Journal of Computers Vol. 33 No. 3, June 2022, pp. 17-33
doi: 10.53106/199115992022063303002

Pedestrian Inertial Navigation with Building Floor Plans for Indoor Environments via Particle Filter

Yushuai Zhang, Jianxin Guo*, Rui Zhu,
Zhengyang Zhao, Feng Wang, Liping Wang

School of Information Engineering, Xijing University, 710123, Xi'an, China
apacheyu@yeah.net, guojx_516@126.com, zhu_r10@163.com, zhaozhengyang6910@163.com
wangfengisn@163.com, wangliping@xijing.edu.cn

Received 20 May 2021; Revised 14 November 2021; Accepted 14 December 2021

Abstract. With the rapid development of smart city, the indoor positioning services became more and more important. During the existing solutions, inertial measurement unit (IMU) with pedestrian dead reckoning (PDR) was a promising scheme since they did not require external equipment in the environment. However, the orientation drift of low-cost IMU limited their application in practical. To address this problem, a zero-velocity update (ZUPT) framework included Kalman filter and particle filter is designed based on the foot-based low-cost IMU and digital floor plan to provide the service of personal navigation. In the designed Smoothing for ZUPT-aided INSs framework, the Kalman filter is used to estimate the position and attitude by zero velocity correction technique. Then, the particle filter is used to improve the localization and heading accuracy by map matching. The position estimation presented in this study achieves an average position error of 1.16 m. The experimental results show that the designed framework can solve the personal navigation problem in the case of building plan information assistance and help improve the accuracy and reliability of continuous position determination of personal navigation systems effectively.

Keywords: pedestrian dead reckoning, zero-velocity update, particle filter, map matching

1 Introduction

In recent years, there has been an increasing interest in the Location Based Services (LBS) [1] with the development of various industries utilizing the information of individual location. In outdoor positioning systems, the Global Position System (GPS) is frequently prescribed for the Location Based Services. However, the primary disadvantage of the GPS is that the position accuracy largely becomes very limited in indoor environments because the walls of the building block the satellite signals.

The high-precision indoor navigation systems can help people save time and increase productivity in cities, where people live and work primarily in indoor environments. Nowadays, the demand of indoor location-related services for Location Based Services (LBS) has growing dramatically with the development of smart city. In technical terms, indoor navigation systems can be divided into two types: infrastructure-based system and infrastructure-free system [2]. Wireless Sensor Networks, such as Wi-Fi [3], ultra-wideband (UWB) [4], and Bluetooth [5], belong in this type. Despite its popularity, the infrastructure-based system is a laborious task that requires a considerable amount of time to install and commission the equipment. The infrastructure-based system requires a large number of devices to be pre-installed in a small space in order to guarantee the accuracy of navigation. For example, five Bluetooth sensors are required for a 6*8 square meter area when using Bluetooth location [5], and Six UWB beacons were installed in each of the three UWB systems to cover a 24 m*14 m indoor area [4]. The second type does not require any additional hardware to be installed and can be more cost effective, such as inertial navigation systems [6-7].

Compared to other navigation systems, inertial navigation systems (INSs) don't require the installation of additional external equipment [8-9]. The inertial navigation systems are not limited by external environmental conditions, making it a fully autonomous navigation system. In addition, microelectromechanical systems (MEMS)-based inertial measurement unit (IMU) sensors offer the advantages of low cost, miniaturization, and low power consumption. Such approaches, however, have failed to address the navigation errors of IMU grow significantly with time. To calibrate the sensor drift errors, the IMU was mounted on a foot to utilize the framework of the zero velocity update (ZUPT) [10-13]. For pedestrian dead reckoning with foot-mounted inertial sensors, this gives undesirable discontinuities in the estimated trajectory at the end of each step. DS Colomar et al. proposed method

* Corresponding Author

is based on a 3-pass mixed open-closed-loop filter to eliminate the discontinuities [14]. However, the low appreciable heading error of a standalone inertial navigation system will lead to the drift errors in the final navigation solution. Therefore, inertial navigation systems should be integrated with other sensors or aiding information to improve the navigation accuracy [15]. References [16-21] use Bluetooth, UWB, Wi-Fi sensor information to correct inertial navigation drift errors, respectively. Multi-sensor fusion positioning methods are all effective in providing users with efficient indoor positioning navigation. However, these methods have high infrastructure costs and require a large workforce to build a complete indoor positioning navigation system. Another way to improve the navigation accuracy is using maps that need no additional infrastructure or equipment [22-25].

The common solution of inertial navigation systems (INSs) and auxiliary information integration is using extended Kalman filter (EKF). However, Extended Kalman filter is not an effective method to provide accurate positioning for the nonlinear integrated navigation systems anymore. As a completely nonlinear state estimator, particle filter can improve the performance of INS/map-integration systems by utilizing information from indoor building structures. In view of the advantages and disadvantages of the two filters, this paper pro-poses an inertial navigation algorithm combining extended Kalman and particle filter which are combined using pedestrian motion features and indoor map information.

Our main contribution can be summarized as follows:

Firstly, Only IMU and indoor map information are available in the proposed algorithm, and the system does not involve pre-measurement or installation of structures, which can significantly reduce the time and economic cost of indoor navigation systems.

Secondly, the Smoothing for ZUPT-aided INSs and map matching method are innovatively combined. An auxiliary particle filter algorithm fused with map matching techniques is used to correct the heading drift error of the Smoothing for ZUPT-aided INSs.

Finally, experimental results in different scenarios are used to show the advantages of the proposed algorithm.

This paper has been divided into three parts. The first part deals with an overview of the system for pedestrian inertial navigation, and details the algorithms involved in the system. In the second part, the experimental results of the indoor navigation system proposed in this paper are analyze. The third part presents the conclusions and future work.

2 Methods

The specific objective of this study was to combination of the Smoothing for ZUPT-aided INSs (SINS) and map matching for indoor positioning and navigation system, whose structure is shown in Fig. 1. Firstly, Preliminary navigation and positioning using ZUPT-aided INSs, which includes zero velocity detection, zero velocity update and inertial navigation algorithm [26] in this system framework. Then, the paper used a smoothing algorithm for ZUPT-aided INSs that removes the discontinuity at the end of each step [11]. However, the errors in position and heading still increase with time. Finally, the system optimizes the estimated trajectories by map matching algorithm. Based on the above parts, we implement a complete indoor navigation system. We'll go through each phase in detail in the following sections.

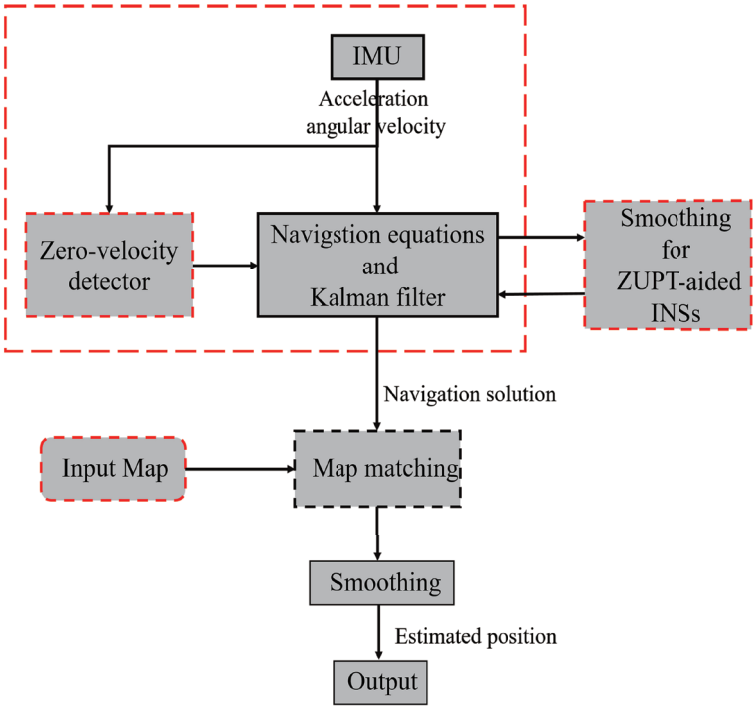


Fig. 1. System overview

2.1 Navigation Algorithm

The input data of the inertial navigation systems are acceleration and angular velocity expressed in time series. Firstly, the rotation matrix of the navigation coordinate system is calculated from the gyroscope data.

$$[\phi\ \theta\ \psi]_k^T = [\phi\ \theta\ \psi]_{k-1}^T + \omega_{k-1}\Delta t \tag{1}$$

where $\omega = [\phi\ \theta\ \psi]^T \in R^3$ is the Euler angle that describes the direction of IMU, i.e., roll, pitch and yaw. And Δt indicates the sampling time.

Next, removing the Earth's gravity, the following velocity is obtained.

$$v_k = v_{k-1} + (R_B^N a_{k-1} - [0\ 0\ g]^T)\Delta t \tag{2}$$

where $v \in R^3$ denotes the velocity, and $a \in R^3$ denotes the x-y-z acceleration and the subscript k denotes the time.

Finally, the position is obtained by integrating the velocity.

$$r_k = r_{k-1} + v_{k-1}\Delta t \tag{3}$$

where r_k denotes the position.

After two integration operations, the increasing position error due to sensor data noise and sensor bias variation. Therefore, the Smoothing for ZUPT-aided INSs algorithm is adopted to correct the inertial navigation error in this study. The zero velocity detection algorithm is the key step of the zero velocity update algorithm.

2.2 Zero Velocity Detection

In each gait cycle, the pedestrian's velocity should theoretically be zero when the sole of the foot completely touches the ground, as shown in Fig. 2. The zero velocity detection algorithm discriminates whether the pedestrian is stationary or not [27]. According to Isaac Skog, the Stance Hypothesis Optimal Detector (SHOE) is far more cost effective [27].

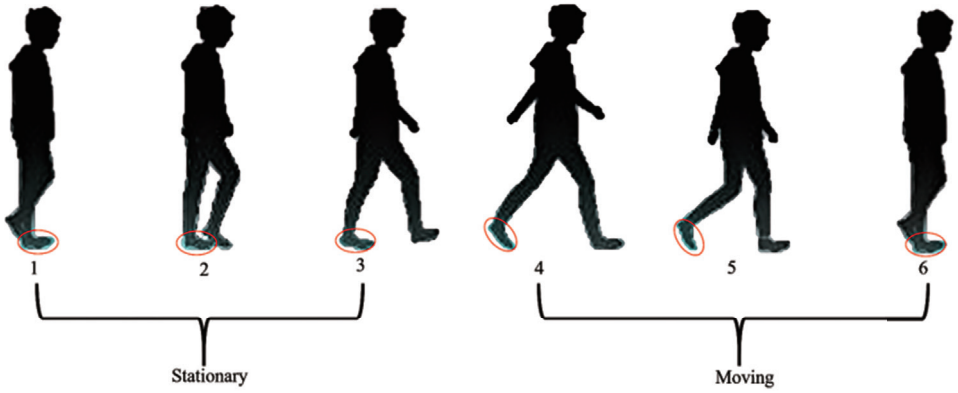


Fig. 2. Pedestrian gait cycle

The goal of the zero velocity detection is to determine whether the IMU is moving or stationary, during a time epoch consisting of $n \in N$ observation between the time instants n and $n+W-1$. The measured values of the acceleration vector and angular rate vector at this moment are denoted as Z_n^a and Z_n^ω, respectively.

$$Z_n^a \triangleq \left\{ y_k^a \right\}_{k=n}^{n+W-1} \tag{4}$$

where $y_k^a \in R^3$ denote the at the time instant $n \in N$ measured acceleration vector.

$$Z_n^\omega \triangleq \left\{ y_k^\omega \right\}_{k=n}^{n+W-1} \tag{5}$$

where $y_k^\omega \in R^3$ denote the at the time instant $n \in N$ measured angular rate vector.

The problem of detecting stationary or moving can be formulated as a binary hypothesis testing problem. The Stance Hypothesis Optimal Detector (SHOE) chooses the hypothesis that IMU is stationary if

$$T(Z_n^a, Z_n^\omega) < \gamma \tag{6}$$

where $T(Z_n^a, Z_n^\omega)$ denote the test statistics of the detector. And denote the detection threshold.

The Stance Hypothesis Optimal Detector (SHOE)

$$T(Z_n^a, Z_n^\omega) = \frac{1}{W} \sum_{k=n}^{n+W+1} \frac{1}{\sigma_a^2} \left\| y_k^a - g \frac{\overline{y}_n^a}{\left\| \overline{y}_n^a \right\|} \right\|^2 + \frac{1}{\sigma_\omega^2} \left\| y_k^\omega \right\|^2 < \gamma \tag{7}$$

where σ_a^2 denote the variance of the measurement noise of the accelerometers, and σ_ω^2 denote the variance of

the measurement noise of the gyroscopes. where $\overline{y}_n^a$ denoted the sample mean.

The zero velocity detection algorithm determines the stationary phase and uses the zero velocity update algorithm to correct the inertial navigation error during the stationary phase.

2.3 Zero Velocity Update

The zero velocity update algorithm can effectively suppress the integral accumulation error of the inertial navigation system. The structure of the zero velocity update algorithm based on Kalman filter is shown in Fig. 3. Firstly, acceleration and gyroscope data are collected via IMU. Secondly, the Stance Hypothesis Optimal Detector algorithm is used to determine the zero velocity interval. Finally, the position, velocity and attitude errors of the system are continuously corrected separately during the zero velocity interval using the zero velocity update algorithm.

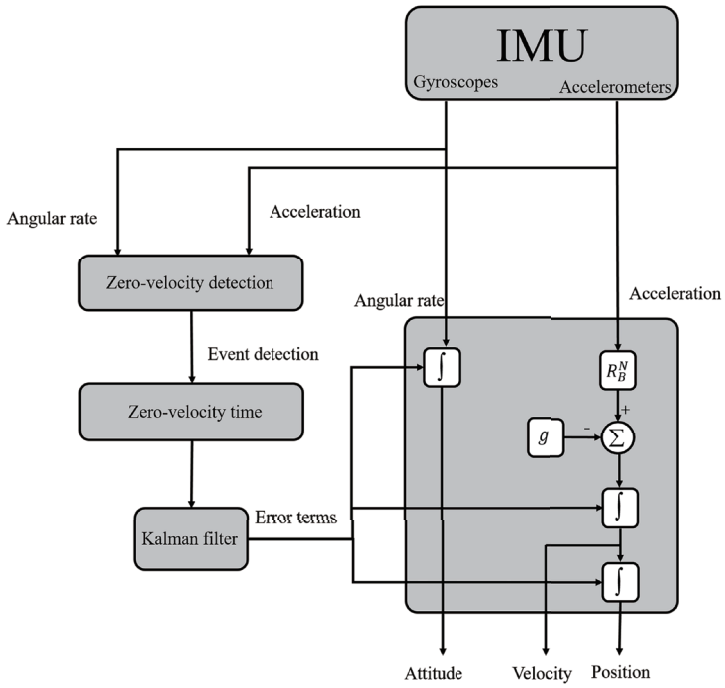


Fig. 3. Structure diagram of zero velocity update algorithm based on Kalman filter

If the 9-element error state vector at time k is

$$\delta X_k = \left[\delta L_k^T \;\; \delta \mathrm{v}_k^T \;\; \delta \theta_k^T \right]^T \tag{8}$$

where $L_n^T \in R^3$, $\mathrm{v}_n^T \in R^3$, $\theta_n^T \in R^3$ denotes the position, velocity, and attitude errors, respectively.

Then, the linearized state transition model is

$$\delta X_{k+1} = \begin{bmatrix} I_3 & T_S I_3 & 0_3 \\ 0_3 & I_3 & T_S \left[s^n \right] \times \\ 0_3 & 0_3 & I_3 \end{bmatrix} \delta X_n + \begin{bmatrix} 0_3 & 0_3 \\ \mathrm{R}_p^n & 0_3 \\ 0_3 & \mathrm{R}_p^n \end{bmatrix} w_n^{(1)} \tag{9}$$

where T_S is the sampling period, R_p^n is the rotation matrix transforming a vector from platform coordinates to

navigation coordinates. $\left[s^n\right]\times$ is the skew-symmetric matrix representation of the specific force vector. And, $w_n^{(1)} \in R^6$ is the covariance matrices of process noise and the observation noise. Here I_3 and 0_3 denote a three by three identity and zero matrices, respectively.

The measurement model is

$$Z_k = H\delta X_k + w_k \tag{10}$$

where Z_k is the error measurements, H is the measurement matrix, and w_k is the measurement noise.

In an error-state Kalman filter, inertial navigation algorithm form the process model along with error covariance matrix propagation.

$$P_k^- = \Phi_k P_{k-1} \Phi_k^T + Q_k \tag{11}$$

where P_k^- is error covariance matrix. Moreover $\Phi_k \in R^{9\times9}$ denote the state transition matrix. And, $Q_k \in R^{9\times9}$ denote stands for process covariance matrix that includes σ_ω and σ_a, gyroscope and accelerometer noises, respectively. The process covariance matrix is

$$Q_k = \begin{bmatrix} \sigma_a^2, I_3 & 0_3 \\ 0_3 & \sigma_\omega^2, I_3 \end{bmatrix} \tag{12}$$

The Kalman gain is computed as

$$K_k = P_k^- H^T (H P_k^- H^T + R)^{-1} \tag{13}$$

The Kalman gain is applied to update the error covariance.

$$P_k = (I_{9\times9} - K_k H) P_k^- \tag{14}$$

The measurement matrix $H = [0_{3\times3} \quad \mathrm{I}_{3\times3} \quad 0_{3\times3}]$ observes only velocity error, so error-state e is formed as

$$e_n = K_n v_n \tag{15}$$

where v denote the observed velocity, and $e_n \in R^{9\times9}$ is lined up as position, velocity, and attitude errors.

2.4 Smoothing for ZUPT-aided INSs

Pedestrian dead reckoning systems built with ZUPT-aided INSs has exhibited outstanding results. Such approaches, however, have failed to address the “collapses” when the step ends. This condition is shown in Fig. 4, where the large corrections leading to significant discontinuities at the end of each step.

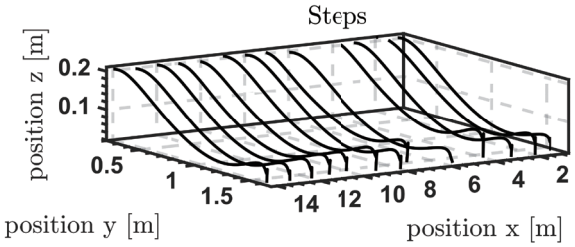


Fig. 4. Steps from a straight-line trajectory as tracked by a ZUPT-aided INS

For foot-mounted inertial navigation systems, this gives undesirable discontinuities at the end of each step. According to D. Simón Colomar, 3-pass algorithm is far more cost effective [14]. The 3-pass algorithm is given in Alg. 1.

Algorithm 1. Pseudo code for the proposed 3-pass smoothing algorithm

Initialization: $\hat{X} = E[X_0], \delta\hat{X}_0 = 0, \mathbf{P}_0 = \mathrm{var}(X_0),$
$c = 0, s_{start} = 1, s_{end} = "end \text{ of data}"$

Loop While: $s_{start} < s_{end}$

%Forward Kalman filter

Loop: $n = s_{start}$ **to** s_{end}

%Time update

$\hat{X}_n = f_{mech}(\hat{X}_{n-1}, \mathbf{f}_n, \omega_n)$

$\delta\hat{X}_{n|n-1} = \mathbf{F}_n \delta\hat{X}_{n-1|n-1}$

$\mathbf{P}_{n|n-1} = \mathbf{F}_n \mathbf{P}_{n-1|n-1}\mathbf{F}_n^T + \mathbf{G}\mathbf{Q}\mathbf{G}^T$

%Measurement update

if $T(\{\boldsymbol{\omega}^i, \mathbf{f}^i\}) < \gamma$

$\mathbf{K}_n = \mathbf{P}_{n|n-1}\mathbf{H}^T(\mathbf{H}\mathbf{P}_{n|n-1}\mathbf{H}^T + \mathbf{R})^{-1}$

$\delta\hat{X}_{n|n} = \delta\hat{X}_{n|n-1} - \mathbf{K}_n(\delta\hat{\mathrm{v}}_{n|n-1} - \hat{\mathrm{v}}_n)$

$\mathbf{P}_{n|n} = \mathbf{P}_{n|n-1}(\mathbf{I} - \mathbf{K}_n\mathbf{H})$

%Segementation rule

if $c > 0$

$c = c + T_s$

if $\left\|diag(\mathbf{P}_{n-1}^v)\right\| > \gamma_s \wedge \left\|diag(\mathbf{P}_n^{vel})\right\| \le \gamma_s \wedge c = 0$

$c = T_s$

if $c > \tau_s$

$s_{end} \leftarrow n$

break loop

%Smoothing

Loop: $n = s_{end} - 1$ **to** s_{start}

$\mathbf{A}_n = \mathbf{P}_{n|n}\mathbf{F}^T\mathbf{P}_{n+1|n}^{-1}$

$\delta\hat{X}_{n|s_{end}} = \delta\hat{X}_{n|n} + \mathbf{A}_n(\delta\hat{X}_{n+1|s_{end}} - \delta\hat{X}_{n+1|n})$

$\mathbf{P}_{n|s_{end}} = \mathbf{P}_{n|n} + \mathbf{A}_n(\mathbf{P}_{n+1|s_{end}} - \mathbf{P}_{n+1|n})\mathbf{A}_n^T$

%Internal state compensation

Loop: $n = s_{start}$ **to** s_{end}

$\begin{bmatrix}\hat{\mathbf{p}}_n \\ \hat{\mathbf{v}}_n\end{bmatrix} \leftarrow \begin{bmatrix}\hat{\mathbf{p}}_n \\ \hat{\mathbf{v}}_n\end{bmatrix} + \begin{bmatrix}\delta\hat{\mathbf{p}}_{n|s_{end}} \\ \delta\hat{\mathbf{v}}_{n|s_{end}}\end{bmatrix}$

$\hat{\mathbf{R}}_n \leftarrow (\mathbf{I}_3 - \Delta_{n|s_{end}})(\hat{\mathbf{R}}_n)$

$\delta\hat{X}_n \leftarrow 0$

$s_{start} = s_{\mathrm{end}} + 1, s_{\mathrm{end}} = "\text{end of data}", c = 0$

2.5 Program Code

Measurement of pedestrian foot movements by an IMU mounted on the pedestrian's foot. Pedestrian position update information with optimal estimation can be obtained by using dead reckoning algorithm. The map matching algorithm based on the particle filter [28-29] can determine the coordinate position and the trajectory of the pedestrian after the pedestrian walks a section of the path in the indoor. In this paper, we combine Map Matching algorithm and Smoothing for ZUPT-aided INSs algorithm (MM-SINS), thus improving the positioning accuracy.

This paper performs two-dimensional plane navigation, so the navigation position after Smoothing for ZUPT-aided INSs calculation is

$$L_k = \begin{bmatrix} x_k \\ y_k \end{bmatrix} \quad \text{k=1,2,3}\cdots N \tag{16}$$

where x_k is the x-axis coordinate point and y_k is the y-axis coordinate point.

In the particle position update algorithm, the position update information of the pedestrian optimal estimated trajectory is

$$\begin{aligned} \Delta pos_k &= L_k - L_{k\text{-}1} \\ &= (x_k - x_{k-1}, y_k - y_{k-1}) \\ &= (\Delta x_k - \Delta y_k) \end{aligned} \tag{17}$$

where Δpos_k denoted position update information for the optimal estimated trajectory for pedestrians.

During map matching, the position update information of the navigation coordinate system is converted to the position update information of the map coordinate system by

$$R_n^{map} = \begin{bmatrix} \cos(\beta) & \sin(\beta) \\ -\sin(\beta) & \cos(\beta) \end{bmatrix} \tag{18}$$

where β is the rotation angle from the navigation coordinate system to the map coordinate system.

Thus, the position of the pedestrian in the map coordinate system is updated by the vector expressed as:

$$\begin{aligned} \Delta p_k &= R_n^{map} \bullet \Delta pos_n \\ &= \left(\Delta x_k^{map}, \Delta y_k^{map}\right)^T \end{aligned} \tag{19}$$

where Δp_k is used as the input to the particle filter to update the state of the sampled particles. The flow and overall framework of map matching based on particle filter is shown in Fig. 5.

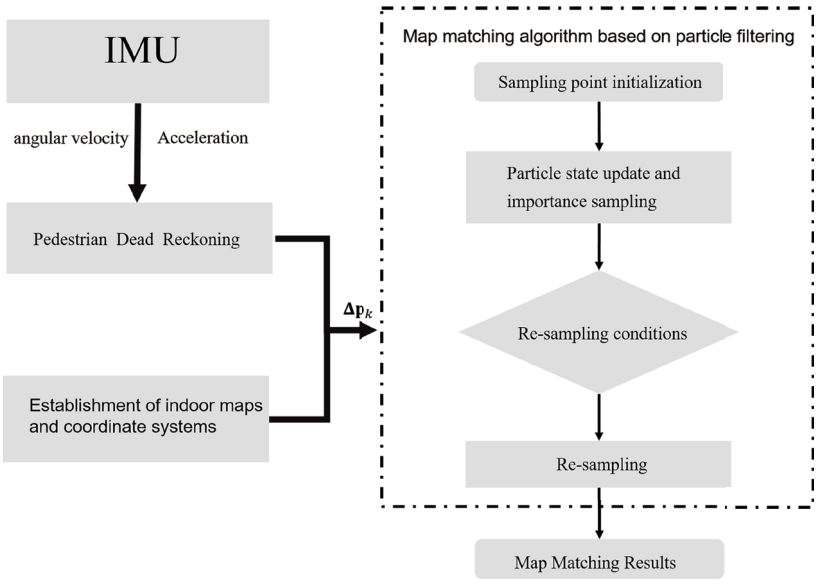


Fig. 5. Map matching algorithm based on particle filter

The particle filter method is based on Monte Carlo simulation theory, which uses a statistical state for estimating the system state, and its corresponding probability density function is numerically approximated. To define the problem during location estimation, the target state and measurement evolves according to the following discrete-time stochastic model:

$$x_k = f_k(x_{k-1}) + W_k \tag{20}$$

$$z_k = h_k(x_{k-1}) + V_k \tag{21}$$

where $f(\cdot)$ and $h(\cdot)$ are the state equation and observation equation of the system, respectively. Where x_k is the system state vector at moment k, and z_k is the observation value. Moreover W_k and V_k denote the system noise and observation noise, respectively.

The state prediction equation is

$$P(x_k \mid z_{1:k-1}) = \int P(x_k \mid x_{k-1}) P(x_{k-1} \mid z_{1:k-1}) dx_{k-1} \tag{22}$$

where $P(x_k \mid x_{k-1})$ is the transition probability density function.

The state update equation is

$$P(x_k \mid z_{1:k}) = \frac{P(z_k \mid x_k) P(x_k \mid z_{1:k-1})}{P(z_k \mid z_{1:k-1})} \tag{23}$$

with the normalizing constant is

$$P(z_k \mid z_{1:k-1}) = \int P(z_k \mid x_k) P(x_k \mid z_{1:k-1}) dx_k \tag{24}$$

where $P(z_k \mid x_k)$ represents the likelihood function, $P(x_k \mid z_{1:k-1})$ is the previous posterior distribution.

Equations (22) and (23) are recursive methods for finding the posterior probability, but they are difficult to implement due to integration. Thus, the particle filter directly estimates the posterior probability density function of the state using the following equation:

$$P(x_k \mid z_k) \approx \sum_{i=1}^{N} \omega_t \delta(x_k - x_k^i) \tag{25}$$

Regardless of system noise and observation noise, the state vector of each particle is the coordinate position in its position diagram. The equation for updating the state of each particle is

$$x_k^i = x_{k-1}^i + \Delta p_k \tag{26}$$

Map matching algorithm based on particle filter implements a fairly straightforward idea. After a given map constraint, new particles should not occupy impossible positions. This paper adopts the line segment intersection algorithm. The line segment intersection is passing through the wall, as shown in Fig. 6.

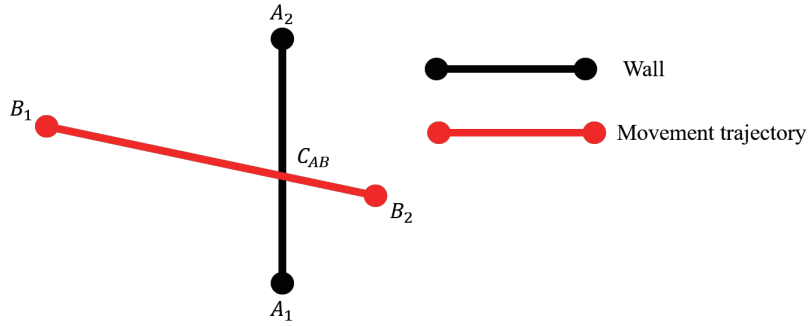


Fig. 6. Schematic diagram of line segment intersection

where the lines A and B and their intersection C is related as follows:

$$C_{AB} = B_1 + \gamma(B_2 - B_1) = A_1 + \lambda(A_2 - A_1) \tag{27}$$

Here, γ and λ obtained from equations (27) and (28), respectively.

$$\gamma = \frac{(A_1 - A_2)_y (A_1 - B_1)_x - (A_1 - A_2)_x (A_1 - B_1)_y}{(A_1 - A_2)_y (B_2 - B_1)_x - (A_1 - A_2)_x (B_2 - B_1)_y} \tag{28}$$

$$\lambda = \frac{(B_2 - B_1)_x (A_1 - B_1)_y - (B_2 - B_1)_y (A_1 - B_1)_x}{(A_1 - A_2)_y (B_2 - B_1)_x - (A_1 - A_2)_x (B_2 - B_1)_y} \tag{29}$$

If γ and λ are greater than 0 and 1 at the same time, the two line segments intersect. The line segment intersection algorithm is used to detect the through wall points, and the particle through wall detection results are shown in Fig. 7.

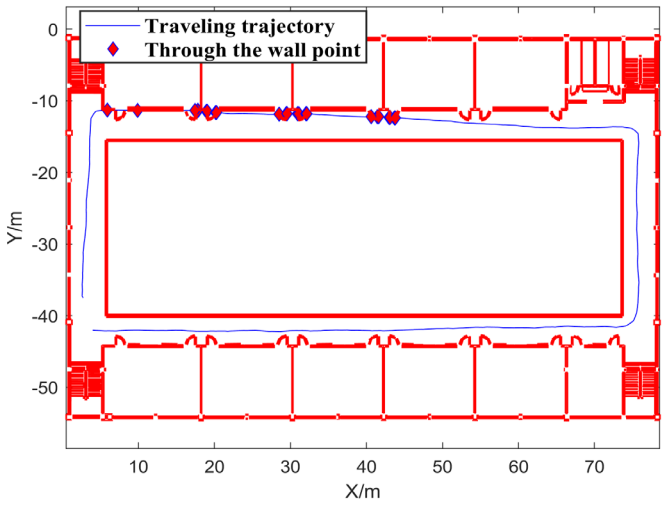


Fig. 7. Motion trajectory through wall detection

Map matching algorithm based on particle filter implements a fairly straightforward idea. After a given map constraint, new particles should not occupy impossible positions. After the particle update, the remaining particle weights are then normalized. the particle weight will also be changed according to the following rule:

$$\omega_k^i = \begin{cases} 0, & \text{if new particle crpssed a wall} \\ 1/N, & \text{otherwise} \end{cases} \tag{30}$$

The position update information of the pedestrian is obtained according to Equation (19), and the sampled particle state is updated in each iteration step. It also determines whether the particle passes through the wall or not, and then reassigns the weights according to the Equation (30) in the importance sampling.

3 Lab Experiment

The performance of the combined Smoothing for ZUPT-aided INSs and Map matching based pedestrian inertial indoor localization method was evaluated using NGIMU from x-io Technologies Limited as an inertial navigation sensor device in an actual indoor test. As shown in Fig. 8, the NGIMU is mounted on the foot.

Fig. 8. The NGIMU mounted on the foot

In the first test, we evaluated our system in an indoor building. As shown in Fig. 9, pedestrians were asked to start from the blue point, walk along the red arrow, and finally walk back to the starting point along the red arrow. In this experiment, the width of the corridor of the indoor building was about 2 m and the distance traveled was about 215.94 m.

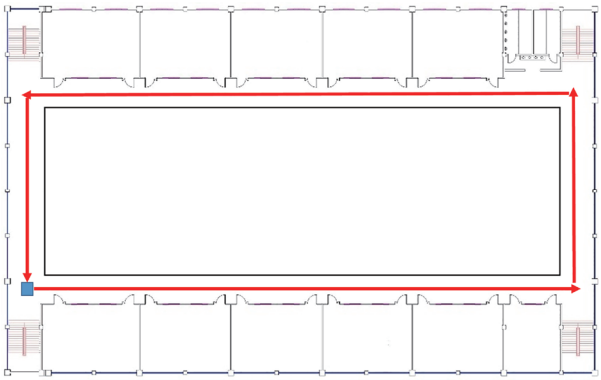


Fig. 9. Designed test 1 trajectory

The collected three-axis acceleration data and three-axis gyroscope data are input into the Stance Hypothesis Optimal Detector (SHOE). The zero velocity interval is detected by the zero velocity detector algorithm. The zero velocity detector results of test 1 are shown in Fig. 10. The x-axis is the experimental row of people walking time, and the y-axis is the switch of the zero velocity interval.

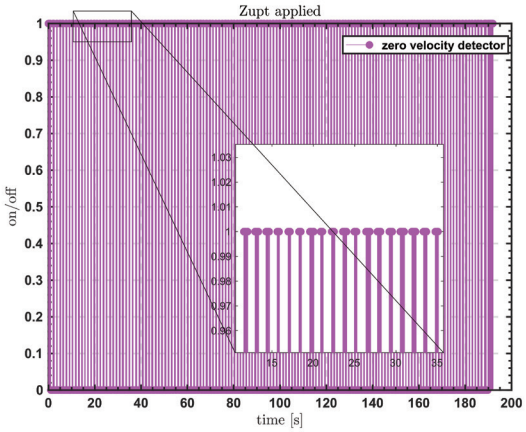


Fig. 10. Designed test 1 trajectory

The acceleration and gyroscope data are collected via IMU. The Stance Hypothesis Optimal Detector algorithm is used to determine the zero-velocity interval. Then, the position, velocity and attitude errors of the system are continuously corrected separately using the smoothing for ZUPT-aided INSs algorithm. Fig. 11 illustrates the result of the smoothing for ZUPT-aided INS.

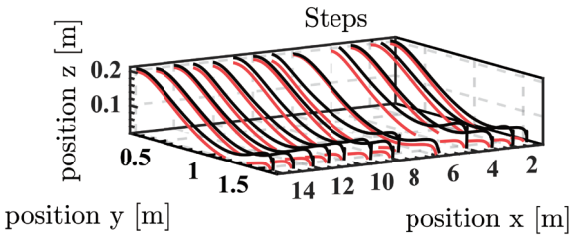


Fig. 11. The result of the smoothing for ZUPT-aided INS

The simulation results of test 1 are shown in Fig. 12. The black trajectory is the trajectory without smooth treatment, and the red trajectory is the trajectory with smooth treatment. It can be clearly observed from the partial enlargement in the figure that the smoothing process can make the trajectory smooth.

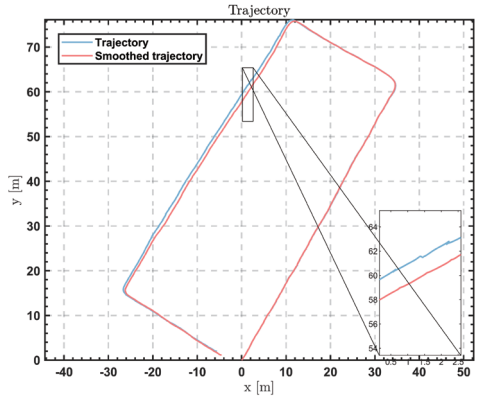


Fig. 12. The estimated trajectory of Smoothing for ZUPT-aided INSs for test 1

Set the number of particles to 350. The results of test 1 map matching are shown in Fig. 13. The blue trajectories are the results of Smoothing for ZUPT-aided INSs, and the green trajectories are the results of MM-SINS method.

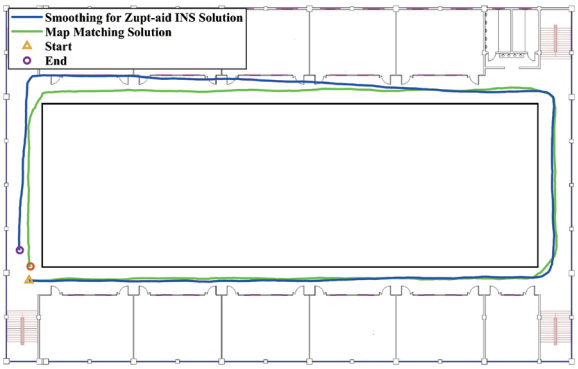


Fig. 13. Estimated trajectory for test 1

We further validate the effectiveness of the MM-SINS method by increasing the walking distance. The test 2 walked one more lap according to the walking track of test 1, and the walking distance of test 2 was about 431.88 m. The test 3 walked three laps, and the walking distance of test 3 was about 647.82 m. The test 4 walked three laps, and the walking distance of test 4 was about 863.76 m. The test 2, test 3, test 4 walking average speeds of 1.20m/s, 0.93m/s,0.93m/s, respectively. The number of particles used in this test is 350. The test 2,3,4 trajectory simulation results are shown in Fig. 14, Fig. 15, Fig. 16, respectively.

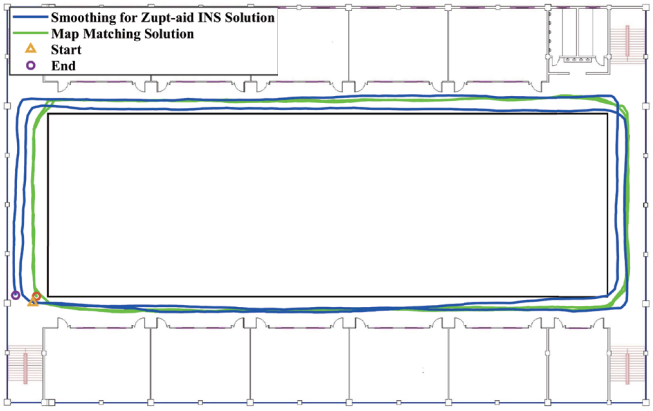


Fig. 14. Estimated trajectory for test 2

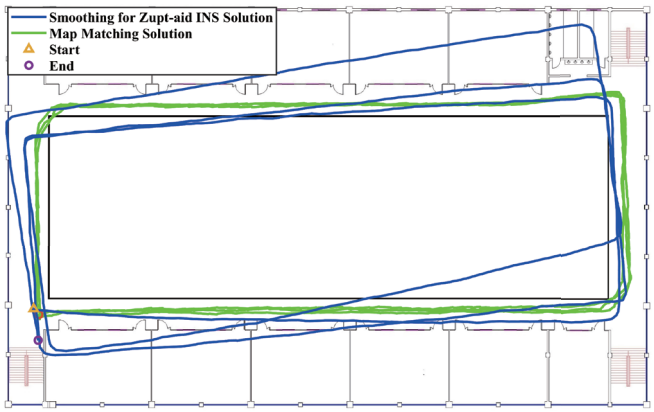


Fig. 15. Estimated trajectory for test 3

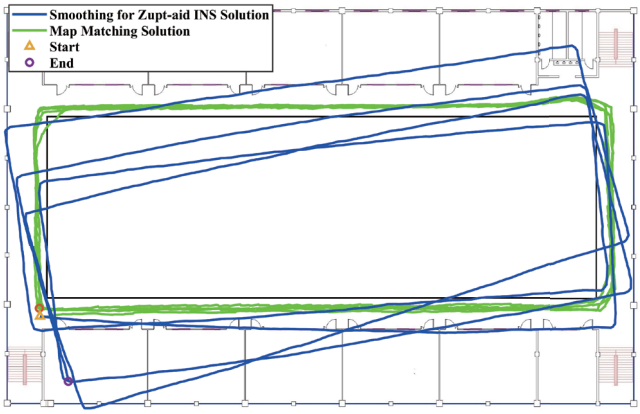


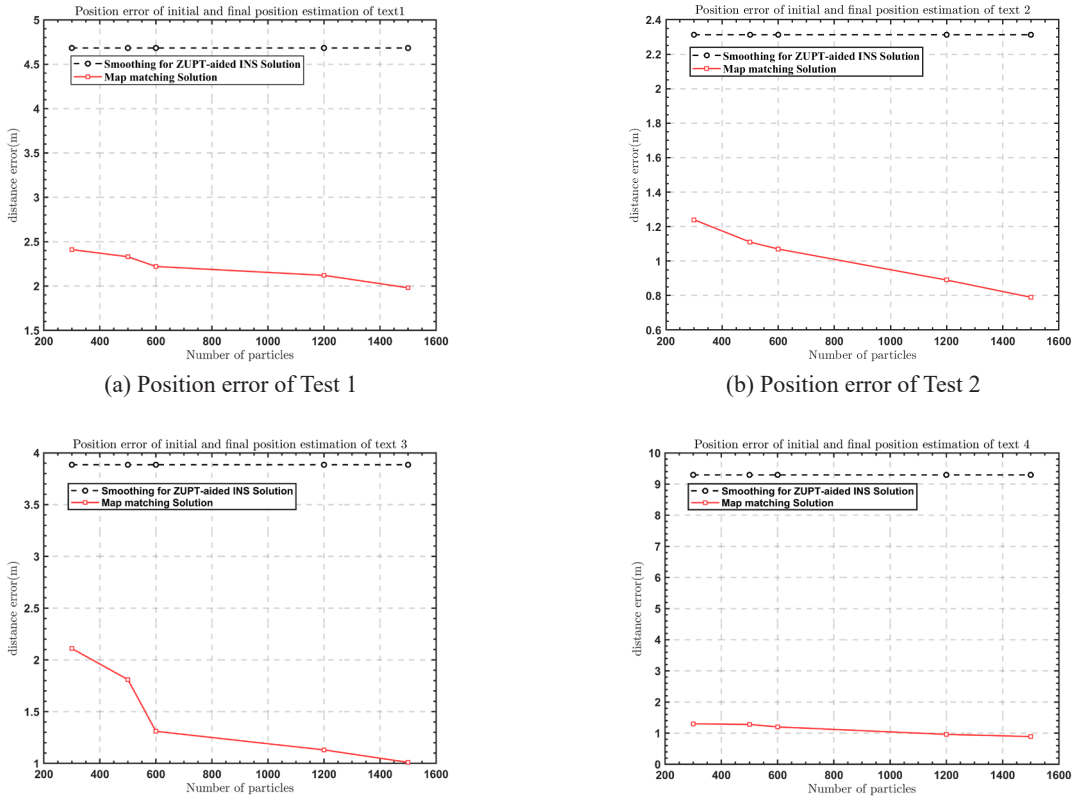
Fig. 16. Estimated trajectory for test 4

All tests start and end at the same position. Hence, the position error of the final position estimate is evaluated by the difference between the initial and final position estimates. In order to analyze the effect of the MM-SINS method on the navigation accuracy, different numbers of particles (N=300, 500, 600, 1200, 1500) were used in the four tests respectively to verify its effect on the localization error. Table 1 illustrates the results of the MM-SINS and SINS estimation solution.

Table 1. Indoor positioning performance using different particles

Particles	Algorithm	Error(m)			
		Test 1	Test 2	Test 3	Test 4
300	MM-SINS	2.41	1.24	2.11	1.30
	SINS	4.68	2.31	3.88	9.29
500	MM-SINS	2.33	1.11	1.81	1.28
	SINS	4.68	2.31	3.88	9.29
600	MM-SINS	2.22	1.07	1.31	1.20
	SINS	4.68	2.31	3.88	9.29
1200	MM-SINS	2.12	0.89	1.13	0.96
	SINS	4.68	2.31	3.88	9.29
1500	MM-SINS	1.98	0.79	1.01	0.89
	SINS	4.68	2.31	3.88	9.29

As shown in Table 1, the error of MM-SINS estimation decreases with the increasing of the number of particles. From the expression of equation (25), we can also deduce that the right part of equation (25) is almost same as its left part when the number of particles N tends to infinite. Both the experiment results and theorical analysis indicate that the particles in PF represent the probability density function of the system. The error of the MM-SINS method and SINS method are shown in Fig. 17(a) to Fig. 17(d).



(a) Position error of Test 1

(b) Position error of Test 2

(c) Position error of Test 3

(d) Position error of Test 4

Fig. 17. Position error of initial and final position

When the particle number is 1500, the average position error of MM-SINS method for four tests is 1.16 m compared to the SINS method. The distance error of MM-SINS method is less than that of SINS method. This shows that the MM-SINS algorithm can effectively correct the position error of the SINS algorithm. For four indoor experiments, the distance error of the MM-SINS method decreases with the increase of the number of particles.

4 Conclusions

This study set out to build a complete personal indoor positioning navigation system. Based on the result of real-world experiments, the following conclusion can be summarized as: an ideal navigation calculating precision can be achieved by the proposed method; and the accuracy of the SINS method navigation solution can be improved sharply.

This study has shown that an effective and accurate indoor positioning and navigation system can be built by fusing inertial measurement unit information with map information. The findings will be of interest to indoor pedestrian navigation system. It is un-fortunate that the study did not include reducing the time redundancy of the particle filter algorithm. Despite its limitations, the study will undoubtedly deepen our under-standing of inertial measurement units and plane map information fusion navigation systems. Further research should be conducted to explore reducing the computational burden of the particle filter algorithm thereby improving navigation efficiency. The results of this paper have practical implications for solving indoor navigation problems, where only simple indoor plane information is needed to obtain high navigation accuracy.

5 Data Availability

The data can be obtained at https://github.com/Evangear/Pedestrian-Inertial-Navigation-with-Building-Floor-Plans-for-Indoor-Environments-via-particle-filte.

6 Disclosure

Funders have no role in research design; collect, analyze or interpret data; At the time of writing or deciding to publish the results.

7 Acknowledgments

We are very grateful to all members of the research team led by J.-X Guo for their valuable suggestions. This research was funded by the foundation of Shaanxi Key Laboratory of Integrated and Intelligent Navigation [grant number SKLIIN-20190102]. Natural Science Foundation of Shaanxi Province [grant numbers 2021JM-537, 2019JQ-936], and the Research Foundation for Talented Scholars of Xijing University [grant numbers XJ20B01, XJ19B01, XJ17B06], and the Key R&D Program of Shaanxi Province, China [grant number 2021GY-341].

References

[1] B. Meng, K. Choi, Tourists' intention to use location-based services (LBS): Converging the theory of planned behavior (TPB) and the elaboration likelihood model (ELM), International Journal of Contemporary Hospitality Management 31(8) (2019) 3097-3115.
[2] C. Lu, H. Uchiyama, D. Thomas, A. Shimada, R.I. Taniguchi, Indoor positioning system based on chest-mounted IMU, Sensors 19(2)(2019) 420.
[3] T. Xie, H. Jiang, X. Zhao, C. Zhang, A Wi-Fi-Based wireless indoor position sensing system with multipath interference mitigation, Sensors 19(18)(2019) 3983.
[4] A.R.J. Ruiz, F.S. Granja, Comparing ubisense, bespoon, and decawave uwb location systems: Indoor performance analysis, IEEE Transactions on instrumentation and Measurement 66(8)(2017) 2106-2117.
[5] Y. Wang, X. Yang, Y. Zhao, Y. Liu, L. Cuthbert, Bluetooth positioning using RSSI and triangulation methods, in: Proc. 2013 IEEE 10th Consumer Communications and Networking Conference (CCNC), 2013.
[6] Q. Wang, M. Cheng, A. Noureldin, Z. Guo, Research on the improved method for dual foot-mounted Inertial/Magnetometer pedestrian positioning based on adaptive inequality constraints Kalman Filter algorithm, Measurement 135(2019) 189-198.
[7] W. Zhang, D. Wei, H. Yuan, The improved constraint methods for foot-mounted PDR system, IEEE Access 8(2020) 31764-31779.
[8] A. Patarot, M. Boukallel, S. Lamy-Perbal, A case study on sensors and techniques for pedestrian inertial navigation, in: Proc. 2014 International Symposium on Inertial Sensors and Systems (ISISS), 2014.
[9] D. Xu, Y. Ding, S. Ma, J. Wang, H. Zhao, Anti-Magnetic Disturbance Pedestrians Navigation System Based on MEMS Inertial Sensors, in: Proc. 2018 IEEE Sensors, 2018.
[10]W. Zhang, X. Li, D. Wei, X. Ji, H. Yuan, A foot-mounted pdr system based on imu/ekf+ hmm+ zupt+ zaru+ hdr+ compass algorithm, in: Proc. 2017 International conference on indoor positioning and indoor navigation (IPIN), 2017.
[11]X. Niu, Y. Li, J. Kuang P. Zhang, Data fusion of dual foot-mounted IMU for pedestrian navigation, IEEE Sensors Journal 19(12)(2019) 4577-4584.
[12]Y. Wang, A. Chernyshoff, A.M. Shkel, Error analysis of ZUPT-aided pedestrian inertial navigation, in: Proc. 2018 International Conference on Indoor Positioning and Indoor Navigation (IPIN), 2018.
[13]H. Ju, C.G. Park, A pedestrian dead reckoning system using a foot kinematic constraint and shoe modeling for various motions, Sensors and Actuators A: Physical 284(2018) 135-144.
[14]D.S. Colomar, J.O. Nilsson, P. Händel, Smoothing for ZUPT-aided INSs, in: Proc. 2012 International Conference on Indoor Positioning and Indoor Navigation (IPIN), 2012.
[15]C. Yu, N. El-Sheimy, H. Lan, Z. Liu, Map-based indoor pedestrian navigation using an auxiliary particle filter, Micromachines 8(7)(2017) 225.
[16]P.K. Yoon, S. Zihajehzadeh, B.S. Kang, E.J. Park, Adaptive Kalman filter for indoor localization using Bluetooth Low Energy and inertial measurement unit, in: Proc. 2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), 2015.
[17]J.D. Ceron, F. Kluge, A. Küderle, B.M. Eskofier, D.M. López, Simultaneous Indoor Pedestrian Localization and House Mapping Based on Inertial Measurement Unit and Bluetooth Low-Energy Beacon Data, Sensors 20(17)(2020) 4742.
[18]L. Jian, K. Yuchen, WiFi/Inertial fusion positioning using linear Kalman filter, Electronic Measurement Technology, 2017.
[19]Z.A. Deng, Y. Hu, J. Yu, Z. Na, Extended Kalman filter for real time indoor localization by fusing WiFi and smartphone inertial sensors, Micromachines 6(4)(2015) 523-543.
[20]S. Zhong, K. Zhang, S. Liu, UWB-Inertial Fusion Location Algorithm Based on Kalman Filtering, in: Proc. 2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV), 2018.
[21]L. Yao, Y.W. A. Wu, L. Yao, Z.Z. Liao, An integrated IMU and UWB sensor based indoor positioning system, in: Proc.

2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN), 2017.
[22]A. Perttula, H. Leppäkoski, M. Kirkko-Jaakkola, P. Davidson, J. Collin, J. Takala, Distributed indoor positioning system with inertial measurements and map matching, IEEE Transactions on Instrumentation and Measurement 63(11)(2014) 2682-2695.
[23]S. Lamy-Perbal, N. Guénard, M. Boukallel, A. Landragin-Frassati, A HMM map-matching approach enhancing indoor positioning performances of an inertial measurement system, in: Proc. 2015 International Conference on Indoor Positioning and Indoor Navigation, 2015.
[24]N. Kim, Y. Kim, Indoor Positioning System with IMU, Map Matching and Particle Filter, Recent Advances in Electical Engineering and Computer Science, 2015.
[25]C. Lu, H. Uchiyama, D. Thomas, A. Shimada, R. I. Taniguchi, Indoor positioning system based on chest-mounted IMU, Sensors 19(2)(2019) 420.
[26]H. Lan, C. Yu, Y. Zhuang Y. Li, N. El-Sheimy, A novel kalman filter with state constraint approach for the integration of multiple pedestrian navigation systems, Micromachines 6(7)(2015) 926-952.
[27]I. Skog, P. Handel, J.O. Nilsson, J. Rantakokko, Zero-velocity detection–An algorithm evaluation, IEEE Transactions on Biomedical Engineering 57(2010) 2657-2666.
[28]C. Yu, N. El-Sheimy, H. Lan, Z. Liu, Map-based indoor pedestrian navigation using an auxiliary particle filter, Micromachines 8(7)(2017) 225.
[29]M. Klepal, S. Beauregard, A backtracking particle filter for fusing building plans with PDR displacement estimates, in: Proc. 2008 5th Workshop on Positioning, Navigation and Communication, 2008.