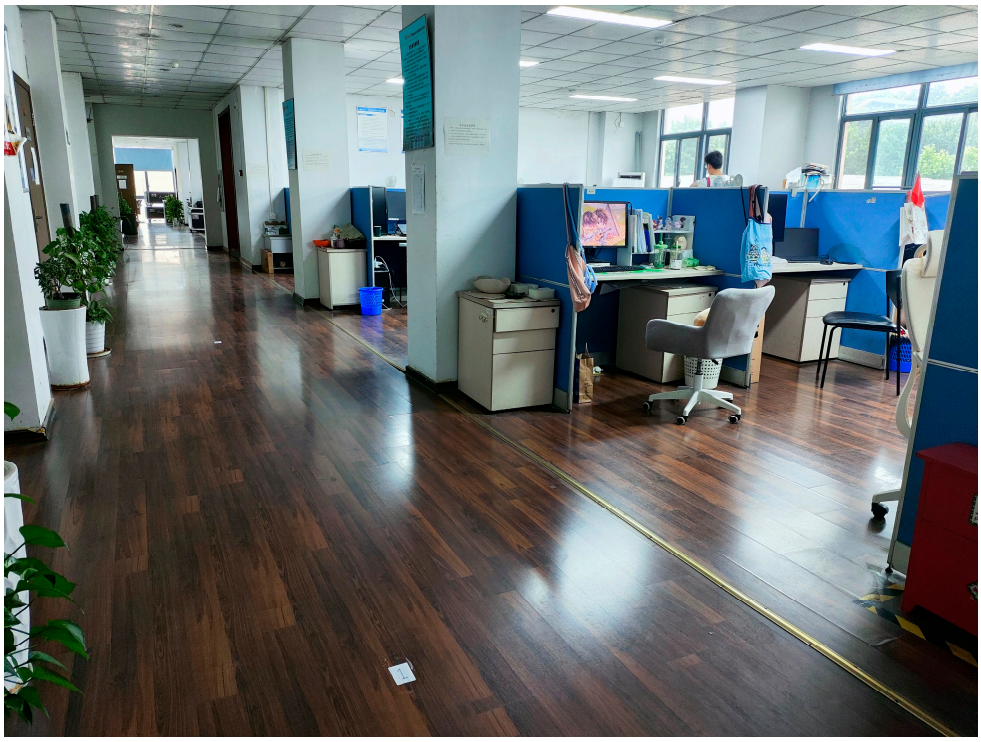


Figure 15. Environment for indoor field tests.

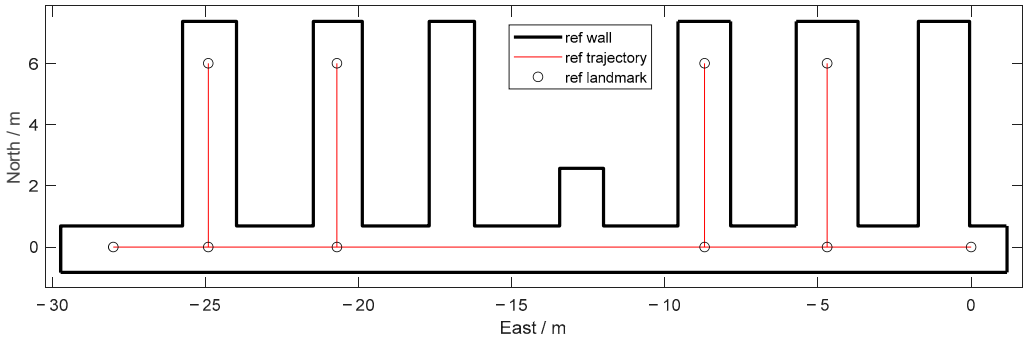


Figure 16. Indoor map and the positions of the landmark points for indoor field tests.

Prior to each experiment, gyroscope biases are estimated during a stationary initialization phase of 30 s and subtracted before operation [29]. A subject wore a single IMU node on their left foot and walked in straight lines between the landmark points. When arriving at a landmark point, the subject was instructed to roughly align their left foot with the marker and stand still for 2–3 s. This procedure allowed for accurate determination of when the IMU was positioned over the landmark point, ensuring that the actual position of the IMU node and the corresponding landmark point would not differ by more than a few centimeters.

Two sets of experiments were conducted, covering total distances of 308 m and 388 m, referred to as Path A and Path B, respectively. The trajectories resulting from the four algorithms are shown in Figures 17 and 18, where the starting point of each trajectory is located at the origin (coordinates 0, 0). It can be observed that due to the discontinuity of the zero-velocity updates, the Kalman Filter states are prone to sudden changes when transitioning from the swing phase to the stance phase, leading to a jagged

trajectory in the ZUPT algorithm, particularly prominent in the results of Path B. This significantly affects its navigation accuracy. In contrast, the other three methods exhibit smoother trajectories, although the FB algorithm also shows small non-smooth transitions, suggesting that applying RTS to FB can still improve navigation accuracy. Consistent with the observations from the simulation experiments, the proposed FB-RTS algorithm also demonstrates slightly improved heading accuracy over the other algorithms. This clearly showcases the superiority of the FB-RTS algorithm in real environments, which is quantitatively detailed in Table 3.

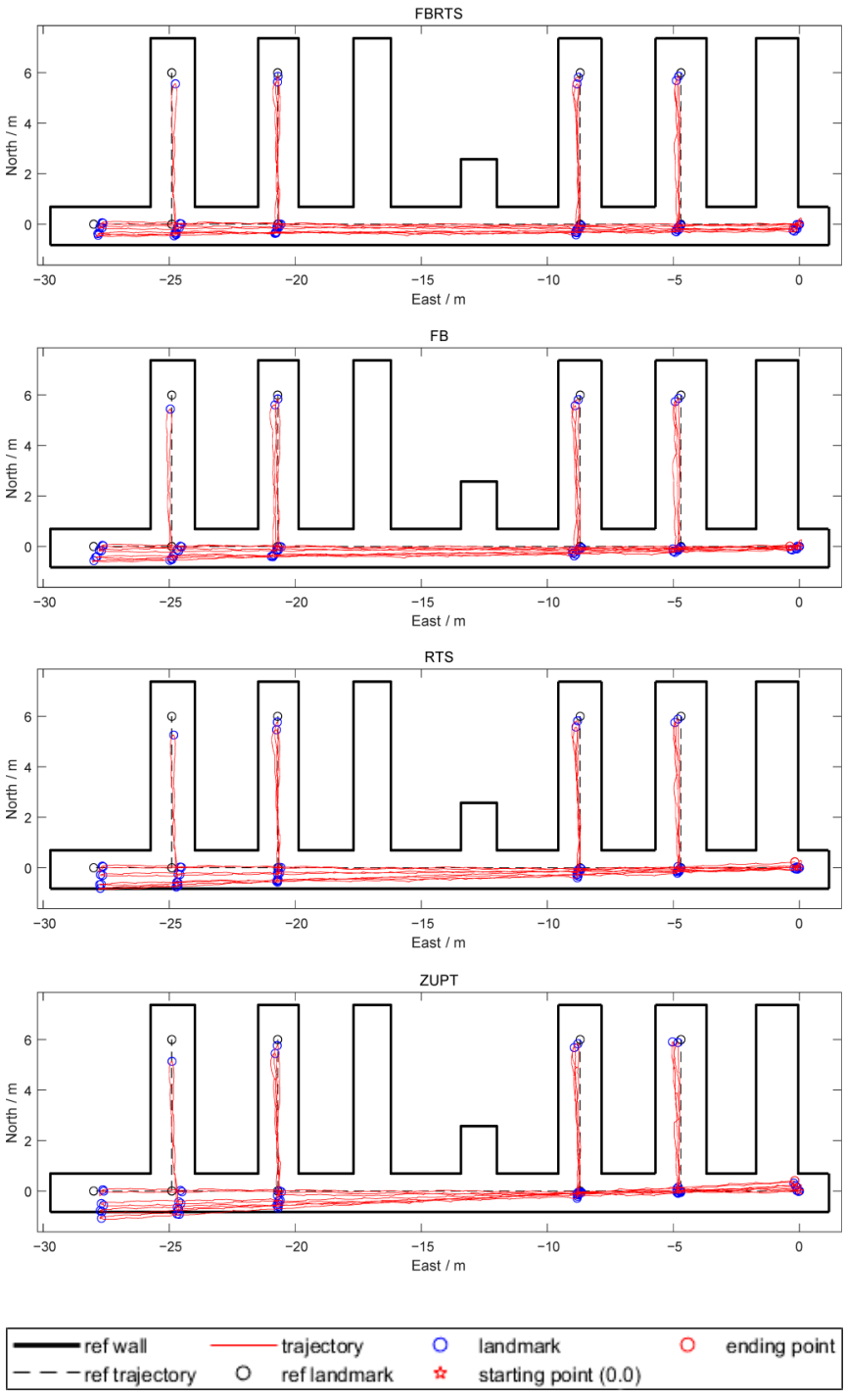


Figure 17. Trajectories of the four algorithms for Path A.

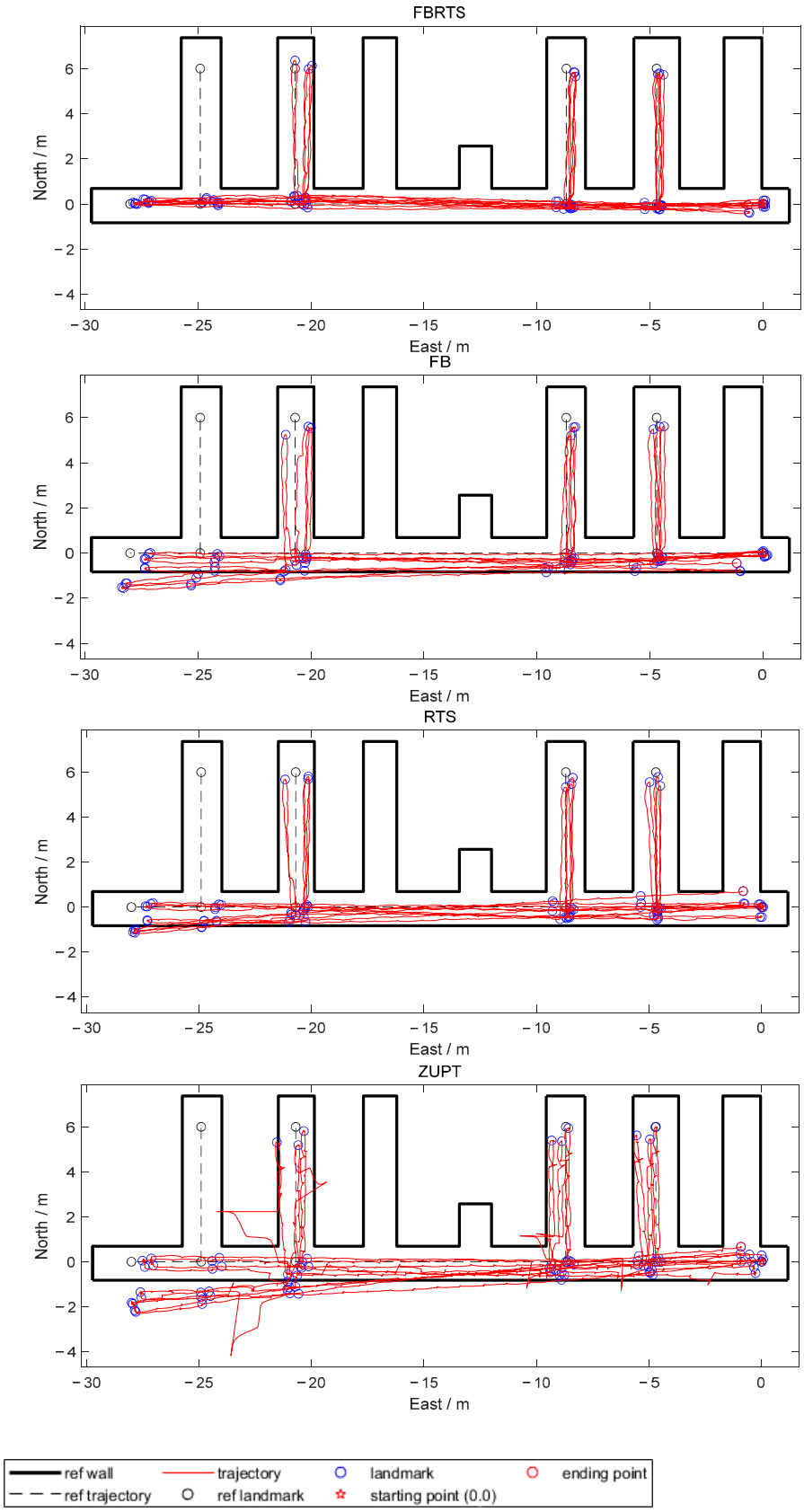


Figure 18. Trajectories of the four algorithms for Path B.

The CDF and D% plots for the four algorithms are presented in Figures 19 and 20. It is evident that for both sets of experiments, the error data and curves indicate that the proposed FB-RTS performs the best, while the ZUPT, our baseline, performs the worst,

with FB and RTS accuracies falling in between. Compared to ZUPT, the position RMSE of FB-RTS is improved by 34.1% and 51.6% in the two sets of experiments, respectively. However, the performance of FB and RTS is not consistent across the two experiments; for Path A, FB outperforms RTS, consistent with the simulation results, but for Path B, FB underperforms RTS. This discrepancy is due to variations in pedestrian gait or walking speed. Regardless, the FB-RTS algorithm significantly outperforms the traditional ZUPT algorithm in all experiments, while also demonstrating its superior robustness over the standalone application of FB or RTS.

Table 3. Error results of different algorithms in indoor field tests.

	Algorithm	Mean Error/*m*	RMSE/*m*	Maximum Error/*m*
Path A	ZUPT	0.361	0.457	1.12
	RTS	0.319	0.389	0.858
	FB	0.285	0.323	0.560
	FB-RTS	0.272	0.301	0.472
Path B	ZUPT	0.788	0.942	2.24
	RTS	0.557	0.620	1.15
	FB	0.706	0.812	1.57
	FB-RTS	0.393	0.456	0.957

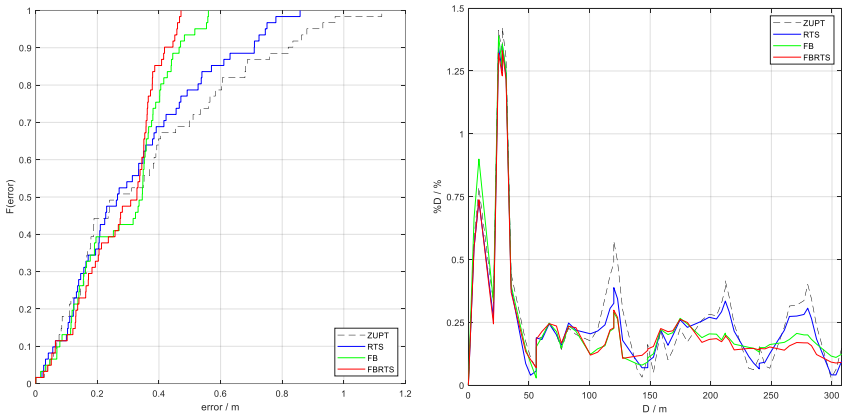


Figure 19. CDF and D% plot for Path A.

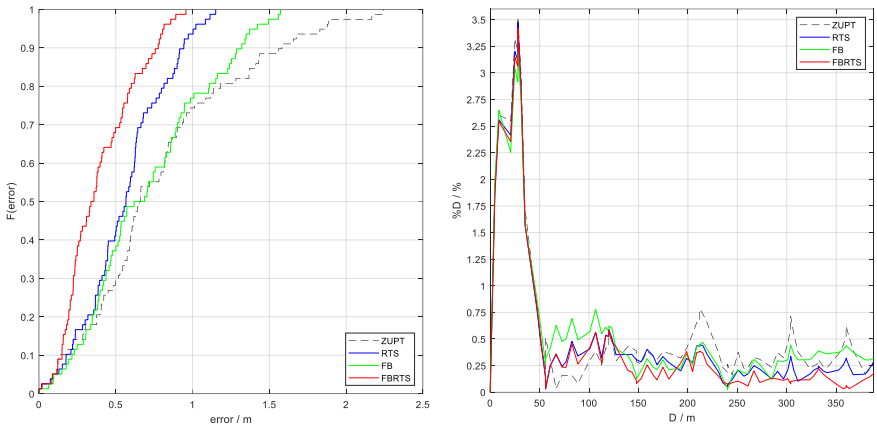


Figure 20. CDF and D% plot for Path B.

Similarly to the findings from the simulation experiments, the initial 100 m of the D% plot reveal that the performance of the four methods does not necessarily align with the aforementioned precision ranking. This discrepancy can be attributed to the short

accumulation time of errors, during which the differences among the algorithms are not pronounced. However, as the walking distance and duration increase, the advantages of the proposed FB-RTS method become more pronounced, and the D% curve becomes more stable than those of other algorithms. This observation underscores the effectiveness of the FB-RTS algorithm in mitigating error accumulation over longer navigation periods, thereby offering a more significant improvement in terms of robustness and high availability over traditional methods as the duration of navigation extends.

4.4. Outdoor Field Tests

Outdoor field tests were further conducted in the Sipailou Campus of Southeast University, aiming to provide long-distance evaluation of the proposed algorithm. To establish accurate ground truth for trajectory validation, a GNSS RTK (Real-Time Kinematic) setup was employed, providing centimeter-level positioning accuracy when RTK is enabled.

Prior to each experiment, IMU nodes underwent the same calibration and bias estimation procedures as in the indoor tests. A subject wore a single IMU node on their left foot and traversed two distinct paths, Path C and Path D, with a length of 2100 m and 2760 m, respectively. The subject was instructed to stop at locations with minimal GNSS signal obstruction for approximately 5 s. These stops served as reference points for evaluating the navigation accuracy of the algorithms. In total, 18 waypoints were recorded for Path C and 24 waypoints for Path D, distributed along the trajectories to ensure comprehensive coverage.

The trajectories generated by the four algorithms (ZUPT, RTS, FB, FB-RTS) are shown in Figures 21 and 22, and their error results are detailed in Table 4. The CDF and D% plots (Figures 23 and 24) further confirmed the superiority of FB-RTS. For both paths, its position RMSE was 26.3% (Path C) and 39.8% (Path D) lower than ZUPT.

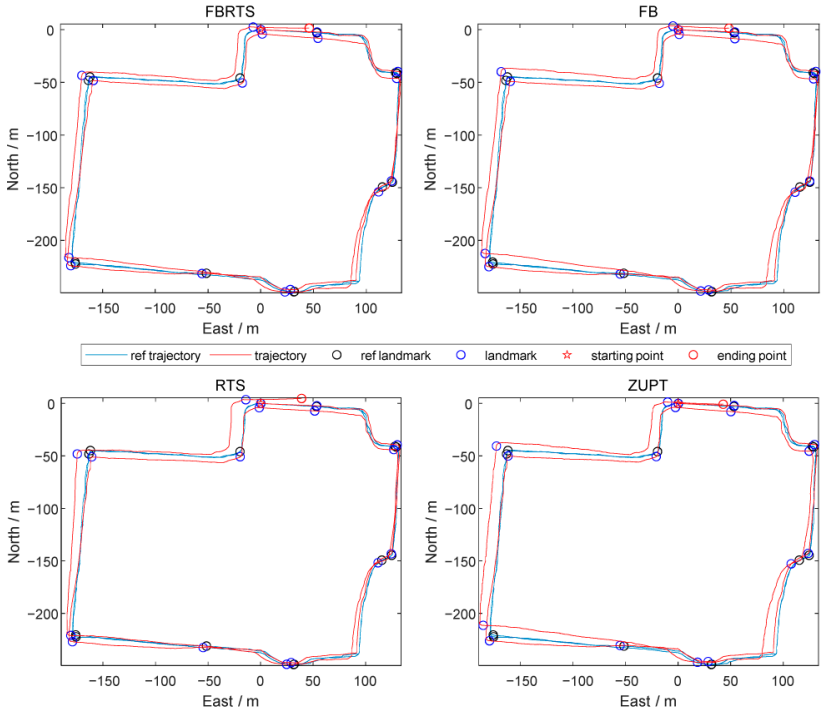


Figure 21. Trajectories of the four algorithms for Path C.

These results highlight the robustness of FB-RTS in long navigation sessions, further confirming the conclusions drawn from the indoor field tests. The algorithm's ability to suppress error accumulation over multi-kilometer trajectories underscores its potential for real-world pedestrian navigation systems.

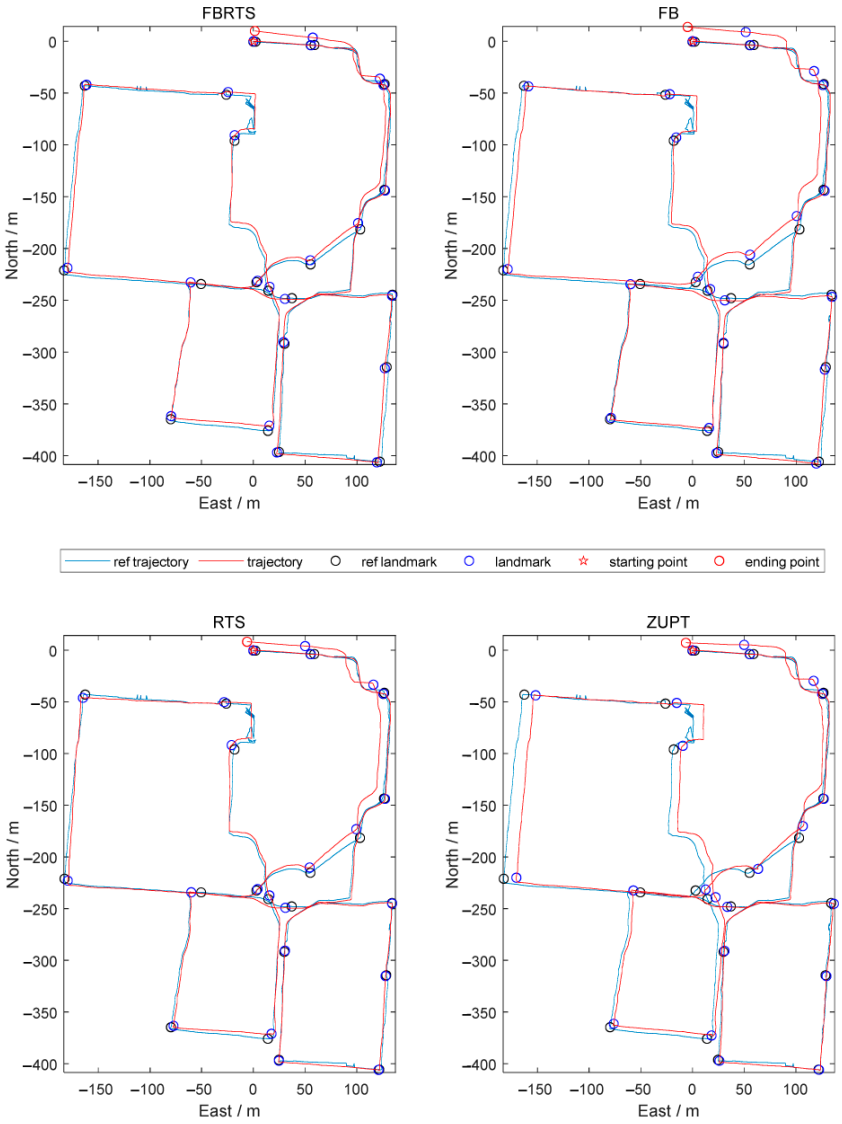


Figure 22. Trajectories of the four algorithms for Path D.

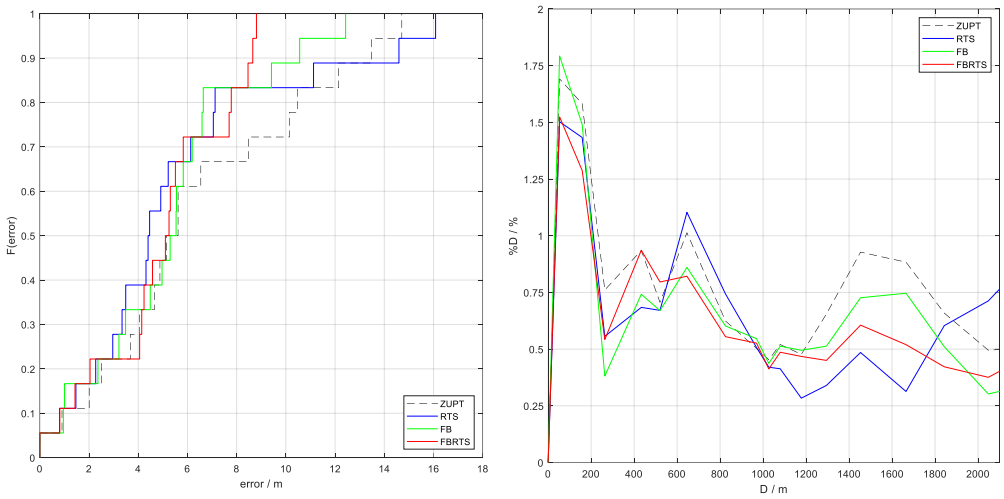


Figure 23. CDF and D% plot for Path C.

Table 4. Error results of different algorithms in outdoor field tests.

	Algorithm	Mean Error/*m*	RMSE/*m*	Maximum Error/*m*
Path C	ZUPT	6.39	7.63	14.7
	RTS	5.54	7.00	16.1
	FB	5.25	6.14	12.4
	FB-RTS	4.98	5.62	8.80
Path D	ZUPT	6.31	7.92	15.0
	RTS	4.29	5.84	13.2
	FB	5.32	7.21	15.7
	FB-RTS	3.81	4.77	10.4

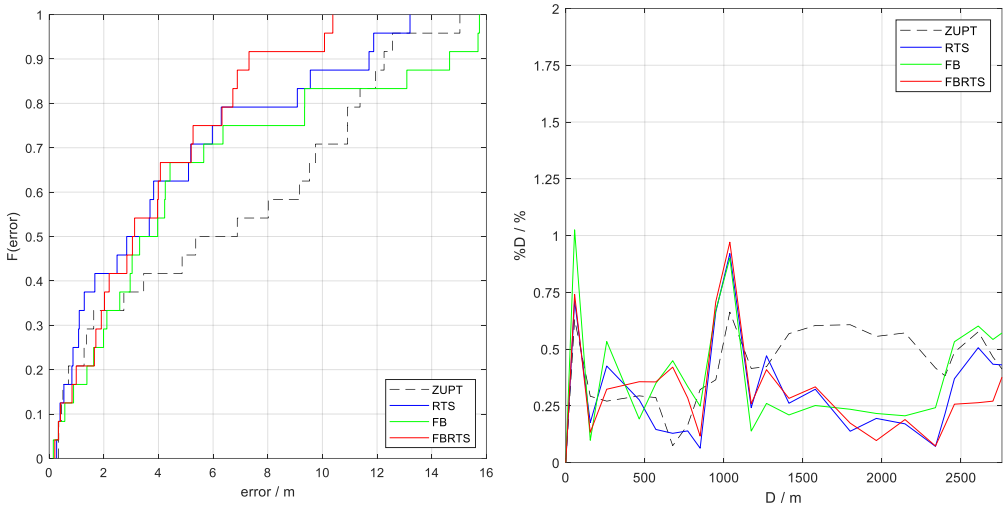


Figure 24. CDF and D% plot for Path D.

5. Discussion

A novel Forward–Backward navigation and RTS smoothing (FB-RTS) navigation scheme is proposed in this paper, designed to enhance the navigation accuracy of MIMU-based pedestrian navigation systems, without the need for additional sensors or specialized conditions. The FB-RTS algorithm harnesses the complete gait cycle of a pedestrian, utilizing historical data during both the stance and swing phases, to enhance navigation accuracy and reliability. First, both forward and backward integration method and the corresponding error equations are constructed, based on which the corresponding RTS smoother is applied. Second, both backward and forward RTS algorithms are established, where the system model and observation model are built under the output correction mode. Finally, both navigation results are combined to achieve the final estimation of attitude and velocity, where the position is recalculated by the optimized data.

Simulation experiments and two sets of field tests were conducted to validate the performance of the FB-RTS algorithm. The results demonstrated the algorithm's superior performance in reducing navigation errors and smoothing pedestrian trajectories, offering a significant improvement over the traditional ZUPT method and both the FB and the RTS method. The FB-RTS algorithm's ability to provide smoother and more accurate navigation trajectories was particularly evident in the real-world tests, offering a robust, adaptable and universal solution that is aligned with the growing demand for accurate and reliable indoor positioning technologies, well-suited for applications in smart buildings, wayfinding in complex indoor environments, and emergency response operations where precise indoor positioning is critical.

Author Contributions: Conceptualization, Y.Y. and C.Y.; methodology, Y.S.; software, Y.S. and X.X.; validation, Y.S.; formal analysis, Y.S.; investigation, Y.S.; resources, C.Y. and X.X.; data curation, Y.S.; writing—original draft preparation, Y.S.; writing—review and editing, Y.Y., C.Y. and X.X.; visualization, Y.S.; supervision, Y.Y.; project administration, Y.Y.; funding acquisition, Y.Y. and C.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded in part by the National Natural Science Foundation of China, grant number 52471358, 52271346, 62471132; in part by the Young Elite Scientists Sponsorship Program by CAST, grant number 2023QNRC001, 2022QNRC001; in part by the Jiangsu Provincial Department of Science and Technology, grant number BM2023013; in part by the Zhishan Youth Scholar Program of Southeast University, grant number 2242024RCB0023.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data will be available upon request.

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

Abbreviations

$\dot{\cdot}$	Derivative of
$\boldsymbol{C}_b^n$	Attitude matrix, orthonormal rotation matrix from body frame to navigation frame
$\boldsymbol{\omega}_{nb}^b$	Angular rate of the IMU relative to navigation frame in body frame
$[\cdot\times]$	Skew-symmetric matrix of
$\boldsymbol{V}^n$	Velocity in navigation frame, $\boldsymbol{V}^n = \begin{bmatrix} V_E & V_N & V_U \end{bmatrix}^T$
$\boldsymbol{f}^b$	Specific force in body frame
$\boldsymbol{g}^n$	The gravitational vector in navigation frame
L	Latitude in the Earth ellipsoid coordinate system
R_M	The Earth's equatorial radius of curvature
h	Height in the Earth ellipsoid coordinate system
λ	Longitude in the Earth ellipsoid coordinate system
R_N	The Earth's meridional radius of curvature
δ	Error of
$\boldsymbol{\phi}^n$	Misalignment angle in navigation frame
$\boldsymbol{\omega}_{in}^n$	Angular rate of navigation frame relative to inertial frame in navigation frame
$\boldsymbol{\varepsilon}^n$	Bias errors from gyroscopes in navigation frame
∇^n	Bias errors from accelerometers in navigation frame
$\boldsymbol{X}$	State vector of Kalman Filter
$\boldsymbol{P}$	Position in the Earth ellipsoid coordinate system, $\boldsymbol{P} = \begin{bmatrix} L & \lambda & h \end{bmatrix}^T$
$\boldsymbol{X}_k$	State vector at time k
$\boldsymbol{\Phi}_{k,k-1}$	State transition matrix from time $k-1$ to k
$\boldsymbol{\Gamma}_{k-1}$	Noise distribution matrix at time $k-1$
$\boldsymbol{W}_{k-1}$	System noise vector at time $k-1$
$\boldsymbol{Z}_k$	Measurement vector at time k
$\boldsymbol{H}_k$	Measurement matrix at time k
$\boldsymbol{V}_k$	Measurement noise vector at time k
$\boldsymbol{v}_k$	Velocity calculated by SINS at time k
$\leftarrow$	Denotes variables used in the backward navigation process
$\boldsymbol{\omega}_{ib}^b$	Angular rate of the IMU relative to inertial frame in body frame
$\boldsymbol{\omega}_{ie}^n$	Angular rate of Earth frame relative to inertial frame in navigation frame
$\boldsymbol{\omega}_{en}^n$	Angular rate of navigation frame relative to Earth frame in navigation frame
^{F}	Represents the variables of the forward filtering process
^{B}	Represents the variables of the backward RTS process
$\boldsymbol{K}_{k/N}^B$	RTS gain at time k of N

References

1. Indoor Location Market Size, Share, Growth Analysis, By Offering (Hardware, Solutions, Services), Technology (BLE, UWB, Wi-Fi), Application, Vertical (Retail, Healthcare & Pharmaceuticals, Manufacturing) and Region—Global Forecast to 2029. Available online: www.marketsandmarkets.com/Market-Reports/indoor-location-market-989.html (accessed on 17 October 2024).
2. Brigante, C.M.; Abbate, N.; Basile, A.; Faulisi, A.C.; Sessa, S. Towards miniaturization of a MEMS-based wearable motion capture system. *IEEE Trans. Ind. Electron.* **2011**, *58*, 3234–3241. [CrossRef]
3. Jiménez, A.R.; Seco, F.; Prieto, J.C.; Guevara, J. Indoor pedestrian navigation using an INS/EKF framework for yaw drift reduction and a foot-mounted IMU. In Proceedings of the 2010 7th Workshop on Positioning, Navigation and Communication, Dresden, Germany, 10–11 March 2010; pp. 135–143.
4. Bird, J.; Arden, D. Indoor navigation with foot-mounted strapdown inertial navigation and magnetic sensors [emerging opportunities for localization and tracking]. *IEEE Wirel. Commun.* **2011**, *18*, 28–35. [CrossRef]
5. Qiu, S.; Wang, Z.; Zhao, H.; Qin, K.; Li, Z.; Hu, H. Inertial/magnetic sensors based pedestrian dead reckoning by means of multi-sensor fusion. *Inf. Fusion* **2018**, *39*, 108–119. [CrossRef]
6. Shi, L.F.; Zhao, Y.L.; Liu, G.X.; Chen, S.; Wang, Y.; Shi, Y.F. A robust pedestrian dead reckoning system using low-cost magnetic and inertial sensors. *IEEE Trans. Instrum. Meas.* **2018**, *68*, 2996–3003. [CrossRef]
7. Harle, R. A survey of indoor inertial positioning systems for pedestrians. *IEEE Commun. Surv. Tutor.* **2013**, *15*, 1281–1293. [CrossRef]
8. El-Sheimy, N.; Li, Y. Indoor navigation: State of the art and future trends. *Satell. Navig.* **2021**, *2*, 7. [CrossRef]
9. Foxlin, E. Pedestrian tracking with shoe-mounted inertial sensors. *IEEE Comput. Graph. Appl.* **2005**, *25*, 38–46. [CrossRef] [PubMed]
10. Niu, X.; Li, Y.; Kuang, J.; Zhang, P. Data fusion of dual foot-mounted IMU for pedestrian navigation. *IEEE Sens. J.* **2019**, *19*, 4577–4584. [CrossRef]
11. Qi, L.; Yu, Y.; Liu, Y.; Gao, C.; Feng, T.; Hui, J.; Wang, S. A robust foot-mounted positioning system based on dual IMU data and ultrasonic ranging. *IEEE Sens. J.* **2023**, *23*, 4085–4095. [CrossRef]
12. Borenstein, J.; Ojeda, L. Heuristic drift elimination for personnel tracking systems. *J. Navig.* **2010**, *63*, 591–606. [CrossRef]
13. Jiménez, A.R.; Seco, F.; Zampella, F.; Prieto, J.C.; Guevara, J. Improved Heuristic Drift Elimination (iHDE) for pedestrian navigation in complex buildings. In Proceedings of the 2011 International Conference on Indoor Positioning and Indoor Navigation, Guimarães, Portugal, 21–23 September 2011; pp. 1–8.
14. Wagstaff, B.; Peretroukhin, V.; Kelly, J. Robust data-driven zero-velocity detection for foot-mounted inertial navigation. *IEEE Sens. J.* **2019**, *20*, 957–967. [CrossRef]
15. Chen, Z.; Pan, X.; Chen, C.; Wu, M. Contrastive learning of zero-velocity detection for pedestrian inertial navigation. *IEEE Sens. J.* **2021**, *22*, 4962–4969. [CrossRef]
16. Chen, C.; Lu, C.X.; Wahlström, J.; Markham, A.; Trigoni, N. Deep neural network based inertial odometry using low-cost inertial measurement units. *IEEE Trans. Mob. Comput.* **2019**, *20*, 1351–1364. [CrossRef]
17. Kim, W.Y.; Seo, H.I.; Seo, D.H. Nine-Axis IMU-based Extended inertial odometry neural network. *Expert Syst. Appl.* **2021**, *178*, 115075. [CrossRef]
18. Qian, W.; Zhou, Z.; Xie, F. Pedestrian navigation method based on machine learning and gait feature assistance. *J. Chin. Inert. Technol.* **2019**, *27*, 433.
19. Liu, H.; Jiang, X.; Yue, Y.; Gao, G. Application of reverse processing in SINS initial alignment and SINS/GPS integrated navigation. *Aircr. Eng. Aerosp. Technol.* **2018**, *90*, 320–327. [CrossRef]
20. Gong, X.; Zhang, J.; Fang, J. A modified nonlinear two-filter smoothing for high-precision airborne integrated GPS and inertial navigation. *IEEE Trans. Instrum. Meas.* **2015**, *64*, 3315–3322. [CrossRef]
21. Zhang, X.; Zhu, F.; Tao, X.; Duan, R. New optimal smoothing scheme for improving relative and absolute accuracy of tightly coupled GNSS/SINS integration. *GPS Solut.* **2017**, *21*, 861–872. [CrossRef]
22. Han, H.; Wang, J.; Du, M. A fast SINS initial alignment method based on RTS forward and backward resolution. *J. Sens.* **2017**, *2017*, 7161858. [CrossRef]
23. Niu, X.; Liu, T.; Kuang, J.; Zhang, Q.; Guo, C. Pedestrian trajectory estimation based on foot-mounted inertial navigation system for multistory buildings in postprocessing mode. *IEEE Internet Things J.* **2021**, *9*, 6879–6892. [CrossRef]
24. Colomar, D.S.; Nilsson, J.O.; Händel, P. Smoothing for ZUPT-aided INSs. In Proceedings of the 2012 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Sydney, Australia, 13–15 November 2012; pp. 1–5.
25. Skog, I.; Handel, P.; Nilsson, J.O.; Rantakokko, J. Zero-velocity detection—An algorithm evaluation. *IEEE Trans. Biomed. Eng.* **2010**, *57*, 2657–2666. [CrossRef] [PubMed]
26. Rauch, H.E.; Tung, F.; Striebel, C.T. Maximum likelihood estimates of linear dynamic systems. *AIAA J.* **1965**, *3*, 1445–1450. [CrossRef]

27. Song, E.; Zhu, Y.; Zhou, J.; You, Z. Optimal Kalman filtering fusion with cross-correlated sensor noises. *Automatica* **2007**, *43*, 1450–1456. [CrossRef]
28. Zhang, Z.; Scaramuzza, D. A tutorial on quantitative trajectory evaluation for visual (-inertial) odometry. In Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 7244–7251.
29. Zhang, Y.; Yang, X.; Xing, X.; Wang, Z.; Xiong, Y. The standing calibration method of MEMS gyro bias for autonomous pedestrian navigation system. *J. Navig.* **2017**, *70*, 607–617. [CrossRef]

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