

Article

Research into Kinect/Inertial Measurement Units Based on Indoor Robots

Huixia Li ^{1,2} , Xi Wen ¹ , Hang Guo ^{1,*}  and Min Yu ³

¹ Institute of Space Science and Technology, Nanchang University, Nanchang 330031, China; hxli0601@gmail.com (H.L.); wuhuawujiu@163.com (X.W.)

² School of Resources Environment & Chemical Engineering, Ministry of Education Key Laboratory of Poyang Lake Environment and Resource Utilization, Nanchang University, Nanchang 330031, China

³ College of Computer Information and Engineering, Jiangxi Normal University, Nanchang 330022, China; myu821@163.com

* Correspondence: hguo@ncu.edu.cn; Tel.: +86-135-7610-2896

Received: 13 January 2018; Accepted: 7 March 2018; Published: 12 March 2018

Abstract: As indoor mobile navigation suffers from low positioning accuracy and accumulation error, we carried out research into an integrated location system for a robot based on Kinect and an Inertial Measurement Unit (IMU). In this paper, the close-range stereo images are used to calculate the attitude information and the translation amount of the adjacent positions of the robot by means of the absolute orientation algorithm, for improving the calculation accuracy of the robot's movement. Relying on the Kinect visual measurement and the strap-down IMU devices, we also use Kalman filtering to obtain the errors of the position and attitude outputs, in order to seek the optimal estimation and correct the errors. Experimental results show that the proposed method is able to improve the positioning accuracy and stability of the indoor mobile robot.

Keywords: indoor navigation; location; Kinect; IMU; Kalman filters

1. Introduction

In mobile robot navigation, stable and reliable positioning results are the key prerequisite for planning paths. In recent years, Kinect has increasingly been applied in robot obstacle avoidance [1,2], target reconstruction [3,4], target tracking [5,6], attitude control [7,8] and other fields due to its advantageous features. Kinect is a novel 3D stereoscopic camera developed by Microsoft, which can provide RGB and depth information of the mobile robot's environment for users, and is low in price, making it suitable for replacing conventional ultrasonic radar and laser radar as a distance sensor. Additionally, the acquired environment depth value is continuous, contains a large amount of information, is minimally influenced by light and other features, and can be used in limited-cost positioning situations with certain accuracy requirements.

Kinect is a visual sensor, and has limitations with respect to accuracy [9] and speed [10,11]. In terms of accuracy, there is a positioning error in the visualization positioning method due to the uncertain location estimation of the spatial feature points.

Take the visual odometer as an example: the position and attitude of the robot is cumulatively calculated based on the change in position between the current frame and the previous one, meaning that the accumulation of estimation errors per frame will reduce the positioning accuracy as the number of frames increases [12].

In extreme cases, feature point extraction and matching algorithms may fail in purely visual autonomous navigation due to insufficient or excessive external light. While in terms of speed, due to the huge amount of image data collected per frame, complicated processing algorithms and lower degrees of parallelism of the corresponding image mean that the speed can not be improved,

thus limiting real-time response to positioning accuracy requirements. In contrast, IMU, which has simple, real-time and all-weather completely autonomous navigation functionality, can compensate for visual measurement errors in accuracy and speed [13], although it still has many serious problems, such as accumulated positioning error.

Therefore, this paper investigated indoor mobile robot integrated navigation and positioning based on Kinect and IMU. For high-precision positioning, we firstly use feature point extraction, match the RGB image between the target frame obtained by Kinect and the reference frame, implement the Random Sample Consensus (RANSAC) algorithm to remove the mismatching points, and then use the absolute orientation algorithm to obtain the Kinect posture and offset (here, Kinect pose represents the pose of the robot). Thereby, we obtain the trajectory of mobile robot movement. The visual positioning results and the inertial navigation system (INS) data are fused by Kalman filter algorithm to improve the self-positioning accuracy of the indoor mobile robot. The detailed design process is described in the following sections.

2. Independent Localization Based on Kinect and INS

2.1. Kinect Method

2.1.1. Kinect Obtaining 3D Point Cloud Data

The pixel coordinates of the feature points in the RGB image coordinates (x_i, y_i) are known, after the angle correction in depth and RGB image, the three-dimensional coordinates of a point in the Kinect v1 coordinate system can be obtained from the depth of the image through the calculation of Equation (1):

$$\begin{aligned}x_c &= (x_i - u_0) * z / f_x \\y_c &= (y_i - v_0) * z / f_y \\z_c &= z\end{aligned}\quad (1)$$

where z is the depth value of a feature point, and u_0, v_0, f_x and f_y are the internal orientation elements of the Kinect calibrated RGB camera. The sensor Kinect operates (runs) usually at a range from 0.5 m to 4 m.

2.1.2. Absolute Orientation Algorithm

In order to improve the image of the absolute orientation accuracy of the digital close-range accuracy image, Nanshan Zheng et al. [14] describe an absolute orientation method for close-range images based on the successive correlative condition adjustment model. If the deformation of the model itself is not considered, an absolute orientation model is a spatial similarity conversion issue, including the rotation and displacement of the model coordinate system relative to the target coordinate system, and the scale factor of the model scale. The basic relation of the absolute orientation is shown in Equation (2):

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \lambda \begin{pmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{pmatrix} \begin{pmatrix} U \\ V \\ W \end{pmatrix} + \begin{pmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{pmatrix}\quad (2)$$

Among them, (U, V, W) are the points in the local reference frame (i.e., the model coordinates) of the corresponding points (X, Y, Z) , which are in the International Terrestrial Reference Frame (ITRF) (i.e., the corresponding target coordinates). Seven absolute orientation elements are: the shift of coordinate origin $(\Delta X, \Delta Y, \Delta Z)$, scale factor λ , the three rotation angles between the two coordinate systems $\{\Phi, \Omega, K\}$, and a_i, b_i, c_i are the transform coefficients between the model and target coordinate system. Absolute orientation is a method obtaining seven absolute orientation elements with known control points. Suppose $M(X_M, Y_M, Z_M), N(X_N, Y_N, Z_N), P(X_P, Y_P, Z_P), Q(X_Q, Y_Q, Z_Q)$ are the initial position coordinates of the Kinect, $m(U_m, V_m, W_m), n(U_n, V_n, W_n), p(U_p, V_p, W_p), q(U_q, V_q, W_q)$ are the adjacent initial position coordinates, wherein λ is 1. Putting coordinates of the four same-named points

into Equation (3), and using the equations listed at points N, P, Q , separately subtracting the equations listed at point M to eliminate the shift parameters, the result is:

$$\begin{bmatrix} U_N - U_M & V_N - V_M & W_N - W_M \\ U_P - U_M & V_P - V_M & W_P - W_M \\ U_Q - U_M & V_Q - V_M & W_Q - W_M \end{bmatrix} R_0 = \begin{bmatrix} X_N - X_M & Y_N - Y_M & Z_N - Z_M \\ X_P - X_M & Y_P - Y_M & Z_P - Z_M \\ X_Q - X_M & Y_Q - Y_M & Z_Q - Z_M \end{bmatrix}. \quad (3)$$

Among them:

$$R_0 = \begin{pmatrix} a_1^0 & b_1^0 & c_1^0 \\ a_2^0 & b_2^0 & c_2^0 \\ a_3^0 & b_3^0 & c_3^0 \end{pmatrix} \quad (4)$$

where R_0 is the transformation matrix transforming between the model (the local reference frame) and the target coordinate system (ITRF). The initial rotation angle of the absolute orientations are:

$$\begin{aligned} \Phi &= -\arctan(a_3^0/c_3^0) \\ \Omega &= -\arcsin(b_3^0) \\ K &= \arctan(b_1^0/b_2^0) \end{aligned} \quad (5)$$

After obtaining the scale factor and the initial rotation angle, the shift parameter ($\Delta X, \Delta Y, \Delta Z$) can be obtained by substituting arbitrary points into the equation.

2.1.3. Implementation of Kinect Self-Localization Algorithm

The scale-invariant feature transform (SIFT) matching algorithm [15] has a strong matching ability, and can match the image feature points when any two images are translated, rotated and affine transformed. In this paper, we mainly use SIFT matching, the RANSAC algorithm [16], and the absolute orientation algorithm to realize the robot self-localization scheme. In this section, a method is proposed for how to use the above algorithms to solve the robot autonomous navigation problem. This method consists of four parts: same-named point extraction, abnormal point elimination, region selection, motion parameter calculation.

Same-named point extraction: Obtain the SIFT matching points of two adjacent images from the RGB image, and then obtain the 3D coordinates of all matching point pairs from the depth image. The specific steps include using the SIFT method to extract the common key points in Image1 and Image2 first, and then obtaining the pixel coordinates of SIFTData1 and SIFTData2 in the respective RGB images.

Although the SIFT algorithm is able to extract stable feature points, there will still be some mismatching points. In order not to affect the calculation and improve the accuracy of parameters, the RANSAC algorithm is used to eliminate mismatch points in the initial matching point pair. Then the 3D coordinates of Depth1 and Depth2 of the key points are obtained from the depth images by using new positional information of Data1 and Data2.

Since Kinect is utilizes an infrared camera, it is easily influenced by noise, such as strong light, high density in the layout of feature points, and other factors, which can cause deviations in the depth information obtained by the Kinect, reducing positioning accuracy. Hence, one can use the bubble sorting method to select four intervals of relatively large feature points (generally the area formed by selected points accounting for more than 50% of the entire color image) for the absolute orientation calculation, and take the 3D coordinates of the points around these four feature points, summed and averaged together, thereby improving the accuracy of the motion parameters calculated by the absolute orientation method.

After removing the mismatching points and using regional screening, remaining at least two pairs of high-precision matching points, and so the reliability of the data is increased.

Assuming the feature point set 3D coordinates acquired by the Kinect in the first position is Data1, and the set acquired in the second position is Data2, combining the coordinates of Data1 and Data2, one can use the absolute orientation algorithm to calculate the rotation matrix and the offset vector in these two positions. Setting the robot's initial position as the origin of the coordinates, when the robot moves to the third position, the feature point set 3D coordinates acquired by the Kinect in the second position are used as the new Data1, and those from the third position as the new Data2. The new Data1 and Data2 coordinates are used to calculate the relative motion parameters from the second to the third position of the robot, and the trajectories of the robot motion can be obtained by iteration. The flowchart is shown in Figure 1.

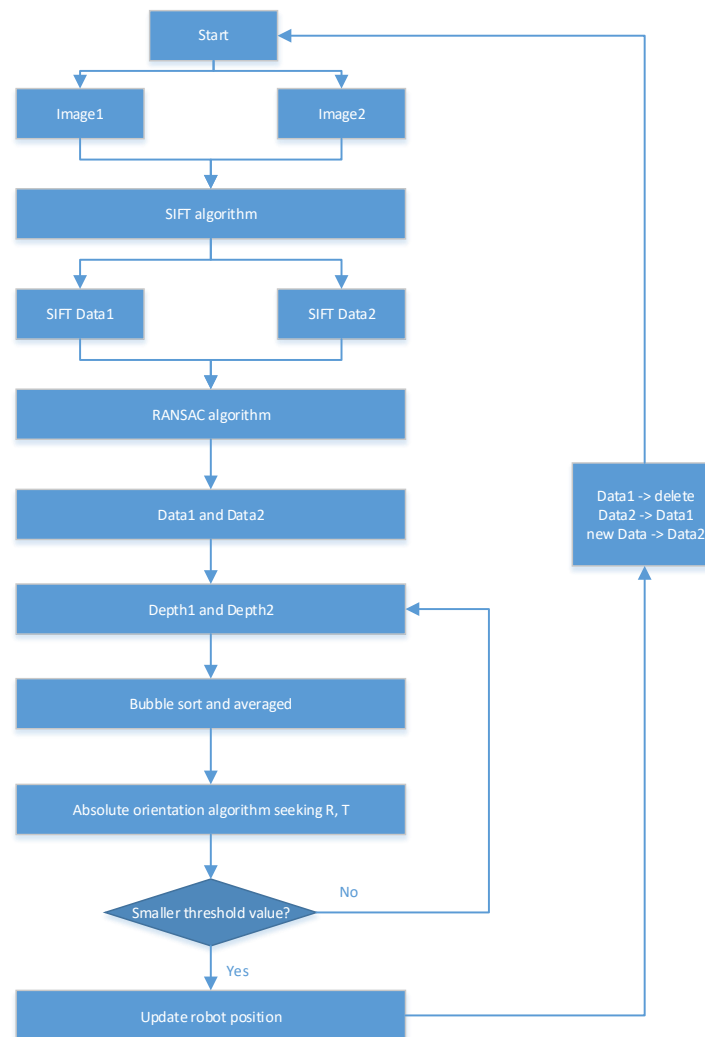


Figure 1. Flowchart of the robot self-localization method.

2.2. Principle and Algorithm Design of SINS

The trapdown Inertial Navigation System (SINS) algorithm [17,18] uses a mathematical platform, and the core part of it is the attitude updating solution. As the angular velocity vector is not a real vector, integrating the angular velocity will produce a noncommutativity rotation error. Thus, the equivalent rotation vector method is used [19]. Due to the superiority of the quaternion algorithm, a quaternion-based rotation vector algorithm is used. Considering that the accuracy and the amount are not high, a three-component algorithm is used to process the equivalent rotation vector problem.

The basic idea of SINS updating is to take the previous navigation parameters (attitude, velocity and position) as initial values, and to calculate the current navigation parameters by using the outputs

(angular velocity and acceleration) of the inertial device from the previous time to the current time. Then it takes recursively the current time as the initial values for the next moment calculation. The basic formulas are as follows.

Velocity update algorithm:

$$v_m^n = v_{m-1}^n + F_{v \rightarrow q} \left(\frac{-\omega_{inm-1}^n T_m}{2} \right) q_{bm-1}^n \Delta v_{sfm} + [g_{m-1} - (2\omega_{iem-1}^n + \omega_{enm-1}^n) \times v_{m-1}^n] T_m \quad (6)$$

$$\Delta v_{sfm} = \Delta v_m + 0.5 \Delta \theta_m \times \Delta v_m + \left\{ \left[\sum_{i=1}^{n-1} k_i \Delta \theta_m(i) \times \Delta v_m(n) \right] + \left[\sum_{i=1}^{n-1} k_i \Delta v_m(i) \times \Delta \theta_m(n) \right] \right\} \quad (7)$$

Location update algorithm:

$$L_m = L_{m-1} + \frac{T_m v_{Nm, m-1}^n}{R_M} \quad (8)$$

$$\lambda_m = \lambda_{m-1} + \frac{T_m \sec L_{m-1} v_{Em, m-1}^n}{R_N} \quad (9)$$

$$h_m = h_{m-1} + T_m v_{Um, m-1}^n \quad (10)$$

Attitude update algorithm:

$$q_{bm}^n = q_{bm-1}^n F_{v \rightarrow q} (\Phi_m - q_{bm-1}^n * \omega_{inm-1}^n T_m) \quad (11)$$

where L_m is the longitude, λ_m is the latitude, h_m is the height, T_m is the sampling period, Φ_m is equivalent rotation matrix, $*$ is common rail, and Δv_{sfm} is the speed compensation amount caused by the force (Sculling compensation term), as the gyroscope outputs are the signals of angular increment and velocity increment, they cannot be calculated directly. Equation (7) is a quadratic fitting of the angular rate and acceleration; that is, a three-subsample algorithm.

As the SINS posture updating with a coning motion, it will lead to a serious drift of the mathematical platform, so it is necessary to further optimize the rotation vector algorithm [20,21]. The following formula is the rotation vector optimized by the three-subsample algorithm.

$$\Phi_m = \Delta \theta_1 + \Delta \theta_2 + \Delta \theta_3 + \frac{9}{20} \Delta \theta_1 \times \Delta \theta_3 + \frac{27}{40} \Delta \theta_2 \times (\Delta \theta_3 - \Delta \theta_1) \quad (12)$$

The longitude, latitude, and height have to be transformed into the local coordinate system. For a coordinate transformation from the International Terrestrial Reference Frame (ITRF), such as the World Geodetic System (WGS) 84, to the local coordinate system, several points with the coordinates of both systems were calculated to support the indoor positioning using a local coordinate system.

3. Integrated Navigation Scheme

Simple visual navigation will have a cumulative error over a long time, and Global Positioning System (GPS) loses signal indoors and cannot be positioned, so it is necessary to introduce other navigation systems to amend traditional navigation systems. In conclusion, a Kalman filter integrated navigation system based on Kinect/IMU is used to reduce the visual error, so that the positioning accuracy of mobile robots in indoor autonomous navigation is improved.

The Kalman filter [22] is an optimal linear estimation method, which uses the state-space description method to describe the system and makes continuous estimations through the estimation errors of various sensor outputs to optimally combine the information of all the sensors. This optimally estimated value minimizes the linear quadratic loss of the estimation error, including the mean square error of any linear combination of state estimation errors. Note that the Kalman gain is the optimal weighting matrix.

The Kalman filter includes time updating and measurement updating. Measurement updating is performed by measuring the new information obtained from the sensor in order to update the estimated value and error. Time updating mainly considers the uncertain dynamic image, update estimation and estimation error. Since the used state equation is based on the errors, this paper does not deal with the extended Kalman filter. A filter that constitutes a feedback pulse is used here. Basic equations of the discrete Kalman filter are shown below.

One-step state prediction equation:

$$\hat{X}_{k|k-1} = \Phi_{k|k-1} \hat{X}_{k-1} \quad (13)$$

where \hat{X}_{k-1} is the system state estimation value at t_{k-1} moment, $\hat{X}_{k|k-1}$ is the system state predictive value at t_k .

State estimation equation:

$$\hat{X}_k = \hat{X}_{k|k-1} + K_k(Z_k - H_k \hat{X}_{k|k-1}) \quad (14)$$

Filter gain equation:

$$K_k = P_{k|k-1} H_k^T [H_k P_{k|k-1} H_k^T + R_k]^{-1} \quad (15)$$

One-step prediction MSE (mean square error):

$$P_{k|k-1} = \Phi_{k|k-1} P_{k-1} \Phi_{k|k-1}^T + \Gamma_{k|k-1} Q_k \Gamma_{k|k-1}^T \quad (16)$$

Estimated MSE (mean square error):

$$P_k = [I - K_k H_k] P_{k|k-1} [I - K_k H_k]^{-1} + K_k R_{k-1} K_k^T \quad (17)$$

Using the Kalman filter linear recursive algorithm to calculate the next moment measurement and state estimates, the initial value of \hat{X}_0 and P_0 should be known first. The designed Kalman filter model is used to fuse the attitude and position information of the Kinect and IMU, and can be used to modify the trajectory of IMU and make up for errors accumulating over time, thus improving the self-localization accuracy of the indoor robot. The system and filter structure are shown in Figure 2.

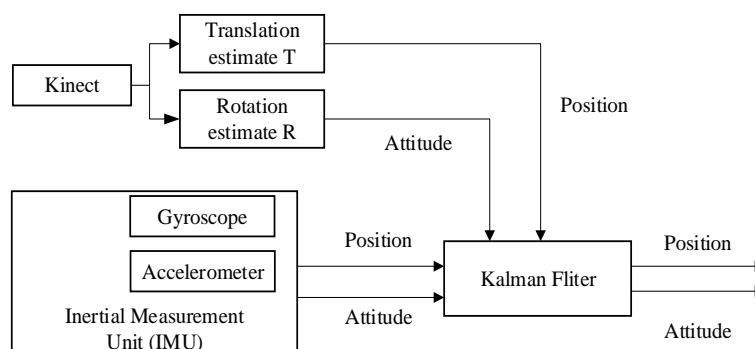


Figure 2. System and filter structure.

4. Indoor Positioning Experiment

In order to verify the positioning accuracy of the Kalman filter after fusion, an indoor experiment was designed in the study room of our institute, in which the robot went along the wall of the room. Using a WX-DP203 (designed and developed by our navigation group) mobile robot installed with a Kinect sensor and IMU, the robot tracked a wall and landmarks, as shown in Figure 3. Kinect image acquisition frequency is 1 Hz, IMU data collection frequency is 100 Hz. In order to compare the final

positioning accuracy, we preplanned the relevant path, set up three control points including the start and the end, and obtained the exact positions of the control points under the local coordinate system by using the total station. With the starting point coordinate set as 0, the measured control point coordinates are shown in Table 1.



Figure 3. Kinect and experimental platform WX-DP203.

Table 1. Control point coordinates (in meters).

Number of Control Points	1	2	3
Control Points	(0, 0)	(5.61, 0.01)	(5.60, 5.61)

During the experiment, the robot moved 11.2 m, moved no more than 20 cm per second, and obtained 116 frames of color and depth images in total. The results of the pure visual positioning and Kalman filter combination positioning are shown in Figure 4. The corresponding visual positions and the control point positions of the three set points are shown in Table 2, Kalman filter error analysis is shown in Table 3.

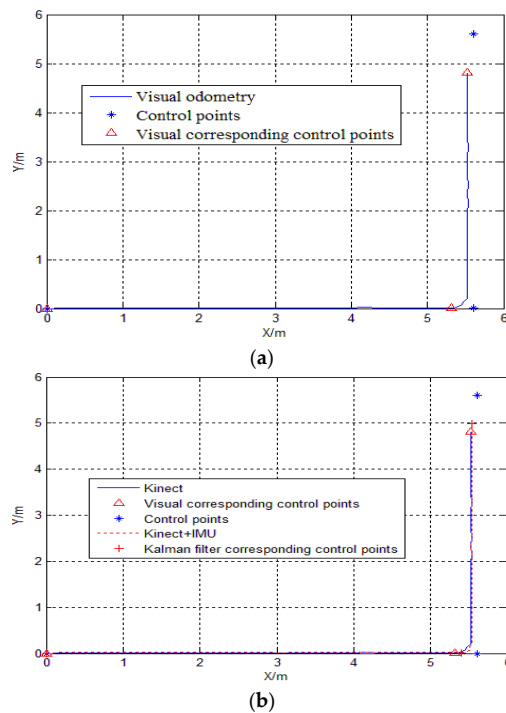


Figure 4. (a) Kinect positioning results; (b) Comparison of Kalman filter positioning trajectory.

Table 2. Position comparison of visual and control points (in meters).

Number of Control Points	1	2	3
The position of the control point	(0.00, 0.00)	(5.61, 0.01)	(5.60, 5.61)
Visual position	(0.00, 0.00)	(5.321, 0.0062)	(5.5248, 4.8182)
Distance errors (Positioning errors)	0.00	0.2890	0.7954

Table 3. Kalman filter error analysis (in meters).

Number of Control Points	1	2	3
Visual odometry	0.00	0.2890	0.7954
Kalman filter of Kinect/IMU	0.00	0.2077	0.6078

It can be seen from Table 2 that when the robot moves in a straight line of 5.61 m, the error accumulates slowly, the positioning accuracy was $0.2890/5.61 \times 100\% = 5.16\%$, which is higher. When the robot turns, the error accumulates faster, and the positioning accuracy was $0.7954/5.61 = 14.2\%$. Considering that the robot itself may sideslip during operation and Kinect sensors are susceptible to noise and other factors, there is some degree of the errors to be expected in the experimental results.

From Table 3, it can be seen that the Kinect/IMU integrated Kalman filter plays a very good role in making up for the cumulative error of the visual range. The combination of these two sensors can achieve higher positioning accuracy. The cumulative error before turning decreased from 0.2890 m to 0.2077 m, the relative visual improved by 1.36%, the cumulative error after turning decreased from 0.7954 m to 0.6078 m, and the relative visual acuity increased by 3.4%. These results show that the cumulative positioning error of the short-distance visual range is very small, while the positioning accuracy is high. We have known that the IMU strapdown solution goes from an unstable to a stable phase in the initial calculation stage, and when a large change in the posture (e.g., a turning along the route) occurs, the strapdown solution will change significantly, and the improvement in the position accuracy after following a short distance is not obvious. However, the accuracy of the orientation angle is better improved, and the stability of the orientation is also improved.

5. Conclusions

This paper combined the Kinect sensor with an IMU and the Kalman filter in order to conduct integrated positioning system research. The integrated positioning results of the Kalman filter reduce the trajectory errors of the visual range with an IMU, modify the accumulative IMU errors with the visual range, and improve the positioning accuracy of the indoor robot. The Kinect has the advantages that it utilizes an ordinary RGB camera and depth sensor, and is a cost-effective, simple, and valuable sensor. Since an IMU has been used to increase the data collection numbers with its high sampling rate and relatively high accuracy, the cumulative errors of the integrated system of our method decreased from 0.2890 m to 0.2077 m, and the relative accuracy improved by 1.36% (for accuracy improvement, see the Table 3), compared with the vision system alone. The Kinect and IMU integrated system benefited from each other, which overcame the drawbacks of them, so that improve the system stability. However, the method has still the accuracy limitation, as the research has only just commenced, and much work is still to be done. Our next step is to increase the Kinect measurement range and improve the accuracy of counting motion parameters, as this is easily affected by the noises. In addition, we will try to reduce the swing amplitude when the robot turns, thereby decreasing the attitude errors.

Acknowledgments: The paper was supported by the projects of the National Key Technologies R&D program (2016YFB0502002) and the National Natural Science Foundation of China (Nos. 41764002 and 41374039). The corresponding author is Hang Guo.

Author Contributions: Hang Guo guided the research work, Hang Guo, Huixia Li, and Xi Wen conceived and designed the experiments, Xi Wen performed the experiments, and analyzed the data; Hang Guo and Huixia Li revised the paper; Xi Wen wrote the paper. Min Yu made the verification for the paper.

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

References

1. Wu, H.B.; Huang, J.F.; Yang, X.N.; Ye, J.H.; He, S.M. A Robot Collision Avoidance Method Using Kinect and Global Vision. *Telkomnika* **2017**, *15*, 4–17. [[CrossRef](#)]
2. Cunha, J.; Pedrosa, E.; Cruz, C.; Neves, A.J.; Lau, N. Using a Depth Camera for Indoor Robot Localization and Navigation. *Ind. Organ. India* **2011**, *116*, 823–831.
3. Huai, J.; Zhang, Y.; Yilmaz, A. Real-time large scale 3D reconstruction by fusing Kinect and IMU data. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2015**, *II-3-W5*, 491–496. [[CrossRef](#)]
4. Um, D.; Ryu, D.; Kal, M. Multiple intensity differentiation for 3-D surface reconstruction with mono-vision infrared proximity array sensor. *IEEE Sens. J.* **2011**, *11*, 3352–3358. [[CrossRef](#)]
5. Pan, S.W.; Shi, L.W.; Guo, S.X. A Kinect-Based Real-Time Compressive Tracking Prototype System for Amphibious Spherical Robots. *Sensors* **2015**, *15*, 8232–8252. [[CrossRef](#)] [[PubMed](#)]
6. Noel, R.R.; Salekin, A.; Islam, R.; Rahaman, S.; Hasan, R.; Ferdous, H.S. A natural user interface classroom based on Kinect. *IEEE Learn. Technol.* **2011**, *13*, 59–61.
7. Zhou, H.K. A Study of Movement Attitude Capture System Based on Kinect. *Rev. Fac. Ing.* **2017**, *32*, 210–215.
8. Stowers, J.; Hayes, M.; Bainbridge-Smith, A. Altitude control of a quadrotor helicopter using depth map from Microsoft Kinect sensor. In Proceedings of the 2011 IEEE International Conference on Mechatronics (ICM), Istanbul, Turkey, 1 August 2011; pp. 358–362.
9. Zhang, Y.; Gao, J.C.; Xu, S.M. Research on Vision Location Algorithm Based on Kinect in Complex Condition. *Mach. Electron.* **2017**, *35*, 72–80.
10. Wang, Z.Y.; He, B.W. Study of Self-localization of Indoor Robot Based on Kinect Sensor. *Mach. Build. Autom.* **2014**, *5*, 154–157.
11. Clark, R.A.; Bower, K.J.; Mentiplay, B.F.; Paterson, K.; Pua, Y.-H. Concurrent validity of the Microsoft Kinect for assessment of spatiotemporal gait variables. *J. Biomech.* **2013**, *46*, 2722–2725. [[CrossRef](#)] [[PubMed](#)]
12. Amidi, O.; Kanade, T.; Fujita, K. A visual odometer for autonomous helicopter flight. *Robot. Autonom. Syst.* **1999**, *28*, 185–193. [[CrossRef](#)]
13. Feng, G.; Huang, X. Observability analysis of navigation system using point-based visual and inertial sensors. *Opt. Int. J. Light Electron Opt.* **2014**, *125*, 1346–1353. [[CrossRef](#)]
14. Zheng, N.-S.; Yang, H.-C.; Zhang, S.-B. An Absolute Orientation Method Suitable for Digital Close-Range Image. *Surv. Mapp.* **2008**, *33*, 111–112.
15. Lowe, D. Distinctive Image Features from Scale-Invariant Keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110. [[CrossRef](#)]
16. Raguram, R.; Frahm, J.M.; Pollefeys, M. A comparative analysis of RANSAC techniques leading to adaptive real-time random sample consensus. In Proceedings of the Computer Vision—ECCV 2008, European Conference on Computer Vision (DBLP), Marseille, France, 12–18 October 2008; pp. 500–513.
17. Bortz, J.E. A new mathematical formulation for strapdown inertial navigation. *IEEE Trans. Aerosp. Electron. Syst.* **2007**, *AES-7*, 61–66. [[CrossRef](#)]
18. Miller, R.B. A new strapdown attitude algorithm. *J. Guid. Control Dyn.* **2012**, *6*, 287–291. [[CrossRef](#)]
19. Liu, J.-Y.; Zeng, Q.-H.; Zhao, W. *Navigation System Theory and Application*; Northwestern Polytechnical University Press: Xi'an, China, 2010.
20. Lee, J.G.; Mark, J.G.; Tzartas, D.A.; Yong, J.Y. Extension of strapdown attitude algorithm for high-frequency base motion. *J. Guid. Control Dyn.* **2012**, *13*, 738–743. [[CrossRef](#)]

21. Wang, C.; Wang, T.; Liang, J.; Chen, Y.; Wu, Y. Monocular vision and IMU based navigation for a small unmanned helicopter. In Proceedings of the 2012 7th IEEE Conference on Industrial Electronics and Applications (ICIEA), Singapore, 18–20 July 2012; pp. 1694–1699.
22. Sirtkaya, S.; Seymen, B.; Alatan, A.A. Loosely coupled Kalman filtering for fusion of Visual Odometry and inertial navigation. In Proceedings of the 2013 16th International Conference on Information Fusion (FUSION), Istanbul, Turkey, 9–12 July 2013; pp. 219–226.



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).