Research on multi-sensor data fusion algorithm for unmanned vehicles under extreme conditions

Ruiqi Zhang
School of Electronic and Optical Engineering, Nanjing University of Science and Technology, Nanjing, China.
zhangruiqi@mail.niit.edu.cn

Abstract. Aiming at the problem of insufficient accuracy in Simultaneous Localization and Mapping of vehicle robot using a single sensor in extreme environment scenes, according to the characteristics of the sensors, a method of fusing the data of lidar and inertial sensors is proposed, the vehicle robot system is designed, and the positioning principle of lidar and inertial sensor is elaborated. Two data fusion algorithms, weighted fusion and Kalman filter, are mainly studied, and the experiments prove that the Kalman filter algorithm has higher positioning accuracy.

Keywords: Extreme environment; sensor fusion; Kalman filter.

1. Introduction
With the development of modern technology, mobile robots play an important role in various transportation and exploration tasks. It can carry out tasks that humans can hardly carry out in special environmental conditions.[1] Simultaneous Localization and Mapping (SLAM) technology enables robots to perceive the environment they are in, construct maps of corresponding scenes and locate themselves.[2] According to different sensors, SLAM can be divided into Lidar-based SLAM (Lidar SLAM), vision-based SLAM (Visual SLAM), inertial sensor-based SLAM, and ultrasonic-based SLAM.[3] The lidar sensor emits a laser beam, and then compares the received signal reflected from the target with the transmitted signal to obtain relevant information about the target. Its main error is instantaneous error.[4] The inertial measurement unit (IMU) measures the angular velocity and acceleration of the object in space through the built-in three-axis gyroscope and accelerometer to estimate the position and attitude of the object. Its main error is the cumulative error.[5] Different from the common indoor environment, the unstructured terrain and various obstacles in the complex environment will have a significant impact on the autonomous movement of the robot. If it is not possible to accurately recognize the various elements in the surrounding environment and obtain the corresponding position contour information It may lead to wrong motion control decisions and put the system in danger.[6] In this case, the sensor alone cannot complete the global map construction without taking into account the measurement requirements such as speed, real-time performance, and accuracy. Lidar SLAM has high accuracy and reliability, and has the advantages of high resolution, anti-interference, small size and light weight, long measurement distance, and rich information.[7] IMU can obtain measurement data with only internal sensors, and is basically not interfered by changes in the external environment.[8] It has the advantages of high update frequency and high estimation accuracy.
in a short period of time, which can well rectify the pose information of the lidar sensor.[9] In the face of complex environments, this paper chooses lidar sensors and inertial sensors for information fusion to obtain more accurate, efficient, and adaptable results.

2. Module principle and scheme design

2.1. Module principle

2.1.1. IMU. Inertial Measurement Unit (IMU) is a device used to detect the three-axis attitude angle and acceleration of an object. Normally, an IMU includes a three-axis gyroscope and a three-axis accelerometer. The gyroscope detects the deflection angular velocity signal of the moving object relative to the initial position direction, and the accelerometer detects the independent acceleration signal of the object in the directions of the three coordinate axes. [10]

The speed measurement models of IMU are as follows:

Gyroscope measurement equation:
\[
\tilde{w}_b(t) = w_b(t) + b_g(t) + n_g
\]

Accelerometer measurement equation:
\[
\tilde{a}_b(t) = a_b(t) + b_a(t) + R_{BW}(t)g_w + n_g
\]

In the equations, \(B\) and \(W\) represent the IMU's own coordinate system and the world coordinate system, respectively, \(R_{BW}\) is the spatial rotation matrix between the two coordinate systems, \(b_a\) and \(b_g\) are random walk noise that obeys Brownian motion, \(n_a\) and \(n_g\) are White noise, \(g_w\) is the gravity component in the world coordinate system.

2.1.2. Lidar. The original measurement data of planar lidar is a series of distances from lidar to the corresponding reflection points. [11] After defining the rectangular coordinate system, the distance can be converted to coordinates. According to the relationship between the increase and decrease of the distance, and the distance difference between two adjacent ranging points and the magnitude of the local error produced by laser, the lidar data can be tested for connectivity and partitioned by region. On the basis of regional segmentation, the slope of obstacles in the region is calculated according to the resolution of the data and the consistency test is carried out, thus judging the structural characteristics of the environment. The feature database is constructed based on the data characterized by the geometric curves of nodes and obstacles. The feature map is the set of geometric characteristic parameters. In indoor environment, the geometric features considered include straight line segments and corner features.[12]

Matching process of straight-line segments

Step 1 Traversal and matching of line segment angle features

According to the slope of the line segment, the included angle between the line segment and the x axis in the local coordinate system can be obtained. Use the positioning parameters \((X_{t_1}, Y_{t_1}, \theta_{t_1})\) of \(t_1\) at the previous time and the heading angle increment provided by IMU to calculate the rotation angle from \(t_1\) to \(t_2\), thus obtaining the angle between the line segment and the x axis in the global coordinate system as

\[
\beta_{gi} = \beta_i + \theta_{t_1} + \theta_e
\]

Set the threshold of line segment matching to \(\delta\).

If \(|(\beta_{gi})_{t_2} - (\beta_{gi})_{t_1}| \leq \delta\), the line segment is considered to be the same line segment as the current line segment in the feature library list, and the matching condition is met.
If \( |(\beta_{\text{gr}})_{2} - (\beta_{\text{gr}})_{1} | > \delta \), the matching condition is not met. Continue to judge the next line segment until all line segments are checked, and the unmatched line segments are added as new line segment features.

Step 2 Distance test of line segment features

Draw the normal of the line segment through the origin of the lidar coordinate system, and find the intersection point \( P(x_{\text{n}}, y_{\text{n}}) \) of the normal and the line segment. The positioning parameters at time \( t_{2} \) are calculated using the positioning parameters at the previous time, and then the local coordinate system is converted to the global coordinate system, so that the intersection point of the normal line and the line segment at the current time is at the position \( (x_{\text{g},2}, y_{\text{g},2}) \) of the global coordinate system.

\[
\begin{pmatrix}
  x_{\text{g},2} \\
  y_{\text{g},2}
\end{pmatrix}
=
\begin{pmatrix}
  \cos(\theta + \theta_{e}) & -\sin(\theta + \theta_{e}) \\
  \sin(\theta + \theta_{e}) & \cos(\theta + \theta_{e})
\end{pmatrix}
\begin{pmatrix}
  x_{\text{n}} \\
  y_{\text{n}}
\end{pmatrix}
+
\begin{pmatrix}
  T_{x_{1}} + S_{x_{e}} \\
  T_{y_{1}} + S_{y_{e}}
\end{pmatrix}
\]

Calculate the geometric distance between the two points according to the coordinate value, and set the threshold value for comparison until all the straight lines are matched.

When the two steps both meet the matching, it can be considered that the straight-line segments match the parameters in the feature list, and the increment used in the matching is used to determine the parameters of the carrier for the next moment.

Matching process of corner features

Distance is used as a basis to the matching of corner features. Set the distance threshold, and calculate the distance \( L_{j} (j = 1, 2, \ldots, n) \) between the point \((x_{j}, y_{j})_{1}\) and the point \((x_{j}, y_{j})_{2}\), where \( n \) is the number of corners in the map. Iterate through all corner features to find out matching points and new corner points. By matching the line segments and corner features of local map and global map, autonomous positioning can be realized relative to the environment measured.

2.2. Scheme design

2.2.1. General structure of system. After the data from the lidar and IMU fused by the upper computer installed with the ROS operating system, the upper computer transmits the data with the lower computer through serial port communication, so as to further control the brushless DC motor and steering gear. In addition, the robot car is equipped with a router, which is convenient for remote control. Intelligent vehicle system includes modules such as power supply module, main controller module, sensor module, driver module, communication module, etc. The concrete structural block diagram is shown below.
2.2.2. **System hardware and software design.** The hardware circuit adopts the modular design, each module completes the corresponding function and keep independent relatively, modules like single-chip microcomputer minimum system, radar acquisition module, power supply module, motor module and communication module are mainly included.

The main framework of the software is the ROS robot operating system, based on which drivers of lidar and IMU attitude sensors are added to obtain the information of the surrounding environment and the overall attitude information of the vehicle. After the process of the collected data, the map is constructed through the SLAM algorithm, and then the vehicle can be controlled to reach the desired location through route planning and autonomous navigation. In the software, we can also use SSH to achieve the remote control of the entire system.

3. **Algorithm research**

3.1. **Weighted fusion**

In the multi-source fusion system, the weight matrix of n class indexes extracted by the American Electrical Equipment Manufacturers Association protocol is designed and its consistency is tested. Then the weight vector of multiple sensors is calculated to get the weight vector set \( W = (w_1, w_2, w_3, \ldots, w_n) \) based on the index, according to which the result of weighted fusion is calculated.[13] The process diagram is as follows.

![Diagram](image)

**Fig.2** Define the trajectory data set of n multi-sensors as

\[
X = \{X_1, X_2, X_3, \ldots, X_n\}
\]

Where \( X_i \) represents the data set of the sensor number i, and the trajectory data of a single sensor is

\[
X_i = \{x_{i1}, x_{i2}, x_{i3}, \ldots, x_{in}\}
\]

\( x_{ij} \) in the formula is the sample point data at the time i.

Weighted fusion of multi-sensor data adopts one-dimensional multi-source fusion model, i.e

\[
\hat{X} = \sum_{i=1}^{n} \hat{X}_i \cdot W_i
\]

Since each sensor has different influence on navigation accuracy in different scenes, weight \( W_i \) can be respectively fused by analytic hierarchy process (AHP) to obtain the weighted fusion data \( \hat{X} \) after final fusion.
3.2. Kalman filter
Kalman filter is a modern optimal estimation theory which is linear, unbiased, and takes the minimum error variance as the estimation criterion.\[14\] It includes two steps: prediction and correction. Through prediction, the impact of uncertain system dynamics generated during the measurement process is mainly considered to update the estimation error. And update the estimation error by obtaining new information from the sensor measurement through correction.

In the calculation method, the Kalman filter adopts the recursive form, that is, on the basis of the estimation at the previous time, the state estimation $\hat{X}(t)$ at the time $t$ is obtained by recursion according to the measured value $Z(t)$ at the time $t$. The estimation $\hat{X}(t)$ obtained by this recursive algorithm can also be considered to be obtained by comprehensively using all measurement information before and at the time $t$, and only process the measurement value of one moment at a time, so that the amount of calculation is greatly reduced. Kalman filter is mainly applicable to linear dynamic systems because it describes the system and the measurement value by using the state equation and linear measurement equation.\[15\]

Suppose the state equation and measurement equation of the discrete system are as follows:

\[
\begin{align*}
X_k &= \Phi_{k/k-1}X_{k-1} + \Gamma_k W_k \\
Z_k &= H_k X_k + V_k
\end{align*}
\]

In the equations, $X_k$ is the estimated n-dimensional state matrix at time $k$; $Z_k$ is the n-dimensional measurement matrix at time $k$; $\Phi_{k/k-1}$ is a transition matrix $(n \times n)$ from time $k$ to $k+1$; $W_k$ is the system noise matrix at time $k$; $H_k$ is the measurement matrix $(m \times n)$ at time $k$; $\Gamma_k$ is the weighting matrix of system noise; $V_k$ is the n-dimensional measurement noise matrix at time $k$.

In Kalman filtering, it is required that $W_k$ and $V_k$ are uncorrelated zero-mean white noise sequences, which need to meet:

\[
\begin{align*}
E[W_k] &= 0, E[W_kW_k^T] = Q_k \delta_{kk} ; \\
E[V_k] &= 0, E[V_kV_k^T] = R_k \delta_{kk} ; \\
E[W_kV_k^T] &= 0 ;
\end{align*}
\]

In the equation, $Q_k$ is the variance matrix of the system noise vector $W_k$, which must be a non-negative definite matrix; $R_k$ is the variance matrix of the measured noise vector $V_k$, which must be a positive definite matrix; $\delta_{kk}$ is Kronecker function.

The conventional Kalman filter mainly includes two parts: prediction estimation and filtering estimation.

1) Prediction estimation

\[
\begin{align*}
\hat{X}_{k/k-1} &= \Phi_{k/k-1} \hat{X}_{k-1/k-1} \\
P_{k/k-1} &= \Phi_{k/k-1} P_{k-1/k-1} \Phi_{k/k-1}^T + \Gamma_k Q_k \Gamma_k^T
\end{align*}
\]

2) Filtering estimation

\[
\begin{align*}
K_k &= P_{k/k-1} H_k^T (H_k P_{k/k-1} H_k^T + R_k)^{-1} \\
\hat{X}_{k/k} &= \hat{X}_{k/k-1} - K_k (Z_k - H_k \hat{X}_{k/k-1}) \\
P_{k/k} &= (I - K_k H_k) P_{k/k-1} (I - K_k H_k)^T + K_k R_k K_k^T \\
\text{or } P_{k/k} &= (I - K_k H_k) P_{k/k-1}
\end{align*}
\]
In the equation, $K_k$ is the filter gain matrix at time $k$; $P_{k/k-1}$ is the prediction estimation error covariance matrix from time $k-1$ to time $k$; $P_{k/k}$ is the optimal filter value error covariance matrix at time $k$.

4. Experiment and results
This experiment is based on ROS operating system in Linux environment.

Fig.3 Experiment scene:

Fig.4 Experiment platform:
As shown in the figure, with 0.1s as the sampling time interval, the absolute error between the actual path and the planned path during the navigation process is recorded. The results show that, as time goes by, the offset error of the three cases is increasing, but the method of sensor data fusion can reduce the error. In addition, the Kalman filter algorithm has achieved better results than the traditional weighted fusion algorithm.

5. Summary
In the face of autonomous navigation requirements of unmanned vehicles that need to work in extreme environmental conditions, this paper studies the combined navigation method of lidar and inertial sensor, and constructs experiment to verify the algorithm. The results show that the combined navigation of lidar and inertial sensor can effectively improve the accuracy of the system in extreme environment compared with the scheme using only a single lidar sensor. And Kalman filter algorithm can effectively process the fusion data and further reduces the random error compared with the traditional weighted fusion algorithm. Therefore, sensor data fusion using Kalman filter algorithm can make full use of the advantages of sensors, significantly improve the positioning accuracy, and promise a strong application prospect.

References
[1] Hongling Wang. Research on Simultaneous Localization and Mapping (SLAM) Methodology of Mobile Robots Adapting to Search and Rescue Environment[D]. Shandong University, 2018.
[2] Randall C. Smith. On the Representation and Estimation of Spatial Uncertainty[J]. The International Journal of Robotics Research, 1986, 5(4): 56-68.
[3] Shiyan Deng, Chengjun Guo. Research on Simultaneous Localization and Mapping Method Based on Multi-sensor Fusion [A]. Academic Exchange Center of China Satellite Navigation System Management Office. Proceedings of the 11th China Satellite Navigation Annual Conference -- S13 Autonomous Navigation [C]. China Satellite Navigation System Management Office Academic Exchange Center: China Science and Technology Beidouhui (Beijing) Technology Co., Ltd., 2020: 6.
[4] Dezhi Wang. Principles of Lidar[J]. Forum for Science and Technology Association (second half
of the month), 2008(05):40.

[5] Haixia Wang, Qingfeng Wu, Xiangbin Wu, Qiuyi Zhang. Application research of multi-sensor fusion in robot position perception[J]. Mechanical and Electrical Engineering Technology, 2020, 49(12): 89-91.

[6] Jiang Zhu. Research on Mobile Robot Cognition and Navigation Obstacle Avoidance Method in Unstructured Environment [D]. Hunan University, 2011.

[7] LV Bin, XU Hao, WU Jianging, et al. LIDAR-Enhanced Connected Infrastructures Sensing and Broadcasting High-Resolution Traffic Information Serving Smart Cities [J]. IEEE Access, 2019(7):79895-79907.

[8] Bolin Dong, Xu Chai. Research on navigation and positioning algorithm based on IMU/vision fusion [J]. Piezoelectric and Sound and Light, 2020, 42(05): 724-728.

[9] Shiyan Deng, Chengjun Guo. Research on Simultaneous Localization and Mapping methods based on multi-sensor fusion [A]. Academic Exchange Center of China Satellite Navigation System Management Office. Proceedings of the 11th China Satellite Navigation Conference-S13 Autonomous Navigation [ C]. Academic Exchange Center of China Satellite Navigation System Management Office: Zhongke Beidouhui (Beijing) Technology Co., Ltd., 2020: 6.

[10] Yao Xu. Research on multi-sensor fusion positioning method of indoor mobile robot based on QR code [D]. Zhejiang University of Science and Technology, 2020.

[11] Ziwei Liao, Rongbing Li, Tingwan Lei, Yijun Hang. Indoor scan matching SLAM method based on geometric feature association[J]. Navigation and Control, 2016, 15(03): 26-32+56.

[12] Rongbing Li, Zhiqi Wang, Ziwei Liao. Research on Lidar/MEMS Micro-inertial Integrated Indoor Navigation Algorithm[J]. Sensors and Microsystems, 2020, 39(12): 14-17+21.

[13] Yajun Fan, Ping Wang. BDS adaptive navigation data fusion algorithm for complex scenes[J]. Journal of Navigation and Positioning, 2020, 8(04): 44-49+57.

[14] Dong Zhang, Songming Jiao, Yanquan Liu. A fusion attitude calculation method based on complementary filtering and Kalman filtering[J]. Sensors and Microsystems, 2017, 36(03): 62-65+69.

[15] Dongbin Hu, Junqiang Liu, Chunlu Pan. Research on UAV integrated navigation algorithm based on SINS/GPS/CNS[J]. Aeronautical Computing Technology, 2021, 51(01): 41-45.