

Article

An Adaptive Pose Fusion Method for Indoor Map Construction

Jinming Zhang ^{1,*} , Lianrui Xu ² and Cuizhu Bao ¹

¹ College of Computer and Information Engineering, Zhejiang Gongshang University, Hangzhou 310081, China; baocuizhu@zjgsu.edu.cn

² Institute of Geospatial Information, Information Engineering University, Zhengzhou 450052, China; mapviewer@163.com

* Correspondence: zhangjm@zjgsu.edu.cn

Abstract: The vision-based robot pose estimation and mapping system has the disadvantage of low pose estimation accuracy and poor local detail mapping effects, while the modeling environment has poor features, high dynamics, weak light, and multiple shadows, among others. To address these issues, we propose an adaptive pose fusion (APF) method to fuse the robot's pose and use the optimized pose to construct an indoor map. Firstly, the proposed method calculates the robot's pose by the camera and inertial measurement unit (IMU), respectively. Then, the pose fusion method is adaptively selected according to the motion state of the robot. When the robot is in a static state, the proposed method directly uses the extended Kalman filter (EKF) method to fuse camera and IMU data. When the robot is in a motive state, the weighted coefficient is determined according to the matching success rate of the feature points, and the weighted pose fusion (WPF) method is used to fuse camera and IMU data. According to the different states, a series of new poses of the robot are obtained. Secondly, the fusion optimized pose is used to correct the distance and azimuth angle of the laser points obtained by LiDAR, and a Gauss–Newton iterative matching process is used to match the corresponding laser points to construct an indoor map. Finally, a pose fusion experiment is designed, and the EuRoc data and the measured data are used to verify the effectiveness of this method. The experimental results confirm that this method provides higher pose estimation accuracy compared with the robust visual inertial odometry (ROVIO) and visual-inertial ORB-SLAM (VI ORB-SLAM) algorithms. Compared with the Cartographer algorithm, this method provides higher two-dimensional map modeling accuracy and modeling performance.

Keywords: pose estimation; simultaneous localization and mapping (SLAM); adaptive pose fusion; extended Kalman filter; weighted pose fusion; map construction



Citation: Zhang, J.; Xu, L.; Bao, C. An Adaptive Pose Fusion Method for Indoor Map Construction. *ISPRS Int. J. Geo-Inf.* **2021**, *10*, 800. <https://doi.org/10.3390/ijgi10120800>

Academic Editor: Wolfgang Kainz

Received: 15 October 2021

Accepted: 28 November 2021

Published: 30 November 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 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 (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

An accurate and efficient pose estimation system is an essential enabler for motion control and path planning in autonomous robots. In the case of indoor environments, SLAM technology is one of the effective means of solving robotic environmental cognition and positioning navigation [1]. SLAM refers to the process of a robot equipped with a specific sensor to establish an environmental model in motion and estimate its motion at the same time [2].

Depending on the available sensors, SLAM can be divided into visual SLAM, laser SLAM [3], and multi-sensor fusion SLAM [4]. The visual SLAM is based on visual sensors and includes sensor information acquisition, visual odometry, backend optimization, loopback detection, and map construction [5]. With the development of computer vision, image processing, and artificial intelligence, the accuracy of visual SLAM has been improved. Nevertheless, applying the visual SLAM to scenes with poor features, high dynamics, and/or weak light results in low performance [5]. Additionally, the IMU sensor can provide complementarity information to improve the performance of visual SLAM. For example, when the vision sensor is moving at a high speed, the success rate of pose

estimation based on feature matching is low, while IMU can still maintain more accurate pose estimation based on its acceleration sensor and gyroscope [6]. The visual sensor can also effectively reduce the accumulated drift error generated by IMU during its movement.

The fusion of the image data acquired by the camera and the velocity and position data acquired by IMU can also improve the robustness of the pose estimation algorithms. This method of combining camera and IMU is called visual-inertial SLAM (VI-SLAM) [7]. Although the pose estimation based on the fusion of two sensor datasets can effectively improve the estimation accuracy, there are still some problems to be addressed. When we use loose couplings as the method of integration posture, the key part still independently uses a visual algorithm for robot pose estimation [8]. The IMU measurement data are then added to the only independent pose estimation process. This, however, results in the internal state correlation between camera and IMU being ignored, while the effective constraints and compensation for the pose estimation are missed. There may even be a failure to estimate the pose error between camera and IMU. Therefore, the loose-coupling method cannot obtain the optimal accuracy of pose estimation.

As an alternative, the method of tight coupling is proposed, which combines visual algorithm and IMU data to jointly construct the equation of motion and the equation of state and further obtain the pose of the robot [9]. However, the tight coupling method using global optimization or local smoothing over-relies on the initial pose estimation. This leads to high complexity and a lack of real-time performance. At the same time, the monocular camera cannot directly obtain the scale information, hence this method should solely rely on the scale information calculated by IMU. Therefore, the tight coupling method cannot effectively enhance the estimation accuracy of the pose scale. Finally, the Markov property of the filtering based on the VI-SLAM method should be taken into account. In other words, the state relationship at a certain moment and the state relationship at all the previous moments cannot be determined. Therefore, this algorithm results in a limited accuracy improvement. To address the above issues, in this paper, in combination with the complementarity of the vision and inertial sensor, we investigate how to adaptively use the EKF method and WPF method to further improve the fusion pose accuracy, that is, an adaptive pose fusion method. We also examine the use of fusion pose to optimize the accuracy and efficiency of map construction by LiDAR. Firstly, the camera and IMU are used to calculate the position and posture of the robot. Secondly, either the EKF method or the WPF method is adaptively selected according to the motion state to form new pose estimation. Thirdly, the high-precision pose after fusion optimization is matched with the laser points measured at the same time to correct their distance and azimuth information, and Gauss–Newton iterative matching process is used to match the corresponding laser points to construct an indoor map.

The main contributions of this work lie in the fact that it proposes an adaptive method of pose fusion to construct the indoor map. The proposed adaptive method is manifested in two aspects. Firstly, by judging the motion state of the robot, either the EKF or the WPF method is chosen to fuse data acquired by the camera and IMU to form a new estimated pose. In a case where the robot is in a static state, the pose fusion is directly performed by EKF method. Similarly, for a robot in motion, the WPF method is used to fuse the posture fusion. Secondly, in the process of using the WPF method to fuse the pose, the weighted coefficients between the camera pose and IMU pose are determined according to the ratio of the feature points successfully matched by two keyframes to the total feature points. A lower matching success rate indicates a relatively low accuracy of the pose calculated by the camera. Hence, the proportion of the pose calculated by IMU should be increased accordingly. For a high matching success rate, the accuracy of the pose calculated by the camera is relatively high, and the proportion of the pose calculated by the camera should be increased accordingly. Based on this straightforward idea, we choose different weighted coefficients based on the matching success rate of 0.25 and 0.6.

The rest of the paper is organized as follows. In Section 2, a review of previous work is presented. An adaptive pose fusion method for indoor map construction is then presented

in Section 3 using an IMU and a camera. The experimental setup is given in Section 4 and an analysis of the results is given in Section 5. Finally, Section 6 concludes the work and gives future directions.

2. Related Work

Within the field of computer science, Davison proposed one of the first real-time monocular localization and mapping frameworks [10]. The author made use of an EKF method where he estimates the absolute position of landmarks. However, a key issue is improving the consistency of the estimation method that is affected by its inherent nonlinearity [9]. One approach is to use visual odometry for the tracked landmarks and thereby significantly reduce the effect of nonlinearities [11,12].

While most standard visual odometry approaches are based on detected and tracked point landmarks as the source of visual information, Jin et al. propose to model the environment as a collection of planar patches and to derive a corresponding photometric error between camera frames [13]. Molton et al. also track locally image patches in a filter-based SLAM framework [14]. By employing gradient-based image alignment, they also co-estimate surface normals but keep data association separated from the subsequent EKF-based information fusion. Silveira et al. also use planar regions and minimize the photometric error concerning a reference frame in order to estimate the relative motion as well as other parameters such as illumination parameters and patch normality [15]. They subsequently merge the output in an EKF.

Furthermore, fusing other sensor information can construct a more robust visual SLAM system, such as VI-SLAM.

Incorporating inertial measurements in the estimation can significantly improve the robustness of the system, provide the estimation process with the notion of gravity, and allow for a more accurate and high bandwidth estimation of the velocities and rotational rates. By adapting the original EKF proposed by Davison [10], additional IMU measurements can be relatively simply integrated into the estimation, whereby calibration parameters can be co-estimated online [16,17]. Weiss and Siegwart designed the Single-Sensor Fusion (SSF) algorithm which takes an EKF as the basic framework [18,19]. The pose calculated by IMU is also considered as the prediction function, and the pose is calculated by Parallel Tracking and Mapping (PTAM) as the measurement data. These effectively improve the performance of an algorithm. Leutenegger et al. describe a tightly coupled approach in which the robot trajectory and sparse landmarks are estimated in a joint optimization problem using inertial error terms as well as the reprojection error of the tracked landmarks in the camera images [20]. This is carried out in a windowed bundle adjustment approach over a set of keyframe images and a temporal inertial measurement window. Similarly, Mourikis and Roumeliotis estimate the trajectory in an IMU-driven filtering framework using the reprojection error of landmarks as measurement updates [21]. Since inertial measurements are often obtained at a higher rate than image data, methods for combining multiple inertial measurements are desirable to reduce computational costs. Forster et al. presented a concise IMU measurements pre-integration method such that they can be efficiently included in a factor graph framework [22]. Usenko et al. extended their previous work on semi-dense visual odometry to integrate inertial measurements [23]. They minimize a joint energy term composed of visual and inertial error terms to estimate the motion of their sensor.

In addition, the motion prediction of IMU still provides accurate estimation in a short time when the visual pose calculation fails. Li and Mourikis also used an improved nonlinear method to optimize the monocular inertial navigation SLAM scheme which is based on the Monocular Visual-Inertial (VINS-Mono) system [24]. In the initialization process, the initialization of the acceleration Bayesian deviation is added, and the Oriented Fast and Rotated Brief (ORB) feature points are used to detect and match the image at the front end for reference to ORB-SLAM. Using this technique improves the robustness of the scheme and also enables high precision real-time positioning and sparse map construction.

Kelly and Sukhatme further proposed visual-inertial navigation based on SLAM called VI ORB-SLAM [17]. This method supports multiple types of cameras such as monocular, binocular, and RGB-D of KINECT. The VI ORB-SLAM system designed a hybrid map data association mechanism based on two kinds of sensors, which can carry out data association for short, medium, and long-term sensor's collected data. Readers should note that the VI ORB-SLAM algorithm based on features tightly coupled with the VIO system relies on the maximum a posteriori estimation to detect the geometric consistency of the keyframes, and further improves the accuracy of map positioning.

3. Methodology

To improve the accuracy of indoor map construction, the high-precision data obtained from various sensors such as cameras and IMU can be combined. In this paper, by judging the motion state of the robot and according to the dependence of the feature point matching, the EKF method and WPF method are, respectively, used to optimize the pose of the camera and IMU. This enables high-precision modeling of the indoor map.

The overall structure of the adaptive method of pose fusion to construct an indoor map based on camera and IMU is shown in Figure 1. This method includes the following four main steps.

- (1) Pose solution. Camera and IMU are used to obtain the pose of the robot;
- (2) Pose fusion with EKF. In cases where the robot is static, EKF method is used to fuse the pose of the camera and IMU;
- (3) Pose fusion with WPF. In cases where the robot is moving, WPF method is used to fuse the pose of the camera and IMU.
- (4) Map construction based on LiDAR.

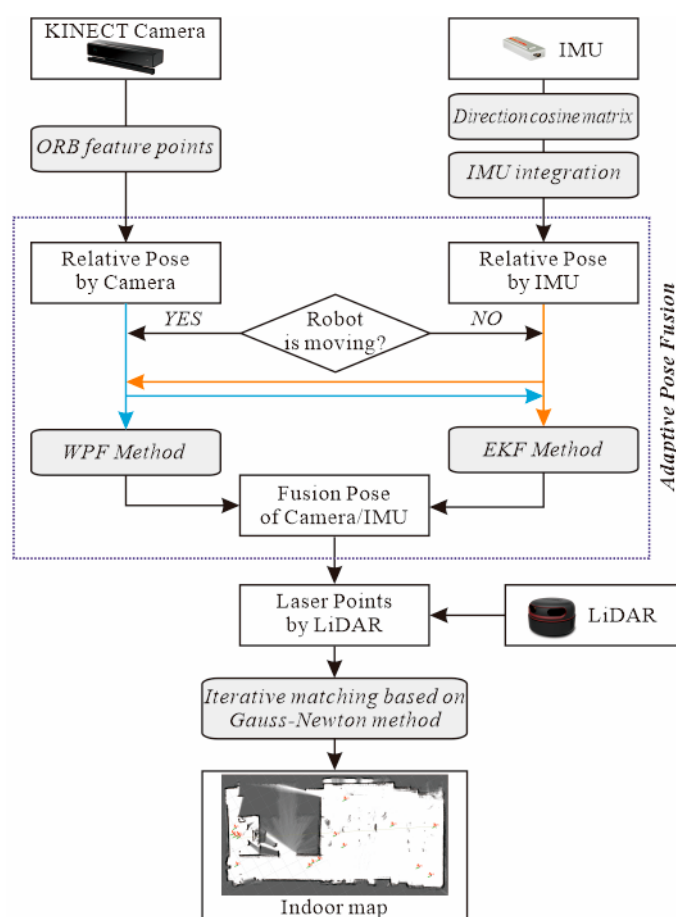


Figure 1. The architecture of the adaptive method of pose fusion for constructing an indoor map.

3.1. Camera Pose Calculation

The robot's position and posture can be calculated based on the image data obtained by the camera which is usually achieved by matching the homonymous feature points of the two partially overlapping images. Firstly, the image feature points are extracted using the Oriented-Fast algorithm [25]. Then, the Fast Library for Approximate Nearest Neighbors (FLANN) method is used to achieve the matching between feature points [26]. According to the corresponding relation of the homonymic feature points in the two images, the camera motion parameters are then calculated.

Assume that the rigid body transformation of the camera motion is (R, T) , where R is the rotation vector and T is the translation vector. The relationship between the coordinates corresponding to the feature points and the moving process is [27]:

$$P_{pm} = RP_{lm} + T \quad (1)$$

where P_{pm} and P_{lm} are the three-dimensional coordinates of the m th feature point of the current frame image and the previous frame image, and $m = 1, 2, 3 \dots n$. Using the FLANN method to match the feature points, we also need to eliminate the feature points with false matching. Here, they are randomly selected from at least four of the feature points. The rigid body transformation quantity (R, T) is then used in $\|P_{pm} - (RP_{lm} + T)\|^2$ and checked to see whether it is less than the threshold. Finally, the least square value of the relationship between the corresponding coordinates of the feature points and the motion process is obtained through iterative calculations. This is then used to estimate the current pose of the robot.

3.2. IMU Pose Calculation

IMU includes a three-axis accelerometer and a three-axis gyroscope. The velocity and position data of IMU can be obtained by quadratic integration of the linear velocity measured by the accelerometer [28]. The posture data of IMU can be obtained by integrating the angular velocity measured by the gyroscope. For a pitch angle of 90° , the differential equation of Euler Angle is singular, and the full posture solution of IMU is unobtainable [29]. The Direction Cosine Matrix (DCM) algorithm uses the direction cosine of the vector to represent the attitude matrix [30]. This effectively avoids the singularity problem when using the Euler Angle to represent the orientation. Therefore, the DCM algorithm is often used to solve IMU posture. The DCM matrix is defined as:

$$C = \begin{bmatrix} i \cdot i' & i \cdot j' & i \cdot k' \\ j \cdot i' & j \cdot j' & j \cdot k' \\ k \cdot i' & k \cdot j' & k \cdot k' \end{bmatrix} \quad (2)$$

where, i', j', k' are unit vectors of the robot coordinates. The derivatives of i', j', k' are:

$$\begin{cases} \frac{di'}{dt} = j'w_z - k'w_y \\ \frac{dj'}{dt} = k'w_x - i'w_z \\ \frac{dk'}{dt} = i'w_y - j'w_x \end{cases} \quad (3)$$

and, $\dot{C}$ is obtained by taking the derivative of matrix C as the following:

$$\dot{C} = C \begin{bmatrix} 0 & -w_z & w_y \\ w_z & 0 & -w_x \\ -w_y & w_x & 0 \end{bmatrix} = C\Omega \quad (4)$$

Equation (4) is then solved to obtain the relative position and the posture transformation relationship between IMU and robot.

3.3. IMU Pre-Integration

The high measurement frequency of IMU requires computing resources. To avoid adding a new state quantity during each IMU measurement, a re-parameterization process is usually added between two frames to realize the motion constraints and avoid repeated integrations. This re-parameterization process is called IMU pre-integration [31]. Assuming that the acceleration and angular velocity measured by IMU are, respectively, expressed as ${}_B\tilde{\omega}_{WB}(t)$ and ${}_B\tilde{\alpha}_{WB}(t)$, the IMU measurement model is:

$$\begin{cases} {}_B\tilde{\omega}_{WB}(t) = {}_B\omega_{WB}(t) + b_g(t) + \eta_g(t) \\ {}_B\tilde{\alpha}_{WB}(t) = R_{WB}^T(t)({}_W\alpha(t) - g(w)) + b_a(t) + \eta_a(t) \end{cases} \quad (5)$$

where W is the world coordinate system, B is the IMU coordinate system, ${}_B\omega_{WB}(t)$ is the instantaneous angular velocity of B relative to W , $R_{WB}^T(t)$ is the rotation matrix from the world coordinate system to the IMU coordinate system, ${}_W\alpha(t)$ is the instantaneous acceleration in the world coordinate system, $b_g(t)$ and $b_a(t)$ are the deviation of the gyroscope from the acceleration, and $\eta_g(t)$ and $\eta_a(t)$ are the random noise. The kinetic integral model is also introduced which contains the rotations R_{WB} and the velocities v_{WB} and the translations p_{WB} as the following:

$$\begin{cases} \dot{R}_{WB} = R_{WB}\widehat{\omega_{WB}} \\ \dot{v}_{WB} = \alpha_{WB} \\ \dot{p}_{WB} = v_{WB} \end{cases} \quad (6)$$

Integrating Formula (6), we obtain the rotation $R_{WB}(t + \Delta t)$, the velocity ${}_Wv(t + \Delta t)$ and the translation ${}_Wp(t + \Delta t)$ within Δt . We then combine the IMU measurement model of the acceleration ${}_B\tilde{\omega}_{WB}(t)$, and angular velocity ${}_B\tilde{\alpha}_{WB}(t)$, to obtain the movement state $R_{WB}(t + \Delta t)$, ${}_Wv(t + \Delta t)$ and ${}_Wp(t + \Delta t)$ of the IMU relative to the world coordinate system within Δt . This is the relative relationship between IMU data at these two moments. The update rates of IMU and camera are not synchronized, so the integration of multi-frame IMU data between two keyframes can effectively constrain them. Considering that the rotation matrix R_{WB}^k is constantly changing with time, the relative motion increments ΔR_{ij} , Δv_{ij} , Δp_{ij} are:

$$\begin{cases} \Delta R_{ij} = R_{BW}^i R_{WB}^j = \prod_{k=i}^{j-1} \text{Exp}\left({}_B\tilde{\omega}_{WB}^k - b_g^k - \eta_{gd}^k\right)\Delta t \\ \Delta v_{ij} = \sum_{k=i}^{j-1} \Delta R_{ik} \left(\tilde{\alpha}^k - b_a^k - \eta_{ad}^k\right)\Delta t \\ \Delta p_{ij} = \sum_{k=i}^{j-1} \left[\Delta v_{ik}\Delta t + \frac{1}{2}\Delta R_{ik} \left(\tilde{\alpha}^k - b_a^k - \eta_{ad}^k\right)\Delta t^2\right] \end{cases} \quad (7)$$

where $\Delta R_{ik} = R_{WB}^i R_{WB}^k$, and $\Delta v_{ik} = R_{WB}^i \left({}_Wv_B^j - {}_Wv_B^i - g_W\Delta t_{ij}\right)$. The left side of Equation (7) is the relative motion increments and the right side is the IMU data. Furthermore, the relative motion increments ΔR_{ij} , Δv_{ij} , Δp_{ij} of two keyframes are [32]:

$$\begin{cases} R_{WB}^j = R_{WB}^i \Delta R_{ij} \text{Exp}\left(J_{\Delta R}^g b_g^i\right) \\ v_{WB}^j = v_{WB}^i + g_W\Delta t_{ij} + R_{WB}^i \left(\Delta v_{ij} + J_{\Delta v}^g b_g^i + J_{\Delta R}^a b_a^i\right) \\ p_{WB}^j = p_{WB}^i + v_{WB}^i \Delta t_{ij} + \frac{1}{2}g_W\Delta t_{ij}^2 + R_{WB}^i \left(\Delta p_{ij} + J_{\Delta p}^g b_g^i + J_{\Delta p}^a b_a^i\right)\Delta t^2 \end{cases} \quad (8)$$

As shown, the relative motion increments ΔR_{ij} , Δv_{ij} , Δp_{ij} between two keyframes are obtained. In other words, the re-parameterized results of IMU pre-integration between two keyframes prevent the repeated integration of IMU observations and improve the reliability of the algorithm.

3.4. Extended Kalman Filter Method to Fuse Camera/IMU Pose

To improve the accuracy of the estimated pose of the robot, the EKF method is used to fuse camera poses and IMU poses. EKF solves nonlinear problems using the first-order Taylor series and updates the state and prediction equations by incorporating a variety of sensor data with a high level of controllability. EKF is divided into the prediction process and the update process [33]. For the prediction process, the error state vector $\tilde{x}$ is defined as:

$$\begin{cases} \tilde{x} = x - \hat{x} \\ \tilde{x} = [\delta p_W^i, \theta p_W^i, v p_W^i, \delta b_a, \delta b_g]^T \\ \hat{x} = [\hat{p}_W^i, \hat{q}_W^i, \hat{v}_W^i, \hat{b}_a, \hat{b}_g]^T \end{cases} \quad (9)$$

where $\hat{x}$ is the expected value of the state vector. Readers should note that $[n_a, n_{b_a}, n_g, n_{b_g}]$ is used to represent the state noise n . According to the definition of the error state vector $\tilde{x}$, the acceleration and angular velocity measured by IMU are linearized. The expressions in the error equations for translation, rotation, velocity, the accelerometer, and the gyroscope's zero deviation are also given as $\delta p_W^i, \theta p_W^i, \delta b_a, \delta b_g$, respectively, about the experimental equations of state. The state transitions φ are also given by the Jacobian matrix as:

$$\varphi = \begin{bmatrix} \frac{\partial f(x_{k-1})}{\partial q} & \frac{\partial f(x_{k-1})}{\partial \omega_b} \\ 0_{3 \times 3} & I_{3 \times 3} \end{bmatrix} \quad (10)$$

Considering the system noise, matrix G and the state transition matrix φ , the system noise covariance matrix Q_d is then obtained. The calculation error of EKF state prediction $X_{k+1|k}$ and the system state covariance matrix forecast $P_{k+1|k}$ are also obtained as the following [34]:

$$\begin{cases} X_{k+1|k} = \varphi X_{k|k} \\ P_{k+1|k} = \varphi P_{k|k} \varphi + Q_d \end{cases} \quad (11)$$

where $X_{k|k}, P_{k|k}$ are the best estimates obtained in the previous time instant. IMU errors continue to accumulate over time. Therefore, it is necessary to use the low-frequency output camera pose as the observed value z of the filter to correct the results of IMU calculation. In other words, the pose conversion matrix of the camera and IMU relative to the robot chassis is determined. This is to obtain the rotation and translation matrices p_c^i , and q_c^i and the system observation equations z_p, z_q between camera and IMU as follows [34]:

$$\begin{cases} z_p = p_W^c = p_W^i + C_W^i p_c^i \\ z_q = q_W^c = q_W^i \otimes q_c^i \end{cases} \quad (12)$$

The update process of EKF is deduced from the observation matrix and the observation noise covariance matrix. The gain matrix K of the system is then obtained as:

$$K = P_{k+1|k} H^T (H P_{k+1|k} H^T + R)^{-1} \quad (13)$$

After obtaining the gain matrix K , the system state variable $X_{k+1|k+1}$ is updated as:

$$X_{k+1|k+1} = X_{k+1|k} + K \tilde{z} \quad (14)$$

and the system covariance matrix $P_{k+1|k+1}$ is also updated as:

$$P_{k+1|k+1} = (I_{15} - KH) P_{k+1|k} (I_{15} - KH)^T + KRK^T \quad (15)$$

After updating the system state quantity $X_{k+1|k+1}$ and the system covariance matrix $P_{k+1|k+1}$, the quaternion calculated by the system state quantity is normalized. The quaternion is then converted into Euler angles for easy pose display and represents the pose output after the application of EKF fusion sensor data.

3.5. Adaptive Weighted Fusion Method to Fuse Camera/IMU Pose

The pose fusion mode depends on the motion state of the robot. The motion state of the robot is either moving or stationary. In this paper, the IMU and odometer data are used to judge the motion and stationary condition of the robot. If the line acceleration of IMU is 0 and the odometer data are not increased, then the robot is in a static state. The pose fusion of camera and IMU is directly fused using the EKF method. Similarly, if the line acceleration of IMU is not 0, or the odometer data are increased, the robot is in motion. The pose fusion of camera and IMU are fused by the WPF method.

For a robot in the moving state, there is great uncertainty regarding the texture features captured by the camera, while the IMU error is relatively stable, although it is accumulated. Therefore, in the process of pose fusion, the weighted coefficient between the camera and IMU needs to be properly considered. The matching success rate of the keyframes feature points can be also analyzed. This is to determine the degree of dependence of camera and IMU pose on the matching of feature points. Different weighted coefficients can be accordingly adopted for different pose states to process the pose fusion of camera and IMU.

For a robot in motion, IMU measures the linear velocity, acceleration, and angular velocity. The camera also provides the corresponding image based on keyframes. Here, the weighted coefficient is set according to the matching success rate of feature points between two keyframes obtained by the camera. If the matching success rate is low, the correlation between the current pose and the camera pose is low. Hence, the proportion of IMU pose should be increased. Similarly, if the matching success rate is high, the correlation between the current pose and the camera pose is high. Hence, the proportion of camera pose should be increased.

Here, we define the matching success rate μ as the ratio of feature points successfully matched by two keyframes to the total feature points. The matching success rate is then used to determine the weighted coefficient of pose fusion between camera and IMU. This process is referred to as obtaining the weighted coefficient of pose fusion.

3.6. LiDAR Mapping

During the construction of indoor maps, unstructured scenes may lead to the tilt of the 2D LiDAR composition plane and may reduce the quality of construction. In this paper, the laser triangulation rangefinder technology is used for LiDAR mapping [35]. After calculating the distance and azimuth angle, the measured laser points and LiDAR obtain the relative coordinates. Before the mapping, the estimation pose after fusion and optimization is also replaced by the estimation pose in a mapping algorithm. This is used as a new data source to calculate the relative relationship between the laser points and LiDAR. Based on this, the distance and azimuth information of the laser points are updated. The iterative matching process based on Gaussian–Newton method is then used to match the coordinates of the laser points and a high-precision indoor map is established [36].

The indoor environment mostly presents vertical structures, so the oblique radar data needs to be approximately projected. Laser data obtained by scanning (ρ, θ) are expressed as (P_x, P_y, P_z) in the form of map coordinates:

$$\begin{cases} P_x = \rho \cdot \cos\theta \\ P_y = \rho \cdot \sin\theta \\ P_z = 0 \end{cases} \quad (16)$$

where $(\theta_x, \theta_y, \theta_z)$ denotes roll angle, pitch angle, and yaw angle. The coordinates (X, Y, Z) of the scanning laser point are also obtained according to the initial pose and angle transformation as the following:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = R_z \cdot R_y \cdot R_x \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} \quad (17)$$

After corresponding the optimized pose estimation and the relative coordinates of the laser points, the Gaussian–Newton method is used to match the laser points and improve the accuracy of map construction. To do this, the minimum value of the rigid body transformation T is obtained as T^* ,

$$T^* = \operatorname{argmin}_T \sum_{i=1}^n [1 - G(S_i(T))]^2 \quad (18)$$

where, $S_i(T)$ is the plane coordinate of the laser point, and $G(S_i(T))$ is the occupation value of $S_i(T)$. If all the laser points are matched completely, $G(S_i(T))$ is 1. The measurement error of the radar data within a given period is then optimized as shown in Formula (19).

$$2 \sum_{i=1}^n \left[1 - G(S_i(T)) - \nabla G(S_i(T)) \frac{\partial S_i(T)}{\partial T} \Delta T \right] \cdot \left[\nabla G(S_i(T)) \frac{\partial S_i(T)}{\partial T} \right]^T \rightarrow 0 \quad (19)$$

Through Taylor expansion and partial derivations, the solution of ΔT is transformed into the following Gaussian–Newton minimization problem:

$$\Delta \zeta = H^{-1} \sum_{i=1}^n \left[\nabla G(S_i(T)) \frac{\partial S_i(T)}{\partial T} \Delta T \right]^T \cdot [1 - G(S_i(T))] \quad (20)$$

where $H = \left[\nabla G(S_i(T)) \frac{\partial S_i(T)}{\partial T} \Delta T \right]^T \left[\nabla G(S_i(T)) \frac{\partial S_i(T)}{\partial T} \right]$. Obtaining the least square solution based on the Gaussian–Newton method, the results of the beam point matching are also determined according to this method. The distance and azimuth angle of laser points corresponding to the optimized pose is then substituted into the mapping process as data sources to construct the two-dimensional indoor map. This effectively removes the distortion caused by the LiDAR movement, hence the mapping efficiency.

4. Adaptive Pose Fusion Experiment

4.1. Experimental Platform

Autolabor is used as the experimental platform for the adaptive pose fusion experiment. This robot is equipped with a KINECT V2 camera (its equipment parameters are shown in Table 1), an AH-100B inertial measurement unit (its equipment parameters are shown in Table 2), and a Rplidar A2 LiDAR (its equipment parameters are shown in Table 3). The three sensors are connected to Autolabor in a solid connection mode (Figure 2).

Table 1. Device parameters of Kinect V2 camera.

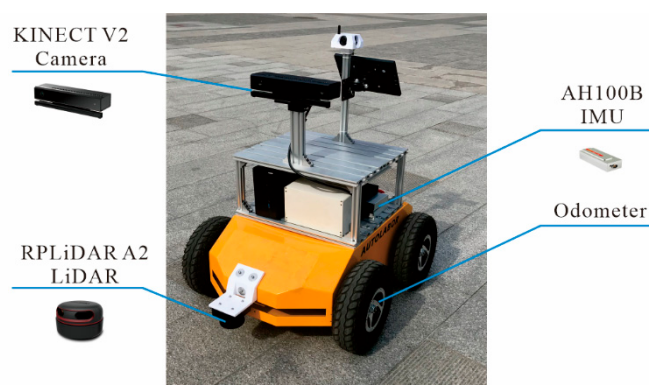
Item	Attributes
Working principle	Time of flight (TOF)
RGB stream	1920 × 1280/30 FPS
Depth stream	512 × 424/30 FPS
Depth distance	0.5 m–4.5 m
Horizontal/vertical viewing angle	70°/60°

Table 2. Device parameters of AH-100B inertial measurement unit.

Item	Gyroscope	Accelerometer	Magnetometer
Measuring range	±2000 °/s	±2 G	±4 gauss
Resolution	<0.1 °/s	<10 mg	<2.5 mgauss
Frequency	40 Hz	37 Hz	50 Hz

Table 3. Device parameters of Rplidar A2 LiDAR.

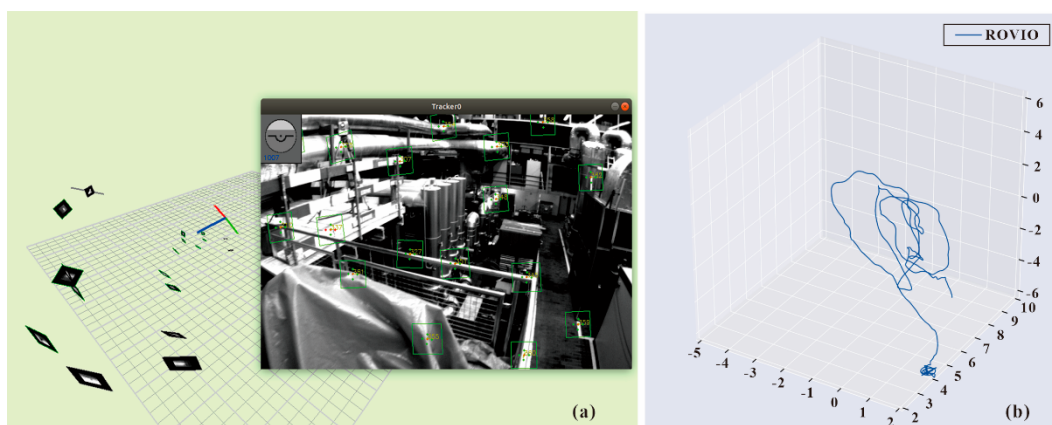
Item	Attributes
Measuring radius	0.2 m–16 m
Sampling frequency	8 K
Angle resolution	0.9 °
Ranging resolution	$\leq 1\%$ of actual distance (range ≤ 12 m) $\leq 2\%$ of actual distance (range ≈ 12 m–16 m)
Ranging accuracy	1% of actual distance (range ≤ 3 m) 2% of actual distance (range ≈ 3 m–5 m) 2.5% of actual distance (range ≤ 5 m–16 m)

**Figure 2.** The experimental platform: Autolabor.

4.2. Experimental Data

Here, we verify the validity of the adaptive pose fusion method, the accuracy of camera/IMU fusion pose, and the effect of optimized LiDAR map construction. To achieve this, we designed two groups of different experiments. The first group uses EuRoC public dataset as the data source, and the other group collects the environmental data from a laboratory as the data source.

The EuRoC public dataset was collected by the ETH Zurich based on the unmanned aerial vehicle in the auditorium and the empty room [37]. As shown in Figure 3, the sensors carried by the unmanned aerial vehicle mainly include a camera and an IMU, with the camera frequency being 30 Hz and the IMU frequency being 40 Hz. The experimental data are provided in four folders, and each folder contains the transformation of the sensor relative to the coordinate system. The EuRoC public dataset is mainly used for comparative analysis of the position and posture accuracy.

**Figure 3.** EuRoC public dataset (a) and trajectory (b).

the ratio of dependence of the feature points matching, and the optimized pose estimation is obtained.

Step 5: 2D map construction.

The updated pose correction RPLIDAR is then optimized to obtain the distance and azimuth information of the laser points, and the two-dimensional environment map is constructed by Gaussian–Newton method.

Step 6: Experimental analysis.

Based on the Euroc public dataset, we compare the accuracy of the ROVIO and APF algorithm in pose fusion optimization [39]. ROVIO is a fully robocentric and direct visual-inertial odometry framework which runs in real-time on computationally constrained platforms. To increase robustness and usability, it implements multicamera support (with or without overlapping field of view) and enables online calibration of camera-IMU extrinsics. It tightly couples the visual information and IMU output and fuses the data through iterative Kalman filtering to update the filtering state and pose. In terms of algorithm framework and data fusion method, the ROVIO algorithm is similar to the APF algorithm presented in this paper.

Based on the data collected in the laboratory scene, the accuracy of the VI ORB-SLAM and APF algorithm is also compared in pose fusion optimization. The VI ORB-SLAM algorithm is composed of ORB-SLAM and IMU module. Through maximum a posteriori estimation, the geometric consistency detection of keyframes is carried out to obtain the optimal pose estimation. Therefore, the VI ORB-SLAM algorithm has a high contrast with the APF algorithm.

The modeling accuracy and effectiveness of Cartographer algorithm [40] and APF algorithm are compared in terms of two-dimensional map construction. The Cartographer is a classical mapping algorithm based on LiDAR. It predicts the pose, based on the information of IMU and odometer by filtering and transforming the scanning points into raster coordinates using voxel filtering. The results show that the Cartographer algorithm is stable and highly robust.

5. Results

5.1. Weighted Coefficient in WPF

As described in Section 3, the indoor map construction method is based on the optimization of pose fusion, which mainly selects different pose fusion strategies according to the motion state of the robot. If the robot is in a static state, the position and posture of camera and IMU are fused using the EKF method. In cases where the robot is in a motive state, the position and posture of the camera and IMU are fused using the WPF method.

For the WPF method, the key is to determine the proportion of camera pose and IMU pose. In this paper, the weighted coefficient between the camera and IMU poses is determined according to the ratio of the feature points successfully matched by two keyframes to the total feature points, which is the weighted coefficient λ of the pose fusion. The matching success rate μ represents the significance of the environment features, hence it determines the accuracy of the pose solution.

We select 19 different scenes including representative positions such as aisles, corners, tables and chairs. Based on the ORB feature point detection, the fast approximate nearest neighbor method is used to match the feature points.

As shown in Figure 5, the three images on the left, respectively, represent the scene of the aisle, corner, desk, and chair in the laboratory environment. The colored dots in the image represent the ORB feature points detected in the different scenes. The three images on the right also represent the images of the aisles, corners, desk, and chair obtained from different angles. The color line represents the ORB feature point pairs which are successfully matched between two images. During the experiment, 533, 149, and 653 feature points are, respectively, detected in the aisle, corner, desk, and chair scenes. Among these features, 50 pairs of feature points are successfully matched in the aisle scene, four pairs

are successfully matched in the corner scenes, and 76 pairs are successfully matched in the desk and chair scenes.



Figure 5. Matching of feature points of aisles (a), corners (b), tables and chairs (c).

The matching success rates of the aisle, corner, desk, and chair scenes are 9.38%, 2.68%, and 11.63%, respectively. This indicates that the scenes with rich textures, obvious structures, and strong characteristics include more feature points and have a higher matching success rate. On the contrary, the scenes with weak texture and a single structure have fewer feature points detected hence a lower matching success rate. It is also seen that in these 19 different scenarios, the highest matching success rate is 43% and is never larger than 50%. Based on the results above, we added further 7 experimental scenes with artificial features that improve the matching success rate to 87.37%, as shown in Figure 6.

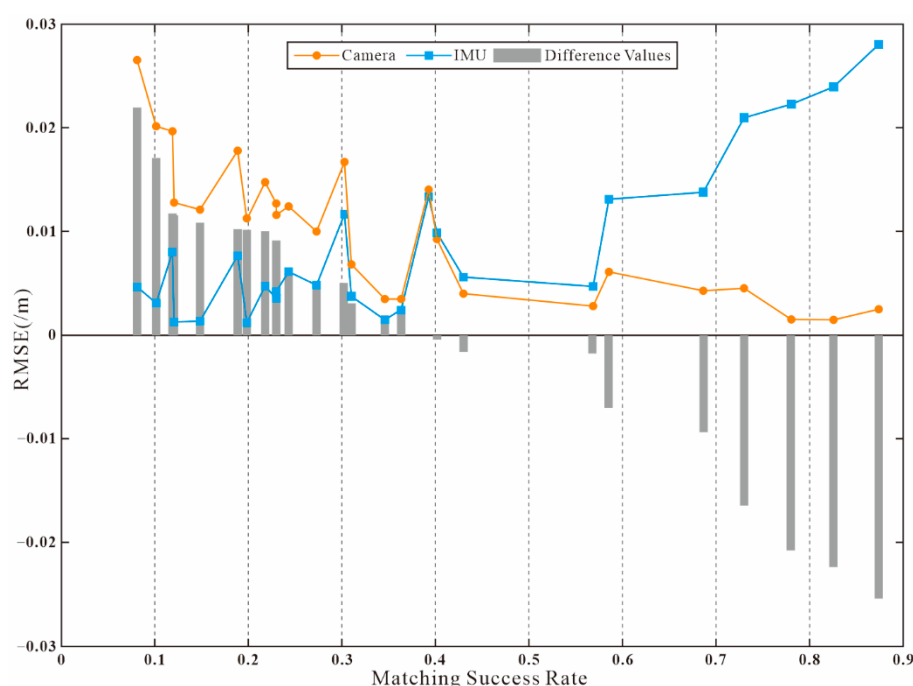


Figure 6. The relationship between the matching success rate and the pose error.

Secondly, the RSME between the pose determined by camera and IMU and the pose solved by EKF is presented in Figure 6. The experimental results show the following: (1) In cases where the matching success rate of the camera is lower, the RMSE of the pose between the camera and solved by EKF is higher. This indicates that the reliability of the pose calculation by the camera is lower. (2) In cases where the matching success rate of the camera is higher, the RMSE of the pose between the camera and solved by EKF is lower. This indicates that the reliability of pose calculation by the camera is higher. (3) Although the higher matching success rate is partly due to adding extra artificial feature points, this effect can be ignored at an appropriate time. If the matching success rate is lower than 50%, the pose accuracy calculated by the camera is gradually decreased, while the pose accuracy calculated by IMU is not significantly changed. This is consistent with the actual situation.

From the perspective of the RMSE deviation value and the matching success rate, the two are in a straight line relationship. Furthermore, if the relationship between the two is described by a cubic curve, the coefficient of determination can even reach about 0.9774, as shown in Figure 7. Taking the cubic curve as the benchmark, when the matching success rates are 0.2462 and 0.6362, respectively, the slope of the curve changes. Therefore, the matching success rate can be divided into three segments with 0.2462 and 0.6362 as nodes. Then, we study the ratio of RMSE between camera and IMU, and always divide the larger value by the smaller ones, and can obtain a graph as shown in Figure 8. The experimental results show that the matching success rate is approximately 0.25 and 0.6 as nodes, and it is also divided into three segments. In summary, we can divide the matching success rate into three intervals, that is $0 \leq \mu < 0.25$, $0.25 \leq \mu < 0.6$ and $0.6 \leq \mu \leq 1$. Finally, based on the experimental results and relaxing the standards, different weighted coefficients can be used for each segment. Through multiple experiments and comparative analysis, the weighted coefficient λ in the pose weighted fusion is set according to the following steps.

- (1) For $0.6 \leq \mu \leq 1$, the weighted coefficient λ of the pose calculated by the camera is 0.5 and the weighted coefficient λ of the pose calculated by IMU is 0.5.
- (2) For $0.25 \leq \mu < 0.6$, the weighted coefficient λ of the pose calculated by the camera is 0.33 and the weighted coefficient λ of the pose calculated by IMU is 0.67.
- (3) For $0 \leq \mu < 0.25$, the weighted coefficient λ of the pose calculated by the camera is 0.2 and the weighted coefficient λ of the pose calculated by IMU is 0.8.

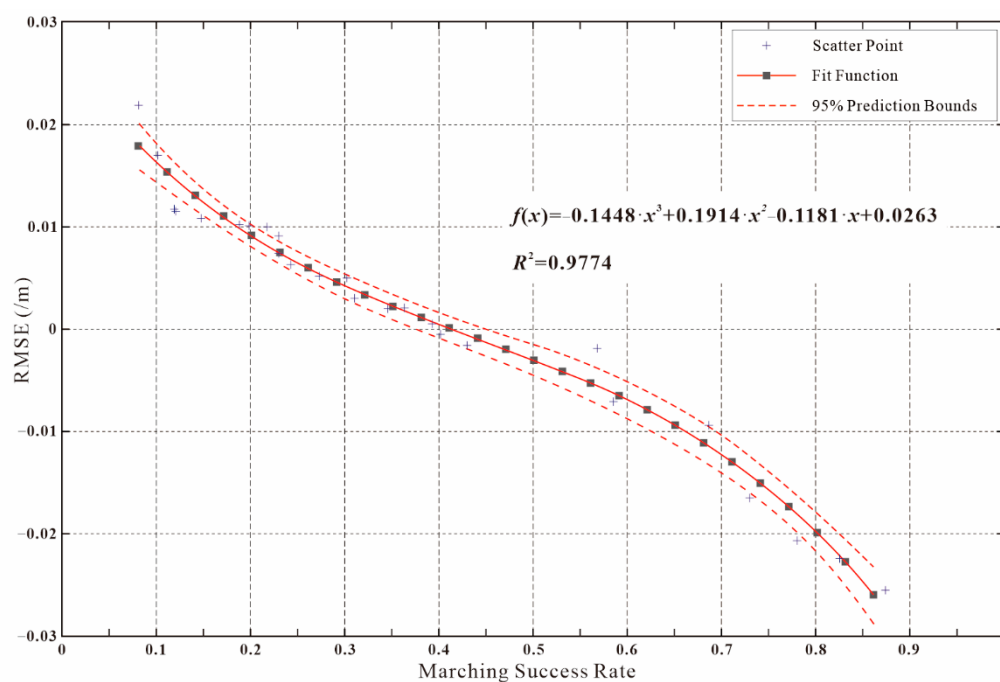


Figure 7. The relationship between the matching success rate and the pose error.

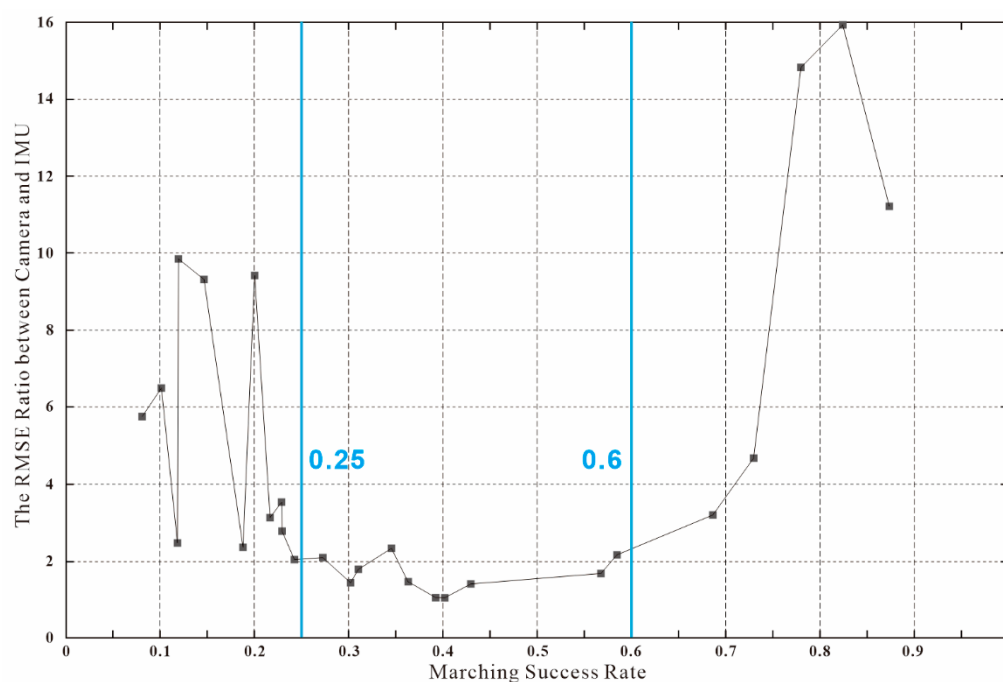


Figure 8. The RMSE ratio between the camera and IMU.

5.2. Optimized Pose Analysis

The first experiment is based on the Euroc public dataset where we compare the pose estimation accuracy of the ROVIO algorithm with the APF algorithm. The ROVIO algorithm tightly couples the visual information and the IMU output, and fuses data through iterative Kalman filtering to update the filtering state and pose. In terms of algorithm framework and data fusion method, the ROVIO algorithm is similar to the APF method. The total duration of this experiment is 149.96 s, and a total of 29,993 postures are selected. The experimental errors are shown in Table 1. The maximum error (Max), minimum error (Min), mean error (Mean), RMSE, and standard deviation (Std) are used as

the performance metrics. The offset diagrams of the robot along the x -axis direction, y -axis direction and z -axis direction are presented in Figure 9.

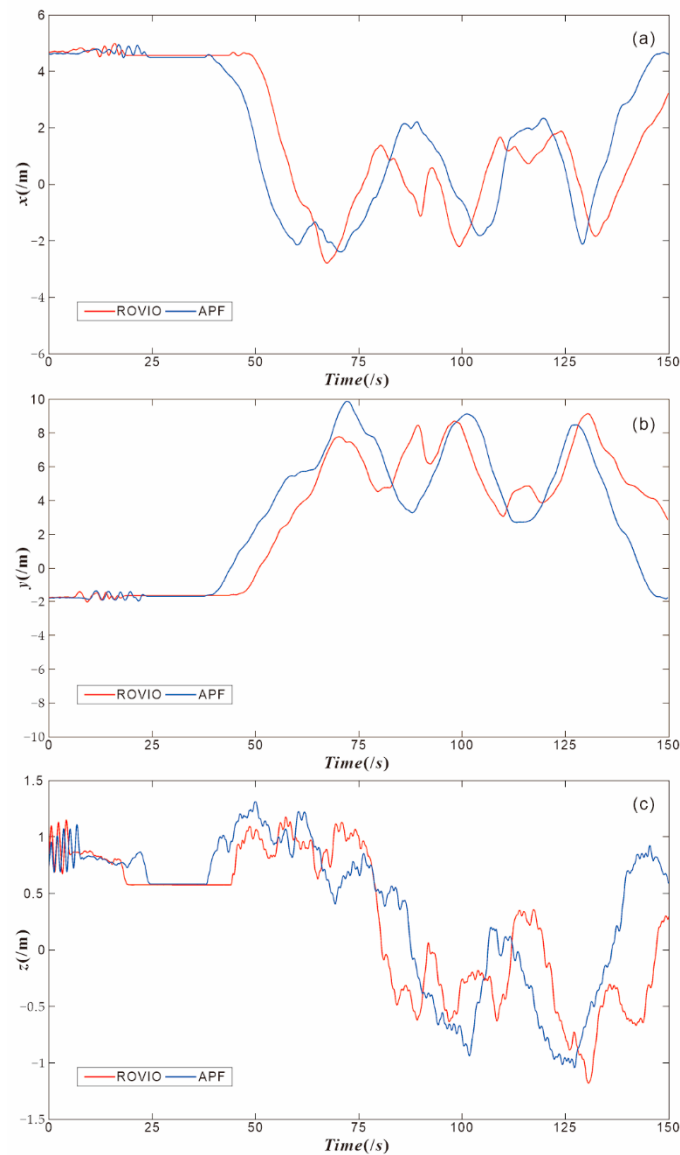


Figure 9. Displacement deviation along the x -axis (a), y -axis (b) and z -axis (c) direction.

Comparing the proposed algorithm with the ROVIO algorithm shows that the absolute RMSE is 2.5944 m, and the absolute standard deviation error is 1.5223 m (Table 4). We further examine the displacement deviation along the x -axis direction and y -axis and z -axis. It is seen that the proposed algorithm can effectively filter the redundant offset and vibration compared with the ROVIO algorithm and effectively improve the estimation accuracy of the robot's pose using the integrated optimization mechanism.

Table 4. Error analysis between the proposed algorithm and the ROVIO algorithm (pose: 29,993).

Algorithm	Max	Min	Mean	RMSE	Std
APF	6.0808	0.2562	2.1008	2.5944	1.5223
ROVIO	6.0945	0.2830	2.3133	2.7281	1.7636

The second experiment is based on the real-time laboratory scene dataset that was collected by the robot to compare and analyze the pose estimation accuracy of the VI ORB-SLAM algorithm with that of the APF algorithm. The total duration of the experiment is 367.30 s, and a total of 3648 poses are selected. The experimental errors are shown in Table 5. The Max, Min, Mean, RMSE, and Std are used as the performance metrics. The offset diagrams of the robot along the x -axis direction and y -axis direction are also shown in Figure 10 (Figure 10 shows the experimental results for a part of the total time).

Table 5. Error analysis of the proposed algorithm and VI ORB-SLAM algorithm.

Algorithm	Max	Min	Mean	RMSE	Std
APF	1.5829	0.0409	0.7543	0.8662	0.4259
VI ORB-SLAM	1.6442	0.0500	0.8903	1.1345	0.7003

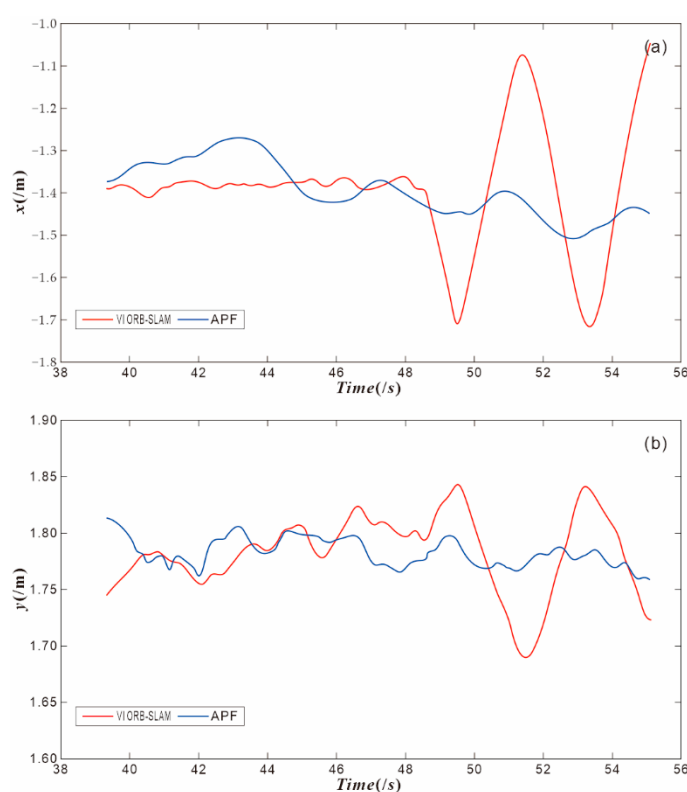


Figure 10. Displacement deviation along the x -axis (a) and y -axis (b) direction.

In the case with 3648 pose estimations, compared with the VI ORB-SLAM algorithm, the proposed algorithm reduces the root mean square error by 0.26 m and the standard deviation by 0.27 m. Compared with the pose solution results of VI ORB-SLAM algorithm, the proposed algorithm is more accurate (see the error analysis in Table 5). As shown in Figure 10, in the process of moving, the displacement deviations of the robots along the x -axis direction and y -axis direction through fusion pose optimization gradually approach the same motion trajectory. The position drift is also decreased with time, indicating that fusion pose optimization can also effectively improve the positioning accuracy of the robot in an indoor scene.

The two groups of experiments that we conducted show that the fusion optimization of the pose based on camera/IMU solves the problem of difficult pose estimation and localization. This is important as in an indoor environment satellite signal is missing. The proposed method also improves the accuracy of pose estimation. Furthermore, it also addresses the issues attributed to the low output frequency and poor dependability of KINECT camera,

and the high-frequency output of IMU used for pose optimization fusion, while meeting the requirements of high-precision pose estimation in highly dynamic situations.

5.3. Map Construction Analysis

The trajectory of the robot during the experiment is shown in Figure 11; it traveled 68.96 m, generated 3648 poses and built a complete indoor environment map. In Figure 11, their poses are measured based on the odometer that comes with the robot directly. Due to the measurement accuracy of the odometer itself, its trajectory can only be used as the reference trajectory data of the experiment, and cannot be used as the true value of the trajectory.

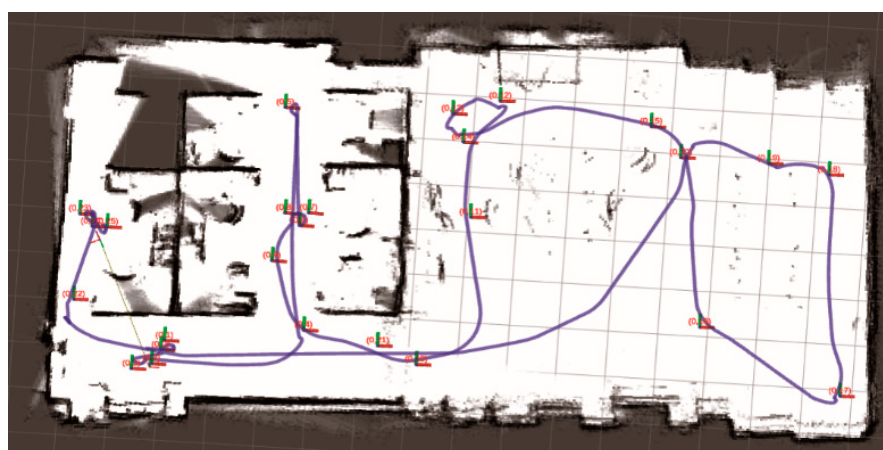


Figure 11. The trajectory of the robot during the experiment (shown as a purple line).

As mentioned above, first of all, the mapping experiment uses a Cartographer algorithm to construct a two-dimensional occupancy grid map of the indoor environment. The experimental result is shown in Figure 12a. Then, the APF algorithm is used to estimate the pose of the robot, and the re-estimated fused pose is used to correct LiDAR data to regenerate a two-dimensional occupancy grid map of the indoor environment, as shown in Figure 12b.

Here, we compare the accuracy and effectiveness of an indoor map construction method based on the APF algorithm with that based on the Cartographer algorithm.

Both the APF algorithm and the Cartographer algorithm use LiDAR data to construct an occupancy grid map. The difference lies in the pose estimation method. The Cartographer algorithm uses a pose estimation method of IMU and odometer, while the APF algorithm uses a pose fusion estimation method based on camera and IMU. Therefore, it can be considered that if there is a difference in the effect of the occupancy grid map, it is most likely caused by the method of pose estimation.

A detailed analysis of the occupancy grid maps constructed by two methods reveals that there is a big difference between the two maps.

Firstly, there may be a phenomenon that the edges are not fitted. That is to say, the probability grids obtained from two different poses do not overlap, meaning that the edges cannot be fitted, and there is a phenomenon of ghosting. Examples are shown in Figure 13B,F. Using the pose fusion algorithm proposed based on camera/IMU in this paper, the 2D laser points are updated according to the robot's pose to form the optimization of laser points distance and azimuth. This improves the accuracy of the estimation and provides a better two-dimensional map edge fitting. However, it does not mean that the optimized pose is used, a good visualization effect can be obtained. Figure 13C is one exception, for example.

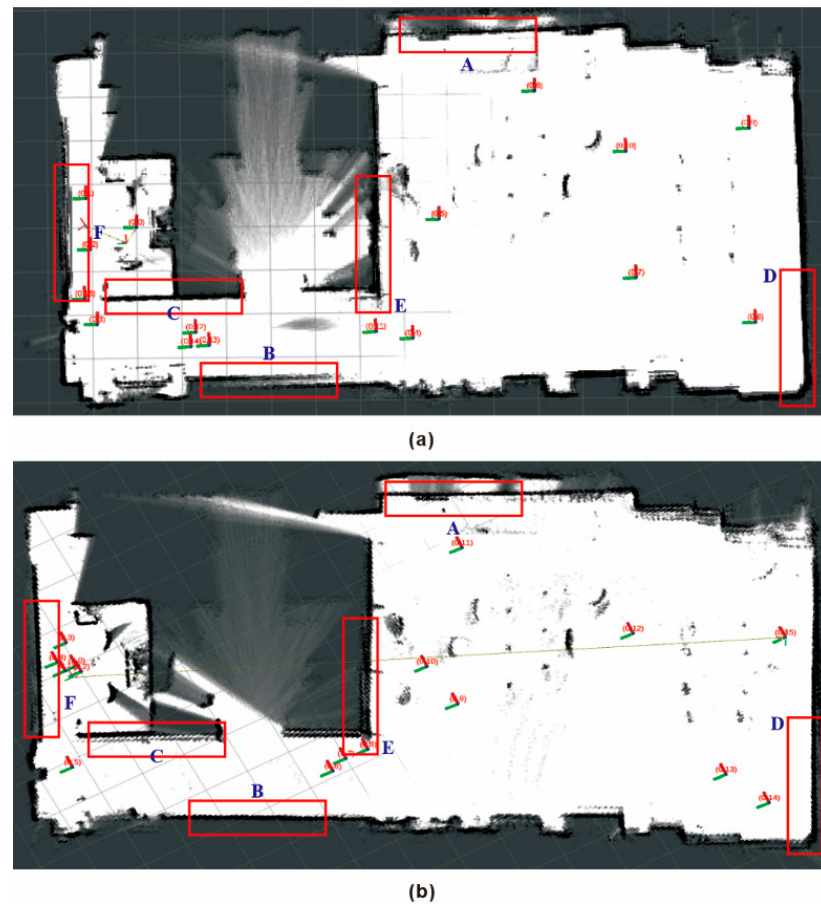


Figure 12. Comparison of mapping effects, and A–F are six selected comparison sub-regions. (a) The map was constructed by Cartographer algorithm. (b) The map was constructed by APF algorithm.

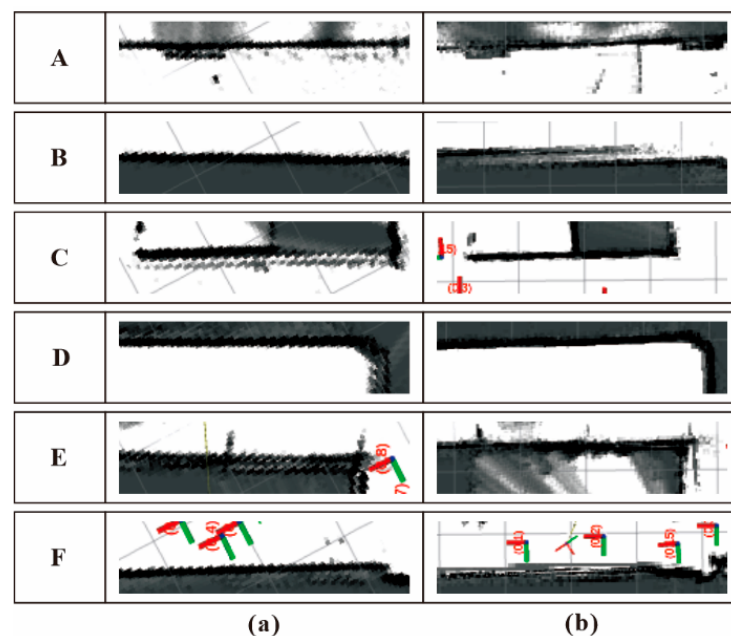


Figure 13. Comparison of map details created by different methods, and A–F are six selected comparison sub-regions. (a) The map was constructed by APF algorithm. (b) The map was constructed by Cartographer algorithm.

Secondly, there may be a phenomenon of right-angle distortion. This phenomenon often occurs in the corner position of an indoor environment. It is also the probability grid obtained by two different poses. The probability grid is fitted, but due to the deviation of the pose direction, the probability grid fitted appears to display a right-angle distortion, or the right-angle edges tilt slightly, such as in Figure 13A,D. As for Figure 13E, although there is ghosting, the right-angle distortion is small.

It is seen that the overall mapping efficiency of the edge details of the map that is constructed by the proposed algorithm is higher in general. The actual wall mapping lines on the two-dimensional plane map are perpendicular to each other, and the relative relationship between the edges has almost no distortion.

However, it can be seen that the overall mapping accuracy and visualization effect of the edge details of the map that is constructed by the proposed algorithm is higher in general. The actual wall mapping lines on the two-dimensional plane map are perpendicular to each other, and the relative relationship between edges has almost no distortion. The proposed algorithm outperforms the Cartographer algorithm.

6. Conclusions

Simultaneous Localization and Mapping is one of the effective methods for robotic environment recognition, localization, and navigation. In this paper, we focus on the indoor environment. We then combine the pose accuracy requirements of indoor mapping and the characteristics of an indoor environment to propose an adaptive pose fusion method. Our proposed method is based on camera/IMU fusion which is applied to two-dimensional map modeling.

In this method, the camera and IMU are used to calculate the position and posture of the robot. Then, by determining the motion state of the robot, the data from the IMU and camera are fused using the EKF method or WPF method to form a new highly accurate pose estimation. The obtained pose is then matched with the laser points measured at the same time to correct the distance and azimuth information of the laser points in the coordinate system. This is then substituted into an iterative matching process based on Gaussian–Newton method to construct the two-dimensional indoor map. Finally, we perform a camera/IMU pose fusion experiment to evaluate the proposed algorithm and verify its performance against Euroc public dataset and the measured dataset.

The experimental results show that the proposed algorithm improves the pose estimation accuracy compared with the ROVIO and VI ORB-SLAM algorithms. Compared with the Cartographer algorithm, the proposed method also provides higher accuracy in two-dimensional map modeling.

Author Contributions: Jinming Zhang contributed to the conception of the study and wrote the manuscript; Lianrui Xu contributed significantly to analysis and performed the data analyses; Cuizhu Bao helped perform the analysis with constructive discussions. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China, grant number 41371383.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data used to support the findings of this study are available from the corresponding author upon request. AWF-Method, the source codes are available for download at the link: <https://github.com/VGEZhangJM/AWF-Method.git> (accessed in 2021).

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

References

- Jin, X.; Wang, H.; Zong, W.; Wang, P.; Wang, C. Research status of obstacle avoidance technologies for autonomous mobile robots. *Transducer Microsyst. Technol.* **2018**, *37*, 5–9.
- Durrant-Whyte, H.; Bailey, T. Simultaneous Localization and Mapping: Part I. *IEEE Robot. Amp Autom. Mag.* **2006**, *13*, 99–110. [\[CrossRef\]](#)
- Cadena, L.; Carlone, H.; Carrillo, Y.; Latif, Y.; Scaramuzza, D.; Neira, J.; Reid, I.; Leonard, J.J. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Trans. Robot.* **2016**, *32*, 1309–1332. [\[CrossRef\]](#)
- Poulose, A.; Han, D.S. Hybrid indoor localization using IMU sensors and smartphone camera. *Sensors* **2019**, *19*, 5084. [\[CrossRef\]](#) [\[PubMed\]](#)
- Fuentes-Pacheco, J.; Ruiz-Ascencio, J.; Rendón-Mancha, J.M. Visual simultaneous localization and mapping: A survey. *Artif. Intell. Rev.* **2015**, *43*, 55–81. [\[CrossRef\]](#)
- Lin, Y.; Gao, F.; Qin, T.; Gao, W.; Liu, T.; Wu, W.; Yang, Z.; Shen, S. Autonomous aerial navigation using monocular visual-inertial fusion. *J. Field Robot.* **2018**, *35*, 23–51. [\[CrossRef\]](#)
- Keivan, N.; Sibley, G. Asynchronous adaptive conditioning for Visual-Inertial SLAM. In *Experimental Robotics*; Hsieh, M., Khatib, O., Kumar, V., Eds.; Springer Tracts in Advanced Robotics: Cham, Switzerland, 2015; pp. 1573–1589.
- Engel, J.; Sturm, J.; Cremers, D. Scale-aware navigation of a low-cost quadcopter with a monocular camera. *Robot. Auton. Syst.* **2014**, *62*, 1646–1656. [\[CrossRef\]](#)
- Lupton, T.; Sukkarieh, S. Visual-inertial-aided navigation for high-dynamic motion in built environments without initial conditions. *IEEE Trans. Robot.* **2011**, *28*, 61–76. [\[CrossRef\]](#)
- Davison, A.J. Real-time simultaneous localisation and mapping with a single camera. In Proceedings of the Ninth IEEE International Conference on Computer Vision, Nice, France, 13–16 October 2003; pp. 1403–1410.
- Castellanos, J.A.; Neira, J.; Tardós, J. Limits to the consistency of EKF-based SLAM. *IFAC Proc.* **2004**, *37*, 716–721. [\[CrossRef\]](#)
- Civera, J.; Grasa, O.G.; Davison, A.J.; Montiel, J. 1-point RANSAC for EKF-based structure from motion. In Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, St. Louis, MO, USA, 10–15 October 2009; pp. 3498–3504.
- Jin, H.; Favaro, P.; Soatto, S. A semi-direct approach to structure from motion. *Vis. Comput.* **2003**, *19*, 377–394. [\[CrossRef\]](#)
- Molton, N.; Davison, A.J.; Reid, I. Locally Planar Patch Features for Real-Time Structure from Motion. In Proceedings of the British Machine Vision Conference, Nottingham, UK, 1–5 September 2004; pp. 1–10.
- Silveira, G.; Malis, E.; Rives, P. An efficient direct approach to visual SLAM. *IEEE Trans. Robot.* **2008**, *24*, 969–979. [\[CrossRef\]](#)
- Jones, E.S.; Soatto, S. Visual-inertial navigation, mapping and localization: A scalable real-time causal approach. *Int. J. Robot. Res.* **2011**, *30*, 407–430. [\[CrossRef\]](#)
- Kelly, J.; Sukhatme, G.S. Visual-inertial sensor fusion: Localization, mapping and sensor-to-sensor self-calibration. *Int. J. Robot. Res.* **2011**, *30*, 56–79. [\[CrossRef\]](#)
- Bailey, T.; Nieto, J.; Guivant, J.; Stevens, M.; Nebot, E. Consistency of the EKF-SLAM algorithm. In Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, Beijing, China, 9–15 October 2006; pp. 3562–3568.
- Weiss, S.; Siegwart, R. Real-time metric state estimation for modular vision-inertial systems. In Proceedings of the 2011 IEEE International Conference on Robotics and Automation, Shanghai, China, 9–13 May 2011; pp. 4531–4537.
- Leutenegger, S.; Lynen, S.; Bosse, M.; Siegwart, R.; Furgale, P. Keyframe-based visual-inertial odometry using nonlinear optimization. *Int. J. Robot. Res.* **2015**, *34*, 314–334. [\[CrossRef\]](#)
- Mourikis, A.I.; Roumeliotis, S.I. A multi-state constraint Kalman filter for vision-aided inertial navigation. In Proceedings of the 2007 IEEE International Conference on Robotics and Automation, Rome, Italy, 10–14 April 2007; pp. 3565–3572.
- Forster, C.; Carlone, L.; Dellaert, F.; Scaramuzza, D. On-manifold preintegration theory for fast and accurate visual-inertial navigation. *IEEE Trans. Robot.* **2015**, 1–18.
- Usenko, V.; Engel, J.; Stücker, J.; Cremers, D. Direct visual-inertial odometry with stereo cameras. In Proceedings of the 2016 IEEE International Conference on Robotics and Automation (ICRA), Stockholm, Sweden, 16–21 May 2016; pp. 1885–1892.
- Li, M.; Mourikis, A.I. High-precision, consistent EKF-based visual-inertial odometry. *Int. J. Robot. Res.* **2013**, *32*, 690–711. [\[CrossRef\]](#)
- Rosten, E.; Porter, R.; Drummond, T. Faster and better: A machine learning approach to corner detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2008**, *32*, 105–119. [\[CrossRef\]](#)
- Muja, M.; Lowe, D.G. Fast approximate nearest neighbors with automatic algorithm configuration. *VISAPP* **2009**, *2*, 331–340.
- Gao, X.; Zhang, T.; Liu, Y.; Yan, Q. *14 Lectures on Visual SLAM: From Theory to Practice*; Electronic Industry Press: Beijing, China, 2017.
- Poulose, A.; Eyobu, O.S.; Han, D.S. An Indoor Position-Estimation Algorithm Using Smartphone IMU Sensor Data. *IEEE Access* **2019**, *7*, 11165–11177. [\[CrossRef\]](#)
- Zhao, R.; Hu, B.; Lv, X.; Tian, J. Filtering algorithm of UKF integrated navigation based on dual-Euler angles. *Syst. Eng. Electron.* **2021**, *43*, 1912–1920.
- Klump, A.R. Singularity-free extraction of a quaternion from a direction-cosine matrix. *J. Spacecr. Rocket.* **1976**, *13*, 754–755. [\[CrossRef\]](#)
- Qin, T.; Li, P.; Shen, S. Vins-mono: A robust and versatile monocular visual-inertial state estimator. *IEEE Trans. Robot.* **2018**, *34*, 1004–1020. [\[CrossRef\]](#)

-
32. Liu, F.; Su, X.; He, Y.; Luo, F.; Gao, H. IMU Preintegration for Visual-Inertial Odometry Pose Estimation. In Proceedings of the 2018 37th Chinese Control Conference (CCC), Wuhan, China, 25–27 July 2018; pp. 5305–5310.
 33. Zhou, X.S.; Roumeliotis, S.I. Robot-to-robot relative pose estimation from range measurements. *IEEE Trans. Robot.* **2008**, *24*, 1379–1393. [[CrossRef](#)]
 34. Xiong, M.; Lu, H.; Xiong, D.; Xiao, J.; Lv, M. Pose estimation of UAV based on monocular vision and inertial navigation. *J. Comput. Appl.* **2017**, *37*, 127–133.
 35. Prozorov, A.; Priorov, A. Three-dimensional reconstruction of a scene with the use of monocular vision. *Meas. Tech.* **2015**, *57*, 1137–1143. [[CrossRef](#)]
 36. Zhu, S.; Mao, J.; Wang, Y.; Liu, C.; Lin, S. Indoor laser SLAM method based on inertial navigation angle compensation. *J. Electron. Meas. Instrum.* **2019**, *33*, 1–7.
 37. Burri, M.; Nikolic, J.; Gohl, P.; Schneider, T.; Rehder, J.; Omari, S.; Achtelik, W.A.; Siegwart, R. The EuRoC micro aerial vehicle datasets. *Int. J. Robot. Res.* **2016**, *35*, 1157–1163. [[CrossRef](#)]
 38. Fernandez, E.; Crespo, L.S.; Mahtani, A.; Martinez, A. *Learning ROS for Robotics Programming*; Packt Publishing Ltd.: Birmingham, UK, 2015.
 39. Bloesch, M.; Burri, M.; Omari, S.; Hutter, M.; Siegwart, R. Iterated extended Kalman filter based visual-inertial odometry using direct photometric feedback. *Int. J. Robot. Res.* **2017**, *36*, 1053–1072. [[CrossRef](#)]
 40. Goczyla, K.; Grabowska, T.; Waloszek, W.; Zawadzki, M. The cartographer algorithm for processing and querying description logics ontologies. In *International Atlantic Web Intelligence Conference*; Springer: Berlin, Heidelberg, 2005; pp. 163–169.