

Article

Autonomous Integrated Navigation for Indoor Robots Utilizing On-Line Iterated Extended Rauch-Tung-Striebel Smoothing

Yuan Xu, Xiyuan Chen * and Qinghua Li

Key Laboratory of Micro-Inertial Instrument and Advanced Navigation Technology, Ministry of Education, School of Instrument Science and Engineering, Southeast University, Nanjing 210096, China; E-Mails: xy_abric@126.com (Y.X.); liqinghua1977@163.com (Q.L.)

* Author to whom correspondence should be addressed; E-Mail: chxiyuan@seu.edu.cn; Tel./Fax: +86-25-8379-2010.

Received: 24 September 2013; in revised form: 7 November 2013 / Accepted: 11 November 2013 /

Published: 25 November 2013

Abstract: In order to reduce the estimated errors of the inertial navigation system (INS)/Wireless sensor network (WSN)-integrated navigation for mobile robots indoors, this work proposes an on-line iterated extended Rauch-Tung-Striebel smoothing (IERTSS) utilizing inertial measuring units (IMUs) and an ultrasonic positioning system. In this mode, an iterated Extended Kalman filter (IEKF) is used in forward data processing of the Extended Rauch-Tung-Striebel smoothing (ERTSS) to improve the accuracy of the filtering output for the smoother. Furthermore, in order to achieve the on-line smoothing, IERTSS is embedded into the average filter. For verification, a real indoor test has been done to assess the performance of the proposed method. The results show that the proposed method is effective in reducing the errors compared with the conventional schemes.

Keywords: inertial navigation systems (INS); integrated navigation; iterated extended Kalman filter (IEKF); extended Rauch-Tung-Striebel smoother (ERTSS); average filtering

1. Introduction

Autonomous mobile robots have increasingly been used in a wide range of applications [1]. One key issue for mobile robots is the ability to obtain their navigation information (such as position, velocity and so on). In order to obtain accurate navigation information indoors, a number of methods

for localization with various sensors and different degrees of precision were proposed over the past few decades [1–3].

The main research approaches for indoor localization include beacon-based solutions and beacon-free solutions [4]. Beacon-based solutions employ reference nodes (RNs) with known location to complete the localization of blind nodes (BNs). Its principle is similar to that of global positioning systems (GPS), but the communication technology used is short-range radio, such as WiFi, UWB, RFID, ZigBee, ultrasound, *etc.* Many researchers have attempted to employ beacon-based solutions for indoor localization. For instance, Shirehjini *et al.* proposed an RFID-based position and orientation measurement system for mobile objects in [5]; Park *et al.* employed ZigBee for an indoor location system [6]. Most of the abovementioned attempts employ the measurement of one or several physical parameters of the radio signal transmitted between the RNs and BNs to complete the wireless localization. Due to the influence of the building structures in indoor environments, the accuracy is about one meter. In order to obtain higher accuracy, some researchers employ ultrasonic waves, and the time of arrival (TOA) mode to complete the distance measurement. For instance, Minami *et al.* proposed a fully distributed localization system based on ultrasound, where the localization accuracy was about 20 cm with 24 devices [7], and Saad *et al.* proposed high-accuracy reference-free ultrasonic location estimation in [2]. Ultrasonic sensors have been shown to be a simple but powerful system for this mode, however, it needs has a high density of RNs to maintain the localization accuracy, which is not practical for large localization areas. Differing from beacon-based solutions relying on RNs, beacon-free ones are a self-contained system capable of providing positioning information independently [8]. Some attempts using inertial navigation system (INS)-based beacon-free solutions have been used in integrated outdoor navigation [9]. For example, a GPS/INS navigation system for launchers and re-entry vehicles was described by Boulade *et al.* in [10], and Xu *et al.* proposed a novel hybrid of least squares support vector machine (LS-SVM) and Kalman filter for GPS/INS integration in [11]. Like the integration navigation mentioned above, several approaches also employ INS-based beacon-free solutions for indoor navigation. For example, Ruiz *et al.* employed inertial measuring units (IMUs)/radio frequency identification (RFID) integrated navigation for pedestrian indoor navigation in [4]. Evannou *et al.* proposed a WiFi/INS integration navigation system for indoor mobile positioning in [12]. However, it should be pointed out that beacon-free solutions are poor in long-term self-contained navigation since the accuracy deteriorates with time [13,14], thus it is just a short-term compensation and therefore, it is not suitable for the precision control of indoor mobile robots.

In the data fusion for the integrated navigation system, the integration filter plays an important role in the navigation accuracy. One of the most popular information fusion algorithms is the Kalman filter (KF). However, its optimality heavily depends on linearity [15]. In order to overcome this problem, the extended KF (EKF) is used [14], but the linearization of a nonlinear system by Taylor series expansion and neglecting of the truncated high-order terms will introduce a truncation error [15]. Then, the unscented KF (UKF) and iterated Extended Kalman filter (IEKF) are proposed. Although the UKF is able to overcome the shortcomings of the EKF, it needs more time to compute large numbers of samples [16]. The IEKF is able to reduce the bias and the estimation errors by adding only a few simple iterative operations. In order to obtain high accuracy, smoothing algorithms have been effectively applied for integrated navigation systems [17]. The Rauch-Tung-Striebel smoothing (RTSS) is one of the most popular methods [18]. RTSS was first proposed in [19], and it includes one

forward data processing and one backward data processing step. Due to its robustness and effectiveness, it is widely used in navigation applications. However, it is only suitable for linear systems. In order to overcome this problem, some researchers employ the so-called Extended RTSS (ERTSS). In this mode, the forward data processing of ERTSS is implemented by EKF instead of KF, such as in [20]. In order to reduce the estimated error of the INS/WSN navigation indoors for mobile robots, this work proposes the design and implementation of an on-line iterated ERTSS (IERTSS). The IEKF is used to improve the filtering output accuracy of the ERTSS algorithm. Then, in order to achieve the on-line smoothing, the IERTSS is used in an average filter to smooth the errors of the INS during the output period. A real indoor test is used to evaluate the performance of the proposed method. The remainder of the paper is organized as follows: Section 2 gives the IERTSS embedded average filter design. The unbiased tightly-coupled integrated model for mobile robot navigation indoors is illustrated in Section 3. Section 4 gives the real indoor tests and performance. Finally, conclusions are given in Section 5.

2. On-Line Iterated Extended Rauch-Tung-Striebel Smoothing

In this section, a brief introduction to the IEKF and ERTSS will be given, and then, an on-line IERTSS will be proposed.

2.1. Iterated Extended Kalman Filter

It is assumed that a discrete-time model of a nonlinear system is given by Equations (1) and (2):

$$\mathbf{X}_k = \mathbf{f}(\mathbf{X}_{k-1}) + \mathbf{B}_{k-1}\boldsymbol{\omega}_{k-1} \quad (1)$$

$$\mathbf{y}_k = \mathbf{h}(\mathbf{X}_k) + \mathbf{v}_k \quad (2)$$

where \mathbf{X}_k is the state vector at time k , $\mathbf{f}(\mathbf{X}_k)$ is the system nonlinear function, \mathbf{B}_k is the process noise input matrix, \mathbf{y}_k is the observation vector, and $\mathbf{h}(\mathbf{X}_k)$ is the observation nonlinear function. $\boldsymbol{\omega}_k$ is the process noise, and \mathbf{v}_k is the observation noise. It is assumed that $\boldsymbol{\omega}_k$ and \mathbf{v}_k are independent zero-mean white Gaussian sequences with covariance \mathbf{Q} and \mathbf{R} , respectively. The IEKF used in this paper involves the following recursive relations [17,19]:

$$\hat{\mathbf{X}}_{k|k-1} = \mathbf{A}_{k-1}\hat{\mathbf{X}}_{k-1|k-1} \quad (3)$$

$$\mathbf{P}_{k|k-1} = \mathbf{A}_{k-1}\mathbf{P}_{k-1}\mathbf{A}_{k-1}^T + \mathbf{Q} \quad (4)$$

where $\mathbf{A}_k = \frac{\partial \mathbf{f}(\hat{\mathbf{X}}_{k|k})}{\partial \hat{\mathbf{X}}_{k|k}}$. Compared with the EKF, the IEKF employs a few simple iterative operations to reduce the bias and the estimation error after getting $\hat{\mathbf{X}}_{k|k-1}$ in Equation (3) and $\mathbf{P}_{k|k-1}$ in Equation (4). The corresponding recursive relations are:

$$\hat{\mathbf{X}}_{k|k}^1 = \hat{\mathbf{X}}_{k|k-1} \quad (5)$$

$$\mathbf{P}_{k|k}^1 = \mathbf{P}_{k|k-1} \quad (6)$$

$$\mathbf{K}_k^n = \mathbf{P}_{k|k-1} \mathbf{H}^n (\hat{\mathbf{X}}_{k|k}^n)^T \left[\mathbf{H}^n (\hat{\mathbf{X}}_{k|k}^n) \mathbf{P}_{k|k-1} \mathbf{H}^n (\hat{\mathbf{X}}_{k|k}^n)^T + \mathbf{R} \right]^{-1} \quad (7)$$

$$\hat{\mathbf{X}}_{k|k}^{n+1} = \hat{\mathbf{X}}_{k|k}^n + \mathbf{K}_k^n \left[\mathbf{y}_k - \mathbf{h}^n (\hat{\mathbf{X}}_{k|k}^n) - \mathbf{H}^n (\hat{\mathbf{X}}_{k|k}^n) \times (\hat{\mathbf{X}}_{k|k-1}^n - \hat{\mathbf{X}}_{k|k}^n) \right] \quad (8)$$

$$\mathbf{P}_{k|k}^n = \left[\mathbf{I} - \mathbf{K}_k^n \mathbf{H}^n (\hat{\mathbf{X}}_{k|k}^n) \right] \mathbf{P}_{k|k-1} \left[\mathbf{I} - \mathbf{K}_k^n \mathbf{H}^n (\hat{\mathbf{X}}_{k|k}^n) \right]^T + \mathbf{K}_k^n \mathbf{R} (\mathbf{K}_k^n)^T \quad (9)$$

here, $\mathbf{H}^n (\hat{\mathbf{X}}_{k|k}^n) = \frac{\partial \mathbf{h} (\hat{\mathbf{X}}_{k|k}^n)}{\partial \hat{\mathbf{X}}_{k|k}^n}$, n is the number of iteration and $n = 1, 2, \dots, j$. Then:

$$\hat{\mathbf{X}}_{k|k} = \hat{\mathbf{X}}_{k|k}^j \quad (10)$$

$$\mathbf{P}_{k|k} = \mathbf{P}_{k|k}^j \quad (11)$$

2.2. Extended Rauch-Tung-Striebel Smoothing

Consider the nonlinear system given by Equations (1) and (2), the forward data processing of ERTSS is utilizing a set of equations as follows:

$$\hat{\mathbf{X}}_{k|k-1} = \mathbf{A}_{k-1} \hat{\mathbf{X}}_{k-1|k-1} \quad (12)$$

$$\mathbf{P}_{k|k-1} = \mathbf{A}_{k-1} \mathbf{P}_{k-1} \mathbf{A}_{k-1}^T + \mathbf{Q} \quad (13)$$

$$\mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{H}_k^T \left[\mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T + \mathbf{R} \right]^{-1} \quad (14)$$

$$\hat{\mathbf{X}}_{k|k} = \hat{\mathbf{X}}_{k|k-1} + \mathbf{K}_k \left[\mathbf{z}_k - \mathbf{h} (\hat{\mathbf{X}}_{k|k-1}) \right] \quad (15)$$

$$\mathbf{P}_{k|k} = \left[\mathbf{I} - \mathbf{K}_k \mathbf{H}_k \right] \mathbf{P}_{k|k-1} \quad (16)$$

where $\mathbf{A}_k = \frac{\partial \mathbf{f} (\hat{\mathbf{X}}_{k|k})}{\partial \hat{\mathbf{X}}_{k|k}}$, $\mathbf{H}_k = \frac{\partial \mathbf{h} (\hat{\mathbf{X}}_{k|k})}{\partial \hat{\mathbf{X}}_{k|k}}$. The backward data processing propagates the filtering outputs

and achieves the smoothing results by using the R-T-S formulation. It is computed with the following equations:

$$\mathbf{K}_k^S = \mathbf{P}_{k|k} \mathbf{A}_k^T (\mathbf{P}_{k+1|k})^{-1} \quad (17)$$

$$\hat{\mathbf{X}}_{k|k}^S = \hat{\mathbf{X}}_{k|k} + \mathbf{K}_k^S \left[\hat{\mathbf{X}}_{k+1|k+1}^S - \hat{\mathbf{X}}_{k|k-1} \right] \quad (18)$$

$$\mathbf{P}_k^S = \mathbf{P}_{k|k} + \mathbf{K}_k^S (\mathbf{P}_{k+1}^S - \mathbf{P}_{k|k-1}) (\mathbf{K}_k^S)^T \quad (19)$$

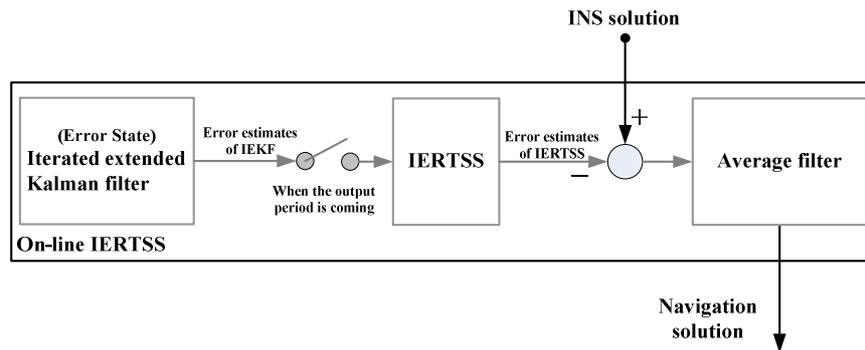
where the superscript S denotes the smoothing, and the recursion [Equations (17)–(19)] is started from the filtering output at the final time.

2.3. On-Line Iterated Extended Rauch-Tung-Striebel Smoothing

As mentioned above, it can be seen that the filtering output accuracy of ERTSS is dependent on the EKF, however, the EKF will generate truncated errors since it employs Taylor series expansion to

linearize the nonlinear system. In this work, in order to obtain a higher accuracy solution, the IEKF mentioned above is used in the forward data processing part of ERTSS (called IERTSS). Moreover, in order to achieve on-line smoothing, this work proposes an on-line IERTSS. The flow chart of this on-line IERTSS is shown in Figure 1. In this mode, the IEKF is used for optimal state estimation. When the output periods are coming, firstly, the IERTSS is employed to smooth the filtering output of IEKF between two data output periods. Then, the average value of the INS state estimation is computed with the INS solution and IERTSS solution.

Figure 1. The flow chart of the on-line IERTSS.



3. Unbiased Tightly-Coupled Integrated Model for Mobile Robot Navigation Indoors

The unbiased tightly-coupled integrated model proposed in [21] is employed in this work. The continuous-time state equation of the filter is illustrated in Equation (20):

$$\begin{bmatrix} \delta \dot{P}_{E,k} \\ \delta \dot{V}_{E,k} \\ \delta \dot{Acc}_{E,k} \\ \delta \dot{P}_{N,k} \\ \delta \dot{V}_{N,k} \\ \delta \dot{Acc}_{N,k} \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 & T & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & T \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}}_{A^*} \begin{bmatrix} \delta P_{E,k-1} \\ \delta V_{E,k-1} \\ \delta Acc_{E,k-1} \\ \delta P_{N,k-1} \\ \delta V_{N,k-1} \\ \delta Acc_{N,k-1} \end{bmatrix} + W_k^* \quad (20)$$

where $(\delta P_{E,k}, \delta P_{N,k})$, $(\delta V_{E,k}, \delta V_{N,k})$ and $(\delta Acc_{E,k}, \delta Acc_{N,k})$ are the errors of position, velocity and acceleration measured by the INS in east and north direction at moment k . T is the sample time; W_k^* is the Gaussian process noise. Equation (20) can be transferred into a discrete-time state equation:

$$\begin{bmatrix} \delta P_{E,k} \\ \delta V_{E,k} \\ \delta Acc_{E,k} \\ \delta P_{N,k} \\ \delta V_{N,k} \\ \delta Acc_{N,k} \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & T & T^2/2 & 0 & 0 & 0 \\ 0 & 1 & T & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & T & T^2/2 \\ 0 & 0 & 0 & 0 & 1 & T \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}}_A \begin{bmatrix} \delta P_{E,k-1} \\ \delta V_{E,k-1} \\ \delta Acc_{E,k-1} \\ \delta P_{N,k-1} \\ \delta V_{N,k-1} \\ \delta Acc_{N,k-1} \end{bmatrix} + W_k \quad (21)$$

where \mathbf{W}_k is the Gaussian process noise. Here, the position of the robot measured by the INS is denoted as (P_E^{INS}, P_N^{INS}) , and the position of the RN is denoted as (x, y) . Thus, the distance between the robot and the RN measured by the INS can be expressed as Equation (22):

$$d_i^{INS} = \sqrt{(P_E^{INS} - x_i)^2 + (P_N^{INS} - y_i)^2}, i = 1, 2, \dots, m \tag{22}$$

where m is the number of the RN. Theoretically, the real distance between the robot and the RN is expressed as Equation (23):

$$d_i^{Real} = \sqrt{(P_E^{Real} - x_i)^2 + (P_N^{Real} - y_i)^2}, i = 1, 2, \dots, m \tag{23}$$

where d_i^{Real} is the real distance between the robot and the RN, (P_E^{Real}, P_N^{Real}) is the real position of the robot. The difference between $(d_i^{INS})^2$ and $(d_i^{Real})^2$ is denoted as Δd_i^2 , and it is expressed as Equation (24):

$$\Delta d_i^2 = (d_i^{INS})^2 - (d_i^{Real})^2 = (P_E^{INS} - x_i)^2 + (P_N^{INS} - y_i)^2 - [(P_E^{Real} - x_i)^2 + (P_N^{Real} - y_i)^2], i = 1, 2, \dots, m \tag{24}$$

The real robot position can be computed by Equation (25):

$$P_E^{Real} = P_E^{INS} - \delta P_E, P_N^{Real} = P_N^{INS} - \delta P_N \tag{25}$$

Thus, the Equation (24) can be transferred to Equation (26):

$$h_{d_i}(\delta P_E, \delta P_N) = \Delta d_i^2 = 2(P_E^{INS} - x_i)\delta P_E + 2(P_N^{INS} - y_i)\delta P_N - (\delta P_E^2 + \delta P_N^2), i = 1, 2, \dots, m \tag{26}$$

The final matrix of the measurement equation at k moment is illustrated in Equation (27):

$$\underbrace{\begin{bmatrix} \Delta V_{E,k} \\ \Delta V_{N,k} \\ \Delta d_{1,k}^2 \\ \Delta d_{2,k}^2 \\ \vdots \\ \Delta d_{m,k}^2 \end{bmatrix}}_{\mathbf{y}_k} = \underbrace{\begin{bmatrix} \delta V_{E,k} \\ \delta V_{N,k} \\ h_{d_1}(\delta P_{E,k}, \delta P_{N,k}) \\ h_{d_2}(\delta P_{E,k}, \delta P_{N,k}) \\ \vdots \\ h_{d_m}(\delta P_{E,k}, \delta P_{N,k}) \end{bmatrix}}_{\mathbf{h}(\mathbf{X}_k)} + \mathbf{v}_k \tag{27}$$

where \mathbf{v}_k is the Gaussian process noise, $(\Delta V_E, \Delta V_N)$ are the differences between INS velocity and WSN velocity in the east and north direction respectively. It is assumed that $\boldsymbol{\omega}_k$ and \mathbf{v}_k are independent zero-mean white Gaussian sequences with covariance \mathbf{Q} and \mathbf{R} , respectively.

The configuration of the data fusion for the integrated navigation in this work is shown in Figure 2.

4. Indoor localization Tests and Performance

4.1. Real Indoor Test Environment

In this work, two real indoor tests were done to assess the performance of the proposed method. The testbed composes of one robot and six RNs. The prototype of the robot used in this work is shown in

Figure 3. The robot is composed of an IMU, an ultrasonic sender and a wireless communication board. Its size is 380 mm × 380 mm × 400 mm (length × width × height). The performance characteristics of the IMU used in this work are listed in Table 1. The robot is the carrier of the IMU and the ultrasonic sender. It is able to collect the IMU datum and the distances between the robot and the RNs by using the PC fixed on the robot. Figure 4 shows the implemented prototype of the RN. Its size is 120 mm × 60 mm × 80 mm (length × width × height). Here, the RN is used to receive the ultrasonic ranging signal sent by the ultrasonic sender and the distance between the RNs and robot can be calculated. It is also able to send the sensor data to the ultrasonic sender when it gets the command. The real indoor test environment is shown in Figure 5, and the positions of the RNs are also marked in Figure 5.

Figure 2. The configuration of the data fusion for the integrated navigation.

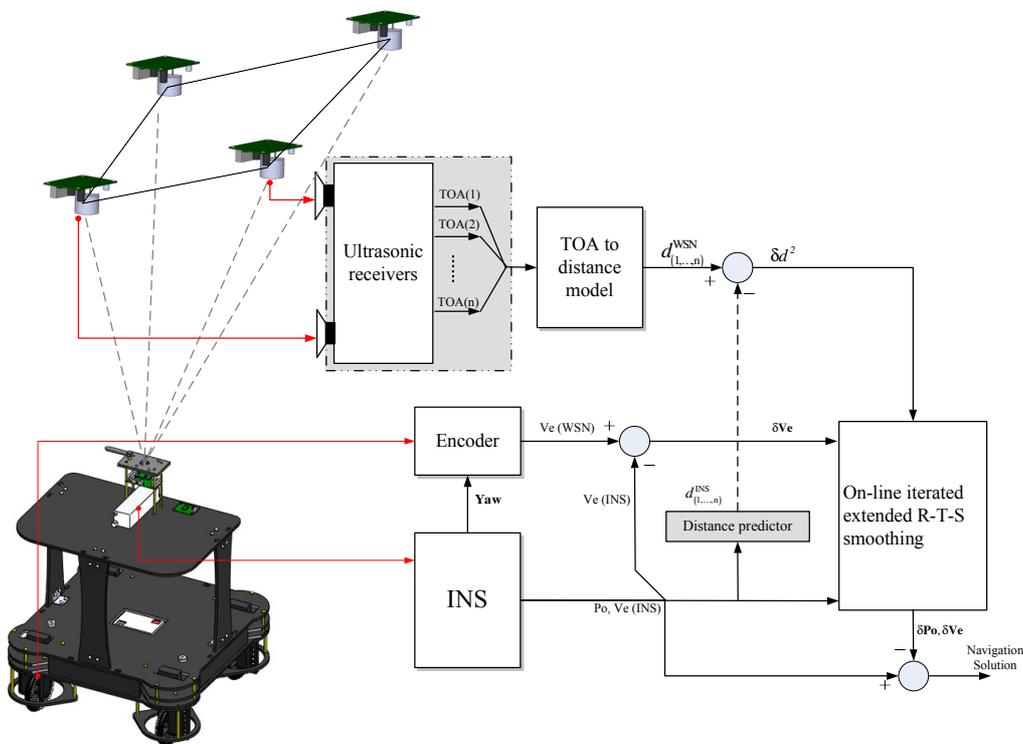


Figure 3. The prototype of the robot.

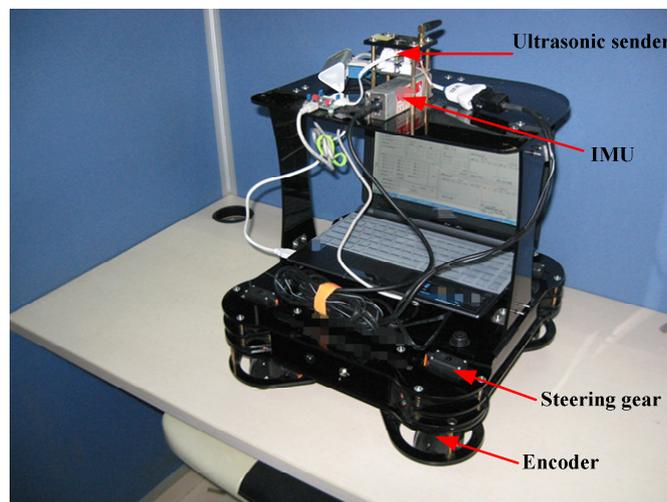


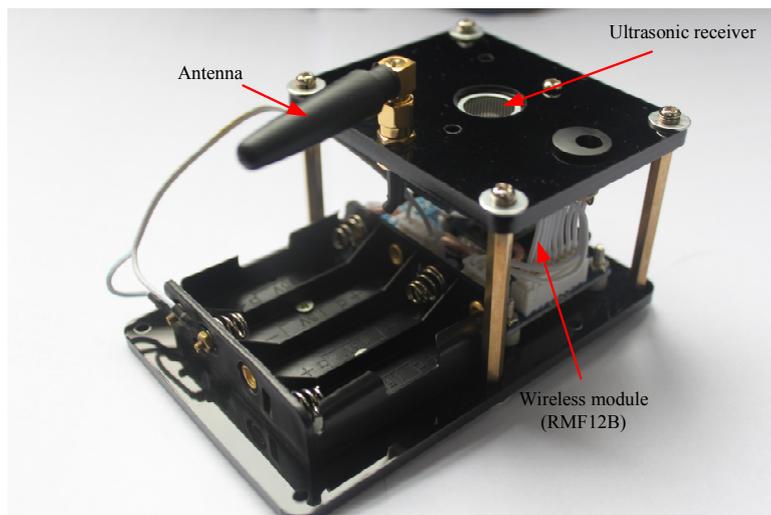
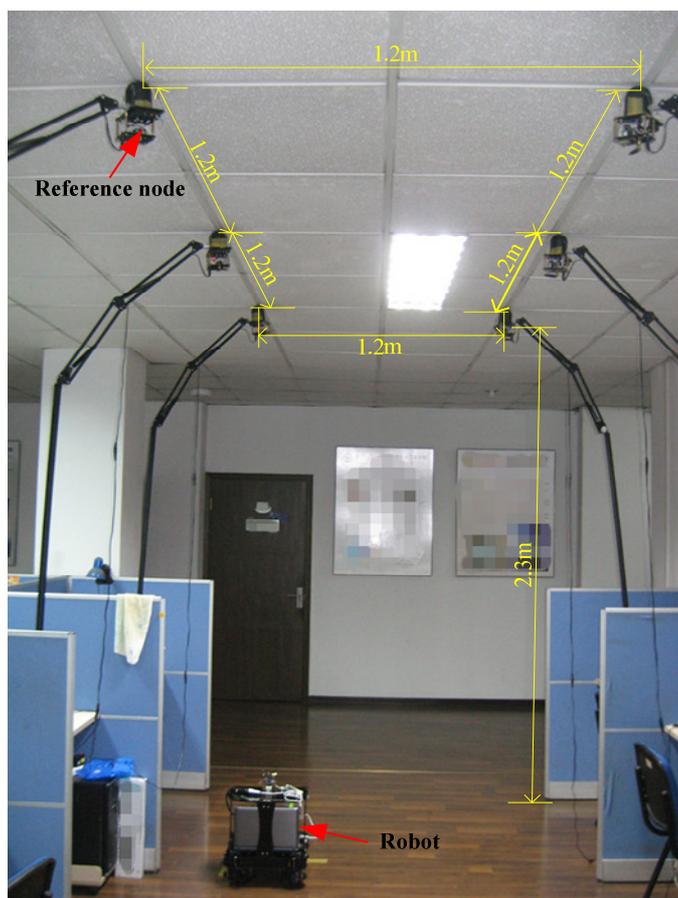
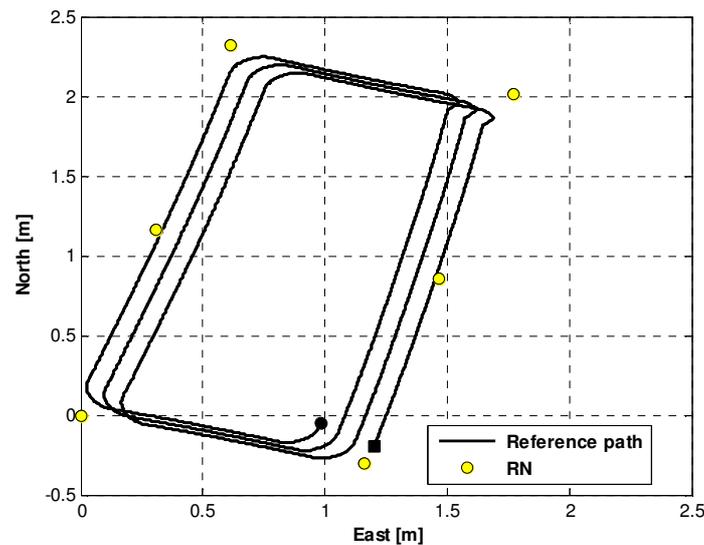
Figure 4. The prototype of the RN.**Figure 5.** Real indoor test environment.

Figure 6 displays the trajectory of the real test. The robot runs from the beginning point (denoted by a black square) to the end point (denoted by a black circle) at a speed of 0.33 m/s. Meanwhile, the RNs are denoted by yellow circles in Figure 6.

Figure 6. The trajectory of the real test.**Table 1.** The performance parameters for the IMU used in this work.

Parameters	Description	Data
Angular Rate	Input Range: Yaw, Pitch, Roll	± 300 deg/s
	Bias	0.02 deg/s RMS
	Scale Factor Accuracy	0.2%
	Non-Linearity	0.1% FS
	Random Walk	6 deg/sqrt (h)
Linear acceleration	Input Range: X/Y/Z	± 2 g
	Bias	0.3 mg RMS
	Scale Factor Accuracy	<0.1%
	Non-Linearity	0.2% FS
	Random Walk	0.06 deg/sqrt (h)

4.2. Algorithm Implementation

The pseudocode of the main program is presented in Table 2. In the pseudocode, the percentage symbol, “%,” is used to mark the comments. The software methodology is implemented in the Matlab programming language. The data refresh rate of the netbook computer is 50 Hz. Sensor data can be stored at the end of each test for subsequent analysis.

4.3. The Performance of the Off-Line IERTSS

In this section, the experimental results when the off-line IERTSS works are discussed. In Figure 7, the position errors for INS-only and WSN in the east direction and north direction are shown in Figure 7a,c, respectively. The position errors for the WSN, EKF, ERTSS and off-line IERTSS in the east direction and north direction are shown in Figure 7b,d, respectively. Furthermore, the root mean square errors (RMSE) of position and velocity for the INS-only, WSN, EKF, ERTSS and off-line IERTSS are shown in Tables 3 and 4, respectively.

Table 2. The pseudocode of the proposed algorithm.

State (s), Covariance (Cov), Position (Po), Velocity (ve), Probability (pp)
1. procedure MAIN % Main program
2. [s, Cov, Po, ve, pp] ← Initialize();
3. Start IMU & ultrasonic measurements ();
4. loop % 50 Hz rate
5. WaitNextIMUSample;
6. [$\omega^b, f^b, \mathbf{Y}_{a\omega}$] ← GetIMUData();
7. [$\mathbf{v}_E^{INS}, \mathbf{v}_N^{INS}$] ← IMUattitudesolution(ω^b, f^b);
8. [$\mathbf{P}_E^{INS}, \mathbf{P}_N^{INS}$] ← DeadReckoning($\mathbf{v}_E^{INS}, \mathbf{v}_N^{INS}$);
9. Data _{Ultrasonic} ← GetUltrasonicData();
10. $TOA_{(1,...,n)}^{Ultrasonic}$ ← MeanTOA(Data _{Ultrasonic});
11. $d_{(1,...,n)}^{WSN}$ ← TOAtoDistanceModel($TOA_{(1,...,n)}^{Ultrasonic}$);
12. $d_{(1,...,n)}^{INS}$ ← INSPositionToDistanceModel($\mathbf{P}_E^{INS}, \mathbf{P}_N^{INS}$);
13. v_e^{WSN} ← GetVelFromCodeWheel();
14. [$\mathbf{v}_E^{WSN}, \mathbf{v}_N^{WSN}$] ← GetNorth&EastVel($v_e^{WSN}, \mathbf{Y}_{a\omega}$);
15. z ← [$\delta\mathbf{v}_e, \delta\mathbf{d}^2$]; % Measurements
16. Cov _z ← GetCovZ(z);
17. s ← [$\mathbf{P}_{o_{i-1}}^+, \mathbf{v}_{e_{i-1}}^+$];
18. Cov ← [$\mathbf{P}_{i-1}^+, \mathbf{P}_{i-1}^-, \text{Cov}_z$];
19. [s, Cov] ← IEKF(z, s, Cov);
21. FilterData ← StoreFilterData(s, Cov);
22. if (OutputData == 1)
23. [$\delta\mathbf{P}_E, \delta\mathbf{P}_N, \delta\mathbf{v}_E, \delta\mathbf{v}_N$] _{IERTSS} ← IERTSS(s, Cov);
24. [$\mathbf{P}_E, \mathbf{P}_N, \mathbf{v}_E, \mathbf{v}_N$] _{IERTSS} ← [$\mathbf{P}_E^{INS}, \mathbf{P}_N^{INS}, \mathbf{v}_E^{INS}, \mathbf{v}_N^{INS}$] − [$\delta\mathbf{P}_E, \delta\mathbf{P}_N, \delta\mathbf{v}_E, \delta\mathbf{v}_N$] _{IERTSS}
25. [$\mathbf{P}_E, \mathbf{P}_N, \mathbf{v}_E, \mathbf{v}_N$] _{Avg. IERTSS} ← GetAverageValue([$\mathbf{P}_E, \mathbf{P}_N, \mathbf{v}_E, \mathbf{v}_N$] _{IERTSS});
26. OutputData();
27. endif
28. end loop
29. Stop IMU & ultrasonic measurements ();
30. StoreSession(All variables); % For analysis
31. end procedure

Figure 7a shows the east position error of the INS-only and WSN. From the figure, one can easily see that though the INS-only solution is continuous, and the east position INS error is accumulated since the DR-based current position has to depend on the previous moment. In Figure 7a, the east position INS error increases to about 190 m in 65 s without any correction. The RMSE of the INS-only solution is 85.89 m. Thus, it is necessary to correct the INS solution. Moreover, we can see that the WSN is able to maintain the east position error compared with the INS-only solution. Table 3 shows that its RMSE is 11.78 cm, which is lower than the INS solution. The east position errors for the WSN, EKF, ERTSS and off-line IERTSS are shown in Figure 7b. From the figure, it can be seen that the estimation accuracy in terms of east position for EKF is superior to that for WSN. The EKF reduces the RMSE in the east direction by about 41.17% compared with the WSN solution. Regarding the smoothing methods, it is evident that both the ERTSS and the off-line IERTSS are effective to reduce

the RMSE. Table 3 shows that the position RMSE for the off-line IERTSS is lower than that for ERTSS. The off-line IERTSS reduces the position RMSE by about 38.22% compared with ERTSS.

Figure 7. The position error for the INS-only, WSN, EKF, ERTSS and off-line IERTSS. (a) and (b) East direction; (c) and (d) North direction.

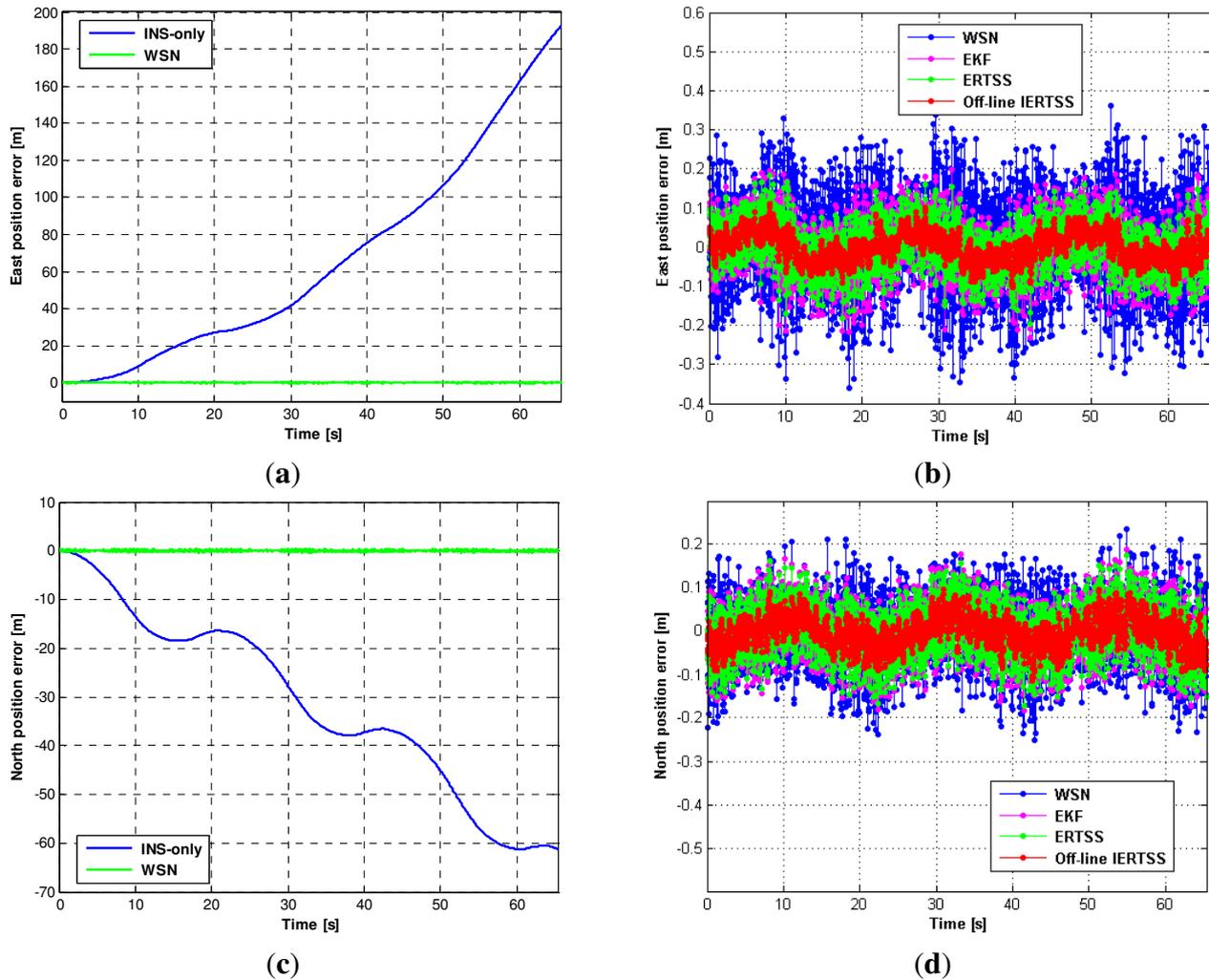


Table 3. Comparison of five estimation strategies in terms of position error.

Method	RMSE (cm)		
	East	North	Mean
INS-only	8589.14	3625.92	6107.53
WSN	11.78	7.71	9.74
EKF	6.93	6.11	6.52
ERTSS	5.73	5.55	5.64
Off-line IERTSS	3.54	3.46	3.50

The north position errors for the INS-only, WSN, EKF, ERTSS and off-line IERTSS are shown in Figure 7c,d, respectively. The trend in this figure is similar to that in Figure 7a,b. The north position error of INS is also accumulated. The WSN solution is also able to maintain the accuracy of the position. From Table 3, we can see that the RMSE of the north position for WSN remains at about 7.71 cm

since the WSN solution just depends on the current measurement. Like Figure 7b, the off-line IERTSS solution also has the lowest error. The RMSE of the north position for the off-line IERTSS is 3.46 cm. The improvement in RMSE is about 43.37% and 37.66% compared with the EKF and ERTSS, respectively. In summary, the off-line IERTSS is the most effective method to reduce the position error compared with the INS-only, WSN, EKF and ERTSS.

The RMSE results of velocity for the INS-only, WSN, EKF, ERTSS and off-line IERTSS are shown in Table 4. From the table, it can be seen that the off-line smoothing-based methods are able to effectively reduce the velocity error of the filter. However, the ERTSS and off-line IERTSS solution are almost the same both in the east and north direction, respectively.

Table 4. Comparison of five estimation strategies in terms of velocity error.

Method	RMSE (cm/s)		
	East	North	Mean
INS-only	343.49	131.48	237.48
WSN	5.73	8.79	7.26
EKF	4.07	6.12	5.10
ERTSS	3.15	4.73	3.94
Off-line IERTSS	2.77	2.79	2.78

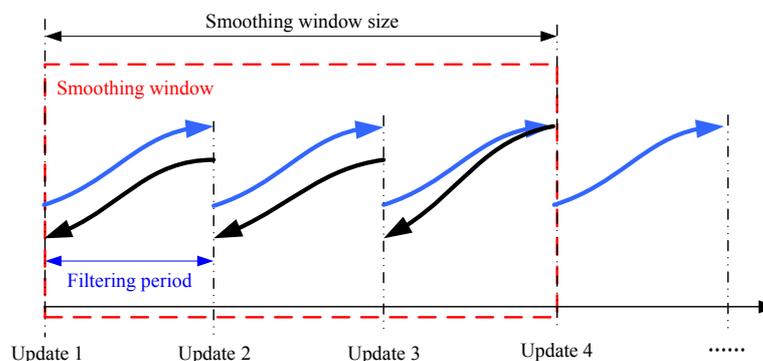
4.4. The Relationship between Smoothing Window Size and Accuracy for the On-Line IERTSS

In this section, the relationship between smoothing window size and accuracy for the on-line IERTSS is discussed. The relation between the smoothing window size and the filtering period is shown in Figure 8. From the figure, we can see that the relation can be expressed by the following equation:

$$(\text{Smoothing window size}) = n \cdot (\text{Filtering period}) = n \cdot T \quad (28)$$

Tables 5 and 6 and Figure 9 show the testing results. From the results, we can see that both the RMSE of position and that of velocity reduce as the smoothing window size increases on the beginning, then the RMSE increases rapidly.

Figure 8. The relation between the smoothing window size and the filtering period.



From Table 5, we can see that RMSE of position for the on-line IERTSS is lowest when the smoothing window size is 6, so for the velocity, the solution is 6. Table 5 displays the RMSE of

position in the east and north direction with the different smoothing window sizes, while, the average position RMSE in the east and north direction is also shown in the last line. The position RMSE in the east and north direction with the different smoothing window size is shown in Table 6, and the average position RMSE in the east and north direction is also shown in the last line. In order to obtain a high accuracy navigation solution, we take the average of the average RMSE for position and velocity, and the result is shown in Table 7. From that table, we can see when the smoothing window size is 6, the average value is lowest.

Table 5. Relationship between smoothing window size and position RMSE (cm).

Size	East	North	Average
1 T	4.37	4.16	4.27
2 T	4.22	3.95	4.09
3 T	4.07	3.84	3.96
4 T	3.94	3.73	3.84
5 T	3.83	3.75	3.79
6 T	3.72	3.73	3.73
7 T	3.71	3.77	3.74
8 T	3.69	3.83	3.76
9 T	3.65	3.97	3.81
10 T	3.60	4.06	3.83
20 T	4.30	5.92	5.11
40 T	7.02	10.59	8.80
60 T	15.56	10.01	12.78
80 T	13.02	20.21	16.61
100 T	15.06	25.47	20.26

Table 6. Relationship between smoothing step and velocity RMSE (cm/s).

Size	East	North	Average
1 T	3.94	4.21	4.07
2 T	3.72	4.07	3.89
3 T	3.78	3.92	3.85
4 T	3.71	3.90	3.81
5 T	3.61	3.98	3.79
6 T	3.68	3.80	3.74
7 T	3.93	3.81	3.87
8 T	4.02	3.94	3.98
9 T	4.52	3.98	4.25
10 T	3.90	3.97	3.93
20 T	4.59	5.24	4.92
40 T	6.52	7.27	6.90
60 T	8.10	7.11	7.61
80 T	9.98	9.76	9.87
100 T	12.35	9.30	10.83

Table 7. The average of average RMSEs for position and velocity.

Size	Position (cm)	Velocity (cm/s)	Average
1 T	4.27	4.07	4.17
2 T	4.09	3.89	3.99
3 T	3.96	3.85	3.90
4 T	3.84	3.81	3.82
5 T	3.79	3.79	3.79
6 T	3.73	3.74	3.73
7 T	3.74	3.87	3.81
8 T	3.76	3.98	3.87
9 T	3.81	4.25	4.03
10 T	3.83	3.93	3.88
20 T	5.11	4.92	5.01
40 T	8.80	6.90	7.85
60 T	12.78	7.61	10.19
80 T	16.61	9.87	13.24
100 T	20.26	10.83	15.55

4.5. Comparison of On-Line and Off-Line IERTSS

In this section, the comparison of on-line and off-line IERTSS is discussed. The position errors for the off-line and on-line IERTSS are shown in Figure 10. The comparison of on-line and off-line mode in terms of position error is shown in Table 8. From the figures, one can see that both in the east direction and north direction, the position error of the on-line and the off-line IERTSS solutions are almost the same, and the off-line mode is a little better than the on-line mode. Table 9 displays the comparison of on-line and off-line mode in terms of velocity error. Like the position error, the on-line and the off-line IERTSS solutions are also almost the same, and the off-line mode is better than the on-line mode. However, the performance of on-line IERTSS is little better than the ERTSS.

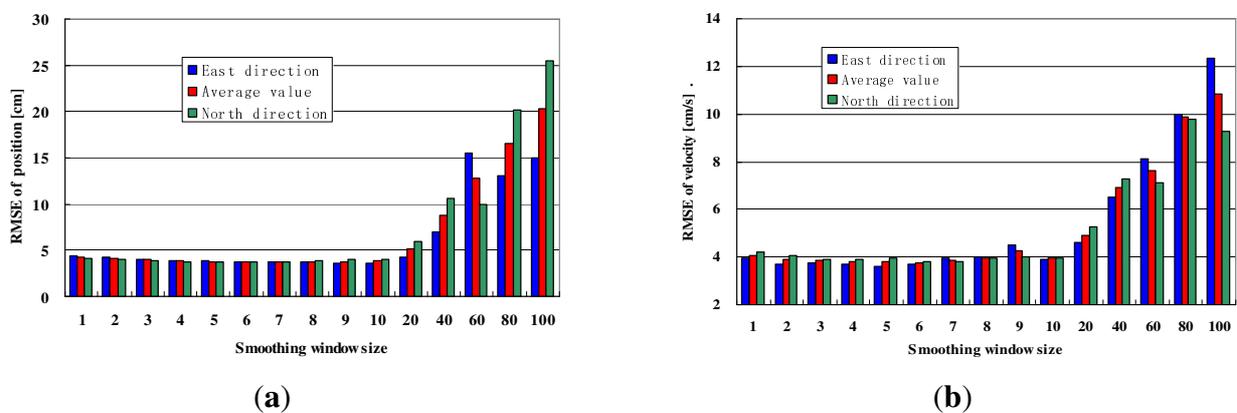
Figure 9. Relationship between smoothing window size and RMSE. (a) Position and (b) Velocity.

Figure 10. The position error for the off-line IERTSS and on-line IERTSS. (a) East direction and (b) North direction.

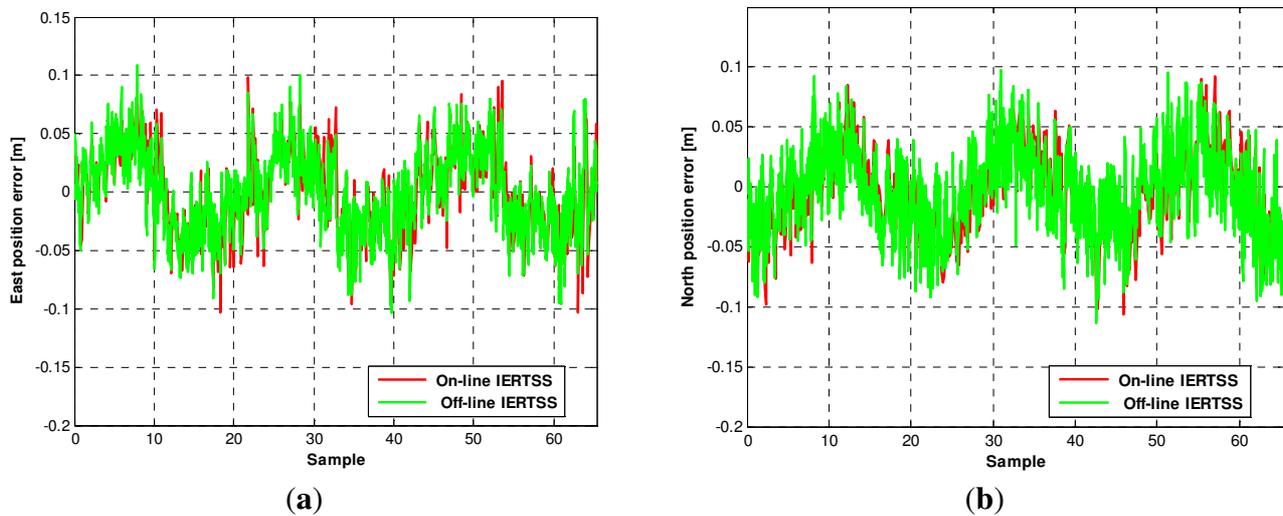


Table 8. Comparison of on-line and off-line mode in terms of position error.

Method	RMSE (cm)		
	East	North	Mean
Off-line IERTSS	3.54	3.46	3.50
On-line IERTSS	3.72	3.73	3.73

Table 9. Comparison of on-line and off-line mode in terms of velocity error.

Method	RMSE (cm/s)		
	East	North	Mean
Off-line IERTSS	2.77	2.79	2.78
On-line IERTSS	3.68	3.80	3.74

5. Conclusions

This work proposed an on-line IERTSS for tightly integrated INS/WSN mobile robot navigation indoors. In this mode, IEKF is employed instead of the EKF in forward data processing of the IERTSS. Then, IERTSS is embedded into average filter for on-line smoothing. The experimental results show that the proposed on-line smoothing outperforms the ERTSS. The performance of on-line smoothing is also comparable to that of off-line smoothing. The results show that the performance of the on-line and off-line mode is almost the same, and the off-line mode is a little better than the on-line mode.

Acknowledgments

This work was supported in part by National Natural Science Foundation of China (No. 51375087, 41204025, 50975049), Ocean Special Funds for Scientific Research on Public Causes (No. 201205035-09), Specialized Research Fund for the Doctoral Program of Higher Education (No. 20110092110039), the 52th China Postdoctoral Science Foundation (No. 2012M520967), and the

Program Sponsored for Scientific Innovation Research of College Graduate in Jiangsu Province, China (No.CXLX_0101).

Conflicts of Interest

The authors declare no conflict of interest.

References

1. Kim, S.J.; Kim, B.K. Accurate hybrid global self-localization algorithm for indoor mobile robots with two-dimensional isotropic ultrasonic receivers. *IEEE Trans. Instrum. Meas.* **2011**, *10*, 3391–3404.
2. Saad, M.M.; Bleakley, C.J.; Ballal, T.; Dobson, S. High-accuracy reference-free ultrasonic location estimation. *IEEE Trans. Instrum. Meas.* **2012**, *6*, 1561–1570.
3. Zhang, J.; Song, G.-M.; Qiao, G.-F.; Meng, T.-H.; Sun, H.-T. An indoor security system with a jumping robot as the surveillance terminal. *IEEE Trans. Consum. Electron.* **2011**, *4*, 1774–1781.
4. Ruiz, A.R.J.; Granja, F.S.; Honorato, J.C.P.; Rosas, J.I.G. Accurate pedestrian indoor navigation by tightly coupling foot-mounted IMU and RFID measurements. *IEEE Trans. Instrum. Meas.* **2012**, *1*, 178–189.
5. Shirehjini, A.A.N.; Yassine, A.; Shirmohammadi, S. An RFID-based position and orientation measurement system for mobile objects in Intelligent Environments. *IEEE Trans. Instrum. Meas.* **2012**, *6*, 1664–1675.
6. Park, W.-C.; Yoon, M.-H. The Implementation of Indoor Location System to Control ZigBee Home Network. In Proceedings of the 2006 SICE-ICASE International Joint Conference, Busan, Korea, 18–21 October 2006.
7. Minami, M.; Morikawa, H.; Aoyama, T. Design and implementation of a fully distributed ultrasonic positioning system. *Electron. Commun. Jpn. Part III Fundam. Electron. Sci.* **2007**, *6*, 17–26.
8. Fang, L.; Antsaklis, P.; Montestruque, L.; McMickell, M.; Lemmon, M.; Sun, Y.; Fang, H.; Koutroulis, I.; Haenggi, M.; Xie, M.; *et al.* Design of a wireless assisted pedestrian dead reckoning system—The NavMote experience. *IEEE Trans. Instrum. Meas.* **2005**, *6*, 2342–2358.
9. Feng, G.-H.; Wu, W.-Q.; Wang, G.-L. Observability analysis of a matrix Kalman filter-based navigation system using visual/inertial/magnetic sensors. *Sensors* **2012**, *12*, 8877–8894.
10. Boulade, S.; Frapard, B.; Biard, A. GPS/INS Navigation System for Launchers and Re-Entry Vehicles-Developments and Adaptation to Ariane 5. In Proceedings of the 5th ESA International Conference on Spacecraft Guidance, Navigation and Control Systems, Frascati, Italy, 22–25 October 2002.
11. Xu, Z.-K.; Li, Y.; Rizos, C.; Xu, X.-S. Novel hybrid of LS-SVM and Kalman Filter for GPS/INS integration. *J. Navig.* **2010**, *2*, 289–299.
12. Evennou, F.; Marx, F. Advanced integration of WiFi and Inertial Navigation Systems for indoor mobile positioning. *EURASIP J. Appl. Signal Process.* **2006**, *2006*, 1–11.

13. Lobo, J.; Lucas, P.; Dias, J.; Traca de Almeida, A. Inertial Navigation System for Mobile Land Vehicles. In Proceedings of the IEEE International Symposium on Industrial Electronics ISIE'95, Athens, Greece, 10–14 July 1995.
14. El-Sheimy, N.; Schwarz, K.P. Integrating Differential GPS Receivers with an Inertial Navigation System (INS) and CCD Cameras for a Mobile GIS Data Collection System. In Proceedings of the International Society for Photogrammetry and Remote Sensing Conference, Ottawa, Canada, 6–10 June 1994.
15. Fang, J.-C.; Gong, X.-L. Predictive iterated Kalman filter for INS/GPS integration and its application to SAR motion compensation. *IEEE Trans. Instrum. Meas.* **2010**, *4*, 909–915.
16. Shademan, A.; Sharifi, F.J. Sensitivity Analysis of EKF and Iterated EKF Pose Estimation for Position-Based Visual Servoing. In Proceedings of the IEEE Conference Control Application, Toronto, Canada, 28–31 August 2005.
17. Lefebvre, T.; Bruyninckx, H.; Schutter, J.D. Kalman filters for nonlinear systems: A comparison of performance. *Int. J. Control* **2004**, *7*, 639–653.
18. Chiang, K.-W.; Duong, T.T.; Liao, J.-K.; Lai, Y.-C.; Chang, C.-C.; Cai, J.-M.; Huang, S.-C. On-line smoothing for an integrated navigation system with low-cost MEMS inertial sensors. *Sensors* **2012**, *12*, 17372–17389.
19. Rauch, H.; Tung, F.; Striebel, C. Maximum likelihood estimates of linear dynamic systems. *AIAA J.* **1965**, *8*, 1445–1450.
20. Gong, X.-L.; Zang, R.; Fang, J.-C. Application of unscented R-T-S smoothing on INS/GPS integration system post processing for airborne earth observation. *Measurement* **2013**, *3*, 1074–1083.
21. Xu, Y.; Chen, X.-Y.; Li, Q.-H. Unbiased tightly-coupled INS/WSN integrated navigation based on extended Kalman filter. *J. Chin. Inert. Technol.* **2012**, *3*, 292–299.

© 2013 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/3.0/>).