An ESKF Based SLAM Approach with Millimeter Wave Radar and IMU

Zhenchang Xu, Huquan Li, Yindong Zi, Shisheng Guo(✉), and Guolong Cui

University of Electronic Science and Technology of China, Chengdu 611731, China
ssguo@uestc.edu.cn

Abstract. In the last decades, numerous simultaneous localization and mapping (SLAM) methods based on different sensors have been proposed. However, it is still an open research problem to achieve robust SLAM in harsh environments. Compared to LiDAR and camera, millimeter wave radar can operate well under different weather and lack of light conditions. In this paper, an error state Kalman filter (ESKF) based SLAM algorithm is proposed. It utilizes a millimeter wave radar and an inertial measurement unit (IMU) to estimate the location of the unmanned vehicle and the mapping result of the surroundings. The unmanned vehicle state is estimated by the state prediction, registration, and the state update step, and the surrounding environment is constructed into a map through the map update step. The proposed algorithm is evaluated via experiment.

Keywords: SLAM · Millimeter wave radar · ESKF · IMU

1 Introduction

The objective of simultaneous localization and mapping (SLAM) technology is to estimate the ego motion of the vehicle while constructing a map of an unknown environment. Existing SLAM schemes are mainly based on LiDAR (LOAM [1], FAST-LIO2 [2]), vision (vins-mono [3], ORB-SLAM3 [4]) and multi-sensors fusion (LVI-SAM [5], R3LIVE [6]), etc. Although LiDAR can acquire high range resolution and angular resolution in most cases, it is sensitive to weather, such as rain and fog [7]. On the other hand, the camera is versatile and inexpensive, but it is susceptible to the changes in the scene, such as lack of light or snow.

Millimeter wave radar could detect the surrounding environment by emitting electromagnetic waves with 1–10 mm wavelength. The millimeter waves have a short wavelength and a wide frequency bandwidth. It can provide relatively high accuracy. In addition, the wavelength of millimeter wave is larger than that of visible light and laser, which can easily penetrate rain and fog. As a result, millimeter wave radar can provide a longer detection distance and more effective anti-interference capability. Besides the price of millimeter wave radar is lower than the LiDAR [8]. In recent years, millimeter wave radar has attracted

W. Fu et al. (Eds.): ICAUS 2022, LNEE 1010, pp. 2301–2310, 2023.
https://doi.org/10.1007/978-981-99-0479-2_214

more and more attention in the SLAM area because of its robustness in harsh environments [7–9]. However, millimeter wave radar has larger measurement errors and more false alarms compared to LiDAR.

In this paper, an ESKF based SLAM method is proposed with millimeter wave radar and IMU. The ESKF is utilized to fuse the measurements of radar and IMU to estimate the ego motion and construct the map of the surrounding environment. We build an experimental platform equipped millimeter wave radar and IMU, and conduct several experiments to validate the proposed method.

The remainder of this paper is organized in the following manner. In Sect. 2, the problem formulation is described. Then we present the ESKF based method to estimate platform motion and construct the map of the surrounding environment in Sect. 3. The experiment results are provided and discussed in Sect. 4. And Sect. 5 concludes this paper.

2 Problem Formulation

Assuming an unmanned vehicle is equipped with millimeter wave radar and IMU, which aims to retrieve the real-time location of the vehicle while constructing a map of the surroundings.

Millimeter wave radar transmits electromagnetic waves to illuminate the surrounding environment. Let $s(t)$ denote the baseband signal transmitted at each radar period and f_c is the carrier frequency. The echo received by the n-th receiver from the m-th transmitter at the k-th radar period is expressed as:

$$r_{mnk}(t) = \sum_{l=1}^{L} \sigma_{mnlk} s(t) \exp\left(-j2\pi f_c \tau_{mnlk}\right) + \phi_{mnk}(t), \tag{1}$$

where σ_{mnlk} is the scattering coefficient for the l-th scatter, τ_{mnlk} denotes the round trip time delay of the l-th scatter for the m-th transmitter and n-th receiver, $\phi_{mnk}(t)$ denotes the noises. After demodulation, sampling and beamforming [10], the point cloud generated by the millimeter wave radar at the k-th radar period could be written as $\mathcal{P}_k = \left\{\bar{\boldsymbol{p}}_{k,1}, ..., \bar{\boldsymbol{p}}_{k,N}\right\}$. Here the component $\bar{\boldsymbol{p}}_{k,j}$ is:

$$\bar{\boldsymbol{p}}_{k,j} = \boldsymbol{p}_{k,j} + \boldsymbol{n}_{k,j}, \tag{2}$$

where $\boldsymbol{p}_{k,j}$ is the true positon of the j-th scatter, and the number of points in the point cloud $\mathcal{P}_k$ is denoted as N.

The IMU is utilized to measure the angular velocity $\bar{\boldsymbol{\omega}}^I$ and the acceleration $\bar{\boldsymbol{a}}^I$ of the unmanned vehicle. $\bar{\boldsymbol{\omega}}^I$ and $\bar{\boldsymbol{a}}^I$ are affected by not only additive Gaussian white noise $\boldsymbol{n}_{\boldsymbol{\omega}}$ and $\boldsymbol{n}_{\boldsymbol{a}}$ but also slowly varying sensor bias $\boldsymbol{b}_{\boldsymbol{\omega}}$ and $\boldsymbol{b}_{\boldsymbol{a}}$, respectively:

$$\begin{aligned} \bar{\boldsymbol{\omega}}^I &= \boldsymbol{\omega}^I + \boldsymbol{b}_{\boldsymbol{\omega}} + \boldsymbol{n}_{\boldsymbol{\omega}} \\ \bar{\boldsymbol{a}}^I &= \boldsymbol{R}_I^{W,T}\left(\boldsymbol{a}^W - \boldsymbol{g}^W\right) + \boldsymbol{b}_{\boldsymbol{a}} + \boldsymbol{n}_a, \end{aligned} \tag{3}$$

where the superscript W, I, B denotes the world coordinate frame (the first radar coordinate frame), the IMU coordinate frame, and the radar coordinate

frame, respectively. $\boldsymbol{\omega}^I$ is the true angular velocity of the sensor and $\boldsymbol{a}^W$ and $\boldsymbol{g}^W$ denote the true acceleration of the sensor and the gravity in the world coordinate frame, respectively. In addition, $\boldsymbol{R}_I^W$ denotes the rotation matrix of the IMU coordinate frame represented in the world coordinate frame. To make it more clear to understand, the relationship among the mentioned three coordinate frames is shown in Fig. 1.

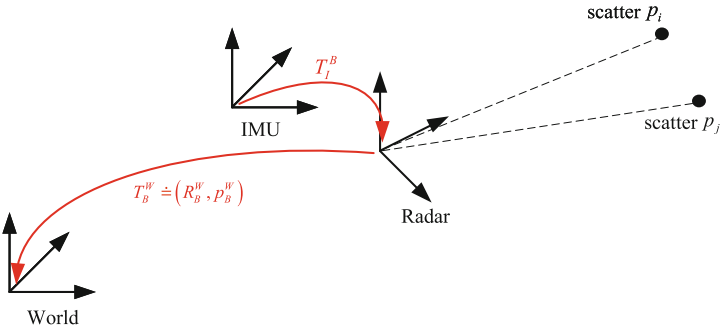


Fig. 1. The relation of three coordinate frames.

Assume that the radar is connected to the IMU with a known extrinsic $\boldsymbol{T}_I^B \doteq \left(\boldsymbol{R}_I^B, \boldsymbol{p}_I^B\right)$. The motion of the platform could be modeled by [11]:

$$\begin{aligned} \dot{\boldsymbol{p}}_B^W &= \boldsymbol{v}_B^W, \quad \dot{\boldsymbol{v}}_B^W = \boldsymbol{R}_B^W \boldsymbol{R}_I^B \left(\bar{\boldsymbol{a}}^I - \boldsymbol{b}_{\boldsymbol{a}} - \boldsymbol{n}_{\boldsymbol{a}}\right) + \boldsymbol{g}^W, \\ \dot{\boldsymbol{R}}_B^W &= \boldsymbol{R}_B^W \left\lfloor \boldsymbol{R}_I^B \left(\bar{\boldsymbol{\omega}}^I - \boldsymbol{b}_{\boldsymbol{\omega}} - \boldsymbol{n}_{\boldsymbol{\omega}}\right) \right\rfloor_\wedge, \quad \dot{\boldsymbol{b}_{\boldsymbol{a}}} = \boldsymbol{n}_{\boldsymbol{ba}}, \quad \dot{\boldsymbol{b}_{\boldsymbol{\omega}}} = \boldsymbol{n}_{\boldsymbol{b\omega}}, \end{aligned} \tag{4}$$

where $\lfloor \boldsymbol{a} \rfloor_\wedge$ denotes the skew symmetric matrix of vector $\boldsymbol{a} \in \mathbb{R}^3$. The sensor bias $\boldsymbol{b}_{\boldsymbol{\omega}}$ and $\boldsymbol{b}_{\boldsymbol{a}}$ are modeled as the random walk process with white noises $\boldsymbol{n}_{\boldsymbol{b\omega}}$ and $\boldsymbol{n}_{\boldsymbol{ba}}$.

The continuous kinematic model in (4) is discretized with the IMU sampling period Δt:

$$\boldsymbol{x}_{i+1} = \boldsymbol{x}_i \boxplus \left(\Delta t \boldsymbol{f}\left(\boldsymbol{x}_i, \boldsymbol{u}_i, \boldsymbol{w}_i\right)\right), \tag{5}$$

where i denotes the index of IMU measurements sampled between two consecutive radar periods at time $\tau_i \in [t_{k-1}, t_k]$. The state $\boldsymbol{x}$, function $\boldsymbol{f}$, input $\boldsymbol{u}$, and noise $\boldsymbol{w}$ are defined as:

$$\begin{aligned} \boldsymbol{x} &= \begin{bmatrix} \boldsymbol{p}_B^{W,T} & \boldsymbol{v}_B^{W,T} & \boldsymbol{R}_B^{W,T} & \boldsymbol{b}_{\boldsymbol{a}}{}^T & \boldsymbol{b}_{\boldsymbol{\omega}}{}^T \end{bmatrix}^T, \\ \boldsymbol{u} = \begin{bmatrix} \bar{\boldsymbol{a}}^{I,T} & \bar{\boldsymbol{\omega}}^{I,T} \end{bmatrix}^T, \boldsymbol{w} &= \begin{bmatrix} \boldsymbol{n}_{\boldsymbol{a}}{}^T & \boldsymbol{n}_{\boldsymbol{\omega}}{}^T & \boldsymbol{n}_{\boldsymbol{ba}}{}^T & \boldsymbol{n}_{\boldsymbol{b\omega}}{}^T \end{bmatrix}^T, \\ \boldsymbol{f}\left(\boldsymbol{x}_i, \boldsymbol{u}_i, \boldsymbol{w}_i\right) &= \begin{bmatrix} \boldsymbol{v}_{B,i}^W \\ \boldsymbol{R}_{B,i}^W \boldsymbol{R}_I^B \left(\bar{\boldsymbol{a}}_i^I - \boldsymbol{b}_{\boldsymbol{a}_i} - \boldsymbol{n}_{\boldsymbol{a}_i}\right) + \boldsymbol{g}^W \\ \boldsymbol{R}_I^B \left(\bar{\boldsymbol{\omega}}_i^I - \boldsymbol{b}_{\boldsymbol{\omega}_i} - \boldsymbol{n}_{\boldsymbol{\omega}_i}\right) \\ \boldsymbol{n}_{\boldsymbol{ba}_i} \\ \boldsymbol{n}_{\boldsymbol{b\omega}_i} \end{bmatrix}, \end{aligned} \tag{6}$$

where $\boldsymbol{p}_B^W$ and $\boldsymbol{v}_B^W$ are the position and velocity of the radar in the world frame, respectively. $\boldsymbol{R}_B^W \in SO(3)$ is the rotation matrix of the radar coordinate frame represented in the world frame.

Operator $\boxplus$ is utilized to establish a projective mapping from the local neighborhood on tangent space $\mathbb{R}^n$ to its manifold $\mathcal{M}$, which is defined as [2]:

$$\begin{bmatrix} \boldsymbol{R} \\ \boldsymbol{a} \end{bmatrix} \boxplus \begin{bmatrix} \boldsymbol{r} \\ \boldsymbol{b} \end{bmatrix} = \begin{bmatrix} \boldsymbol{R}\,\mathrm{Exp}\,(\boldsymbol{r}) \\ \boldsymbol{a}+\boldsymbol{b} \end{bmatrix}, \tag{7}$$

where $\boldsymbol{R} \in \mathcal{M} = SO(3)$, $\boldsymbol{r} \in \mathbb{R}^3$, $\boldsymbol{a}$, $\boldsymbol{b} \in \mathbb{R}^n$, $\mathrm{Exp}\,(\cdot)$ denotes the Rodrigues's transformation from the rotation vector to the rotation matrix [12].

The location of the unmanned vehicle and the map of the surrounding environment could be jointly estimated by calculating the state $\boldsymbol{x}$ based on the measurements of radar and IMU, and subsequently transforming the radar point cloud to the world coordinate frame.

3 Methodology

In the following part, an ESKF-based method is proposed to realize the vehicle ego motion estimation and map construction simultaneously. The flowchart of our method is shown in Fig. 2. The state of the vehicle is predicted continuously when IMU measurements arrive. Once the radar point cloud arrives, the position and rotation of the radar are estimated by registration and state update. Finally, the radar point cloud is converted to the world coordinate frame and then merged into map.

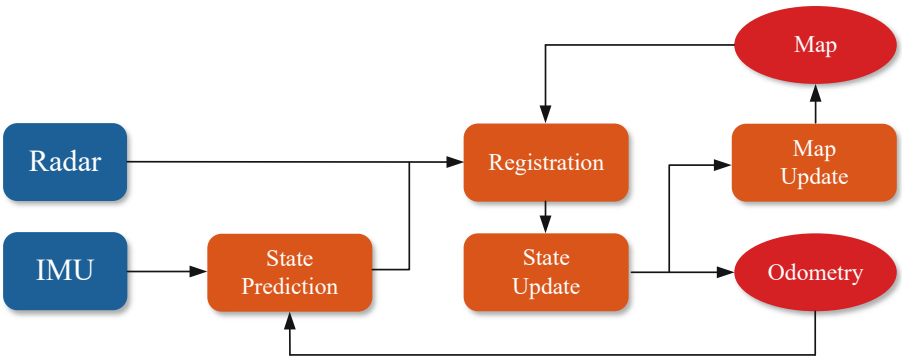


Fig. 2. Flowchart of the proposed algorithm.

3.1 State Prediction

Assume that the state at the $k-1$-th radar period t_{k-1} is $\check{\boldsymbol{x}}_{k-1}$ with covariance matrix $\check{\boldsymbol{P}}_{k-1}$, and $\boldsymbol{x}_{k-1}$ is the true state of $\check{\boldsymbol{x}}_{k-1}$. The error state $\tilde{\boldsymbol{x}}_i$ is written as:

$$\tilde{\boldsymbol{x}}_k = \boldsymbol{x}_k \boxminus \check{\boldsymbol{x}}_k = \begin{bmatrix} \tilde{\boldsymbol{p}}_B^{W,T} & \tilde{\boldsymbol{v}}_B^{W,T} & \delta\boldsymbol{\theta}_B^{W,T} & \tilde{\boldsymbol{b}}_{\boldsymbol{a}}^{W,T} & \tilde{\boldsymbol{b}}_{\boldsymbol{\omega}}^{W,T} \end{bmatrix}^T, \tag{8}$$

where $\delta\boldsymbol{\theta}_B^{W,T} = \mathrm{Log}\left(\hat{\boldsymbol{R}}_B^{W,T}\boldsymbol{R}_B^W\right)$ is the attitude error. $\tilde{\boldsymbol{p}}_B^W$, $\tilde{\boldsymbol{v}}_B^W$, $\tilde{\boldsymbol{b}}_{\boldsymbol{a}}^W$, $\tilde{\boldsymbol{b}}_{\boldsymbol{\omega}}^W$ are the error of position, velocity, accelerometer bias, and gyroscope bias, respectively, which are standard additive errors ($\tilde{\boldsymbol{x}} = \boldsymbol{x} - \check{\boldsymbol{x}}$). Operator $\boxminus$ is utilized to establish a projective mapping from manifold $\mathcal{M}$ to its tangent space $\mathbb{R}^n$ [2]:

$$\begin{bmatrix}\boldsymbol{R}_1\\ \boldsymbol{a}\end{bmatrix} \boxminus \begin{bmatrix}\boldsymbol{R}_2\\ \boldsymbol{b}\end{bmatrix} = \begin{bmatrix}\mathrm{Log}\left(\boldsymbol{R}_2^T\boldsymbol{R}_1\right)\\ \boldsymbol{a}-\boldsymbol{b}\end{bmatrix}, \tag{9}$$

where $\boldsymbol{R}_1$, $\boldsymbol{R}_2 \in \mathcal{M} = SO(3)$, $\boldsymbol{a}$, $\boldsymbol{b} \in \mathbb{R}^n$, and $\mathrm{Log}(\cdot)$ denotes the Rodrigues's transformation from the rotation matrix to the rotation vector [12].

Once the IMU measurement is received, the state is predicted as:

$$\hat{\boldsymbol{x}}_{i+1} = \hat{\boldsymbol{x}}_i \boxplus (\Delta t \boldsymbol{f}(\hat{\boldsymbol{x}}_i, \boldsymbol{u}_i, \boldsymbol{0}))\,; \quad \hat{\boldsymbol{x}}_0 = \check{\boldsymbol{x}}_{k-1}. \tag{10}$$

The error state dynamic model can be derived as:

$$\begin{aligned}\tilde{\boldsymbol{x}}_{i+1} &= \boldsymbol{x}_{i+1} \boxminus \hat{\boldsymbol{x}}_{i+1}\\ &= (\boldsymbol{x}_i \boxplus \Delta t \boldsymbol{f}(\boldsymbol{x}_i, \boldsymbol{u}_i, \boldsymbol{w}_i)) \boxminus (\hat{\boldsymbol{x}}_i \boxplus \Delta t \boldsymbol{f}(\hat{\boldsymbol{x}}_i, \boldsymbol{u}_i, \boldsymbol{0}))\\ &\simeq \boldsymbol{F}_{\tilde{\boldsymbol{x}}}\tilde{\boldsymbol{x}}_i + \boldsymbol{F}_{\boldsymbol{w}}\boldsymbol{w}_i,\end{aligned} \tag{11}$$

where the matrixes $\boldsymbol{F}_{\tilde{\boldsymbol{x}}}$ and $\boldsymbol{F}_{\boldsymbol{w}}$ are defined as:

$$\begin{aligned}\boldsymbol{F}_{\tilde{\boldsymbol{x}}} &= \left.\frac{\partial(\boldsymbol{x}_{i+1} \boxminus \tilde{\boldsymbol{x}}_{i+1})}{\partial \tilde{\boldsymbol{x}}_i}\right|_{\tilde{\boldsymbol{x}}_i=0, \boldsymbol{w}_i=0}\\ &= \begin{bmatrix}\boldsymbol{I}_3 & \boldsymbol{I}_3\Delta t & \boldsymbol{0}_3 & \boldsymbol{0}_3 & \boldsymbol{0}_3\\ \boldsymbol{0}_3 & \boldsymbol{I}_3 & -\hat{\boldsymbol{R}}_{B,i}^W\left\lfloor\hat{\boldsymbol{a}}_i^B\right\rfloor_\wedge \Delta t & -\hat{\boldsymbol{R}}_{B,i}^W\boldsymbol{R}_I^B\Delta t & \boldsymbol{0}_3\\ \boldsymbol{0}_3 & \boldsymbol{0}_3 & \mathrm{Exp}\left(-\hat{\boldsymbol{\omega}}_i^B\Delta t\right) & \boldsymbol{0}_3 & -\mathrm{J_r}\left(\hat{\boldsymbol{\omega}}_i^B\Delta t\right)\Delta t\\ \boldsymbol{0}_3 & \boldsymbol{0}_3 & \boldsymbol{0}_3 & \boldsymbol{I}_3 & \boldsymbol{0}_3\\ \boldsymbol{0}_3 & \boldsymbol{0}_3 & \boldsymbol{0}_3 & \boldsymbol{0}_3 & \boldsymbol{I}_3\end{bmatrix},\\ \boldsymbol{F}_{\boldsymbol{w}} &= \left.\frac{\partial(\boldsymbol{x}_{i+1} \boxminus \tilde{\boldsymbol{x}}_{i+1})}{\partial \boldsymbol{w}_i}\right|_{\tilde{\boldsymbol{x}}_i=0, \boldsymbol{w}_i=0}\\ &= \begin{bmatrix}\boldsymbol{0}_3 & \boldsymbol{0}_3 & \boldsymbol{0}_3 & \boldsymbol{0}_3\\ -\hat{\boldsymbol{R}}_{B,i}^W\boldsymbol{R}_I^B\Delta t & \boldsymbol{0}_3 & \boldsymbol{0}_3 & \boldsymbol{0}_3\\ \boldsymbol{0}_3 & -\mathrm{J_r}\left(\hat{\boldsymbol{\omega}}_i^I\Delta t\right) & \boldsymbol{0}_3 & \boldsymbol{0}_3\\ \boldsymbol{0}_3 & \boldsymbol{0}_3 & \boldsymbol{I}_3\Delta t & \boldsymbol{0}_3\\ \boldsymbol{0}_3 & \boldsymbol{0}_3 & \boldsymbol{0}_3 & \boldsymbol{I}_3\Delta t\end{bmatrix}.\end{aligned} \tag{12}$$

where $\hat{\boldsymbol{a}}_i^B = \boldsymbol{R}_I^B\left(\bar{\boldsymbol{a}}_i^I - \hat{\boldsymbol{b}}_{\boldsymbol{a}_i}\right)$, $\hat{\boldsymbol{\omega}}_i^B = \boldsymbol{R}_I^B\left(\bar{\boldsymbol{\omega}}_i^I - \hat{\boldsymbol{b}}_{\boldsymbol{\omega}_i}\right)$, $\boldsymbol{I}_3$ and $\boldsymbol{0}_3$ are 3×3 identity matrix and 3×3 zero matrix, respectively. $\mathrm{J_r}(\boldsymbol{\phi})$ is defined as:

$$\mathrm{J_r}(\boldsymbol{\phi}) = \frac{\sin\phi}{\phi}\boldsymbol{I}_3 + \left(1 - \frac{\sin\phi}{\phi}\right)\boldsymbol{a}\boldsymbol{a}^T - \frac{1-\cos\phi}{\phi}\lfloor\boldsymbol{a}\rfloor_\wedge, \tag{13}$$

where $\phi = \|\boldsymbol{\phi}\|$ ($\|\boldsymbol{\phi}\|$ is the norm of $\boldsymbol{\phi}$) and $\boldsymbol{a} = \boldsymbol{\phi}/\phi$.

The covariance matrix of the state $\hat{\boldsymbol{x}}_{i+1}$ can be computed by:

$$\hat{\boldsymbol{P}}_{i+1} = \boldsymbol{F}_{\tilde{\boldsymbol{x}}} \hat{\boldsymbol{P}}_i \boldsymbol{F}_{\tilde{\boldsymbol{x}}}^T + \boldsymbol{F}_{\boldsymbol{\omega}} \boldsymbol{Q} \boldsymbol{F}_{\boldsymbol{\omega}}^T; \quad \hat{\boldsymbol{P}}_0 = \hat{\boldsymbol{P}}_{k-1}, \tag{14}$$

where $\boldsymbol{Q}$ is the covariance matrix of the Gaussian white noises $\boldsymbol{w}$.

The prediction continues until a new period at t_k of radar point cloud reaches. The estimation of the state and the covariance matrix at t_k are denoted as $\hat{\boldsymbol{x}}_k$ and $\hat{\boldsymbol{P}}_k$, respectively.

3.2 Registration

The registration step utilizes the point-to-point iterative closest point (ICP) algorithm to calculate the relative transformation between two radar point clouds [13]. The kernel of ICP is aligning two point clouds by iteratively minimizing the Euclidean distance between the corresponding points.

Assume that a radar point cloud $\mathcal{P}^B = \left\{\bar{\boldsymbol{p}}_{k,1}, ..., \bar{\boldsymbol{p}}_{k,m}\right\}$ is aligned with the point cloud of map $\mathcal{Q} = \{\bar{\boldsymbol{q}}_1, ..., \bar{\boldsymbol{q}}_n\}$. Let the pose $\hat{\boldsymbol{p}}_{B,k}^W$ and the rotation matrix $\hat{\boldsymbol{R}}_{B,k}^W$ in the predicted state $\hat{\boldsymbol{x}}_k$ be the initial guess in the ICP algorithm. The estimation of $\boldsymbol{p}_{z,k}$ and $\boldsymbol{R}_{z,k}$ are calculated by iteratively minimizing:

$$\boldsymbol{p}_{z,k}, \boldsymbol{R}_{z,k} = \underset{\boldsymbol{p}_z, \boldsymbol{R}_z}{\arg\min} \sum_{(i,j)} \left\| \boldsymbol{R}_z \bar{\boldsymbol{p}}_{k,i} + \boldsymbol{p}_z - \bar{\boldsymbol{q}}_j \right\|^2, \tag{15}$$

where (i, j) denotes that the nearest point of $\boldsymbol{R}_z \bar{\boldsymbol{p}}_{k,i} + \boldsymbol{p}_z$ in $\mathcal{Q}$ is $\bar{\boldsymbol{q}}_j$.

3.3 State Update

Note that the estimation of state $\boldsymbol{x}_k$ at time t_k is $\hat{\boldsymbol{x}}_k$. The measurement $\boldsymbol{z}_k = \left[\boldsymbol{p}_{z,k}^T, \boldsymbol{R}_{z,k}^T\right]^T$ is calculated in the registration step when the radar point cloud arrives. The measurement $\boldsymbol{z}_k$ could be denoted as:

$$\begin{aligned} \boldsymbol{z}_k &= \boldsymbol{h}\left(\boldsymbol{x}_k\right) \boxplus \boldsymbol{n}_z \\ &\simeq \boldsymbol{h}\left(\hat{\boldsymbol{x}}_k\right) \boxplus \boldsymbol{H}_k \tilde{\boldsymbol{x}}_k \boxplus \boldsymbol{n}_z \\ &= \hat{\boldsymbol{z}}_k \boxplus \boldsymbol{H}_k \tilde{\boldsymbol{x}}_k \boxplus \boldsymbol{n}_z, \end{aligned} \tag{16}$$

where $\hat{\boldsymbol{z}}_k$ is the prediction of the measurement, $\boldsymbol{n}_z$ is the Gaussian white noise. $\boldsymbol{H}_k$ is the Jacobin matrix of $\boldsymbol{h}\left(\boldsymbol{x}_k\right)$, defined as:

$$\begin{aligned} \boldsymbol{H}_k &= \frac{\partial \boldsymbol{h}\left(\boldsymbol{x}_k\right)}{\partial \boldsymbol{x}_k} \frac{\partial \boldsymbol{x}_k}{\partial \tilde{\boldsymbol{x}}_k} \\ &= \begin{bmatrix} \boldsymbol{I}_3 & \boldsymbol{0}_3 & \boldsymbol{0}_3 & \boldsymbol{0}_3 & \boldsymbol{0}_3 \\ \boldsymbol{0}_3 & \boldsymbol{0}_3 & \mathrm{J_r^{-1}}\left(\mathrm{Log}\left(\boldsymbol{R}_B^W\right)\right) & \boldsymbol{0}_3 & \boldsymbol{0}_3 \end{bmatrix}. \end{aligned} \tag{17}$$

The state is updated by the ESKF through the following equations [14]:

$$\begin{aligned} \boldsymbol{K} &= \hat{\boldsymbol{P}}_k \boldsymbol{H}^T \left(\boldsymbol{H} \hat{\boldsymbol{P}}_k \boldsymbol{H}^T + \boldsymbol{N} \right)^{-1}, \\ \check{\boldsymbol{x}}_k &= \hat{\boldsymbol{x}}_k \boxplus \left(\boldsymbol{K} \left(\boldsymbol{z}_k \boxminus \hat{\boldsymbol{z}}_k \right) \right), \\ \check{\boldsymbol{P}}_k &= \left(\boldsymbol{I}_{15} - \boldsymbol{K} \boldsymbol{H} \right) \hat{\boldsymbol{P}}_k, \end{aligned} \tag{18}$$

where $\boldsymbol{N}$ is the covariance matrix of $\boldsymbol{n}_z$ and $\boldsymbol{I}_{15}$ is 15×15 identity matrix.

3.4 Map Update

The map update step aims to transform the radar point clouds to the world frame and merge them to the map. The radar point cloud $\mathcal{P}_k$ is projected from the radar coordinate frame at time t_k to the world coordinate frame with the state update $\check{\boldsymbol{x}}_k$ via:

$$\boldsymbol{p}'_{k,j} = \check{\boldsymbol{R}}^W_{B,k} \bar{\boldsymbol{p}}_{k,j} + \check{\boldsymbol{p}}^W_B; \quad \bar{\boldsymbol{p}}_{k,j} \in \mathcal{P}_k. \tag{19}$$

As the unmanned vehicle moves, repeat the above process and append the radar point cloud to the existing map. Gradually, the map of the surrounding environment is obtained through step-by-step iterative processing.

4 Experiments

4.1 Experimental Platform

The experiment is carried out using a wheeled robot, as shown in Fig. 3. A LiDAR (Livox Horizon) is mounted at the front of the wheeled robot, which is used to provide reference ground truth to verify our proposed algorithm. A 77 GHz millimeter wave radar (Texas Instruments AWR2944EVM) is equipped at the top of the vehicle. The parameters of the radar are shown in Table 1. An IMU (HFI-A9) is mounted next to the millimeter wave radar. The specification of the IMU is given in Table 2.

Table 1. Radar specification.

Parameter	Specifications
Maximum range (m)	200
Range resolution (m)	1.3
Maximum velocity (m/s)	±38.89
Velocity resolution (m/s)	0.1
Azimuth angular resolution (Deg)	9.5
Frame rate (Hz)	4

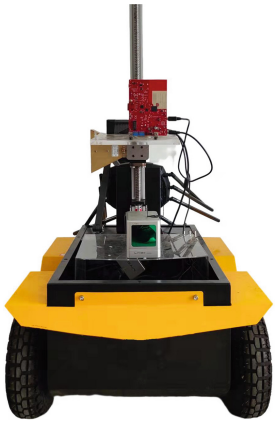


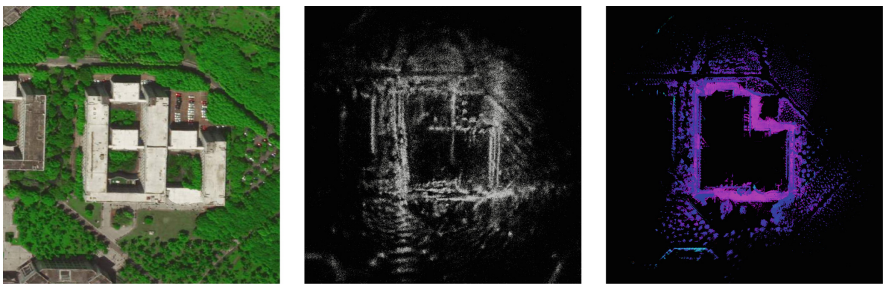
Fig. 3. Experimental platform.

Table 2. IMU specification.

Parameter	Specifications
Gyroscope range (Deg/s)	±2000
Acceleration range (g)	±8
Angle accuracy (Deg)	0.5
Frame rate (Hz)	150

Table 3. Absolute pose error results

Absolute pose error	ICP+IMU	Proposed method
Translation (m)	16.6869	8.5217
Rotation (rad)	0.3344	0.1691



(a) (b) (c)

Fig. 4. Mapping results. (a) Satellite map. (b) Mapping results of the proposed method. (c) Mapping results of FAST-LIO2.

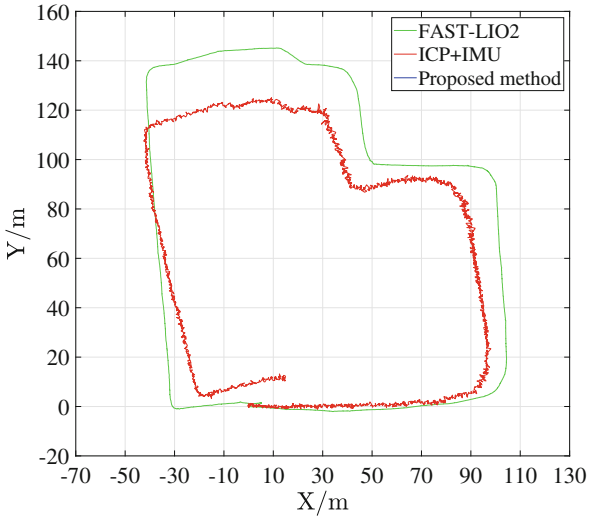


Fig. 5. Trajectories estimated by different algorithms.

4.2 Validation

To verify the effectiveness of the proposed method, we simultaneously collected the data of LiDAR, millimeter wave radar, and IMU data by manually controlling the wheeled robot moving around a building in University of Electronic Science and Technology of China. The experimental scene is shown in Fig. 4(a).

The mapping results of the proposed method and FAST-LIO2 are depicted in Fig. 4(b) and 4(c), respectively. It shows that the proposed method can generate a similar mapping result compared with FAST-LIO2.

The trajectories estimated by different algorithms are presented in Fig. 5. FAST-LIO2 is the reference trajectory generated by FAST-LIO2 [2] using LiDAR and IMU. ICP+IMU is the trajectory estimated by using ICP method which employing IMU to provide an initial rotation matrix guess. From the Fig. 5, we can find that the trajectory of the proposed method is close to the trajectory generated by FAST-LIO2, while the trajectory of ICP+IMU has larger error.

To quantitatively evaluate the performance of the proposed method, we compute the root mean square error (RMSE) of the absolute position error (APE) for the proposed method and ICP+IMU. As shown in Table 3, the proposed method achieves higher translation and rotation accuracies compared with ICP+IMU.

5 Conclusion

In this paper, an ESKF-based SLAM algorithm with millimeter wave radar and IMU is proposed to simultaneously calculate the location of the unmanned vehicle and the mapping of the surrounding environment, where the ESKF is utilized

to fuse the measurements of radar and IMU. The IMU measurements are used to predict state. Subsequently, the predicted state is fed into the registration step as the initial guess to calculate the position and rotation of radar. Finally, the state is corrected after the position and rotation matrix of the radar are calculated, and the map of the surrounding environment is obtained. Experimental result shows that our algorithm can provide reliable location and mapping result.

References

1. Zhang, J., Singh, S.: LOAM: lidar odometry and mapping in real-time. Robot. Sci. Syst. **2**(9), 1–9 (2014)
2. Xu, W., Zhang, F.: FAST-LIO: a fast, robust LiDAR-inertial odometry package by tightly-coupled iterated Kalman filter. IEEE Robot. Autom. Lett. **6**(2), 3317–3324 (2021)
3. Qin, T., Li, P., Shen, S.: VINS-mono: a robust and versatile monocular visual-inertial state estimator. IEEE Trans. Rob. **34**(4), 1004–1020 (2018)
4. Campos, C., Elvira, R., Rodríguez, J.J.G., Montiel, J.M.M., Tardós, J.D.: ORB-SLAM3: an accurate open-source library for visual, visual-inertial, and multimap SLAM. IEEE Trans. Rob. **37**(6), 1874–1890 (2021)
5. Shan, T., Englot, B., Ratti, C., Rus, D.: LVI-SAM: tightly-coupled lidar-visual-inertial odometry via smoothing and mapping. In: IEEE International Conference on Robotics and Automation (ICRA), pp. 5692–5698 (2021)
6. Lin, J., Zhang, F.: R3LIVE: A Robust, Real-time, RGB-colored, LiDAR-Inertial-Visual tightly-coupled state Estimation and mapping package. arXiv, arXiv:2109.07982 (2021)
7. Hong, Z., Petillot, Y., Wang, S.: RadarSLAM: radar based large-scale SLAM in all weathers. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 5164–5170 (2020)
8. Cen, S.H., Newman, P.: Precise ego-motion estimation with millimeter-wave radar under diverse and challenging conditions. In: IEEE International Conference on Robotics and Automation (ICRA), pp. 6045–6052 (2018)
9. Holder, M., Hellwig, S., Winner, H.: Real-time pose graph SLAM based on radar. In: IEEE Intelligent Vehicles Symposium (IV), pp. 1145–1151 (2019)
10. Lim, T.-Y., Markowitz, S.A., Do, M.N.: RaDICaL: a synchronized FMCW radar, depth, IMU and RGB camera data dataset with low-level FMCW radar signals. IEEE J. Sel. Top. Signal Process. **15**(4), 941–953 (2021)
11. Farrell, J.A.: Aided Navigation: GPS with High Rate Sensors. McGraw-Hill, New York (2008)
12. Hertzberg, C., Wagner, R., Frese, U., Schröder, L.: Integrating generic sensor fusion algorithms with sound state representations through encapsulation of manifolds. Inf. Fusion **14**(1), 57–77 (2013)
13. Arun, K.S., Huang, T.S., Blostein, S.D.: Least-squares fitting of two 3-D point sets. IEEE Trans. Pattern Anal. Mach. Intell. **9**(5), 698–700 (1987)
14. Zhen, W., Zeng, S., Soberer, S.: Robust localization and localizability estimation with a rotating laser scanner. In: IEEE International Conference on Robotics and Automation (ICRA), pp. 6240–6245 (2017)