Motion Estimation Algorithm Based on GA-Optimized High-Order Extended Kalman Filter

Yue Cao, Jixin Liu*, Yanbin Cai

School of Automation, Guangdong University of Petrochemical Technology, Maoming 525000, China

Abstract: To address the limitations of the Extended Kalman Filter (EKF) in handling strongly nonlinear systems, such as low accuracy, significant errors in robotic motion estimation, weak anti-interference capabilities, and sub-optimal performance in vehicle motion estimation, this paper introduces a novel vehicle motion estimation method based on a Genetic Algorithm-Optimized Higher Order Extended Kalman Filter (GA-HEKF). The HEKF algorithm enhances traditional EKF by incorporating hidden variables to mitigate round-off errors and pseudo-linearizing the vehicle motion model. This approach establishes a linear relationship between system variables and hidden variables, transforming the system state model into a linear form and equivalently rewriting the observation model. As a result, the vehicle motion model is reformulated into a structure compatible with Kalman filtering, enabling more accurate state estimation. To further improve the performance of the HEKF algorithm, an improved genetic algorithm is employed to optimize the covariance matrices of system noise and observation noise during the high-order extended Kalman filtering process. This optimization enhances the precision of vehicle state parameter estimation by dynamically adjusting the noise characteristics to better reflect real-world conditions. Simulation experiments are conducted to validate the effectiveness of the proposed method. The results demonstrate that the GA-HEKF algorithm outperforms traditional EKF in terms of accuracy, robustness, and computational efficiency. Specifically, it exhibits superior performance in scenarios with high nonlinearity and noise interference, making it a promising solution for advanced vehicle motion estimation applications. The proposed method not only addresses the inherent limitations of EKF but also provides a framework for improving state estimation in other complex nonlinear systems.

Keywords: Motion Estimation; Higher Order Extended Kalman Filter; Genetic Algorithm

DOI: 10.57237/j.cst.2025.02.001

1 Introduction

In recent years, as research on robotics has deepened, the role of robots in modern industrial technology has become increasingly important. The demands on SLAM technology for robots have also been rising, which has attracted attention from scholars across various fields [1]. Traditional SLAM algorithms are divided into vector-based SLAM algorithms and random finite set (RFS) based SLAM algorithms. Smith R. initially proposed us-

Funding: Guangdong Province Sci-ence and Technology Innovation Strategic Special Funding under Grant 2023S003042;
Maoming City Science and Technology Plan Project under Grant 2021002;
Maoming City Science and Technology Plan Project under Grant 2024012;
The Talent Introduction Project for Guangdong University of Petrochemical Technology under Grant 2020rc32.

*Corresponding author: Jixin Liu, liujixin2000@gdupt.edu.cn

Received: 8 January 2025; Accepted: 18 February 2025; Published Online: 17 April 2025 http://www.computscitech.com

ing the Extended Kalman Filter (EKF) to solve SLAM problems, employing maximum likelihood for data association. Their map consists of landmarks, and the robot continuously uses sensors for data observation to improve positioning accuracy. This algorithm suffers from high computational complexity and poor robustness [2]. Montemerlo proposed the FastSLAM algorithm based on the Rao-Blackwellized Particle Filter, which has better scalability than EKF-SLAM [3-5]. Bailey improved the extended Kalman filter algorithm to enhance computational efficiency [6]. Pfingsthorn M. addressed the shortcomings of both algorithms by proposing a HybridSLAM algorithm that combines their advantages: FAST-SLAM handles front-end map processing while EKF-SLAM performs back-end data fusion, thereby enhancing algorithmic feasibility [7]. The RFS-based SLAM algorithm uses RFS theory for digital modeling; RFS is a set form of random variables, replacing vector representations with random finite sets to represent map features and observation data. Mullame proposed PHD-SLAM using probability hypothesis density (PHD) intensity function recursion to jointly estimate the robot's pose and map features, implementing pose estimation with a particle filter and map feature estimation with a PHD filter [8, 9]. To address PHD-SLAM's accuracy limitations, Deusch implemented the labeled multi-Bernoulli filter algorithm to SLAM problems, enhancing the algorithm's estimation accuracy [10]. Zhen Zhou and others proposed a novel EKF-SLAM algorithm based on a forgetting factor to reduce state estimation errors and improve filtering effects [11]. Rauniyar Shyam and others proposed a differential flatness-based linear quadratic regulator, utilizing inertial measurement unit sensor measurements and considering sensor constraints (such as limited sensing range and field of view) for light detection and ranging, maximizing sensor data utilization and performance improvement [12]. Mailky Hamza and others introduced an end-to-end simultaneous localization and mapping architecture based on scan matching and extended Kalman filtering, successfully predicting using laser radar, GNSS, and IMU data sensor fusion [13]. Although improved traditional algorithms have significantly enhanced efficiency and performance in mobile robot localization and mapping, they still exhibit inadequate adaptability and computational precision in complex environments. This paper proposes a vehicle feature estimation algorithm based on higher-order extended Kalman filtering. By fitting nonlinear system models and introducing hidden variables, it ultimately reformulates them into a linearized form, improving the algorithm's precision and anti-interference capabilities.

2 High-Order Extended Kalman Filter

2.1 Extended Kalman Filter

The Kalman filter is a mathematical algorithm used for estimating the state of dynamic systems. It was originally proposed by mathematician Rudolf Kalman in the 1960s [14]. However, it has some limitations, as it is only applicable to linear system problems and cannot solve nonlinear system models. The EKF (Extended Kalman Filter) is an extension of the Kalman filter that can be used for state estimation in nonlinear systems [15]. Unlike the Kalman filter, the EKF approximates nonlinear functions using Taylor expansion to linearize them, and then applies the principles of the Kalman filter for estimation.

Here is the state-space function for a nonlinear discrete system.

$$\begin{cases} x(k+1) = f(x(k), u(k)) + w(k) \\ y(k+1) = h(x(k+1)) + \varphi(k+1) \end{cases} \tag{1}$$

Where $f(x(k), u(k))$ is the state transition function, $h(x(k+1))$ is the observation function, and they possess first-order partial derivatives. $w(k)$ and $\varphi(k+1)$ are the state noise and observation noise, respectively.

The state model and observation model are expanded using Taylor series at points $\hat{x}(k)$ and $\hat{x}(k+1)$, respectively, and terms of second order and higher are discarded.

$$\begin{cases} f(x(k), u(k)) \approx f(\hat{x}(k), u(k)) + \left.\dfrac{\partial f(x(k), u(k))}{\partial x(k)}\right|_{x(k)=\hat{x}(k)} (x(k) - \hat{x}(k)) \\ h(x(k+1)) \approx h(\hat{x}(k+1)) + \left.\dfrac{\partial h(x(k+1))}{\partial x(k+1)}\right|_{x(k+1)=\hat{x}(k+1)} (x(k+1) - \hat{x}(k+1)) \end{cases} \tag{2}$$

The equation for the linearized Extended Kalman Filter is as follows.

The prediction phase:

$$\begin{aligned}\hat{x}(k+1|k) &= f(\hat{x}(k|k),u(k))\\ P(k+1|k) &= A(k+1|k)P(k|k)A^T(k+1|k)+Q(k)\end{aligned} \tag{3}$$

The update phase:

$$\begin{aligned}\hat{x}(k+1|k+1) &= \hat{x}(k+1|k)+K(k+1)[y(k+1)-h(\hat{x}(k+1|k))]\\ K(k+1) &= P(k+1|k)C^T(k+1)[C(k+1)P(k+1|k)C^T(k+1)+R(k+1)]^{-1}\\ P(k+1|k+1) &= \left[I-K(k+1)C(k+1)\right]P(k+1|k)\end{aligned} \tag{4}$$

The EKF uses a Taylor series to expand nonlinear functions and discards their higher-order terms, allowing the Kalman filter to be used for state estimation in nonlinear systems, thereby improving estimation accuracy. Compared to the traditional Kalman filter, the EKF can estimate multiple state variables simultaneously, making it more flexible in dealing with multi-dimensional state estimation problems. However, for systems with strong nonlinearity, discarding the higher-order terms of the Taylor expansion and the complexity of computing the Jacobian matrix during the linearization process can lead to a decrease in filtering performance.

2.2 High-order Extended Kalman Filter

Given a class of strongly nonlinear system equations [16]:

$$\begin{aligned}x(k+1) &= f(x(k))+w(k)\\ y(k+1) &= h(x(k+1))+v(k+1)\end{aligned} \tag{5}$$

where $x(k+1)\in R^{n\times 1}$ is the state vector, $y(k+1)\in R^{m\times 1}$ is the observation vector, $f(x(k))$ is the state transition function, $h(x(k+1))$ is the observation function, $w(k)$ and $\varphi(k+1)$ are state noise and observation noise, respectively.

The state transition function can be expressed as:

$$f(x(k)) = a^{(0)}(k)x(k)+a^{(1)}(k)f^{(1)}(x(k))+\cdots+a^{(r)}(k)f^{(r)}(x(k)) \tag{6}$$

Let

$$\alpha^{(l)}(k) = f^{(l)}(x(k)); l=1,2,\cdots,r \tag{7}$$

and denote $\alpha^{(l)}(k)$ as the hidden variable function of the original variable $x(k)$.

Then the state equation can be rewritten as:

$$x(k+1) = a^{(0)}(k)x(k)+\sum_{l=1}^{r}a^{(l)}(k)\alpha^{(l)}(k)+w(k) \tag{8}$$

Although Equation (8) is formally a linear system, the system still varies over time with respect to the original variables. The introduction of the hidden variable function only changes the form, hence it is referred to as pseudo-linearization. The hidden variables should be considered as new variables of the system and the dimensions of the original and hidden variables are extended. Also the system equations are linearized within the expanded dimensional space.

First, let the hidden variables be time-varying parameters, and establish a linear coupling relationship between $\alpha^{(l)}(k+1)$ and $x(k)$.

$$\alpha^{(l)}(k+1) = \sum_{u=l}^{r}e_l^{(u)}(k)x^{(u)}(k), u=1,2,\cdots,r \tag{9}$$

where $e_l^{(u)}(k), u=1,2,\cdots,r$ are the model parameters to be identified.

Represent the original variables and hidden variables in an expanded dimension:

$$X(k) = \left[x(k)\quad \alpha^{(1)}(k)\quad \cdots\quad \alpha^{(r)}(k)\right]^T \tag{10}$$

Then the original state model can be rewritten as:

$$\begin{bmatrix} x^{(1)}(k+1) \\ \vdots \\ x^{(l)}(k+1) \\ \vdots \\ x^{(r)}(k+1) \end{bmatrix} = \begin{bmatrix} a_1^{(1)} & \cdots & a_1^{(l)} & \cdots & a_1^{(r)} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ b_l^{(1)} & \cdots & b_l^{(l)} & \cdots & b_l^{(r)} \\ \vdots & \cdots & \vdots & \ddots & \vdots \\ b_r^{(1)} & \cdots & b_r^{(l)} & \cdots & b_r^{(r)} \end{bmatrix} \begin{bmatrix} x^{(1)}(k) \\ \vdots \\ \alpha^{(l)}(k) \\ \vdots \\ \alpha^{(r)}(k) \end{bmatrix} + \begin{bmatrix} w^{(1)}(k) \\ \vdots \\ w^{(l)}(k) \\ \vdots \\ w^{(r)}(k) \end{bmatrix} \tag{11}$$

where $w^{(l)}(k), l = 1, 2 \ldots r$ is the random error vector produced by introducing higher-order hidden variables.

If denote:

$$F(k) = \begin{bmatrix} a_1^{(1)} & \cdots & a_1^{(l)} & \cdots & a_1^{(r)} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ b_l^{(1)} & \cdots & b_l^{(l)} & \cdots & b_l^{(r)} \\ \vdots & \cdots & \vdots & \ddots & \vdots \\ b_r^{(1)} & \cdots & b_r^{(l)} & \cdots & b_r^{(r)} \end{bmatrix} \tag{12}$$

$$W(k) = \left[(w^{(1)}(k))^T, \cdots, (w^{(l)}(k))^T, \cdots, (w^{(r)}(k))^T \right]^T \tag{13}$$

Then, from Equation (10), it follows that the nonlinear state model can be linearly represented as:

$$X(k+1) = F(k)X(k) + W(k) \tag{14}$$

If denote:

$$h(x(k+1)) = c^{(0)}(k+1)x(k+1) + c^{(1)}(k+1)h^{(1)}(x(k+1)) + \cdots + c^{(r)}(k+1)h^{(r)}(x(k+1)) \tag{15}$$

According to Equations (7)-(11), the nonlinear observation model can be linearly represented as:

$$H(k+1) = \begin{bmatrix} c^{(l)} & \cdots & c^{(l)} & \cdots & c^{(r)} \end{bmatrix} \tag{16}$$

$$y(k+1) = H(k+1)X(k+1) + v(k+1) \tag{17}$$

3 Cart Motion Model

Two-dimensional model of a front-wheel drive cart is adopted in this paper, as shown in Figure 1. Sensors such as Lidar and Odometer share the same coordinate system with the vehicle body.

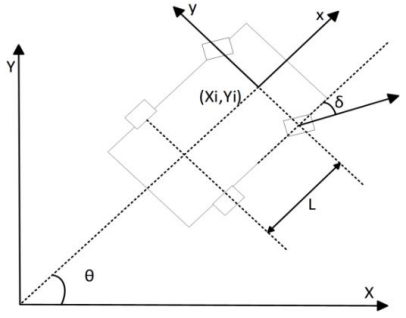


Figure 1 Cart Motion Model

The cart body can be simplified to a line segment with length L, and the center of it is located at the front wheel center point. Assuming that both the rear wheel steering angle and the slip angle are 0, as shown in Figure 2.

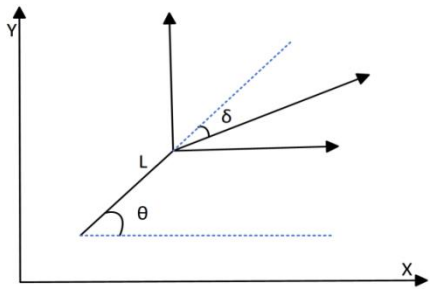


Figure 2 Simplified Cart Model

The single cart model is used to establish the kinematic model, with speed and steering angle as control input quantities. The motion model of the cart can be obtained as follows:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos\theta \\ \sin\theta \\ \tan\delta / L \end{bmatrix} V \tag{18}$$

During the process of cart motion control, the control object is as follows:

$$\mu = \begin{bmatrix} V \\ \omega \end{bmatrix} \tag{19}$$

Where ω is the angular velocity, then equation (18) can be written as:

$$\begin{bmatrix} \dot{\bar{x}} \\ \dot{\bar{y}} \\ \dot{\bar{\theta}} \end{bmatrix} = \begin{bmatrix} \cos\theta \\ \sin\theta \\ 0 \end{bmatrix} V + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \omega \tag{20}$$

Discretize the motion model:

$$\begin{bmatrix} x(k+1) \\ y(k+1) \\ \theta(k+1) \end{bmatrix} = \begin{bmatrix} x(k) \\ y(k) \\ \theta(k) \end{bmatrix} + \begin{bmatrix} \Delta TV(k+1)\cos(\theta(k)) \\ \Delta TV(k+1)\sin(\theta(k)) \\ \Delta T\omega(k) \end{bmatrix} + \begin{bmatrix} q_x(k+1) \\ q_y(k+1) \\ q_\theta(k+1) \end{bmatrix} \tag{21}$$

Where $q(k+1)$ is the system modeling error. The speed V is set as a constant value.

The observation returned by the cart's sensor is the relative distance and azimuth angle between the cart and the road sign. And the following equation can be obtained from Figure 1.

$$\begin{bmatrix} z_r^i(k) \\ z_\varphi^i(k) \end{bmatrix} = \begin{bmatrix} \sqrt{\left(x_i - x(k)\right)^2 + \left(y_i - y(k)\right)^2} \\ \arctan\left(\dfrac{y_i - y(k)}{x_i - x(k)}\right) - \theta(k) \end{bmatrix} + \begin{bmatrix} r_i^i(k) \\ r_j^i(k) \end{bmatrix} \tag{22}$$

where $r^i(k)$ is the observation noise, and (x_i, y_i) is the coordinate of the road sign in the map.

4 Motion Estimation Based on High-Order Extended Kalman Filter

The conventional Extended Kalman Filter (EKF) linearizes nonlinear functions via first-order Taylor series expansion while discarding higher-order terms. Such approximation inevitably introduces truncation errors, leading to suboptimal estimation accuracy. To address this limitation, this paper proposes a Higher-Order Extended Kalman Filter (HEKF) framework specifically designed for vehicle motion estimation. As discussed in the previous section, the equation for the cart motion system is:

$$\begin{cases} x_1(k+1) = x_1(k) + \Delta TV\cos(x_3(k)) + q_1(k+1) \\ x_2(k+1) = x_2(k) + \Delta TV\sin(x_3(k)) + q_2(k+1) \\ x_3(k+1) = x_3(k) + \Delta T\omega(k) + q_3(k+1) \end{cases} \tag{23}$$

$$\begin{cases} y_1 = \sqrt{\left(x_i - x_1(k)\right)^2 + \left(y_i - x_2(k)\right)^2} + r_i^i(k) \\ y_2 = \arctan\left(\dfrac{y_i - x_2(k)}{x_i - x_1(k)}\right) - x_3(k) + r_j^i(k) \end{cases} \tag{24}$$

High-order polynomial fitting is performed separately, and the goodness of fit for each equation is shown in Table 1.

Table 1 Goodness of Fit for each order

Equation \ Order	2	3	4	5	6	7
x_1	0.6273	0.6551	0.9012	0.9832	0.9834	0.9844
x_2	0.4892	0.9085	0.9738	0.9896	0.9898	0.9801
y_1	0.3786	0.8975	0.9544	0.9912	0.9920	0.9922
y_2	0.3549	0.4862	0.8822	0.9764	0.9801	0.9809

From Table 1, it can be seen that during the fitting process, the goodness of fit increases rapidly from second to fifth order, and grows slowly and with a smaller magnitude above the fifth order. Therefore, fifth-order polynomial is chosen for fitting. The state equation after fitting is:

$$\begin{cases} x_1(k+1) = x_1(k) + a_0 + a_1 x_3(k) + a_2 x_3^2(k) + a_3 x_3^3(k) + a_4 x_3^4(k) + q_1(k) \\ x_2(k+1) = x_2(k) + b_0 + b_1 x_3(k) + b_2 x_3^2(k) + b_3 x_3^3(k) + b_4 x_3^4(k) + q_2(k) \\ x_3(k+1) = x_3(k) + \omega(k) + q_3(k) \end{cases} \tag{25}$$

The observation equation is:

$$\begin{aligned} y_1 = {} & c_1 + c_2 x_1 + c_3 x_2 + c_4 x_1^2 + c_5 x_2^2 + c_6 x_1 x_2 + c_7 x_1^3 + c_8 x_1 x_2^2 + c_9 x_1^2 x_2 + c_{10} x_2^3 + c_{11} x_1 x_2^3 + c_{12} x_1^3 x_2 + \\ & c_{13} x_2^4 + c_{14} x_1^4 + c_{15} x_1^2 x_2^2 + r_1(k) \end{aligned} \tag{26}$$

$$\begin{aligned} y_2 = {} & d_1 + d_2 x_1 + d_3 x_2 + d_4 x_1^2 + d_5 x_2^2 + d_6 x_1 x_2 + d_7 x_1^3 + d_8 x_1 x_2^2 + d_9 x_1^2 x_2 + d_{10} x_2^3 + d_{11} x_1 x_2^3 + d_{12} x_1^3 x_2 + \\ & d_{13} x_2^4 + d_{14} x_1^4 + d_{15} x_1^2 x_2^2 - x_3(k) + r_2(k) \end{aligned} \tag{27}$$

Let:

$$\alpha^{(l)}(k) = x_1^m(k) x_2^n(k) x_3^o(k) \tag{28}$$

Then the equation (25) to (27) can be rewritten as:

$$x_i(k+1) = h^{(i)} x_i(k) + \sum_l^r h_l^{(i)}(k) \alpha^{(l)}(k) + q_i(k+1) \tag{29}$$

$$y_i(k+1) = g^{(i)} x_i(k+1) + \sum_l^r g_l^{(i)}(k) \alpha^{(l)}(k+1) + r_l^i(k+1)) \tag{30}$$

Where $h_l^{(i)}$ and $g_l^{(i)}$ represent the coefficients of each term. Let the hidden variable be a time-varying parameter, establishing a linear coupling relationship between $\alpha^{(l)}(k+1)$ and $x(k)$:

$$\alpha^{(l)}(k+1) = \sum_{u=1}^{r} e_l^{(u)}(k) \alpha^{(u)}(k), u = 1, 2, \cdots, r \tag{31}$$

When there are no prior estimates, $e_l^{(u)}$ is given as:

$$e_l^{(u)}(k) = \begin{cases} I, & l = u \\ 0, & l \neq u \end{cases} \tag{32}$$

Let the augmented variable be:

$$X(k) = \left[x_1(k), x_2(k), x_3(k), \alpha^{(4)}(k), \alpha^{(5)}(k), \cdots, \alpha^{(34)}(k), \alpha^{(35)}(k) \right]^T \tag{33}$$

The system transition matrix is:

$$F(k) = \begin{bmatrix} h_1^{(1)} & h_1^{(2)} & \cdots & h_1^{(35)} \\ h_2^{(1)} & h_2^{(2)} & \cdots & h_2^{(35)} \\ h_3^{(1)} & h_3^{(2)} & \cdots & h_3^{(35)} \\ e_4^{(1)} & e_4^{(2)} & \cdots & e_4^{(35)} \\ \vdots & \vdots & \ddots & \vdots \\ e_{35}^{(1)} & e_{35}^{(2)} & \cdots & e_{35}^{(35)} \end{bmatrix} \tag{34}$$

The system modeling error is:

$$W(k) = \left[q_1(k), q_2(k) \cdots, q_{35}(k) \right]^T \tag{35}$$

Thus, the system equation can be written as:

$$X(k+1)=F(k)X(k)+W(k) \tag{36}$$

The observation equation can be written as:

$$y(k+1)=H(k+1)X(k+1)+v(k+1) \tag{37}$$

Where $H(k+1)$ is defined as:

$$H(k+1)=\begin{bmatrix} g_{(1)}^{1} & \cdots & g_{(l)}^{1} & \cdots & g_{(35)}^{1} \\ g_{(1)}^{2} & \cdots & g_{(l)}^{2} & \cdots & g_{(35)}^{2} \end{bmatrix} \tag{38}$$

Then the high-order extened Kalman filter designing procedure is as follows.

Step 1: Set the initial values of the new system as $X(0)$, the initial covariance as $P(0)$;

Step 2: Time update, Obtain the predicted value $\hat{X}(k+1|k)$ from the initial state estimate $\hat{X}(k|k)$:

$$\hat{X}(k+1|k)=F(k)\hat{X}(k|k) \tag{39}$$

estimate the predicted error covariance matrix $P(k+1|k)$ from the prediction error:

$$P(k+1|k)=F(k)P(k|k)F^{T}(k)+Q \tag{40}$$

Step 3: Measurement update, the predicted measurement value is:

$$\hat{Y}(k+1|k)=H(k)\hat{X}(k+1|k) \tag{41}$$

state correction:

$$\hat{X}(k+1|k+1)=\hat{X}(k+1|k)+K(k+1)\left(y(k+1)-\hat{Y}(k+1|k)\right) \tag{42}$$

where $K(k+1)$ is the Kalman gain for the extended new system:

$$K(k+1)=P(k+1|k)H^{T}(k+1)\times\left(H(k+1)P(k+1|k)H^{T}(k+1)+R(k)\right)^{-1} \tag{43}$$

system error covariance matrix estimation:

$$P(k+1|k+1)=[I-K(k+1)H(k+1)]P(k+1|k) \tag{44}$$

Step 4: Repeat Step 3 and Step 4 until the last navigation point is reached.

5 Hekf Parameter Estimation Based on Genetic Algorithm

In the estimation process of the Extended Kalman Filter (EKF) and the Higher-Order Extended Kalman Filter (HEKF) algorithm, the covariance matrices Q of system process noise and R of observation noise play a crucial role in determining the performance of the estimator. Previous studies have usually assumed these matrices to be Gaussian white noise, by setting their fixed values (or mean vectors) to zero and assuming independence from the system state. However, in actual vehicle driving scenarios, the surrounding driving environment is complex, and road surface conditions exhibit stochastic variations, leading to both processing noise and observation noise covariance matrices containing unknown and constantly changing colored noise. The observation noise covariance matrix is related to sensor accuracy and can be characterized through statistical analysis of sensor data. On the other hand, obtaining the processing noise covariance matrix is often challenging and time-consuming because it requires repeated comparative testing. Therefore, it is difficult to obtain optimal values for these noise matrices. To address this issue, this work proposes an enhanced genetic

algorithm to optimize the system processing noise and observation noise covariance matrices during the estimation process of the HEKF and aims to improve the accuracy of vehicle state parameter estimation.

5.1 Generation of the Population

The method of midpoint interpolation is adopted to generate the initial population.

Step 1: The mobile cart must pass through arbitrary grid cell in each row from the starting point to the endpoint. Randomly selected white grid cell in each row forms the cart's path, such as (1, 25, 46, 62, 83, 107, 125, 148, 164, 185). Refer to Figure 3 for illustration.

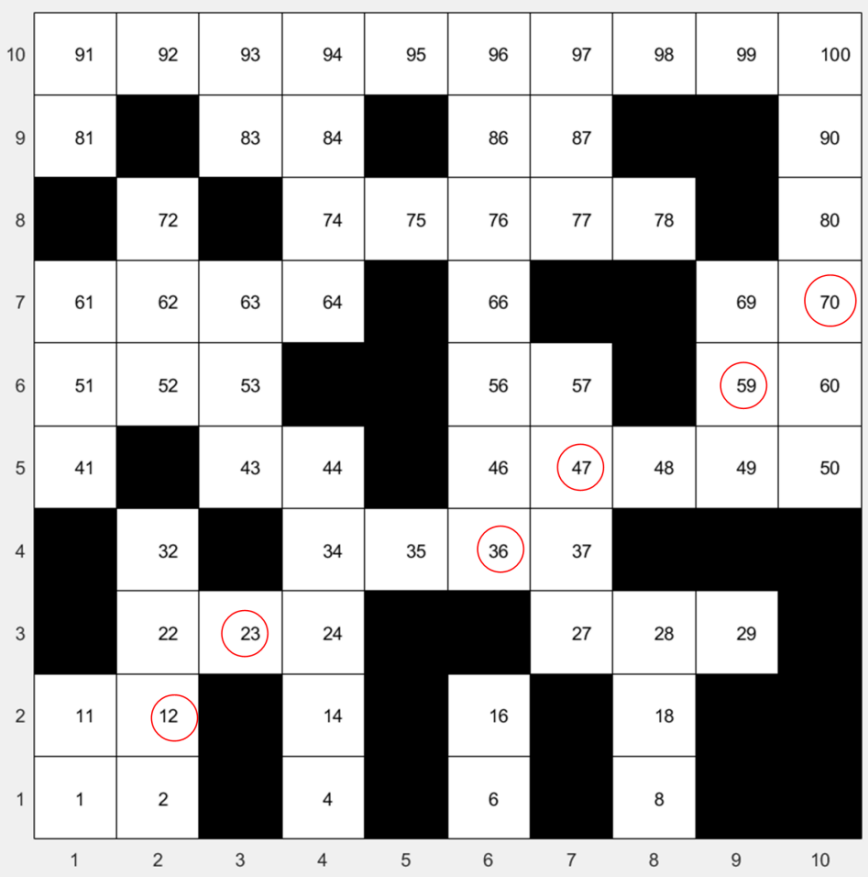


Figure 3 Path diagram before initial individual interpolation

Step 2: Determine whether adjacent path points are continuous. If the straight line connecting the central coordinates of adjacent path points passes through a black obstacle grid cell, then the two paths are not connected. Otherwise, they are connected.

$$\omega = \max\left\{abs\left(x_{i+1} - x_i\right), abs\left(y_{i+1} - y_i\right)\right\} \tag{45}$$

$$\mu = \begin{cases} 1 & \omega > 1 \\ 0 & \omega \le 1 \end{cases} \tag{46}$$

where x, y are the horizontal and vertical coordinates of the grid. If $\omega \le 1$, then $\mu = 0$. This indicates that the two points are continuous; otherwise, they are discontinuous and interpolation is needed.

Step 3: After the first two steps, interpolate between two non-adjacent points to make them continuous. The new grid coordinate points are generated as follows:

$$\begin{cases} x_n = \text{int}\left(\dfrac{x_{i+1} + x_i}{2}\right) \\ y_n = \text{int}\left(\dfrac{y_{i+1} + y_i}{2}\right) \end{cases} \tag{47}$$

where x_n, y_n is the coordinate value of the new grid.

Step 4: Repeat Steps 1-3 until a certain scale of the initial population is generated.

5.2 Selection of Operator

In this paper, a dynamic selection approach that integrates the benefits of elitist selection and roulette wheel selection is employed to enhance algorithmic performance. In the early stages of the algorithm iteration, when superior individuals are scarce, roulette wheel selection is used to increase population diversity. In the later stages, elitist selection is adopted to preserve high-quality individuals.

Step 1: Calculate the fitness value for each individual in the initial population, along with their probability of being selected.

$$P(x_i)=\frac{f(x_i)}{\sum_{j=1}^{N} f(x_j)} \tag{48}$$

Step 2: When the current iteration number is less than half of the total iteration count, generate a random number r between [0, 1] using a random function. If $r<P(x_i)$, select the individual for the next generation.

Step 3: When the current iteration number exceeds half of the total iteration count, compute the fitness values for each individual and sort them, selecting the top-ranked n individuals for the next generation.

Step 4: Repeat Step 2 and Step 3 to select a new population.

By adopting dynamic selection and combining the advantages of elitist selection with roulette wheel selection, the population is prevented from converging to local optima, increase the diversity of the population is increased, and the algorithmic efficiency is improved.

5.3 HEKF Noises Optimization

To optimize system noise q and observation noise r, define them as positive definite diagonal matrices respectively:

$$\begin{cases} q_i = d(q_1, q_2, q_3) \\ r_i = d(r_1, r_2) \end{cases} \tag{49}$$

For computational convenience, system noise and observation noise are combined into a single vector:

$$g=(q_1, q_2, q_3, r_1, r_2)^T \tag{50}$$

where g represents each individual and chromosome. The optimization objective is to minimize the mean square error between the observed true values and the estimated measurements. Therefore, a new fitness function needs to be established:

$$f=abs(x_h-\hat{x}_h) \tag{51}$$

where x_h denotes the true observation variable at time h, and $\hat{x}_h$ is the estimated output variable at time h.

6 Simulation Experiments and Results Analysis

In this section of the simulation experiment, the Matlab 2016a is used. The GA's initial parameters are as follows. Population Size: 300, Iteration Number: 100, Crossover Probability: [0.4, 0.8], Mutation Probability: [0.01, 0.1]. The wheelbase and speed are set to $L=4m, V=4m/s$, respectively. Figure 4 depicts the data generation experiment, where asterisk indicates landmarks totaling 60 in number, black dots represent navigation points with a total of 7, the black dashed line shows the true path, and black circles denote the cart's pose.

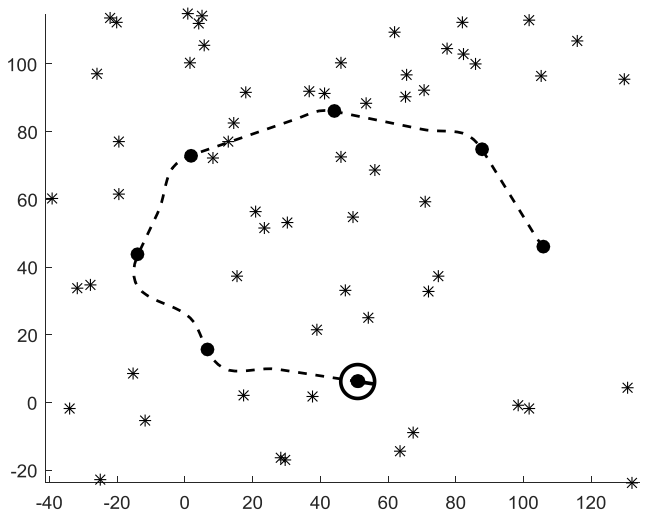


Figure 4 True Path Data Generation

Given initial noise: linear speed error of 1 m, steering angle error of 5°, observed relative distance error of 1 m, and observed relative angle error of 1°. As shown in Figure 5, under the influence of low noise, both EKF and HEKF method showing minimal error compared to the true path.

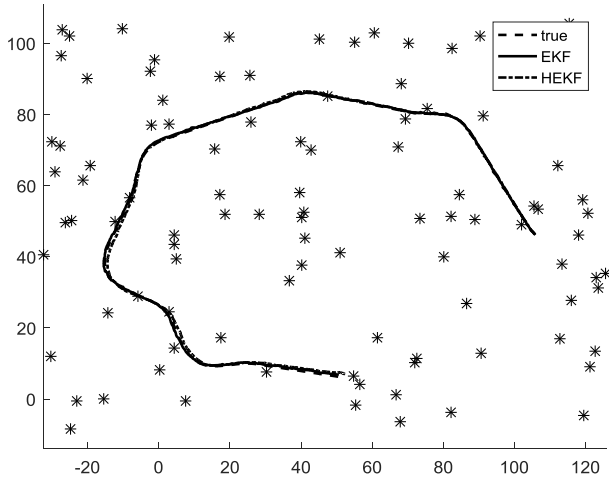


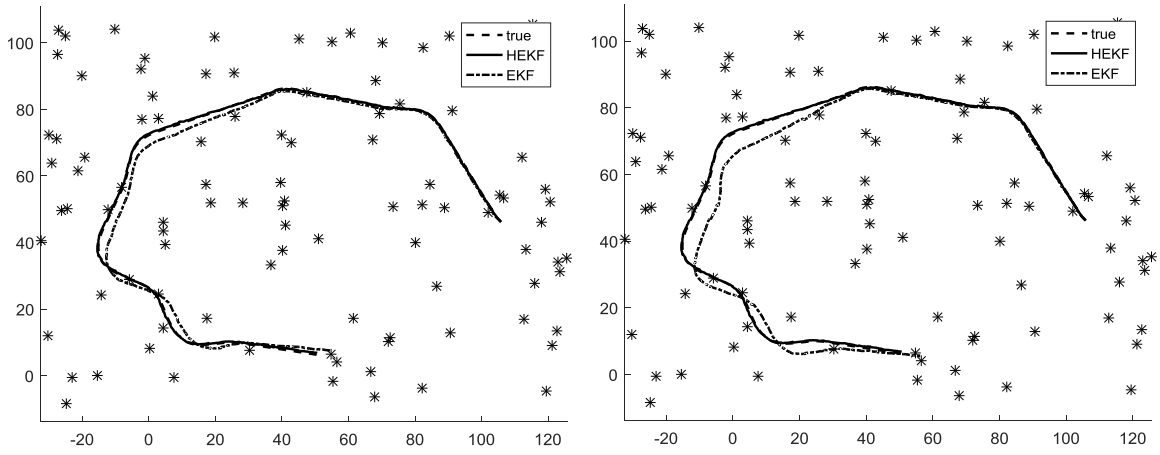
Figure 5 Path Estimation Performance of EKF and HEKF under initial noise

Given the Gaussian Gaussian white noise with a mean of 0 and variance of 2, and the colored noise as follow,

where x_i is a random variable in $[0,1]$ and $e_w(k)$ represents a white noise sequence.

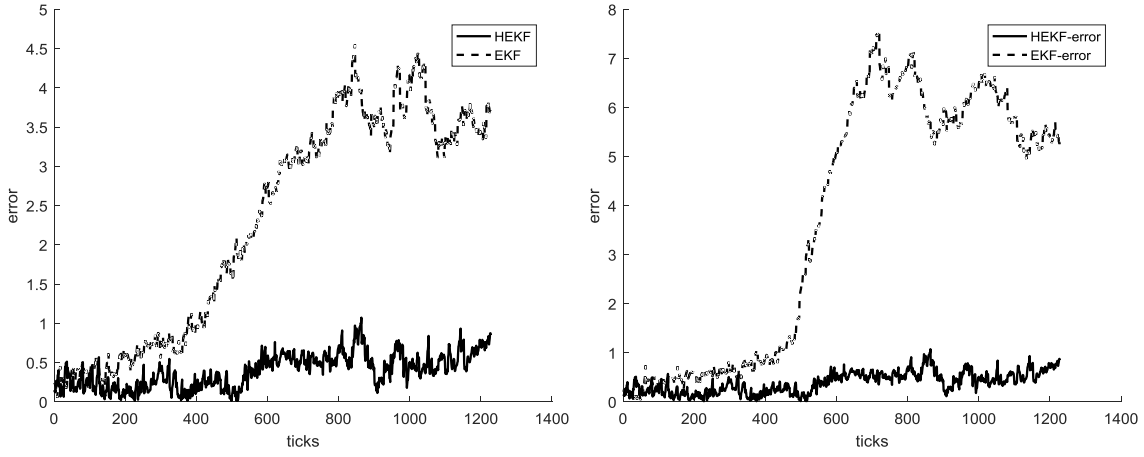
$$e_c(k)=[x_1,x_2,x_3]\begin{bmatrix} e_w(k) \\ e_w(k-1) \\ e_w(k-2) \end{bmatrix} \tag{52}$$

Figure 6 presents the estimated paths using EKF and HEKF method after adding Gaussian white noise and the colored noise. Figure 7 shows the overall error between the estimated and true paths, and Figure 8 displays the error of x, y, θ. As observed in Figure 6. The performance of EKF estimation method declines significantly for the introduction of Gaussian noise, resulting in larger errors from the true trajectory, as shown in Figure 6(b). Whereas HEKF estimation method exhibits smaller errors. Figures 7 and 8 demonstrate that the error of HEKF estimation method compared to EKF is reduced by nearly 80%.



(a) Gaussian Noise (b) Colored Noise

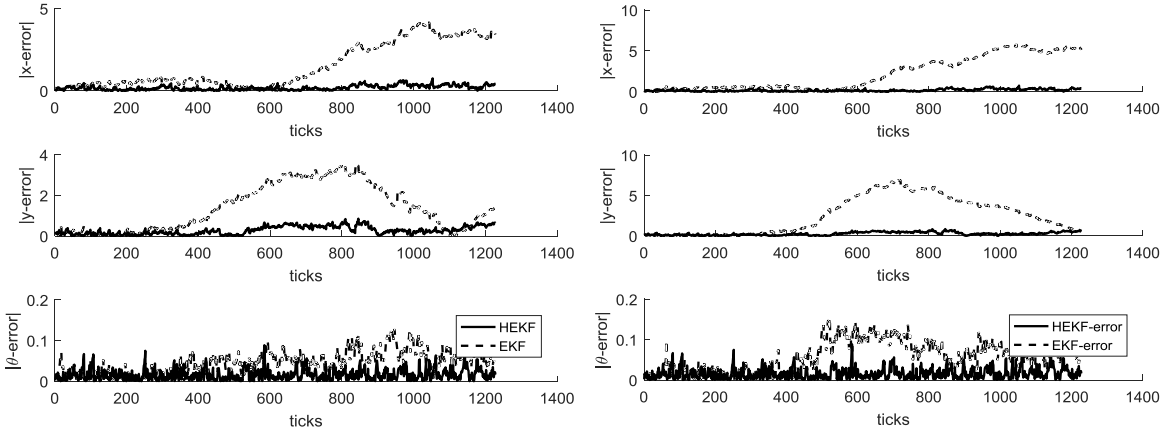
Figure 6 Path Estimation Performance of EKF and HEKF under Gaussian and Colored noise



(a) Gaussian Noise (b) Colored Noise

Figure 7 Path Estimation Trajectory Error of EKF and HEKF under Gaussian and Colored noise

The comparison of the proposed HEKF method, the Iterated Extended Kalman Filter (IEKF) and Extended Kalman Filter under the colored noise is shown in Figures 9-10 [17, 18]. It can be seen that the proposed method has better performance compared to the IEKF and EKF method under the colored noise.



(a) Gaussian Noise (b) Colored Noise

Figure 8 Directional x, y, θ Error of EKF and HEKF for Path Estimation under Gaussian and Colored noise

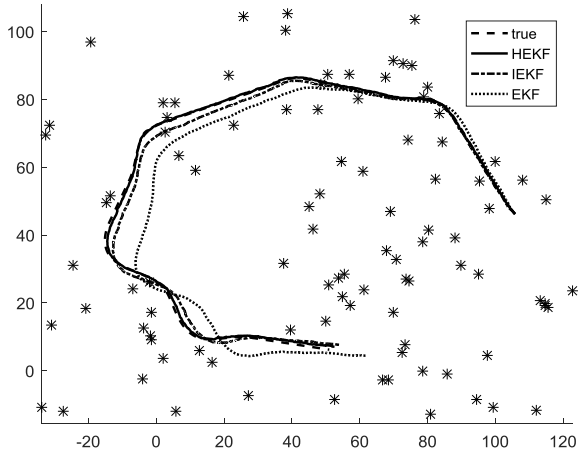
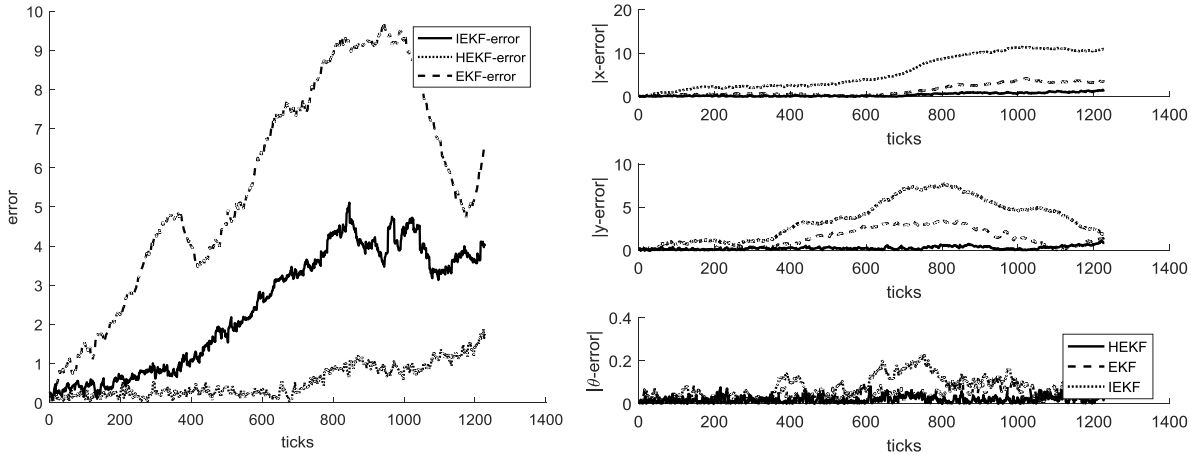


Figure 9 Path Estimation Performance of EKF, IEKF, and the proposed HEKF under colored noise



(a) Trajectory Error (b) Directional x, y, θ Error

Figure 10 Path Estimation Error of EKF, IEKF, and the proposed HEKF under colored noise

The comparison of the trajectory errors and directional errors of path estimation using EKF, IEKF, and proposed HEKF is shown in Table 2. The proposed HEKF algorithm avoids round-off errors of higher-order terms, demonstrating algorithmic performance superior to the standard EKF and IEKF, with smaller algorithmic errors.

Table 2 RMSE Comparison

RSME / Method	x	y	θ	Trajectory
EKF	6.6273	5.6551	0.2012	7.6003
IEKF	2.6273	1.6551	0.1043	2.2773
HEKF	0.6273	0.4551	0.0182	0.5889

7 Conclusion

This paper improves upon the Extended Kalman Filter (EKF) by proposing a cart motion estimation method based on the Higher-Order Extended Kalman Filter (HEKF). The proposed method effectively reduces the round-off errors associated with higher-order terms, offering significant improvements in precision and performance. By modeling the moving cart, nonlinear state functions are defined as hidden variables, which are then transformed into pseudo-linear forms to avoid the truncation errors inherent in traditional EKF. These hidden variables are modeled linearly, allowing the pseudo-linear model to be equivalently rewritten into a linear form. Additionally, the observation model is equivalently rewritten to align with the linearized system. To further enhance the algorithm's robustness, an improved genetic algorithm is employed to optimize the system noise parameters. Simulation results demonstrate that the proposed HEKF algorithm outperforms the traditional EKF method in terms of precision and noise immunity, making it a more reliable solution for motion estimation in nonlinear systems. The significance of this research lies in its potential to advance the field of motion estimation, particularly in applications requiring high accuracy and robustness, such as autonomous vehicles, robotics, and precision navigation systems. By addressing the limitations of EKF in handling nonlinearities and noise, the proposed HEKF method provides a foundation for developing more reliable and efficient estimation techniques. Future research should focus on extending this approach to more complex and dynamic systems, such as multi-agent systems or environments with rapidly changing conditions. Additionally, exploring the integration of machine learning techniques to further optimize the noise covariance matrices and hidden variable modeling could yield even greater improvements. Investigating the real-time implementation of the HEKF algorithm on embedded systems or hardware platforms would also be valuable, as it would demonstrate its practical applicability in real-world scenarios. These advancements could pave the way for broader adoption of the HEKF method in industries where precise motion estimation is critical.

Conflicts of Interests

The authors have no relevant financial or non-financial interests to disclose.

References

[1] Konieczny K. Technologia SLAM (Simultaneous Localization and Mapping) [J]. PRZEGLĄD GEODEZYJNY, 2023, (6): 47-52.

[2] Smith R. C., Cheeseman P. On the Representation and Estimation of Spatial Uncertainty [J]. The International Journal of Robotics Research, 1986, Vol. 5(4): 56-68.

[3] Montemerlo M., Sebastian Thrun, et al. FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem [C]. Proceedings of AAAI-2002, 2002.

[4] Zhao Y, Wang T, Qin W, et al. Improved Rao-blackwellised Particle Filter based on Randomly Weighted Particle Swarm Optimization [J]. Computers and Electrical Engineering, 2018, 71: 477-484.

[5] Daixian Z., Yinan M., Mingbo W., et al. LSO-FastSLAM: A New Algorithm to Improve the Accuracy of Localization and Mapping for Rescue Robots [J]. Sensors, 2022, Vol. 22(3): 16-22.

[6] Bailey T., Durrant-Whyte H.. Simultaneous Localisation and Mapping (SLAM): Part II State of the Art [J]. 2006(3): 15-22.

[7] Pfingsthorn M., Slamet B., Visser A.. A Scalable Hybrid Multi-Robot SLAM Method for Highly Detailed Maps [C]. Robocup: Robot Soccer World Cup XI, July, Atlanta, USA, 2008. 1-48.

[8] Mullane J., B. N. Vo, M. D. Adams. Rao-Blackwellised PHD SLAM [C]. IEEE International Conference on Robotics & Automation, 2010: 5410-5416.

[9] Fei Z., Zijing Z., Luxi Y.. A New PHD-SLAM Method based on Memory Attenuation Filter [J]. Measurement Science and Technology, 2021, Vol. 32(9): 102-108.

[10] Deusch H., Reuter S., Dietmayer K.. The Labeled Multi-Bernoulli SLAM Filter [J]. IEEE Signal Processing Letter. 2015, Vol. 22(10): 89-99.

[11] Zhou Z., Wang D., Xu B.. A Multi-innovation with Forgetting Factor based EKF-SLAM Method for Mobile Robots [J]. Assembly Automation, 2020, Vol. 41(1): 71-78.

[12] Shyam R., Sameer B., Daegyun C., et al. EKF-SLAM for Quadcopter Using Differential Flatness-Based LQR Control [J]. Electronics, 2023, Vol. 12(5): 1113-1113.

[13] Hamza M., Mohamed A., Mustapha R.. An Efficient End-to-end EKF-SLAM Architecture based on LiDAR, GNSS, and IMU Aata Sensor Fusion for Autonomous Ground Vehicles [J]. Multimedia Tools and Applications, 2023, Vol. 83(18): 56183-56206.

[14] R. Bucy, R. Kalman and I. Selin, Comment on "The Kalman Filter and Nonlinear Estimates of Multivariate Normal Processes" [J], IEEE Transactions on Automatic Control, Vol. 10(1): 33-38.

[15] Fariña B., Toledo J., Acosta L.. Sensor Fusion Algorithm Selection for an Autonomous Wheelchair Based on EKF/UKF Comparison [J]. International Journal of Mechanical Engineering and Robotics Research, 2023, Vol. 12(1): 112-121.

[16] Wen C. L., Cheng X. S., Xu D. X., et al. Filter design based on characteristic functions for one class of multi-dimensional nonlinear non-Gaussian systems [J]. Automatica, 2017(82): 171-180.

[17] Jagan, B. O. L., Rao, S. K. Evaluation of DB-IEKF Algorithm Using Optimization Methods for Underwater Passive Target Tracking [J]. Mobile Networks and Applications, 2022, Vol. 27(3): 1-11.

[18] J. Lin and F. Zhang. Loam livox: A fast, robust, high-precision LiDAR odometry and mapping package for LiDARs of small FoV [C]. 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 2020, pp. 3126-3131.