Journal of Systems Engineering and Electronics
Vol. 33, No. 3, June 2022, pp.727 – 736

Heading constraint algorithm for foot-mounted PNS using low-cost IMU

GUI Jing, ZHAO Heming*, and XU Xiang

School of Electronic and Information Engineering, Soochow University, Suzhou 215006, China

Abstract: Foot-mounted pedestrian navigation system (PNS) is a common solution to pedestrian navigation using micro-electro mechanical system (MEMS) inertial sensors. The inherent problems of inertial navigation system (INS) by the traditional algorithm, such as the accumulated errors and the lack of observation of heading and altitude information, have become obstacles to the application and development of the PNS. In this paper, we introduce a heuristic heading constraint method. First of all, according to the movement characteristics of human gait, we use the generalized likelihood ratio test (GLRT) detector and introduce a time threshold to classify the human gait, so that we can effectively identify the stationary state of the foot. In addition, based on zero velocity update (ZUPT) and zero angular rate update (ZARU), the cumulative error of the inertial measurement unit (IMU) is limited and corrected, and then a heuristic heading estimation is used to constrain and correct the heading of the pedestrian. After simulation and experiments with low-cost IMU, the method is proved to reduce the localization error of endpoint to less than 1% of the total distance, and it has great value in application.

Keywords: pedestrian navigation system (PNS), zero velocity update (ZUPT), gait detection, heading constraint.

DOI: 10.23919/JSEE.2022.000067

1. Introduction

With the development of portable sensing technology, pedestrian navigation and positioning technologies have increasingly become a hot topic in the area of industry and academia [1], and there are already a wide range of solutions available.

A traditional practice is to equip with global navigation satellite system (GNSS); however, weak satellite signals in urban canyons and indoor environments render GNSS-based methods ineffective in such scenarios. On top of these GNSS-based methods, current navigation technologies can be roughly divided into three categories: intersection processing technology based on wireless communication signals, map matching technology based on database and dead reckoning (DR) technology based on micro-electro mechanical system (MEMS) inertial sensors [2].

Bluetooth, ZigBee and ultra wide band (UWB) positioning technology, based on wireless communication signals, are mainly used to locate pedestrians by establishing multiple wireless network positioning nodes [2]. Among them, the UWB system has many advantages such as high positioning accuracy, low power consumption and good penetration. However, it requires very precise clock synchronization and high cost, which is not conducive to widespread application.

WIFI fingerprint matching and simultaneous localization and mapping (SLAM) technology [3,4], based on database, are currently successful indoor positioning systems. Nonetheless, these methods have the drawback of requiring the creation and maintenance of the relevant databases.

The DR technology based on MEMS inertial sensors, which offer a beacon-free solution, is an independent navigation and positioning technology [5,6]. Due to its small size, low power consumption, low cost, and easy digitization, MEMS inertial sensors are widely used in the field of navigation and positioning with strict hardware cost and volume requirements. Nevertheless, the output of the MEMS sensors typically contains zero bias, bias instability, and other errors, which are integrated through the navigation equation to generate position errors that accumulate with time [7,8]. Thus, it is necessary to use other methods to reduce the errors due to this integral drift and improve the pedestrian DR (PDR) system using MEMS inertial sensors.

Foxlin proposed a foot-mounted pedestrian navigation method to estimate the position by the zero velocity update (ZUPT) algorithm [9] in 2005, and pointed out that the ZUPT method is effective in reducing the cumulative error of inertial sensors [10,11]. The principle of

Manuscript received April 20, 2021.
*Corresponding author.
This work was supported by the National Natural Science Foundation of China (61803278).

the ZUPT algorithm comes from the fact that when pedestrians walk, a stance phase occurs, and at that moment, the velocity of the foot is practically zero [12]. At this point, an acceleration integral could be avoided. However, the heading angle errors still cannot be well corrected, which is due to the unobservability of the heading angle error itself. Therefore, how to reduce the heading angle error to a reasonable range is a major challenge in pedestrian navigation research. Many methods use additional sensors (e.g., magnetometers [12−14]) to provide accurate heading information. If no additional sensor is equipped, many academic teams also have done various attempts and studies. Based on the assumption that the angular rate change is zero when a stance phase occurs, the zero angular rate update (ZARU) method has been proposed to limit the accumulation of gyroscopic errors. In addition, Borenstein et al. [15] corrected heading errors by taking into account the fact that building usually has four directions and most pedestrians travel along the corridor. This method is known as the heuristic drift elimination (HDE) method, and Jimenez et al. [16] proposed an improved HDE (iHDE) method based on this.

Thereupon, according to the characteristics of pedestrian movement, this paper uses the generalized likelihood ratio test (GLRT) method proposed by Skog et al. [17] and introduces a time threshold to detect the gait of the person, so as to accurately identify the stance phase of the foot. In addition, the ZUPT and ZARU methods are integrated to limit and correct the cumulative errors of the IMU, and then the heuristic heading constraint method is used to correct the direction of pedestrian movement. After testing and verification, this method can well restrain heading divergence and reduce positioning error.

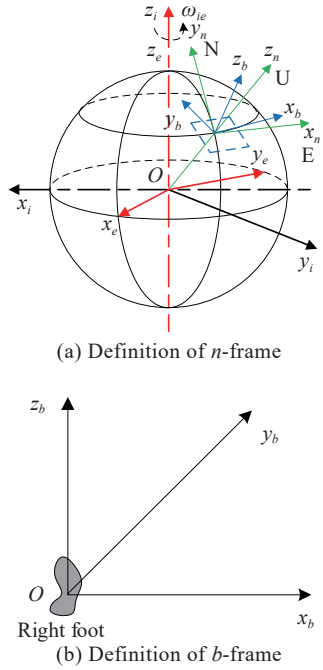
This paper is organized as follows. Section 2 presents the initial information processing. Section 3 introduces the gait detection. Section 4 expounds the proposed heading constraint method and pedestrian navigation framework. The simulation and experimental results are presented in Section 5. Finally, Section 6 concludes this paper.

2. Initial information processing

2.1 Self-alignment

This article selects the East-North-Up geographical coordinate system as the navigation coordinate system (n-frame) and the Right-Front-Up orientation of the pedestrian's right foot as the body coordinate system (b-frame). The definition of n-frame and b-frame are shown in Fig. 1.

Use the initial stationary phase to estimate the initial zero bias of the gyroscope and accelerometer as $\boldsymbol{b}_{g_0}$ and $\boldsymbol{b}_{a_0}$.



(a) Definition of n-frame

(b) Definition of b-frame

Fig. 1 Diagram of different coordinate system

Self-alignment usually includes leveling and compass alignment. The former initializes the roll angle and the pitch angle, while the latter initializes the yaw angle. Normally leveling is done first, followed by compass alignment [18].

According to the leveling principle, the pitch angle θ and roll angle ϕ in the initial state can be obtained by using the accelerometer output in the stationary state.

$$\theta_0 = \arctan\left(\frac{f_y^b}{\sqrt{(f_x^b)^2+(f_z^b)^2}}\right) \tag{1}$$

$$\phi_0 = -\arctan\left(\frac{f_x^b}{f_z^b}\right) \tag{2}$$

Due to the comparatively large gyroscopic drift of the consumer inertial sensor MPU9250, it is not capable of compass alignment. Therefore, it is necessary to assume that the initial heading angle is zero or to obtain the initial yaw angle ψ_0 with additional equipment.

2.2 Navigation calculation

Firstly, the attitude transfer matrix is updated using the angular rate measured by the gyroscope [19,20].

$$\boldsymbol{C}_{b,k+1}^n = \boldsymbol{C}_{b,k}^n(2\boldsymbol{I}_{3\times3}+\boldsymbol{\Omega}\Delta t)(2\boldsymbol{I}_{3\times3}-\boldsymbol{\Omega}\Delta t)^{-1} \tag{3}$$

where $\boldsymbol{C}_b^n$ is the attitude transfer matrix, $\boldsymbol{\Omega}$ is the antisym-

metric array of $\boldsymbol{\omega}_{ib}^{b}$.

Due to the low accuracy of consumer grade MEMS inertial sensors, the Earth's rotation rate can usually be ignored to simplify the calculation. And the specific force measurements are averaged over time, so the coordinate transformation matrix should also be averaged, from which the acceleration values in n-frame can be derived.

$$\boldsymbol{f}^{n}=\frac{1}{2}(\boldsymbol{C}_{b,k+1}^{n}+\boldsymbol{C}_{b,k}^{n})\boldsymbol{f}^{b}, \tag{4}$$

$$\boldsymbol{a}^{n}=\boldsymbol{f}^{n}+\boldsymbol{g}^{n}, \tag{5}$$

where $\boldsymbol{g}^{n}=\begin{bmatrix}0 & 0 & -g\end{bmatrix}$ and g is the local gravity acceleration.

The simplified equations for velocity and position updates are as follows:

$$\boldsymbol{v}_{k+1}^{n}=\boldsymbol{v}_{k}^{n}+\frac{1}{2}(\boldsymbol{a}_{k+1}^{n}+\boldsymbol{a}_{k}^{n})\Delta t, \tag{6}$$

$$\boldsymbol{p}_{k+1}^{n}=\boldsymbol{p}_{k}^{n}+\frac{1}{2}(\boldsymbol{v}_{k+1}^{n}+\boldsymbol{v}_{k}^{n})\Delta t. \tag{7}$$

3. Gait detection

The process of walking can be seen as a cyclical movement of alternately falling and lifting the feet and each complete gait can usually be divided into four processes, as shown in Fig. 2. Or simplify it to two processes of the stance phase and the swing phase.

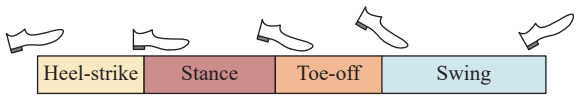


Fig. 2 Pedestrian gait classification

An important prerequisite of pedestrian navigation is to accurately identify the stance phase, which is usually detected by distinguishing between changes in acceleration and angular velocity of the inertial sensors. Generally, the following two conditions must be met.

(i) Angular velocity conditions: $\left\|\boldsymbol{\omega}_{k}^{b}\right\|=0$;

(ii) Acceleration conditions: $\left\|\boldsymbol{a}_{k}^{b}\right\|=g$.

The logic signal is used to divide whether it is in a stance phase. When a stance phase occurs, the detection signal is 1; otherwise, when a swing phase occurs, the detection signal is 0.

The traditional single-threshold gait detection algorithms mainly include acceleration variance detection, acceleration amplitude detection, angular rate energy detection, and GLRT method. In some specific kinematic conditions such as going up and down the stairs or running, it is difficult to achieve accurate gait detection with a single acceleration or angular rate detection method. Therefore, the GLRT method is a good choice. It combines the first three methods through hypothesis testing and it is judged as a stance phase when all three are satisfied at the same time. The final formula is given by the method of statistical test [17] as follows:

$$T_{n}(\boldsymbol{f}_{k},\boldsymbol{\omega}_{k})=\frac{1}{W}\sum_{k=n}^{n+W-1}\left(\frac{1}{\sigma_{f}^{2}}\left\|\boldsymbol{f}_{k}-g\frac{\overline{\boldsymbol{f}}_{k}}{\left\|\overline{\boldsymbol{f}}_{k}\right\|}\right\|^{2}+\frac{1}{\sigma_{\omega}^{2}}\left\|\boldsymbol{\omega}_{k}\right\|^{2}\right). \tag{8}$$

Select T_d as the threshold value for gait detection, when $T_n \leqslant T_d$, the pedestrian gait is judged to be in a stance phase and the output is a logic value of 1; otherwise, the pedestrian is in a swing phase and the output is 0.

Due to the gait characteristics of pedestrians and the performance limitations of consumer-grade inertial sensors, when only the GLRT method is used to detect the stance phase, there will be phenomena as shown in Fig. 3. In order to overcome this shortcoming, after comprehensive consideration, this article adds time threshold and gait classification based on the GLRT method, and performs multiple detections on gait to enhance the accuracy. Alternatively, between the toe-off and the next heel-strike can be considered a swing phase. To avoid outliers, only the stance phase between the heel-strike and the next toe-off is detected. The results of the gait detection and the intercepts are shown in Fig. 4, where the grey area is the stance phase and represents the detection signal of 1.

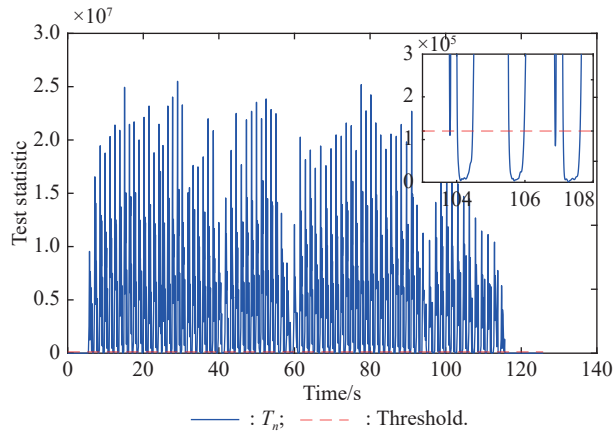


Fig. 3 Phenomenon where the swing state is below a threshold value

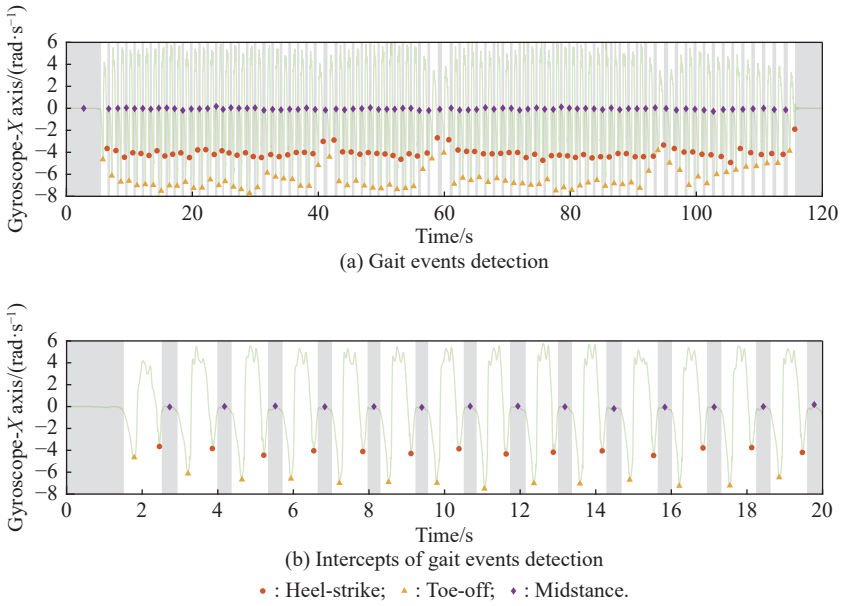


Fig. 4 Results of the gait detection and the intercepts

4. Proposed heuristic heading constraint

4.1 Straight detection

The original HDE method and iHDE algorithm can work well on a linear path in four or eight predefined major directions. Since most of the buildings are frame-type buildings, the corridors and walls are mostly either parallel or vertical. When pedestrians walk indoors, most of them walk along the corridors. Therefore, for the main direction of the building, these eight directions are defined as the main headings of pedestrian movement (as shown in Fig. 5). The difference between the two main directions is $\Delta\theta$, generally $\Delta\theta$ equals 30° or 45°. When pedestrians walk in a straight line, we can use the matched main direction to restrict the pedestrian's heading.

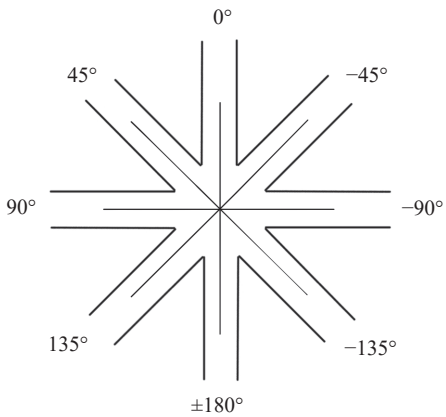


Fig. 5 Common eight pre-defined major directions

However, there are usually numerous irregular paths in the outdoor environment. As shown in Fig. 6, it is difficult for us to predefine the dominant direction of all straight paths.

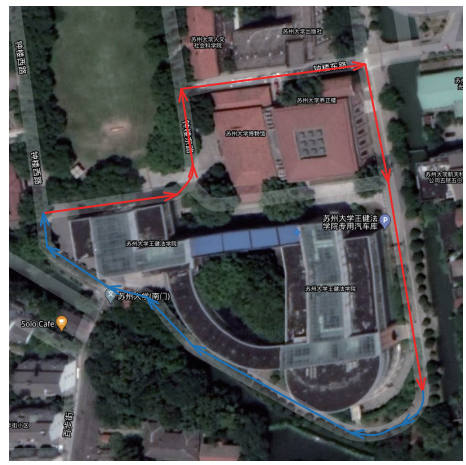


Fig. 6 Diagram of the walking route under Google maps

Thereupon, this article introduces a straight forward detection method to detect whether pedestrians are walking straight. Taking Fig. 6 as an example, the pedestrian walks in a straight line from the starting point, and the average of the initial three-step directions is used to establish the initial dominant direction of the linear path 1. If the proposed algorithm detects that the pedestrian starts to walk along the curved path 2, the heading correction is stopped.

When the proposed algorithm again detects that the pedestrian is walking along a straight path (e.g., straight line path 3), a temporary dominant direction is established. In contrast to the method for determining the initial dominant direction, the subsequent dominant direction is

given by a combination of the dominant direction of the previous straight path and the direction change of the turning path.

Firstly, the walking direction of each step $\psi_s(n)$ is defined as the heading angle at the end of the stance phase.

The dominant direction is calculated by the following formula:

$$\psi_{\text{main}}(m)=\begin{cases}\dfrac{1}{3}\sum_{j=1}^{3}\psi_s(j), & m=1\\ \psi_{\text{main}}(m-1)+\Delta\psi, & m>1\end{cases}. \tag{9}$$

Judge whether the pedestrian is going straight within three steps.

$$\begin{cases}\max\left(|\psi_s(j)-\bar{\psi}_s(j)|\right)<\text{Th}_\theta, j=n:n-2; \text{ straight}\\ \text{otherwise, turning}\end{cases} \tag{10}$$

where Th_θ is the max change value of the heading angle within three steps.

4.2 Pedestrian tracking framework

The pedestrian tracking algorithm that we use is based on the framework proposed by Foxlin [9] and later refined by Jimenez et al. [14]. The extended Kalman filter (EKF) model has 15-element error state vector, and the vector is composed of attitude error $\delta\boldsymbol{\Phi}$, velocity error $\delta\boldsymbol{v}$, position error $\delta\boldsymbol{p}$, the estimated bias of gyroscopes $\delta\boldsymbol{b}_g$ and the estimated bias of accelerometers $\delta\boldsymbol{b}_a$ in the navigation coordinate system.

$$\boldsymbol{x}=\begin{bmatrix}\delta\boldsymbol{\Phi} & \delta\boldsymbol{v} & \delta\boldsymbol{p} & \delta\boldsymbol{b}_g & \delta\boldsymbol{b}_a\end{bmatrix}^{\text{T}} \tag{11}$$

The error model for MEMS pedestrian navigation system (PNS) in the navigation coordinate system [21] is as follows:

$$\delta\dot{\boldsymbol{p}}=\delta\boldsymbol{v}, \tag{12}$$

$$\delta\dot{\boldsymbol{v}}=(\boldsymbol{f}^n\times)\delta\boldsymbol{\Phi}+\boldsymbol{C}_b^n\delta\boldsymbol{b}_a+\boldsymbol{C}_b^n\boldsymbol{m}_a^b, \tag{13}$$

$$\delta\dot{\boldsymbol{\Phi}}=-\boldsymbol{C}_b^n\delta\boldsymbol{b}_g-\boldsymbol{C}_b^n\boldsymbol{m}_g^b, \tag{14}$$

$$\delta\dot{\boldsymbol{b}}_g=-\boldsymbol{\beta}_g\delta\boldsymbol{b}_g+\boldsymbol{w}_g^b, \tag{15}$$

$$\delta\dot{\boldsymbol{b}}_a=-\boldsymbol{\beta}_a\delta\boldsymbol{b}_a+\boldsymbol{w}_a^b, \tag{16}$$

where $\boldsymbol{m}_g^b$ and $\boldsymbol{m}_a^b$ are the gyroscope and accelerometer output noise, $\boldsymbol{w}_g^b$ and $\boldsymbol{w}_a^b$ are the Gaussian white noise of $\delta\boldsymbol{b}_g$ and $\delta\boldsymbol{b}_a$ under the first-order Markov model, $\boldsymbol{\beta}_g$ and $\boldsymbol{\beta}_a$ are the time constants associated with first-order Markov modelling.

There are two stages in the Kalman filter as follows.

(i) Time update

The filter propagates the error covariance matrix $\boldsymbol{P}_{k+1/k}$, which can be expressed as follows:

$$\boldsymbol{P}_{k+1/k}=\boldsymbol{F}_{k+1}\boldsymbol{P}_k\boldsymbol{F}_{k+1}^{\text{T}}+\boldsymbol{Q} \tag{17}$$

where $\boldsymbol{Q}$ is the covariance matrix of the process noise, and $\boldsymbol{F}_k$ is the 15×15 state transition matrix. They can be expressed as follows:

$$\boldsymbol{F}_k=\begin{bmatrix}\boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & -\boldsymbol{C}_{nb} & \boldsymbol{0}_{3\times3}\\ \boldsymbol{f}^n\times & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{C}_{nb}\\ \boldsymbol{0}_{3\times3} & \boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3}\\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{\beta}_g & \boldsymbol{0}_{3\times3}\\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{\beta}_a\end{bmatrix}\Delta t+\boldsymbol{I}_{15\times15}, \tag{18}$$

$$\boldsymbol{Q}=\text{E}\left(\boldsymbol{w}_k\boldsymbol{w}_k^{\text{T}}\right). \tag{19}$$

(ii) Measurement update

The Kalman filter gain, $\boldsymbol{K}_{k+1}$, can be calculated as

$$\boldsymbol{K}_{k+1}=\boldsymbol{P}_{k+1/k}\boldsymbol{H}^{\text{T}}(\boldsymbol{H}\boldsymbol{P}_{k+1/k}\boldsymbol{H}^{\text{T}}+\boldsymbol{R})^{-1} \tag{20}$$

where $\boldsymbol{R}$ is the covariance matrix of the measurement noise and $\boldsymbol{H}$ is the measurement matrix.

The state estimation can be calculated as

$$\boldsymbol{x}_{k+1}=\boldsymbol{K}_{k+1}{z_{k+1}}^{\text{T}} \tag{21}$$

where z_{k+1} is the actual error measurement. There are three error measurements that would be measured.

i) ZUPT: When the feet are stationary, the velocities of the system are expected to be zero, and the estimated velocity at time k can be taken as error measurements.

$$\boldsymbol{m}_k=\boldsymbol{v}_k-[0,0,0] \tag{22}$$

ii) ZARU: ZARU takes into account the fact that the attitude is invariable when a stance phase occurs. Thus any angular rate measured at that point is considered as error measurement.

$$\boldsymbol{m}_k=\boldsymbol{\omega}_k-[0,0,0] \tag{23}$$

iii) Heuristic yaw update: In case of straight ahead, the yaw error measurement can be calculated as follows:

$$\boldsymbol{m}_k=\left[0\ \ 0\ \ \psi(k)-\psi_{\text{main}}(k)\right]. \tag{24}$$

Finally, the corresponding measurement matrix $\boldsymbol{H}$ can be formulated as follows:

$$\boldsymbol{H}=\begin{bmatrix}\boldsymbol{0}_{3\times3} & \boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3}\\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times3}\\ [0\ 0\ 1] & \boldsymbol{0}_{1\times3} & \boldsymbol{0}_{1\times3} & \boldsymbol{0}_{1\times3} & \boldsymbol{0}_{1\times3}\end{bmatrix}. \tag{25}$$

The predictive error covariance is

$$\boldsymbol{P}_{k+1} = (\boldsymbol{I}_{15\times15} - \boldsymbol{K}_{k+1}\boldsymbol{H})\boldsymbol{P}_{k+1/k}. \tag{26}$$

The last step is to correct the attitude, velocity and position estimate by removing the Kalman error estimates.

In addition, at time t_{k+1}, the filter only performs a time update if the pedestrian is in a swing phase, and the filter performs a complete update (time update and measurement update) if the pedestrian is in a stance phase.

4.3 System structure

The overall frame and flow of the proposed algorithm is illustrated in Fig. 7. As described above, the system mainly consists of navigation calculation, gait detection, and Kalman filter.

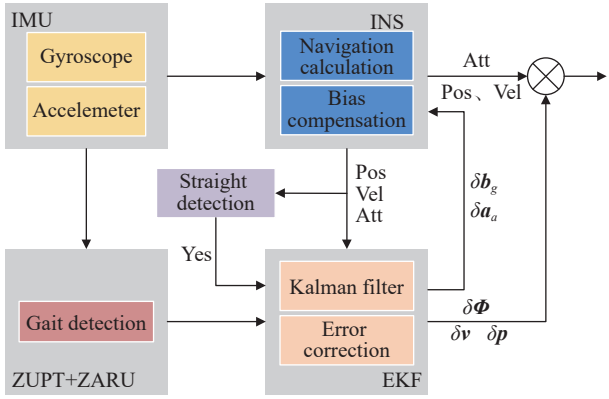


Fig. 7 Structure of the proposed algorithm

5. Experiments and discussion

5.1 Simulation experiment

In order to verify the effectiveness of the algorithm proposed in this paper, a set of simulation experiments are designed to generate simulated pedestrian inertial sensor data when walking [22].

(i) Simulation experiment design

Assume that a person is advancing in a straight line in the horizontal plane, and the initial heading angle is ψ_0. The initial attitude, velocity and position of the right foot in the navigation system are

$$\begin{cases} \boldsymbol{\Psi} = \begin{bmatrix} 0 & 0 & \psi_0 \end{bmatrix} \\ \boldsymbol{v}^n = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix} \\ \boldsymbol{p}^n = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix} \end{cases} \tag{27}$$

where ψ_0 is the heading angle before the turn.

To simplify the experiment, each step of a pedestrian is divided into two stages: the swing and the stance phase, which takes t_{sw} and t_{st} respectively, so that a full gait period is T. In each gait cycle, the change in position of the gait during the swing phase with respect to the initial state (or the previous stationary state) is set to

$$\begin{cases} \Delta\boldsymbol{p} = \begin{bmatrix} p_e & p_n & p_u \end{bmatrix} \\ p_e = \dfrac{l_s}{2}\{1-\cos[\pi(t/t_{sw})]\}\sin\psi_0 \\ p_n = \dfrac{l_s}{2}\{1-\cos[\pi(t/t_{sw})]\}\cos\psi_0 \\ p_u = \dfrac{l_h}{2}\{1-\cos[2\pi(t/t_{sw})]\} \end{cases} \tag{28}$$

where l_s is the length of each step and l_h is the maximum height of the raised foot surface. The velocity of the right foot can be derived from the differential of the change of position.

In each gait cycle, the attitude of the right foot is

$$\begin{cases} \boldsymbol{\Psi} = \begin{bmatrix} \theta & \phi & \psi \end{bmatrix} \\ \theta = \dfrac{\theta_{\max}}{2}\{1-\cos[2\pi(t/t_{sw})]\} \\ \phi = 0 \\ \psi = \psi_0 \end{cases} \tag{29}$$

where $\theta_{\max}$ is the maximum change in pitch angle during travel, θ, ϕ, ψ are pitch, roll and heading angles respectively.

In order to obtain a closed walking trajectory and design experiments as simple as possible, assume that pedestrians can turn 90° without changing their position of the right foot in a turning period t_c, which can be expressed as

$$\begin{cases} \theta = 0 \\ \phi = 0 \\ \psi = \psi_0 + \dfrac{\pi}{4}\{1-\cos[\pi\,(t/t_c)]\} \end{cases}. \tag{30}$$

The local geographical coordinates are $\boldsymbol{P}_0$, $\boldsymbol{P}_0 =$ [31.306 175N 120.644 243E 10], and the simulated gyroscope and accelerometer outputs are as follows:

$$\boldsymbol{\omega}_{ib}^b = \boldsymbol{\omega}_{ib}^b + \boldsymbol{C}_n^b\boldsymbol{C}_e^n\boldsymbol{\omega}_{ie}^e + \boldsymbol{b}_g + \boldsymbol{n}_g, \tag{31}$$

$$\boldsymbol{f}^b = \boldsymbol{C}_n^b(\dot{\boldsymbol{v}}^n - \boldsymbol{g}^n) + \boldsymbol{b}_a + \boldsymbol{n}_a, \tag{32}$$

where $\boldsymbol{b}_g$, $\boldsymbol{n}_g$ are the bias and noise of the gyroscope respectively, and $\boldsymbol{b}_a$, $\boldsymbol{n}_a$ are the bias and noise of the accelerometer respectively.

The overall parameter settings are shown in Table 1, and some of the graphics produced by the simulation are shown in Fig. 8 and Fig. 9.

Table 1 Simulation parameter settings

Symbol	Parameter	Value
l_s	Step length/m	1.2
l_h	Max height/m	0.15
t_{sw}	Swing phase cycle/s	1.0
t_{st}	Stance phase cycle/s	0.5
θ_{max}	Max pitch/rad	0.6
t_c	Turning cycle/s	0.5
$\boldsymbol{b}_g$	Gyroscope bias/(°/s)	$[\ 1.5\ \ 2\ \ 1.2\ ]$
$\boldsymbol{b}_a$	Accelerometer bias/(m/s^2)	$[\ 0.1\ \ 0.2\ \ 0.15\ ]$
$\boldsymbol{n}_g$	Gyroscope noise/(rad/$\sqrt{\text{s}}$)	$[\ 0.1\ \ 0.1\ \ 0.1\ ]$
$\boldsymbol{n}_a$	Accelerometer noise/(m/s^2/$\sqrt{\text{Hz}}$)	$[\ 0.1\ \ 0.1\ \ 0.1\ ]$
ψ_0	Initial yaw/(°)	0
f	Sample frequency/Hz	100

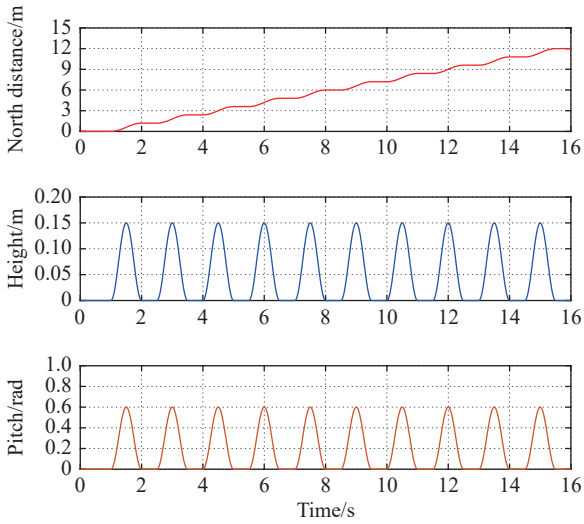


Fig. 8 Partial graphics generated by simulation

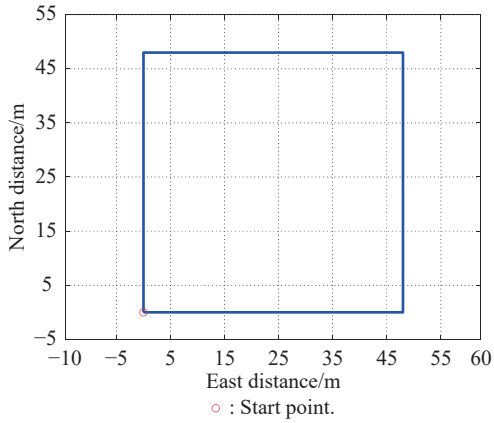


Fig. 9 Generated pedestrian trajectory in simulation

(ii) Analysis of results

For the inertial data generated by the simulation, we use the proposed method in this paper and the traditional ZUPT algorithm to solve at the same time. The comparison of the trajectories is shown in Fig. 10. The endpoint positioning error of the proposed method is about 0.1%, which is less than 1%. And this can verify the effectiveness of the proposed method.

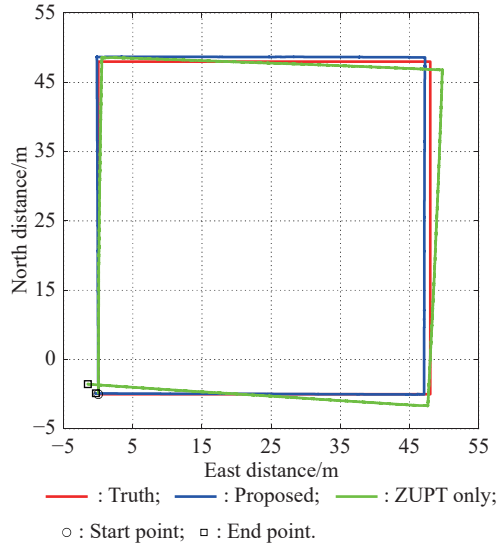


Fig. 10 Comparison of results using different methods

5.2 Field experiments

(i) Experimental conditions

We have designed a real-time PNS hardware and software platform for the acquisition, processing and display of inertial navigation data.

The microcontroller unit (MCU) STM32F405 acquires data from the IMU and sends it to the host computer. The inertial sensor MPU9250 is chosen as the IMU, and its performance parameters are shown in Table 2.

Table 2 MPU9250 sensor performance parameters

Parameter	Gyroscope	Accelerometer
Maximum range	±2000 °/s	±16g
Sensitivity	16.4 LSB/(°/s)	2 048LSB/g
Non-linearity	0.2 %	0.5 %
Initial bias	±5 °/s	±60 mg
Noise density	0.05 rad/$\sqrt{\text{s}}$	0.4 mg/$\sqrt{\text{Hz}}$

(ii) Analysis of results

In order to verify the practical performance of the proposed method, we design two experiments.

In the first experiment, the participant walks around the rectangular indoor field. For the data collected, different methods are used to calculate the navigation information. We compare the navigation performance of ZUPT/ZARU-

based inertial navigation system (INS) using the proposed heuristic heading constraint method, ZUPT/ZARU-based INS and ZUPT-based INS, and the final plane trajectory comparison is shown in Fig. 11.

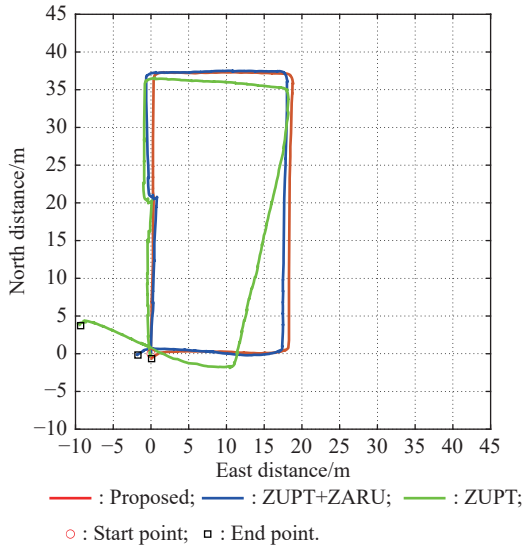


Fig. 11 Plane trajectory comparison of closed rectangular paths using different algorithms

The end-point positioning error is calculated separately (as shown in Table 3), which shows that the end-point positioning error of the proposed method is 0.68% and less than 1%. The formula for calculating the positioning error of the end point is as follows:

$$\delta_{\text{location}} = \frac{d}{D} \tag{33}$$

where d is the distance between the end point and the start point, and D is the total length of route.

The attitude angle and velocity information of this experiment after calculation are shown in Fig.12 and Fig.13. From these two figures, it can be seen how well the method limits errors in velocity and horizontal attitude angle. In addition, it also shows that the proposed method can suppress heading drifts in a straight line to some extent.

In the second experiment, participant walks around the campus as the route shown in Fig. 6. Here we use an ellipsoid fitting algorithm to calibrate the acquired magnetometer data at the initial position, so as to obtain the initial yaw. Refer to experiment above, we use different methods to calculate navigation information from the collected data.

The final calculated plane trajectories are compared, as shown in Fig. 14. It shows that the path calculated by the proposed method is almost the same as the real route in the map. As shown in Table 3, the end-point positioning error of the method proposed in this paper is still less than 1%.

Table 3 Comparison of walking experiment results for different methods

Algorithm	Total distance / m		End-point / m		Positioning error /%	
	Test1	Test2	Test1	Test2	Test1	Test2
Proposed method	110	590	(0.48,−0.58)	(−0.86,−1.73)	0.68	0.33
ZUPT+ZARU	110	590	(−1.97,−0.12)	(42.65,59.23)	1.79	12.37
ZUPT only	110	590	(−8.10,4.82)	(56.41,58.26)	8.56	13.74

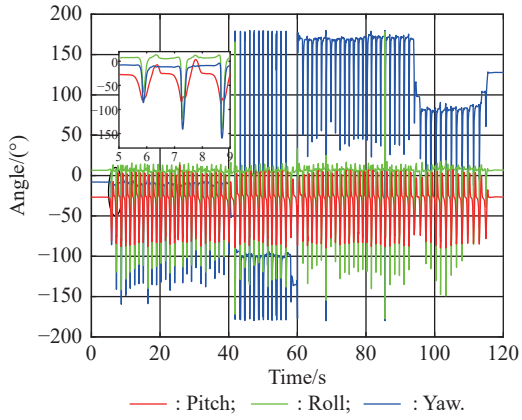


Fig. 12 Attitude angles of a closed rectangular path in the navigation coordinate system

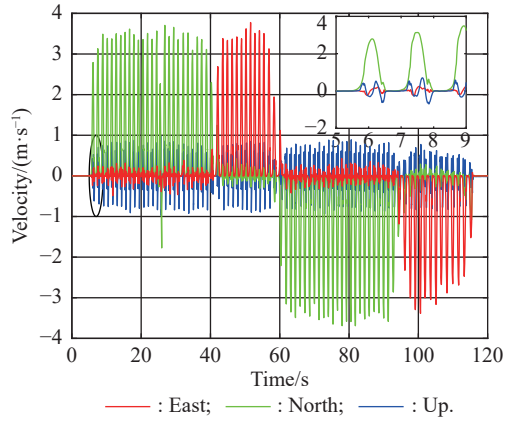


Fig. 13 Velocity of the closed rectangular path in the navigation coordinate system

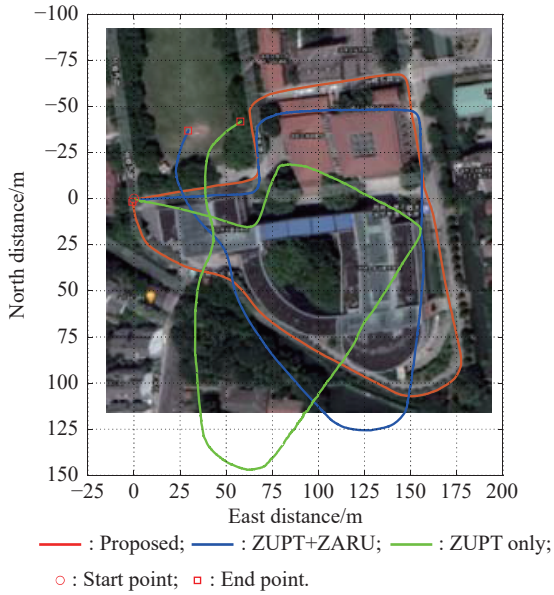


Fig. 14 Trajectory comparison of closed curve paths using different algorithms

The calculated attitude and velocity information of this experiment are presented in Fig. 15 and Fig. 16 separately.

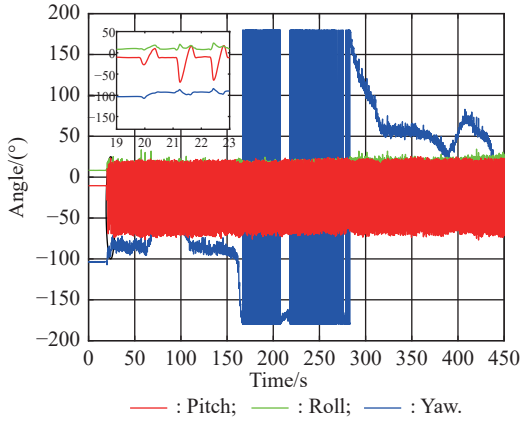


Fig. 15 Attitude angles of closed curve paths in the navigation coordinate system

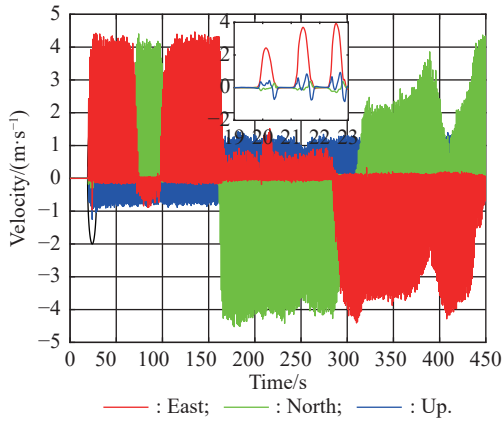


Fig. 16 Velocity of closed curve paths in the navigation coordinate system

6. Conclusions

In this paper, we propose a heuristic heading constraint method for a foot-mounted PNS equipped with low-cost IMU. Due to the limited performance of the consumer grade inertial sensor MPU9250, in order to improve the accuracy of gait detection, this paper adopts a GLRT method and introduces a time threshold, combined with a gait classification method. Analysis of the heuristic heading constraint method shows that, together with ZUPT and ZARU, this method is effective in mitigating heading errors. We compare the navigation performance of ZUPT/ZARU-based INS using the proposed heuristic heading constraint method and ZUPT/ZARU-based INS. The experimental results show that the proposed PNS algorithm can reduce the positioning error of the endpoint to less than 1%.

References

[1] HARLE R. A survey of indoor inertial positioning systems for pedestrians. IEEE Communications Surveys & Tutorials, 2013, 15(3): 1281–1293.

[2] FALLAH N, APOSTOLOPOULOS I, BEKRIS K, et al. Indoor human navigation systems: a survey. Interacting with Computers, 2013, 25(1): 21–33.

[3] HONKAVIRTA V, PERALA T, ALI-LOYTTY S, et al. A comparative survey of WLAN location fingerprinting methods. Proc. of the 6th Workshop on Positioning, Navigation and Communication, 2009: 243–251.

[4] ELLOUMI W, LATOUI A, CANALS R, et al. Indoor pedestrian localization with a smartphone: a comparison of inertial and vision-based methods. IEEE Sensors Journal, 2016, 16(13): 5376–5388.

[5] HSU Y L, WANG J S, CHANG C W. A wearable inertial pedestrian navigation system with quaternion-based extended Kalman filter for pedestrian localization. IEEE Sensors Journal, 2017, 17(10): 3193–3206.

[6] ZHAO Y L, LIANG J Q, SHA X P, et al. Estimation of pedestrian altitude inside a multi-story building using an integrated micro-IMU and barometer device. IEEE Access, 2019, 7: 84680–84689.

[7] MADGWICK S O H, HARRISON A J L, VAIDYANATHAN R. Estimation of IMU and MARG orientation using a gradient descent algorithm. Proc. of the IEEE International Conference on Rehabilitation Robotics, 2011: 1–7.

[8] GROVES P D. Navigation using inertial sensors. IEEE Aerospace and Electronic Systems Magazine, 2015, 30(2): 42–69.

[9] FOXLIN E. Pedestrian tracking with shoe-mounted inertial sensors. IEEE Computer Graphics and Applications, 2005, 25(6): 38–46.

[10] PARK S K, SUH Y S. A zero velocity detection algorithm using inertial sensors for pedestrian navigation systems. Sensors, 2010, 10(10): 9163–9178.

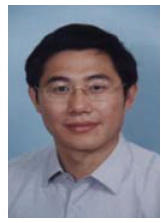
[11] WAHLSTROM J, SKOG I. Fifteen years of progress at zero velocity: a review. IEEE Sensors Journal, 2020, 21(2): 1139–1151.

[12] TJHAI C. Integration of multiple low-cost wearable inertial/magnetic sensors and kinematics of lower limbs for improving

pedestrian navigation systems. Alberta: University of Calgary, 2019.
[13] NORRDINE A, KASMI Z, BLANKENBACH J. Step detection for ZUPT-aided inertial pedestrian navigation system using foot-mounted permanent magnet. IEEE Sensors Journal, 2016, 16(17): 6766–6773.
[14] JIMENEZ A R, SECO F, ZAMPELLA F, et al. PDR with a foot-mounted IMU and ramp detection. Sensors , 2011, 11(10): 9393–9410.
[15] BORENSTEIN J, OJEDA L. Heuristic drift elimination for personnel tracking systems. The Journal of Navigation, 2010, 63(4): 591–606.
[16] JIMENEZ A R, SECO F, ZAMPELLA F, et al. Improved heuristic drift elimination (iHDE) for pedestrian navigation in complex buildings. Proc. of the International Conference on Indoor Positioning and Indoor Navigation, 2011: 1–8.
[17] SKOG I, HANDEL P, NILSSON J O, et al. Zero-velocity detection—an algorithm evaluation. IEEE Trans. on Biomedical Engineering, 2010, 57(11): 2657–2666.
[18] CGROVES P D. Principles of GNSS, inertial, and multi-sensor integrated navigation systems. USA: Artech House Verlag, 2013.
[19] FISCHER C, SUKUMAR P T, HAZAS M. Tutorial: implementing a pedestrian tracker using inertial sensors. IEEE Pervasive Computing, 2012, 12(2): 17–27.
[20] SKOG I, HANDEL P. A low-cost GPS aided inertial navigation system for vehicle applications. Proc. of the 13th European Signal Processing Conference, 2005: 1–4.
[21] WANG Y S, CHERNYSHOFF A, SHKEL A M. Study on estimation errors in ZUPT-aided pedestrian inertial navigation due to IMU noises. IEEE Trans. on Aerospace and Electronic Systems, 2019, 56(3): 2280–2291.
[22] ZHU M, WU Y, LUO S. A pedestrian navigation system by low-cost dual foot-mounted IMUs and inter-foot ranging. Proc. of the German Institute of Navigation Inertial Sensors and Systems, 2020: 1−20.

Biographies

GUI Jing was born in 1997. He received his B.S. degree from Soochow University, Suzhou, China, in 2019. He is currently pursuing his M.S. degree in the School of Electronic and Information Engineering, Soochow University. His current research interests include low-cost inertial navigation systems and pedestrian navigation.
E-mail: jgui716@163.com



ZHAO Heming was born in 1957. He received his M.S. degree from Soochow University, Suzhou, China, in 1982. He is a doctoral tutor and a dean in the School of Electronic and Information Engineering, Soochow University. He is a member of IEEE and a senior member of Chinese Electronic Institute, and he is a member of the editorial boards of signal processing. His current research interests include speech signal processing and intelligent computing.
E-mail: hemzhao@163.com

XU Xiang was born in 1988. He received his M.S. degree from Harbin Engineering University, Harbin, China, in 2014, and his Ph.D. degree from Southeast University, Nanjing, China, in 2018. He is currently a lecturer with the School of Electronic and Information Engineering, Soochow University. His current research interests include initial alignment for inertial navigation, integrated navigation for autonomous underwater vehicle, attitude estimation for low-cost inertial measurement unit and information fusion.
E-mail: hsianghsu@163.com