PDR/GNSS Fusion Algorithm Based on Joint Heading Estimation

Abdul Rehman, Qiang Liu(✉), Zida Wu, Huiping Zhu, Jiuchao Qian,
Yuze Wang, and Peilin Liu

School of Electronic Information and Electrical Engineering,
Shanghai Jiao Tong University, Shanghai, China
lugialiu@sjtu.edu.cn

Abstract. The performance of Pedestrian Dead Reckoning (PDR) based on a smartphone is limited due to the low-cost MEMS IMU. Heading and stride length estimation tend to accumulate errors over time and those errors lead to the failure of PDR. Considering that Global Navigation Satellite System (GNSS) provides absolute location information in outdoor applications, and the characteristics of estimation errors of GNSS are quite different from those of PDR. In this paper, we analyse the characteristics of two different types of errors first, then we propose a fusion positioning framework that can fuse PDR and GNSS information in real time. The joint estimations of heading and stride length are also given. The experimental results show that the fusion algorithm is superior to PDR or GNSS in heading estimation, anti-multipath and noise performance.

Keywords: PDR · Heading estimation · Fusion positioning

1 Introduction

With the development of smartphones, many navigation sensors are integrated into it, such as GNSS module and IMU. However, the performance of low-cost devices on smartphone differs greatly from that of professional devices. For example, the bias instability of the MEMS accelerometer is generally greater than 0.03 m/s^2, the bias instability of the MEMS gyroscope is generally greater than 100°/h, and the GNSS positioning accuracy of smartphones based on single point positioning is usually about 10 m [1, 5]. For this reason, the traditional Strap-down Inertial Navigation System (SINS) algorithm is almost impossible to implement on the MEMS IMU. Although the low accuracy of MEMS will lead to accumulated errors in SINS, in some scenarios, such as a pedestrian or driving vehicles, the MEMS sensors can be used to assist attitude measurement, and the attitude change has a certain regularity. Therefore, some techniques such as PDR [3] and Vehicle Dead Reckoning (VDR) are developed in recent years. Besides GNSS and IMU, there are many other types of sensors integrated on smartphones, such as magnetometer, Bluetooth, WIFI and camera etc., so the fusion scheme can be varied. Taking the GNSS and INS integrated navigation as an example, this is a highly complementary system, so it has achieved success in many applications. However, MEMS is not good enough to support INS because of the above-mentioned disadvantages. According to the level of coupling, integrated navigation can be divided

J. Sun et al. (Eds.): CSNC 2019, LNEE 562, pp. 326–339, 2019.
https://doi.org/10.1007/978-981-13-7751-8_33

into three types: loose coupling at the level of positioning results, tight coupling at the level of observations and deep/ultra-tight coupling at the level of signal tracking. With the open access to the raw GNSS observations of Android smartphones in recent years, the tight coupling between MEMS IMU and low-cost GNSS is theoretically supported, but such research still faces great challenges.

Because the low-cost MEMS IMU and GNSS on smartphones are still unable to achieve GNSS/INS integrated navigation, while the integration of PDR and GNSS based on MEMS is considered to be feasible, so many related research have been published in recent years. As mentioned above, the fusion based on PDR and low-cost GNSS can be either loosely coupled or tightly coupled. Hsu et al. [2] proposed a framework of the fusion algorithm based on 3D map-assisted GNSS and PDR in an urban environment. This algorithm outputs the stride length and heading of PDR as well as the position and accuracy of 3D-GNSS into the Kalman filter to obtain the fusion result, so it is a loosely coupled method. The observations of the Kalman filter come from 3D-GNSS, while the PDR outputs are used to be the control inputs. This is because the trajectory of pedestrians is very uncertain and it is difficult to be described by linear state transition equations. Therefore, the real-time observations of PDR are used to update the state transition equations. However, the accumulated error of PDR still exists in the fusion system, resulting in the final heading estimations. Lan et al. [4] uses the EKF to fuse PDR and GNSS. This algorithm is directly based on the output information of the MEMS sensors, so it can be regarded as a tightly coupled algorithm. This algorithm uses the heading derived by GNSS to assist the heading estimation of the MEMS gyroscope, which improves the overall accuracy of the heading estimation. However, the serious noise problem in the heading of GNSS has not been solved well, so the initial stage of the final trajectory output dependent highly on the heading of GNSS.

Based on the above analysis, this paper focuses on the error analysis of PDR and GNSS, then a fusion framework is proposed, which fully combines the advantages of PDR and GNSS, and effectively eliminates accumulated error and noise. In addition, considering the inconsistency of update frequencies between GNSS and PDR in practice, a synchronization method is also discussed.

The following is organized as follows. The PDR algorithm is briefly introduced in Sect. 2. We discuss the error analysis, data synchronization and fusion framework in Sect. 3. Section 4 shows the experimental setup and results. Section 5 gives the summary and future work.

2 Pedestrian Dead Reckoning

The PDR algorithm typically consists of three steps, they are step detection, stride length estimation and heading estimation, as shown in Fig. 1.

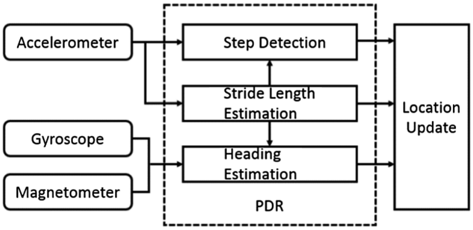

Fig. 1. Block diagram of the PDR algorithm

3-axis accelerometer output is generally used to detect the occurrence of the step and to estimate the stride length, while the 3-axis gyroscope or magnetometer is used to estimate the heading. When PDR detects a step, it will output the heading and stride length estimation {θ, SL} of step k. If the initial position is prior known, we can get the pedestrian position by Eq. 1 as follows.

$$\begin{cases} x_{k+1} = x_k + SL_k \cdot sin\theta_k \\ y_{k+1} = y_k + SL_k \cdot cos\theta_k \end{cases} \tag{1}$$

2.1 Step Detection

Step detection is mainly based on the periodic signal generated by the accelerometer when a pedestrian is walking, in order to determine whether the pedestrian has stepped or not. Common step detection methods include zero-crossing detection, peak detection, the autocorrelation method and the spectrum analysis method. In this paper, the raw data from accelerometer are pre-processed by setting the low-pass filter with the cut-off frequency is 3 Hz to reduce the device noise. Then the dynamic threshold zero-crossing detection method proposed in [7] is used to control the transition between walking and non-walking states through a finite state machine, it can achieve the 99% accuracy.

2.2 Stride Length Estimation

There are several methods for stride length estimation, they are empirical model, the linear model, the non-linear model and the machine learning model. The stride length estimation model used in this paper is a binary linear regression model with the stride frequency f_s and the acceleration variance σ_a^2 as parameters, and this model was first proposed in [8]. This model is expressed as follows.

$$SL = \alpha f_S + \beta\sigma_a^2 + \gamma \tag{2}$$

In Eq. 2, α and β are weighting factors, while γ is a constant, they need to be determined by off-line training. Once determined, it can be used to estimate the stride length of different pedestrians in real-time applications.

2.3 Heading Estimation

Heading estimation is the most important part of the PDR algorithm because the small estimation errors introduced in the heading estimation will accumulate, which will eventually lead to shape changes between estimated trajectory and real trajectory. The heading estimation usually uses the data from gyroscope, magnetometer or both of them. Since the gyroscope generates a cumulative error during integration, and the magnetometer is susceptible to interference from the surrounding magnetic field, these two sensors have complementary characteristics. The heading estimation used in this paper combines gyroscope and magnetometer, and the optimal heading estimation is able to be obtained by using the redundant information from gyroscope and magnetometer.

3 PDR/GNSS Fusion Framework

This section firstly defines the GNSS heading and stride length, then discusses the synchronization of PDR and GNSS data. After that, the analysis for heading error and stride length error of PDR and GNSS is given and the fusion framework is also proposed.

3.1 GNSS Heading and Stride Length

GNSS single point positioning gives the position of the user at a particular time. It is generally considered that the GNSS trajectory can be obtained by connecting the single point positions in time. The receiver outputs the filtered single point positions in general, but there is still the influence of thermal noise. Especially with low-cost GNSS on smartphone, the noise power is much larger than the professional receiver. Because the observed outputs of the PDR are heading and stride length, in order to unify the outputs of GNSS and PDR, it is necessary to define the GNSS heading and stride length. These two concepts are rarely mentioned in the traditional GNSS field. The definitions are described as follows.

Coordinate transformation: The latitude, longitude and height (LLH) output by GNSS are first converted into ECEF XYZ coordinates, and then the ECEF coordinates can be converted to East-North-Up (ENU) coordinates based on a prior known initial position.

Calculate heading and stride length: Assume (E_k, N_k, U_k) and (E_{k+1}, N_{k+1}, U_{k+1}) are the ENU coordinates of time k and k + 1, then the calculation formula of the GNSS heading θ_{k+1}^{GNSS} and the stride length SL_{k+1}^{GNSS} is shown in Eq. 3. It should be noted that the definition of GNSS heading is the same as that of PDR, that is, the angle between the vector of the two positioning results and the northward direction. The range of the heading in the formula is (−π, +π).

$$\begin{cases} \theta_{k+1}^{GNSS} = arctan\left(\frac{N_{k+1}-N_k}{E_{k+1}-E_k}\right) \\ SL_{k+1}^{GNSS} = \left\| \begin{bmatrix} E_{k+1} - E_k \\ N_{k+1} - N_k \end{bmatrix} \right\|_2 \end{cases} \tag{3}$$

3.2 Heading and Stride Length Errors of PDR and GNSS

The Eq. 4 gives an equivalent expression of Eq. 1, where $\{x_1, y_1\}$ is the known initial position. In each iteration process, some errors will be introduced in stride length and heading estimations, which will be reflected in position x and y. Equation 4 can be used to derive the formations of the errors on x and y, but the process is complex, and it is not very helpful to eliminate accumulated errors.

$$\begin{cases} x_{k+1} = x_1 + \sum_{i=1}^{k} (SL_i \cdot sin\theta_i) \\ y_{k+1} = y_1 + \sum_{i=1}^{k} (SL_i \cdot cos\theta_i) \end{cases} \quad (4)$$

In order to combine PDR and GNSS, one idea is based on the fusion of positions. However, according to above analysis, it is not convenient to analyse the influence of accumulated errors by directly fusing their positions. The following will give the error analysis of heading and stride length for two systems and state the aims of fusion algorithm. The necessary symbolic descriptions are given as follows.

$\varepsilon_k^{\theta-PDR}$: heading estimation error of PDR at step k
ε_k^{SL-PDR}: stride length estimation error of PDR at step k
$\varepsilon_k^{\theta-GNSS}$: heading estimation error of GNSS at time k
$\varepsilon_k^{SL-GNSS}$: stride length estimation error of GNSS at time k

In order to analyze the characteristics of the above errors, we designed the following experiment. This experiment requires some priori control points, for PDR, the tester needs to travel along the road with a straight line and step by step along the ground control points in the line. The stride length can be fixed to a constant, and the real-time data can be recorded and analysed by the post-processing software. The characteristics of the four types of errors can be obtained by carrying out a large number of experiments described above. According to these experimental results, the heading and stride length errors of PDR increase with the increase of k, while the heading and stride length errors of GNSS are more like zero-mean Gaussian noise. In this paper, we define a metric called Mean Cumulative Heading Error (MCHE) as shown in Eq. 5 to measure the heading errors of the two systems.

$$MCHE(k) = \frac{1}{k} \sum_{i=1}^{k} \varepsilon_k^{\theta-SYS} \quad (5)$$

In Eq. 5, SYS can be either GNSS or PDR. MCHE can effectively distinguish the characteristics of Gaussian noise and cumulative error. When k increases, the zero-mean noise MCHE of GNSS will tend to zero, while the cumulative error MCHE of PDR will be divergent. Similarly, we define the Cumulative Stride Length Error (CSLE) as shown in Eq. 6 to measure the stride length errors of the two systems.

$$CSLE(k) = \sum_{i=1}^{k} \varepsilon_k^{SL-SYS} \tag{6}$$

According to the above discussion, one of the objectives of the proposed fusion algorithm in this paper is to make the fused estimations are not only able to avoid the accumulated errors, but them also can reduce the noise power.

3.3 Data Synchronization

In PDR, the output is updated when the step is detected, so the outputs of PDR are unpredictable. However, the GNSS updates its results periodically, which is usually an update rate of 1 Hz. In order to fuse these data in real time, it is necessary to make the output frequency of the two systems be consistent. When a person is walking normally, usually at least one complete step can be completed within 1 s, and if a person cannot complete a step within 1 s, it is more likely that this person does not make a decision for next step. The first case is called a fast step, and the second case is a slow step, as shown in Fig. 2. In general, people take fast steps when they are walking continuously, and the slow steps happen accidentally. If there is no step for a long time, it is called a stop, that is, the pedestrian's feet are closed and stay in place. Figure 2 shows a schematic of the asynchronous output of GNSS and PDR.

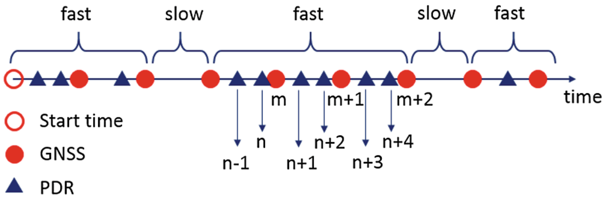

Fig. 2. Data synchronization between PDR and GNSS

The basic idea of the data synchronization method proposed in this paper is to turn the data of PDR into a timing output. In Fig. 2, between time m and m + 1 of GNSS, there are two PDR outputs, they are step n + 1 and n + 2, respectively, corresponding to $\{\theta_{n+1}, SL_{n+1}\}$ and $\{\theta_{n+2}, SL_{n+2}\}$, we define a PDR output $\{\theta_m, SL_m\}$ at time m of GNSS as shown in Eq. 7.

$$\begin{cases} \theta_m = \frac{\theta_{n+1} + \theta_{n+2}}{2} \\ SL_m = \sqrt{SL_{n+1}^2 + SL_{n+2}^2 + 2C} \\ C = SL_{n+1} SL_{n+2} \cdot cos(\theta_{n+1} - \theta_{n+2}) \end{cases} \tag{7}$$

Equation 7 gives a method for calculating the synchronized PDR heading and stride length at a GNSS time. This equation can be extended to the case where there are more than one PDR outputs in one GNSS interval, where the calculation of the heading is averaging, and the calculation of the stride length is not a simple summation, but the distance between the first PDR output at time m and the first PDR output at time m + 1.

3.4 Fusion Framework

Due to the cumulative errors, the trajectory of PDR will have severe distortion with the increase of time. The trajectory of GNSS appears as a non-smooth curve, which is the effect of thermal noise, but it does not produce distortion. This is a very obvious difference between relative positioning and absolute positioning. The goal of the fusion algorithm is to suppress the cumulative errors and reduce the noise power as much as possible so that the final pedestrian trajectory can truly reflect the real dynamic position of the pedestrian, that is, the trajectory is smooth and not shape changed.

Considering that PDR/GNSS fusion algorithm can directly adopt a loose coupled method, as shown in Fig. 3. The synchronized PDR system outputs position and heading information, then they make difference with the outputs of GNSS. After that, the position and heading residuals will be sent to Kalman filter for corrections. Finally, the corrections will be returned to PDR system to form the estimated position and heading. However, the difficulty of this method is that the observation noise of position still needs to be analysed in the design of Kalman filter. In practical applications, the statistical characteristics of these noises and errors are difficult to obtain. Moreover, as discussed in Sect. 3.2, the error and noise components in the final position observation are difficult to separate.

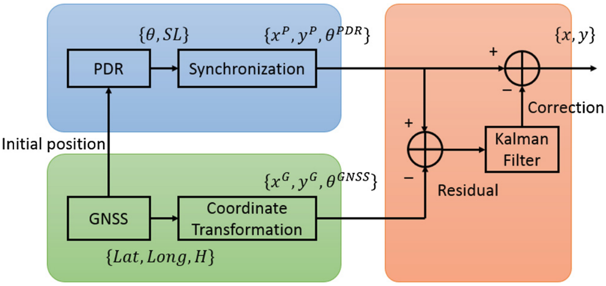

Fig. 3. Traditional loose-coupled framework

In order to reduce the dependence on the selection of Kalman parameters, another fusion idea is presented in this paper. The idea is based on joint heading estimation, which uses PDR heading with low noise power to smooth GNSS heading with high noise. Not only that, because the differential PDR heading is used in the algorithm, the cumulative error is also suppressed. The proposed PDR/GNSS fusion framework is shown in Fig. 4. The initial position is provided by the GNSS at the initial stage.

Then PDR outputs the heading and stride length estimation, and GNSS outputs the latitude and longitude, after coordinate transformation, the GNSS heading and stride length can be obtained according to Eq. 3. At the same time, PDR output the synchronized heading and stride length $\left\{\theta^{PDR}, SL^{PDR}\right\}$.

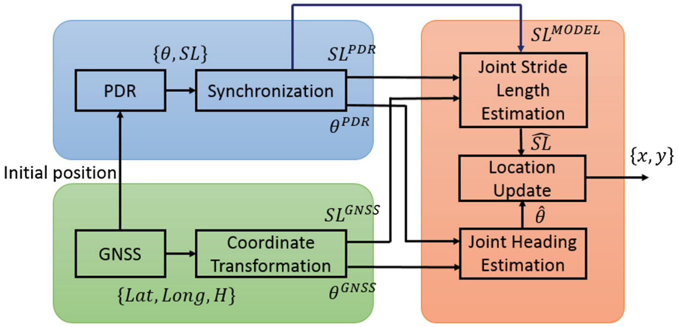

Fig. 4. Joint fusion framework of PDR/GNSS

Then, the heading and stride length estimation of the two systems are respectively sent to the joint heading estimation and joint stride length estimation module, and finally the heading and stride length estimation $\left\{\widehat{\theta}, \widehat{\mathrm{SL}}\right\}$ after the fusion filtering are output by the two modules and sent them to the location update module. The iterative update of the pedestrian location is completed by Eq. 1.

The purpose of the joint heading estimation module is to fuse the PDR heading with cumulative error and the GNSS heading with random noise to achieve the dual effect of reducing the cumulative error and the noise power. The detail is shown in Eq. 8 as follows.

$$\begin{gathered}\widehat{\theta}_{k+1} = \frac{1}{M}\Theta_{k+1} + \frac{M-1}{M}\left(\widehat{\theta}_k + \theta_{k+1}^{PDR} - \theta_k^{PDR}\right) \\ \Theta_{k+1} = p\theta_{k+1}^{GNSS} + (1-p)\theta_{k+1}^{PDR}\end{gathered} \tag{8}$$

The input of the above iterative equation is the heading estimation of two systems θ^{PDR} and θ^{GNSS}, and the output is the joint heading estimation. M and p are the design parameters, where M is the smoothing coefficient, which is used to filter and smooth the GNSS heading with noise. It should not be set too large or too small, if it is too large, the tracking of the heading becomes dull, if it is too small, the filtering effect is not good, and it is usually set to 30 to 50. p is a weighting factor and usually takes a value from 1 to 1.5.

The input of the joint stride length estimation module includes three aspects: the PDR stride length after synchronization, the GNSS stride length, and the output SL^{MODEL} of the PDR synchronization module, whereas the synchronization module determines the mode of the pedestrian, that is, the fast step, slow step and stop mentioned in Sect. 3.3. The final output is the weighted output of the three.

4 Experimental Setup and Results

4.1 Experimental Setup

The experimental environment is shown by the red line in the Fig. 9. The selected route is a straight line with ground marks, which length is about 127 m and the off-angle of the north direction is −19.19°. In order to highlight the characteristics of PDR algorithm error accumulation over time, the tester starts from point A in a straight line, a total of three round trips (where A-B-A is a round trip), in a total time of 540 s.

In the experiment, the tester held the Huawei P10 smartphone in his right hand and walked continuously along the straight line. The walking speed was relatively uniform, and occasionally there were fast and slow. When he reached one of the two endpoints, A or B, his feet closed together, took turn and started moving again. During the experiment, there are occasional vehicles on both sides, and the multipath error of the low-cost GNSS chip in smartphone will become larger in this case, so the anti-interference ability of the fusion algorithm can also be tested.

4.2 Experimental Results

4.2.1 Error Analysis

Figure 5(a) shows the step detection results of PDR. The stop time of three round trips can be clearly seen from the figure, as shown by the red dashed box. Figure 5(b) shows the comparison between the heading estimation of PDR and the true value. It can be found that the heading estimation is relatively accurate in the first 200 s, but less accurate in the next 300 s. The heading estimation has a significant deviation during three round trips.

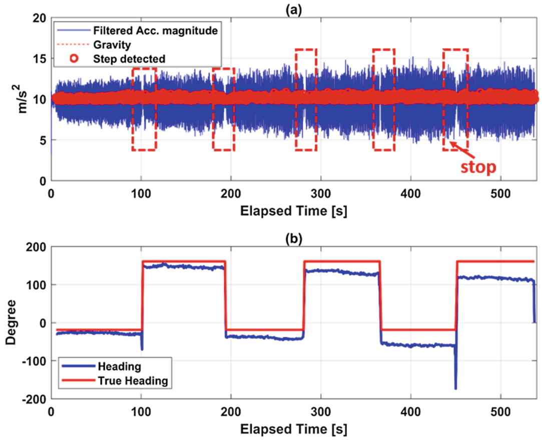

Fig. 5. Step detection and heading estimation

Figure 6 shows the characteristics of the stride length estimation error. Figure 6(a) shows a comparison of the estimated stride length of PDR with the average stride length. It can be seen that the estimated stride length is significantly larger than the average stride length with an increase in time, with an average stride length of approximately 0.73 m. Figure 6(b) shows CSLE of PDR. It can be seen from the figure that since there is an error of about 0.1 m for each PDR stride length estimation, the total error of nearly 50 m at 1000 steps is finally caused. Figure 5(b) and Fig. 6(b) together show that both the stride length estimation and the heading estimation in PDR introduce a large cumulative error.

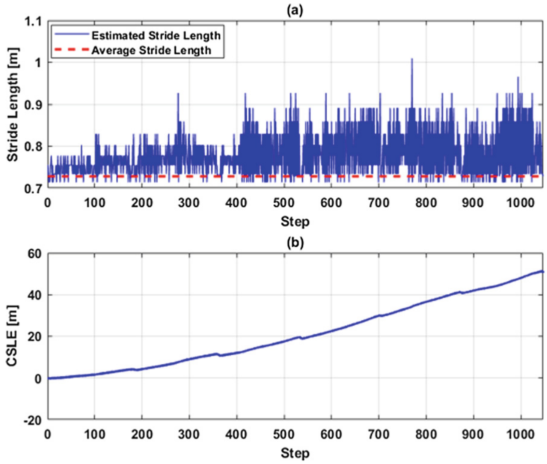

Fig. 6. Stride length error of PDR

Figure 7 shows the error characteristics of PDR heading and GNSS heading. The red line in (a) shows the heading error of GNSS. It can be seen that the GNSS heading is greatly affected by noise, and the maximum error is 20°. The PDR heading indicated by the blue line shows a time accumulation error, but it can be seen that the noise on it, is not very large. The blue line in the figure has several glitches, which is caused by the detection of stop in PDR algorithm, indicating that the pedestrian is stopped at this moment. Figure 7(b) shows the average cumulative heading error of MCHE for GNSS heading and PDR heading. This figure illustrates the completely different characteristics of the two errors, i.e. the GNSS heading error is zero-mean random noise, and the PDR heading error appears the nature of the time function.

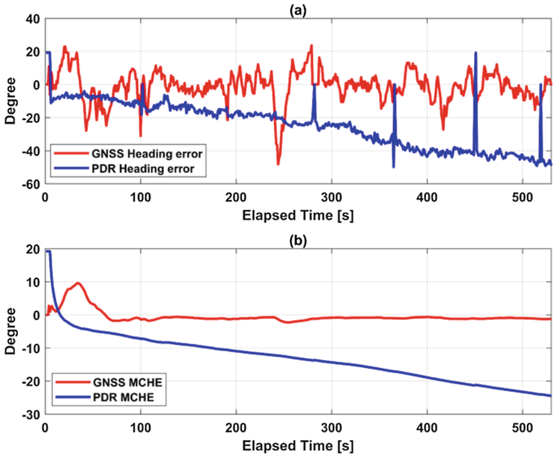

Fig. 7. Heading errors of PDR and GNSS

4.2.2 Fusion Results

The comparison of the GNSS heading, PDR heading and the heading estimation obtained by the fusion algorithm proposed in this paper is shown in Fig. 8. It can be seen from the figure that the GNSS heading does not have time offset, but its noise affects the accuracy of heading estimation, which makes the pedestrian trajectory derived from GNSS not smooth, and GNSS is easily interfered by environmental factors. Where the red circle 1 is marked, there is a large fluctuation in the GNSS data, this is because the occasionally passing of vehicles during the test. This situation will cause the GNSS multipath error at that moment to suddenly increase. At the same time, PDR is not subject to external interference and has high robustness. The combined heading has the respective advantages of the above two kinds of heading estimation, such as no obvious time offset, enhanced anti-interference ability and random noise reduction. The red circle 2 in the figure gives a slow step detected at a certain moment, which shows that the slow step is rarely seen during the normal walking of the pedestrian. Since the PDR heading estimation will detect the slow step, the heading estimation of the fusion algorithm will also show the effect of slow steps, but this will not affect the subsequent heading estimation. That is to say, this phenomenon is harmless.

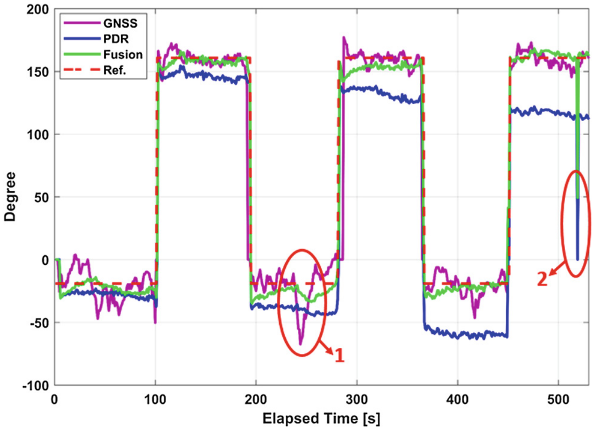

Fig. 8. Comparison of heading estimation by three different methods

Figure 9 shows the final trajectory of the three (GNSS, PDR, Fused) methods, which successively disassembles the three round trips into three graphs (a), (b) and (c). In the above figure, A is the starting point of the path and B is the ending point. As can be seen from the Fig. 9(a), the PDR method has high accuracy in the transient, but it will yaw with an increase in time, while the GNSS trajectory shows greater volatility, and the trajectory of the fusion algorithm shows the high degree of consistency with the real trajectory. Figure 9(b) shows that with the increase in time, the trajectory of PDR produces severe yaw, which can no longer reflect the true trajectory of pedestrians. In this round trip, the distortion of GNSS trajectory is due to the interference of the vehicles during test, even if the GNSS is subject to severe multipath interference and PDR produces severe yaw, in this case, the fusion algorithm can still give relatively smooth, robust and unbiased trajectories. The PDR trajectory in Fig. 9(c) has completely lost its meaning, and the GNSS is less affected by the interference during this round trip. The trajectory of the fusion algorithm can still maintain the non-yaw, indicating that it is affected by the PDR heading estimation at this time, but it also inherits the advantages of PDR low heading noise and the smoother trajectory of PDR. From the above three figures, it can be seen that the fusion algorithm basically eliminates the characteristics of time bias, and obviously enhances the anti-interference ability, and realizes the complementary advantages of PDR and GNSS. However, we have also noticed that compared with the real trajectory, the length of the fusion trajectory is shorter, and it is obvious in the Fig. 9(b) and (c). This is mainly because the stride length estimation in the fusion algorithm still has a small cumulative error. So the performance of the stride length estimation needs to be improved in future work.

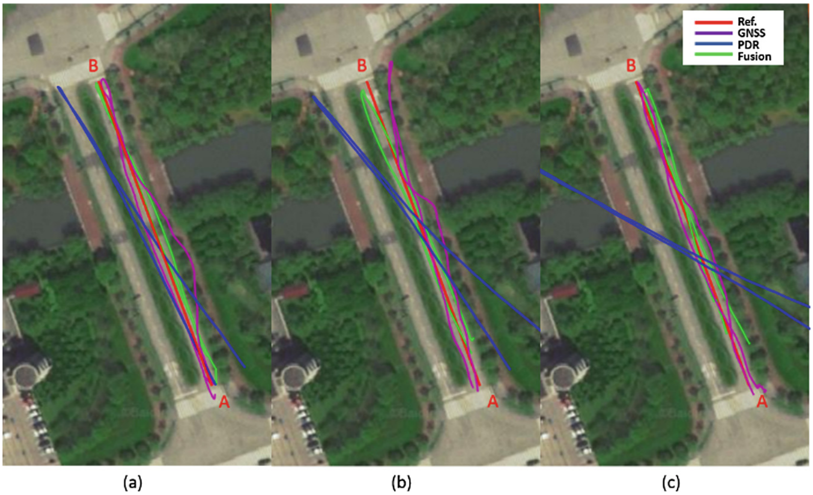

Fig. 9. Comparison of trajectories by three methods

5 Conclusion and Future Work

In this paper, an effective PDR/GNSS fusion framework is proposed by comparing and analyzing the heading and stride length errors of PDR and GNSS. The framework gives a synchronization scheme for PDR and GNSS data and the detailed heading and stride length fusion algorithm. The experimental results show that the fusion algorithm is better than the single GNSS or PDR algorithm in terms of heading estimation, anti-interference and noise performance, which reflects the complementary advantages of absolute positioning and relative positioning. At the same time, however, the accuracy of the stride length estimation needs to be improved. The performance of the fusion algorithm in the urban multipath environment remains to be verified. These are the focus of future work.

Acknowledgements. This work is funded by the China Scholarship Council under grant no. 17B8ECC74F, “The Belt and Road” program.

References

1. Groves PD (2013) Principles of GNSS, inertial, and multi-sensor integrated navigation systems. Ind Robot 67(3):191–192
2. Hsu LT, Gu Y, Huang Y et al (2016) Urban pedestrian navigation using smartphone-based dead reckoning and 3D maps aided GNSS. IEEE Sens J 16(5):1281–1293
3. Jimenez AR, Seco F, Prieto C et al (2009) A comparison of pedestrian dead-reckoning algorithms using a low-cost MEMS IMU. In: IEEE International Symposium on Intelligent Signal Processing. IEEE, pp 37–42
4. Lan H, Yu C, El-Sheimy N (2015) An integrated PDR/GNSS pedestrian navigation system, vol 342, pp 677–690

5. Liu Q, Ying R, Wang Y, Qian J, Liu P (2018) Pseudorange double difference algorithm based on duty-cycled carrier phase smoothing on low-power smart devices. In: Sun J, Yang C, Guo S (eds) China Satellite Navigation Conference (CSNC) 2018 Proceedings, CSNC 2018. Lecture Notes in Electrical Engineering, vol 497. Springer, Singapore
6. Malkos S (2016) User location takes center stage in new android OS: Google to provide raw GNSS measurements. GPS World 27:36
7. Qian J, Ma J, Ying R et al (2013) RPNOS: Reliable Pedestrian Navigation on a Smartphone. Geo-Informatics in Resource Management and Sustainable Ecosystem. Springer, Berlin, pp 188–199
8. Shin SH, Park CG, Kim JW et al (2007) Adaptive step length estimation algorithm using low-cost MEMS inertial sensors. In: 2007 IEEE Sensors Applications Symposium, SAS 2007. IEEE, pp 1–5