

## 激光惯导紧耦合的户外长距离 SLAM 算法研究

孙 渊 陈志金 董昊轩

(上海电机学院机械学院 上海 201306)

**摘 要:** 针对大多数 SLAM 算法在户外长距离环境下轨迹误差大、建图漂移问题,提出一种基于 IEKF 的激光雷达与 IMU 紧耦合的 SLAM 算法,并构建全局一致的激光三维点云地图。首先,构建 IMU 状态模型并通过前向传播预估状态,运用反向传播对点云进行运动补偿,然后采用迭代扩展卡尔曼滤波融合 IMU 数据与雷达数据,得到前端激光里程计;引入回环检测模块,在点云中构建三角描述符,对三角描述符的边进行匹配以实现闭环检测;最后在后端优化部分采用 GTSAM 构建因子图,融合 IMU 预积分因子、里程计因子、回环检测因子,消除累积误差,提高定位精度,降低地图漂移。实验表明,所提算法相较于 FAST-LIO2 算法在 KITTI 数据集与自采集数据集中的 APE RMSE 分别平均下降了 50.06%、33.65%,降低了  $z$  轴上的漂移,能够构建闭合的稠密点云地图。

**关键词:** 激光 SLAM; 迭代扩展卡尔曼滤波; 回环检测; 因子图优化; 点云地图

**中图分类号:** TP242.3; TN98      **文献标识码:** A      **国家标准学科分类代码:** 510.20

Research on outdoor long range SLAM algorithm with laser  
inertial guidance tight coupling

Sun Yuan Chen Zhijin Dong Haoxuan

(School of Mechanical Engineering, Shanghai DianJi University, Shanghai 201306, China)

**Abstract:** Aiming at the problem of large trajectory error and map drift of most slam algorithms in outdoor long-distance environment, a SLAM algorithm based on IEKF tightly coupled lidar and IMU is proposed, and a globally consistent laser 3D point cloud map is constructed. Firstly, the IMU state model is constructed and the state is estimated by forward propagation, and the back propagation is used to compensate the motion of the point cloud, and then the IMU data and radar data are fused by iterative extended Kalman filter to obtain the front-end laser odometer; the loop back detection module is introduced to construct the triangle descriptor in the point cloud and match the edges of the triangle descriptor to achieve closed-loop detection; finally, in the back-end optimization part, GTSAM is used to build a factor map, which integrates IMU pre integration factor, odometer factor and loop detection factor to eliminate cumulative errors, improve positioning accuracy and reduce map drift. Experiments show that compared with FAST-LIO2 algorithm, the APE RMSE of the proposed algorithm in KITTI data set and self collected data set is reduced by 50.06% and 33.65%, respectively, and the drift on the  $z$  axis is reduced, which can build a closed dense point cloud map.

**Keywords:** laser SLAM; iterative extended Kalman filter; loopback detection; factor map optimization; point cloud mapping

## 0 引 言

在未知环境中实时、准确构建稠密的三维地图,同时在地图中进行定位对于移动机器人进行安全导航至关重要<sup>[1-2]</sup>。同步定位与建图(simultaneous localization and mapping, SLAM)技术能够在未知环境中实时计算机器人的 6 个自由度并定位建图。通常根据机器人搭载的传感器

不同,SLAM 技术可以大致的分为视觉 SLAM<sup>[3-4]</sup>与激光 SLAM。基于相机的视觉 SLAM 虽然提供了丰富的色彩信息,但对周围环境的纹理敏感,容易因为光照变化而产生严重的运动模糊,同时依靠机器人自身的计算资源,难以满足图像下构建稠密地图的计算需求。而基于激光的 SLAM 可以依靠激光雷达通过精确的距离测量直接获取周围环境的详细三维结构信息,受光照影响更小,能够更好的适应环

境变化<sup>[5]</sup>。

基于单一雷达传感器的 LOAM<sup>[6]</sup>算法通过提取特征点实现了激光里程计估计,但是该算法无法去除运动过程中产生的点云畸变,并且当移动机器人处于缺少强特征的场景中时,例如长直隧道、走廊以及空旷场景等多种退化场景时,其准确性和鲁棒性存在显著不足。对此,为了提高 SLAM 算法的精度与鲁棒性,近年来一些激光雷达里程计通常会与惯性测量单元(inertial measurement unit, IMU)进行融合。将高频率的 IMU 与激光雷达里程计进行融合,去除雷达点云畸变,提供一个良好的初值使算法避免陷入局部最小值。雷达与 IMU 的融合方式分为松耦合与紧耦合两种,松耦合方式根据雷达与 IMU 的观测数据分别进行自身的运动估计,然后将估计的位姿结果进行融合。例如 Shan 等<sup>[7]</sup>在 LOAM 的基础上提出的 LeGO-LOAM 算法,该算法引入了点云聚类与地面分割的思想以对数据进行处理,加快了点云的配准速度。但是 LeGO-LOAM 的闭环检测效果较差,且在环境复杂、长距离运行的情况下容易失效。同时松耦合的方式虽然能够降低计算负载,但是在无特征的环境下可能在某些方向上导致退化。而紧耦合的方式则将雷达与 IMU 的观测数据融合在一起,共同构建运动方程和观测方程,然后进行状态估计。例如, LINS<sup>[8]</sup>采用了基于误差状态的迭代卡尔曼滤波融合雷达与 IMU 数据,并以机器人中心重新定义了机器人状态。LIO-SAM<sup>[9]</sup>则通过 IMU 预积分去除雷达点云畸变,同时利用因子图框架构建 IMU 预积分因子、雷达里程计因子、回环检测因子以及可选的 GPS 因子进行后端优化。但是 LIO-SAM 的回环检测不良,在开放环境下容易导致退化。Xu 等<sup>[10]</sup>提出的 FAST-LIO 算法采用迭代扩展卡尔曼滤波(iterative extended kalman filtering, IEKF)快速高效地融合雷达点云数据与 IMU 数据,但是该算法适用于固态雷达。随后的

FAST-LIO2<sup>[11]</sup>在 FAST-LIO 基础上去除了特征提取,直接将点云注册到地图,能够快速构建稠密的三维点云地图,但是缺少回环检测与后端优化,无法在户外长距离下消除累计误差、解决地图漂移等问题。李倩等<sup>[12]</sup>等则在 FAST-LIO2 的基础上提出了一种多激光雷达的定位与建图算法,扩展了机器人的感知范围,提高了定位精度。

综上所述,为解决长距离下建图漂移的问题,提高定位精度,本文参考 FAST-LIO2 提出一种基于 IEKF 的激光惯导紧耦合 SLAM 算法。首先利用 IMU 去除点云畸变,然后利用 IEKF 融合雷达点云数据与 IMU 数据,得到激光里程计,为后端优化提供良好的初值。引入回环检测模块,提取稳定三角描述符<sup>[13]</sup>(stable triangle descriptor, STD),以解决回环不良的问题,并在检测到回环时将回环因子加入后端因子图优化。在后端部分采用因子图优化,通过融合 IMU 预积分因子、激光里程计因子与回环检测因子来消除算法累计误差,提高系统定位精度,构建闭合的稠密三维点云地图。

1 总体框架

本文提出的 SLAM 算法的总体框架如图 1 所示,算法主要分为 3 个模块:1)前端里程计模块:首先将输入的雷达数据与 IMU 数据进行预处理,通过前向传播得到雷达点云的估计位置,利用反向传播对点云进行运动补偿,然后通过 IEKF 融合数据,输出里程计与 IMU 坐标系下的子图。2)回环检测模块:在点云数据中提取 STD,并与哈希表中候选的 STD 进行得分匹配,如果得分超过阈值,则认为存在有闭环,将回环因子加入后端优化。3)后端因子图优化模块:利用 GTSAM 库构建因子图,将 IMU 预积分因子、激光里程计因子、回环检测因子融合求解,求得关键帧的准确位姿,随后将关键帧累积构成全局地图。

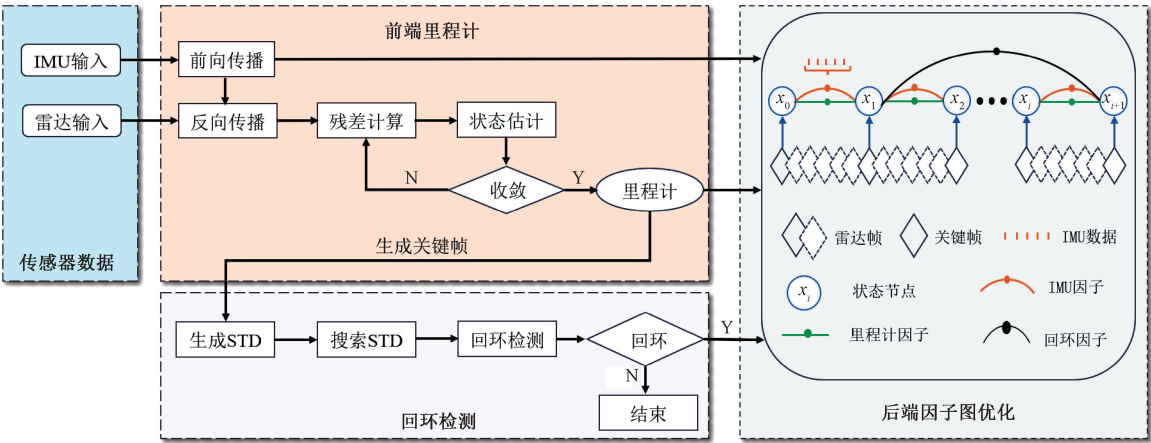


图 1 算法总体框架

Fig. 1 Overall framework of the algorithm

## 2 紧耦合的雷达惯导里程计

### 2.1 IMU 状态模型及前向传播

定义 IMU 坐标系为  $I$ , 雷达坐标系为  $L$ , 世界坐标系为  $W$ , 雷达与 IMU 之间的外参变换矩阵为  ${}^I T_L = ({}^I R_L, {}^I p_L) \in SE(3)$ 。则 IMU 的线性化连续时间运动方程状态模型  $\mathbf{x}$  为:

$$\mathbf{x} = [{}^W \mathbf{p}^T, {}^W \mathbf{v}^T, {}^W \mathbf{R}^T, \mathbf{b}_\omega^T, \mathbf{b}_a^T, \mathbf{g}^T]^T \quad (1)$$

式中:  ${}^W \mathbf{p}^T, {}^W \mathbf{v}^T, {}^W \mathbf{R}^T$  为 IMU 在世界坐标系下的位置、速度与姿态,  $\mathbf{b}_\omega^T, \mathbf{b}_a^T$  为 IMU 陀螺仪与加速度计的零偏,  $\mathbf{g}^T$  为重力加速度。根据论文 FAST-LIO<sup>[10]</sup> 中对  $\boxplus$  与  $\boxminus$  的定义, 使用零阶保持器在 IMU 采样周期  $\Delta t$  处将式(1)表示的连续模型离散化, 生成的离散状态模型为:

$$\mathbf{x}_{i+1} = \mathbf{x}_i \boxplus (\Delta t f(\mathbf{x}_i, \mathbf{u}_i, \mathbf{w}_i)) \quad (2)$$

式中:  $i$  为 IMU 测量数据的索引,  $\mathbf{x}_i, \mathbf{u}_i, \mathbf{w}_i$  为第  $i$  帧 IMU 数据时的状态、输入与噪声。

前向传播的示意图如图 2 所示。一旦接收到 IMU 数据, 将噪声  $\mathbf{w}_i$  设置为 0, 通过式(2)将前一帧激光雷达扫描后得到的最优状态估计值  $\hat{\mathbf{x}}_{k-1}$  以及协方差矩阵  $\hat{\mathbf{P}}_{k-1}$  进行传播:

$$\hat{\mathbf{x}}_{i+1} = \hat{\mathbf{x}}_i \boxplus (\Delta t f(\hat{\mathbf{x}}_i, \mathbf{u}_i, \mathbf{0})); \hat{\mathbf{x}}_0 = \hat{\mathbf{x}}_{k-1} \quad (3)$$

$$\hat{\mathbf{P}}_{i+1} = \mathbf{F}_{\hat{\mathbf{x}}_i} \hat{\mathbf{P}}_i \mathbf{F}_{\hat{\mathbf{x}}_i}^T + \mathbf{F}_{\mathbf{w}_i} \mathbf{Q}_i \mathbf{F}_{\mathbf{w}_i}^T; \hat{\mathbf{P}}_0 = \hat{\mathbf{P}}_{k-1} \quad (4)$$

式中:  $\mathbf{x}_i, \hat{\mathbf{x}}_i$  为状态估计时的真值与预测值。  $\mathbf{Q}_i$  为噪声  $\mathbf{w}_i$  的协方差, 其中矩阵  $\mathbf{F}_{\hat{\mathbf{x}}_i}$  与矩阵  $\mathbf{F}_{\mathbf{w}_i}$  的计算公式如下:

$$\mathbf{F}_{\hat{\mathbf{x}}_i} = \frac{\partial (\mathbf{x}_{i+1} \boxminus \hat{\mathbf{x}}_{i+1})}{\partial \hat{\mathbf{x}}_i} \Big|_{\hat{\mathbf{x}}_i = \mathbf{x}_i, \mathbf{w}_i = 0} \quad (5)$$

$$\mathbf{F}_{\mathbf{w}_i} = \frac{\partial (\mathbf{x}_{i+1} \boxminus \hat{\mathbf{x}}_{i+1})}{\partial \mathbf{w}_i} \Big|_{\hat{\mathbf{x}}_i = \mathbf{x}_i, \mathbf{w}_i = 0} \quad (6)$$

将本次前向传播得到的状态和协方差分别记为  $\hat{\mathbf{x}}_k$ 、 $\hat{\mathbf{P}}_k$ , 作为下一次前向传播的预测状态, 一直持续到扫描结束。

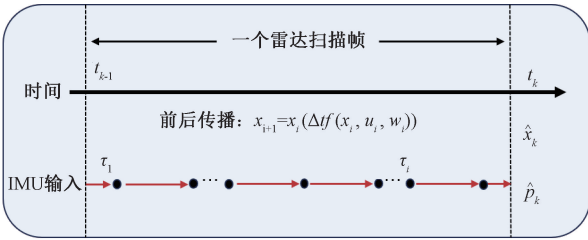


图 2 前向传播

Fig. 2 Forward propagation

### 2.2 测量模型

当激光雷达在扫描周围一圈得到一个雷达扫描帧时, 如果机器人依旧在运动过程中, 点云之间存在有相对运动, 则会产生点云畸变。为了去除掉点云畸变, 需要对点云进行运动补偿。根据前向传播中的 IMU 的状态预测值, 估计扫描中每个点的激光雷达位姿相对于扫描结束时

刻的位姿, 并将其转换到扫描结束时刻的雷达坐标系下。

定义在  $t_{k-1}$  与  $t_k$  相邻两个时刻之间在雷达坐标系  $L$  下累积的点云集合为一帧  ${}^L P$ , 该帧当前点云与帧结束点云之间的位姿变换为  ${}^{t_k} T_{t_{k-1}} = ({}^{t_k} R_{t_{k-1}}, {}^{t_k} p_{t_{k-1}})$ , 则对该帧点云内的每一个点  ${}^L p_j \in {}^L P$  通过式(7)进行运动补偿投影到  $t_k$  时刻:

$${}^{t_k} p_j = {}^{t_k} T_{t_{k-1}}^{-1} {}^{t_k} T_{t_{k-1}} {}^L p_j \quad (7)$$

补偿后的雷达点在雷达坐标系下的真实位置为:

$${}^L p_j^{gt} = {}^L p_j + {}^{t_k} n_j \quad (8)$$

式中:  ${}^L p_j^{gt} \in {}^L P_j, {}^{t_k} n_j$  为雷达测距误差。得到点的真实位置后, 使用其对应雷达姿态  ${}^W T_{t_k} = ({}^W R_{t_k}, {}^W p_{t_k})$  与雷达与 IMU 的外参  ${}^I T_L$  将其投影到世界坐标系  $W$  中, 那么该点在地图中的局部小平面的为<sup>[11]</sup>:

$$h_j(\mathbf{x}_k, {}^L p_j^{gt}) = {}^W \mathbf{u}_j^T ({}^W T_{t_k} {}^I T_L {}^L p_j^{gt} - {}^W q_j) \quad (9)$$

式中:  ${}^W \mathbf{u}_j^T$  为对应局部小平面的法向量,  ${}^W q_j$  是平面上的一点。

### 2.3 残差计算

在通过反向传播对一个雷达扫描帧中全部的雷达点云进行运动补偿后。设当前迭代卡尔曼滤波的迭代次数为  $\kappa$ , 则当前迭代更新时状态  $\mathbf{x}_k^\kappa$  的预测值为  $\hat{\mathbf{x}}_k^\kappa$ :

$$0 = h_j(\mathbf{x}_k, {}^L n_j) = h_j(\hat{\mathbf{x}}_k^\kappa \boxplus \tilde{\mathbf{x}}_k^\kappa, {}^L n_j) \simeq h_j(\hat{\mathbf{x}}_k^\kappa, 0) +$$

$$\mathbf{H}_j^\kappa \tilde{\mathbf{x}}_k^\kappa + v_j = \mathbf{z}_j^\kappa + \mathbf{H}_j^\kappa \tilde{\mathbf{x}}_k^\kappa + v_j \quad (10)$$

式中:  $\tilde{\mathbf{x}}_k^\kappa = \mathbf{x}_k \boxminus \hat{\mathbf{x}}_k^\kappa, \mathbf{H}_j^\kappa$  为  $h_j(\hat{\mathbf{x}}_k^\kappa \boxplus \tilde{\mathbf{x}}_k^\kappa, {}^L n_j)$  关于  $\tilde{\mathbf{x}}_k^\kappa$  在 0 处的雅可比矩阵,  $v_j \in (0, \mathbf{R}_j)$  是由原始噪声引起的,  $\mathbf{z}_j^\kappa$  为残差:

$$\mathbf{z}_j^\kappa = h_j(\hat{\mathbf{x}}_k^\kappa, 0) = \mathbf{u}_j^T ({}^W \hat{T}_{t_k} {}^I \hat{T}_L {}^L p_j - {}^W q_j) \quad (11)$$

### 2.4 状态估计更新

式(3)、(4)的传播状态  $\hat{\mathbf{x}}_k$  与协方差  $\hat{\mathbf{P}}_k$  对未知状态  $\mathbf{x}_k$  提供了先验分布, 各变量相互之间的关系为:

$$\mathbf{x}_k \boxminus \hat{\mathbf{x}}_k = (\hat{\mathbf{x}}_k^\kappa \boxplus \tilde{\mathbf{x}}_k^\kappa) \boxminus \hat{\mathbf{x}}_k = \tilde{\mathbf{x}}_k^\kappa \boxminus \hat{\mathbf{x}}_k + \mathbf{J}^\kappa \tilde{\mathbf{x}}_k^\kappa \sim \mathbf{N}(0, \hat{\mathbf{P}}_k) \quad (12)$$

式中:  $\mathbf{J}^\kappa$  是  $(\hat{\mathbf{x}}_k^\kappa \boxplus \tilde{\mathbf{x}}_k^\kappa) \boxminus \hat{\mathbf{x}}_k$  在  $\tilde{\mathbf{x}}_k^\kappa = 0$  处的偏微分。除先验分布外, 式(13)的状态分布是基于式(10)推导的测量模型计算得到的:

$$-v_j = \mathbf{z}_j^\kappa + \mathbf{H}_j^\kappa \tilde{\mathbf{x}}_k^\kappa \sim \mathbf{N}(0, \mathbf{R}_j) \quad (13)$$

结合式(12)中的先验分布和式(13)中的状态分布, 得到了状态  $\mathbf{x}_k$  的后验分布, 基于最大后验估计, 构造最大后验估计目标函数<sup>[11]</sup>:

$$\min_{\tilde{\mathbf{x}}_k^\kappa} (\|\mathbf{x}_k \boxminus \hat{\mathbf{x}}_k\|_{\hat{\mathbf{P}}_k}^2 + \sum_{j=1}^m \|\mathbf{z}_j^\kappa + \mathbf{H}_j^\kappa \tilde{\mathbf{x}}_k^\kappa\|_{\mathbf{R}_j}^2) \quad (14)$$

式中:  $\|\mathbf{x}\|_M^2 = \mathbf{x}^T \mathbf{M} \mathbf{x}$ , 将该最大后验估计问题进行迭代优化:

$$\mathbf{K} = (\mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} + \mathbf{P}^{-1})^{-1} \mathbf{H}^T \mathbf{R}^{-1} \quad (15)$$

$$\mathbf{x}_k^{\kappa+1} = \hat{\mathbf{x}}_k^\kappa \boxplus (-\mathbf{K} \mathbf{z}_k^\kappa - (\mathbf{I} - \mathbf{K} \mathbf{H}) (\mathbf{J}^\kappa)^{-1} (\hat{\mathbf{x}}_k^\kappa \boxminus \hat{\mathbf{x}}_k)) \quad (16)$$

式中:  $\mathbf{K}$  为卡尔曼滤波增益,  $\mathbf{H} = [\mathbf{H}_1^{\kappa T}, \dots, \mathbf{H}_m^{\kappa T}]^T$ ,  $\mathbf{R} = \text{diag}(\mathbf{R}_1, \dots, \mathbf{R}_m)$ ,  $\mathbf{P} = (\mathbf{J}^{\kappa})^{-1} \hat{\mathbf{P}}_k (\mathbf{J}^{\kappa})^{-T}$ ,  $\mathbf{Z}_k^{\kappa} = [\mathbf{Z}_1^{\kappa T}, \dots, \mathbf{Z}_m^{\kappa T}]^T$ .

上述过程重复进行直到目标函数收敛,然后给出收敛后的最终最优状态和协方差估计如下:

$$\bar{\mathbf{x}}_k = \hat{\mathbf{x}}_k^{\kappa+1}, \bar{\mathbf{P}}_k = (\mathbf{I} - \mathbf{K}\mathbf{H})\mathbf{P} \quad (17)$$

随后,将最优状态估计  $\bar{\mathbf{x}}_k$  和协方差估计  $\bar{\mathbf{P}}_k$  作为下一次扫描的输入,每次重复上述操作,得到紧耦合的激光惯导里程计。

### 3 回环检测

本文引入回环检测模块,以降低算法在长距离运行下产生的累积误差。回环检测能够识别机器人是否到过同一个位置,并利用这一信息计算当前位姿与曾经位姿间的变换,作为约束加入全局优化中,从而提高位姿图整体精度。精准的回环检测可以有效的形成闭环,而错误的回环检测则会导致建图失败。对此,为了提高回环检测的准确性,通常需要在点云中提取刚性不变的描述子。

三角形的形状由边的长度或夹角唯一确定。因此,三角形的形状对刚性变换是完全不变的<sup>[13]</sup>。STD 算法能够在点云中提取局部关键点,并将这些关键点编码为三角描述子,通过匹配点云之间的三角描述子来实现地点识别。一个 STD 的结构如图 3 所示。其中每个顶点  $P_1, P_2, P_3$  对应一个相邻的平面,  $\mathbf{n}_1, \mathbf{n}_2, \mathbf{n}_3$  为邻接平面的法向量。顶点按照  $l_{12} \leq l_{23} \leq l_{13}$  排列。回环检测的步骤如下:

1) 为了提取 STD,首先将一个关键帧划定给指定大小为 2 m 的体素,并计算体素内包含的一组点  $p_i (i = 1, \dots, N)$  的点协方差矩阵  $\sum$ :

$$\bar{\mathbf{p}} = \frac{1}{N} \sum_{i=1}^N p_i; \sum = \frac{1}{N} \sum_{i=1}^N (p_i - \bar{\mathbf{p}})(p_i - \bar{\mathbf{p}})^T \quad (18)$$

令  $\lambda_k$  表示  $\sum$  的第  $k$  个最大特征值,则当  $\lambda_3 < \delta_1$  且  $\lambda_2 > \delta_2$  时认为该体素为体素平面,其中  $\delta_1, \delta_2$  为预设的参数,然后取体素投影到体素平面上距离最大的点为关键点。

2) 在关键帧中提取关键点后,构建 kd 树,每个关键点搜索 20 个邻近点来形成 STD。

3) 根据边长  $l_{12}, l_{13}, l_{23}$  以及法线投影向量的点积  $\mathbf{n}_1 \cdot \mathbf{n}_2, \mathbf{n}_1 \cdot \mathbf{n}_3, \mathbf{n}_2 \cdot \mathbf{n}_3$  计算哈希键,并将候选 STD 加入候选表。

4) 计算当前帧 STD 的哈希值,与候选表中的候选帧进行对比,如果超过了阈值,则认为出现了回环。

### 4 后端因子图优化

本文参考 LIO-SAM<sup>[9]</sup> 算法,后端优化部分采用 GTSAM 库进行因子图优化,以降低激光惯导里程计的累计误差。因子图优化主要由待优化的变量节点与节点间

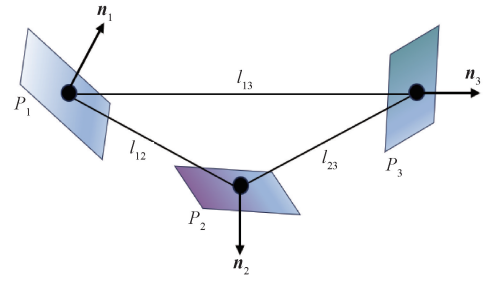


图 3 三角描述符

Fig. 3 Triangle descriptor

的约束因子边构成,其对于一个符合高斯噪声的 SLAM 问题进行最大后验概率推断,转换为如下最小化非线性最小二乘的和:

$$\mathbf{X}^{MAP} = \underset{\mathbf{X}}{\operatorname{argmin}} \sum_i \|F_i(\mathbf{x}_i) - \mathbf{z}_i\|^2 \quad (19)$$

式中:  $F_i(\mathbf{x}_i)$  为观测函数,  $\mathbf{z}_i$  为观测值。

IMU 预积分因子能够构建两个关键帧之间的约束,通过对  $i, j$  两个关键帧之间的 IMU 角速度和加速度的变化量进行积分,得到在两个关键帧之间  $\Delta t$  时间内 IMU 的旋转更新量  $\Delta \mathbf{R}_{ij}$ 、速度更新量  $\Delta \mathbf{v}_{ij}$  与位移更新量  $\Delta \mathbf{p}_{ij}$ , 其 IMU 预积分因子模型为:

$$\mathbf{e}_{imu} = \begin{bmatrix} \operatorname{Log}(\Delta \mathbf{R}_{ij}^T (\mathbf{R}_i^T \mathbf{R}_j)) \\ \Delta \mathbf{v}_{ij} - \mathbf{R}_i^T (\mathbf{v}_j - \mathbf{v}_i - g \Delta t) \\ \Delta \mathbf{p}_{ij} - \mathbf{R}_i^T \left( \mathbf{p}_j - \mathbf{p}_i - \mathbf{v}_i \Delta t - \frac{1}{2} g \Delta t^2 \right) \\ b_{aj} - b_{ai} \\ b_{\omega j} - b_{\omega i} \end{bmatrix} \quad (20)$$

式中:  $\operatorname{Log}()$  将旋转矩阵转换为李代数向量,  $b_a, b_{\omega}$  分别为加速度计与陀螺仪噪声。

激光惯导里程计因子通过估计相邻帧的相对位姿,提供局部运动约束,设关键帧  $i$  与  $j$  的相对位姿为  $\mathbf{T}_{ij} = (\mathbf{R}_{ij}, \mathbf{t}_{ij})$ , 其模型为:

$$\mathbf{e}_{lidar} = \begin{bmatrix} \operatorname{Log}(\mathbf{R}_{ij}^T (\mathbf{R}_i^T \mathbf{R}_j)) \\ \mathbf{t}_{ij} - \mathbf{R}_i^T (\mathbf{t}_j - \mathbf{t}_i) \end{bmatrix} \quad (21)$$

回环检测因子能够识别重复场景,连接历史与当前位姿,消除累积漂移<sup>[14]</sup>。设关键帧  $i$  与  $j$  的相对位姿为  $\mathbf{T}_{loop} = (\mathbf{R}_{loop}, \mathbf{t}_{loop})$ , 其模型为:

$$\mathbf{e}_{loop} = \begin{bmatrix} \operatorname{Log}(\mathbf{R}_{loop}^T (\mathbf{R}_i^T \mathbf{R}_j)) \\ \mathbf{t}_{loop} - \mathbf{R}_i^T (\mathbf{t}_j - \mathbf{t}_i) \end{bmatrix} \quad (22)$$

调用 iSAM2 算法处理新增的激光里程计因子、IMU 预积分因子与回环检测因子,减少一次性线性误差累计,降低陷入局部极小的概率。

### 5 算法评估与实验验证

#### 5.1 实验平台

为了全面评估本文所提出算法在不同环境中在定位



精度、建图质量上的性能表现,本文分别在公开数据集以及自采集数据集上进行了测试。其中公开数据集选择 KITTI 数据集中带有回环与真值轨迹的 00、05、07、09 序列。自采集数据集则使用煜禾森机器人移动底盘作为采集平台,其上搭载有 Livox Mid360 雷达(内置有型号为 ICM40609 的 6 轴 IMU,频率为 200 Hz)、计算处理单元(型号为 NVIDIA Jetson orin Nx 16 G)与 RTK 天线,实验设备如图 4 所示。本文中所有实验均在配置有 Intel Core i5-9300H 处理器与 16 GB 内存的 Ubuntu 20.04 操作系统环境的笔记本上运行。算法采用 C++ 编程语言开发,同时选用开源机器人操作系统框架 ROS(版本为 Noetic)作为软件平台。



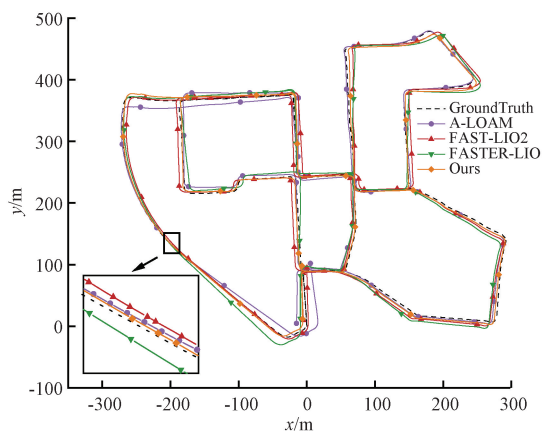
图 4 实验设备

Fig. 4 Experimental equipment

## 5.2 定位精度分析

为了准确地评估算法优劣,刻画轨迹之间的误差,本文采用开源 SLAM 评测工具 EVO 将算法运行的估计轨迹与真实轨迹进行对齐并计算绝对位姿误差(absolute pose error, APE),即每个位姿李代数的均方根误差(root mean square error, RMSE),其定义如下:

$$APE_{trans} = \sqrt{\frac{1}{N} \sum_{i=1}^N \|trans(T_{gt,i}^{-1} T_{est,i})\|_2^2} \quad (23)$$

(a) KITTI 00 序列  
(a) KITTI 00 sequences

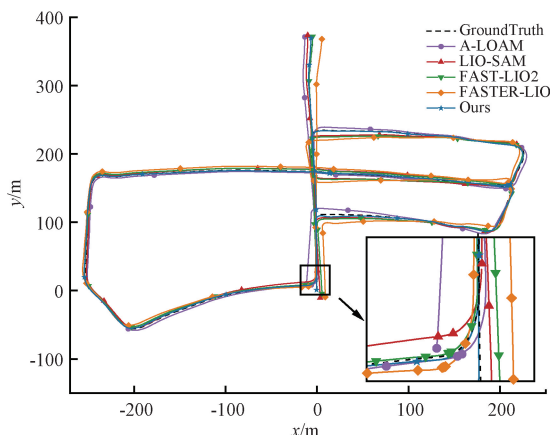
式中:  $APE_{trans}$  为绝对位姿误差在平移的均方根误差  $T_{gt,i}$  为第  $i$  帧的真实位姿,  $T_{est,i}$  为第  $i$  帧的估计位姿,  $trans$  表示平移。

### 1) KITTI 数据集实验

A-LOAM、LIO-SAM、FASTER-LIO<sup>[15]</sup>、FAST-LIO2 与本文算法在 KITTI 数据集序列 00、05、07 和 09 序列上运行的真实轨迹以及对比结果如图 5 所示(由于 LIO-SAM 未能在 00 序列上未能良好运行,因此未在 00 序列中绘制轨迹并计算 APE)。从图 5 中可以观察到,经由回环检测与后端机制的优化,本文算法在整体上展现出更高的准确性,其估计轨迹更紧密地贴合于参考真值轨迹。5 种不同算法的 APE RMSE 误差值统计如表 1 所示。相比于其他 4 种算法,本文算法在 RMSE 上均取得了最低值。其中在 KITTI 00 序列下, RMSE 对比 A-LOAM、FAST-LIO2 与 FAST-LIO 算法分别降低了 15.94%、42.14%、38.61%。在 KITTI 05 序列下, RMSE 对比 A-LOAM、LIO-SAM、FAST-LIO2 与 FAST-LIO 算法分别降低了 60.19%、57.32%、56.24% 和 74.17%。而在 KITTI 07 序列下的 RMSE 分别降低了 47.5%、26.74%、66.31%、49.6%。在 KITTI 09 序列下的 RMSE 分别降低了 72.63%、76.55%、48.92%、47.86%。实验结果表明,在公开数据集 KITTI 序列中,本算法在定位精度上取得了更加良好的效果。

### 2) 自采集数据集实验

为进一步验证本文算法的定位精度与建图效果,使用如图 4 实验设备在东南大学苏州校区采集了两组数据集:校园数据集与户外数据集。为验证算法的回环检测效果,两个数据集的路径均形成闭环,其中校园数据集的卫星地图轨迹投影如图 6(a)所示,在路径中两次经过中间区域,该数据集总路径长度为 2 345 m。户外数据集的卫星地图轨迹投影如图 6(b)所示,该数据集的总路径长度为 2 781 m。两个数据集涵盖了高层建筑物及繁茂树木的复杂环境,且路径持续时间较长,具有较高的挑战性。此外,本文采用

(b) KITTI 05 序列  
(b) KITTI 05 sequences

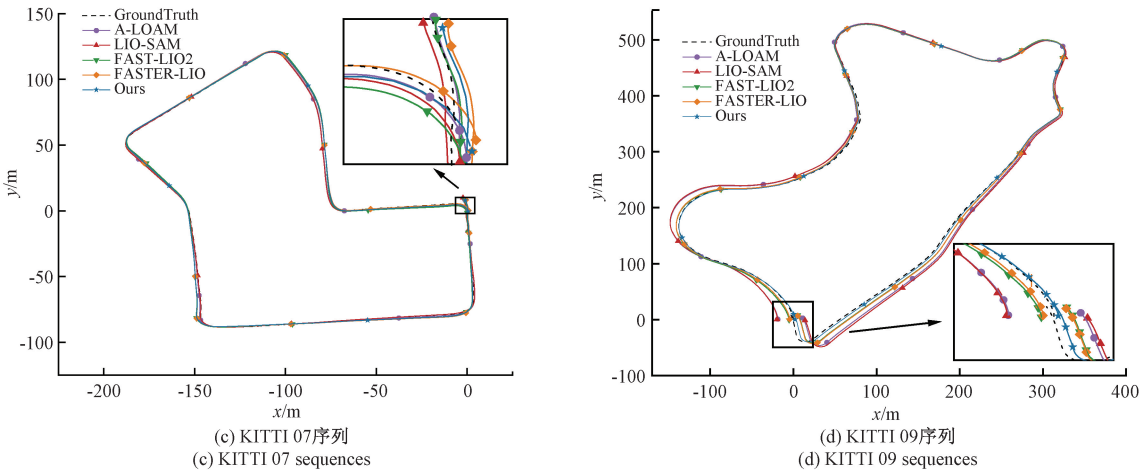


图 5 各算法在 KITTI 数据集上的轨迹对比图  
Fig. 5 Comparison of the trajectories of the algorithms in the KITTI dataset

表 1 各算法在 KITTI 上的 RMSE 值对比

Table 1 Comparison of RMSE values of various algorithms on KITTI

算法	KITTI 00	KITTI 05	KITTI 07	KITTI 09
	序列	序列	序列	序列
A-LOAM	8.72	5.17	1.20	5.90
LIO-SAM	*	4.82	0.86	6.90
FAST-LIO2	12.67	4.71	1.87	2.38
FASTER-LIO	11.94	7.98	1.25	2.42
Ours	<b>7.33</b>	<b>2.06</b>	<b>0.63</b>	<b>0.78</b>

表 2 自采集数据集的 APE 值

Table 2 APE value of self collected dataset

算法	校园数据集			户外数据集		
	rmse max std			rmse max std		
	rmse	max	std	rmse	max	std
LIO-SAM	2.76	5.78	1.25	7.83	14.78	3.31
FASTLIO2	2.31	5.05	1.10	2.95	5.70	1.22
FASTER-LIO	2.32	5.10	1.11	4.28	7.82	1.85
Ours	<b>2.01</b>	<b>2.70</b>	<b>0.29</b>	<b>1.48</b>	<b>2.44</b>	<b>0.53</b>

各算法之间的 RMSE 随时间的变化如图 7 所示。其中 LIO-SAM、FAST-LIO2 和 FASTER-LIO 三种算法在两个数据集上的 RMSE 波动均较大,表明算法在特定时间间隔内误差较大。本文算法则在整个时间范围内 RMSE 波动最小,折线较低且更平滑。这表明与其他 3 个算法相比,本文算法不但始终表现出更低的平移误差,定位性能更佳,而且具有更好的稳健性,能够更好的适应复杂多变的环境。

本文算法在回环优化前后的三维空间轨迹对比如图 8 所示。可以看到在进行回环优化之前,算法在长距离运行后由于里程计的累计误差,出现了一定程度的漂移。而在加入回环检测模块后,在经过历史位置时后端优化将当前帧位姿与历史帧位姿进行了对齐,抑制  $z$  轴方向上的漂移,得到了更加准确的位姿。

5.3 建图效果分析

为了验证本文算法的建图效果,使用校园数据集与户外数据集进行建图试验,将本文算法分别与 LIO-SAM、FAST-LIO2、FASTER-LIO 算法得到的点云地图进行对比。4 种算法在校园数据集上的建图效果如图 9 所示。本文算法的全局地图如图 9(a)所示,算法构建的点云地图建筑物轮廓、道路以及周围树木清晰。LIO-SAM 的局部建图效果如图 9(b)所示,建筑物虽然未出现明显漂移,但房屋轮廓较厚,且周围存在有大量噪点并出现了重影。

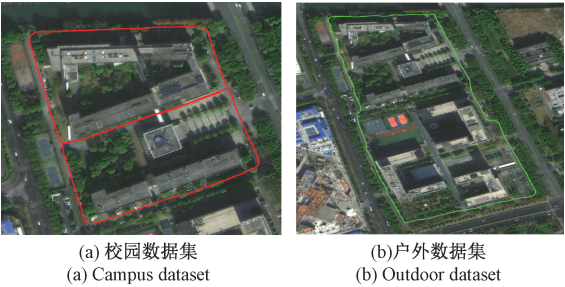


图 6 自采集数据集的卫星地图投影  
Fig. 6 Satellite map projection of self-collected datasets

IMU 与 RTK 组合惯导的方式提供地面轨迹真值。

将本文算法分别与 LIO-SAM、FAST-LIO2、FASTER-LIO 三个算法在自采集数据集上进行了测试;4 种不同算法的 RMSE 对比结果如表 2 所示。实验结果表明,本文算法相对于 LIO-SAM、FAST-LIO2、FASTER-LIO 算法的 RMSE 在校园数据集上分别下降了 27.17%、12.99%、13.36%。在户外数据集上分别下降了 81%.09%、49.83%、65.42%。且在最大值、标准差上均取得了更小的数值,这表明了本文算法在定位精度上的优势。

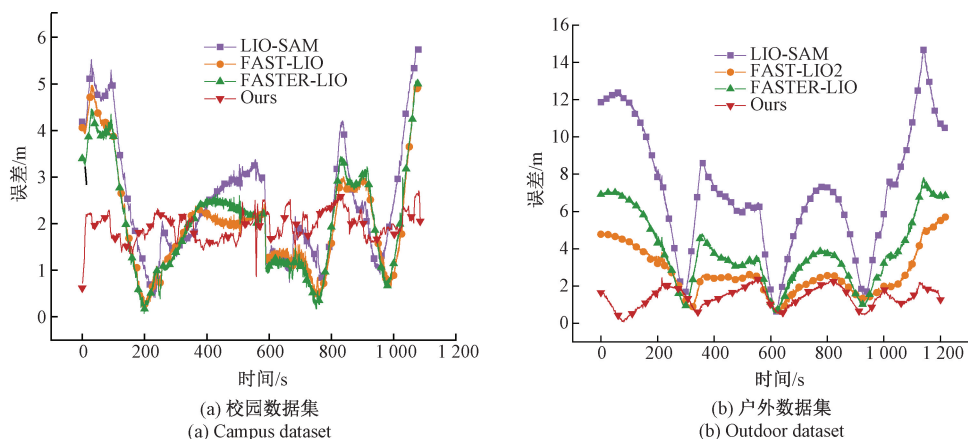


图 7 绝对位姿误差变化对比图

Fig. 7 Comparison of absolute posture error changes

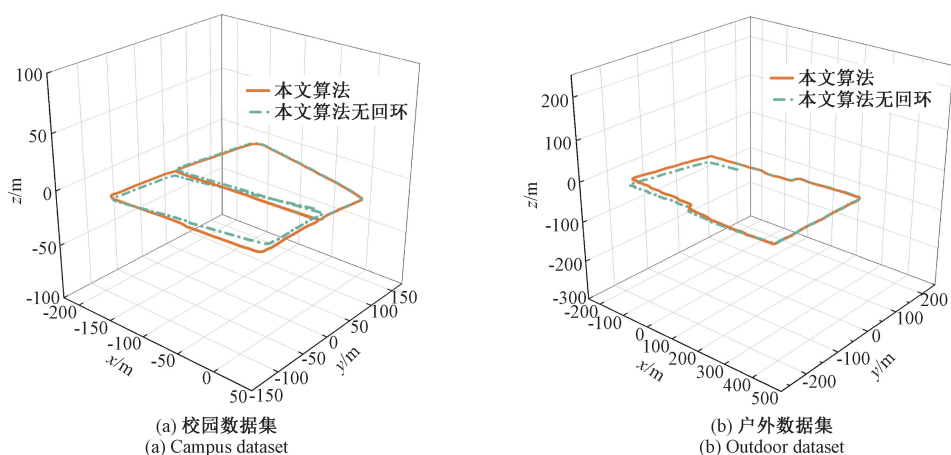


图 8 回环优化前后轨迹图对比

Fig. 8 Comparison of trajectory before and after loopclosure optimization

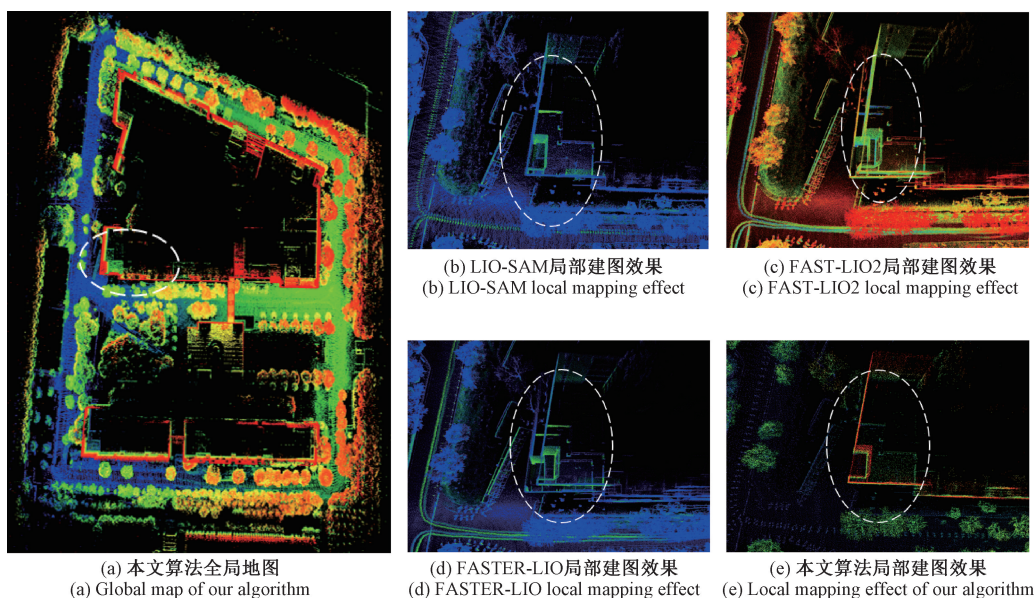


图 9 各算法在校园数据集上的建图效果对比

Fig. 9 Comparison of mapping effects of various algorithms on campus datasets



FAST-LIO2 与 FASTER-LIO 的建图效果分别如图 9(c)、图 9(d)所示,算法构建的建筑物出现了明显的漂移,建筑物的轮廓未能重合。本文算法在局部建图效果图 9(e)所示,算法构建的墙体没有漂移,墙体较薄且无重影。

本文算法全局建图效果及各算法在  $z$  轴上的表现如图 10 所示,其中,LIO-SAM 在  $z$  轴上的表现如图 10(b)所示,点云地图在运行过程中偏移明显,在水平方向更加弯曲,且在起止点处由于闭环检测失败,构建的地图出现了

明显的下沉,点云分层明显。FAST-LIO2 与 FASTER-LIO 建图效果如图 10(c)、(d)所示,两种算法在水平方向上的建图效果虽然有所改善,但是由于缺少回环检测与后端优化,在起止点处仍存在位姿不闭合问题。而本文算法构建的点云地图如图 10(e)所示,得益于回环检测模块与后端优化,在起止点处得到了闭合, $z$  轴上未出现漂移。实验表明,本文算法能够在户外长距离运行下构建全局一致的三维地图。

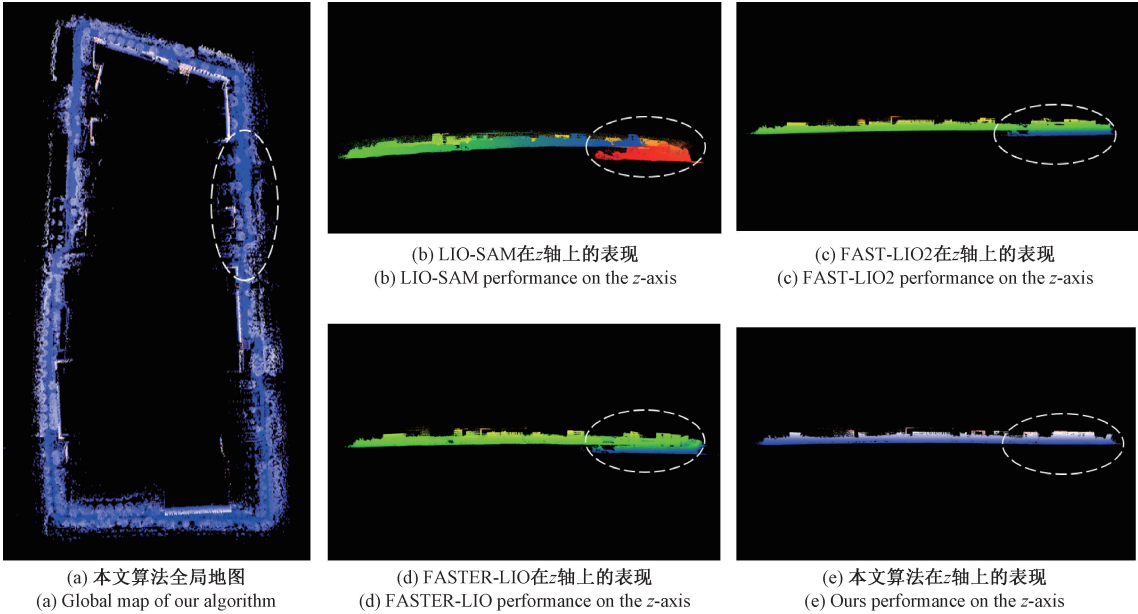


图 10 本文算法全局建图效果及各算法在  $z$  轴上的表现

Fig. 10 The global mapping effect of this algorithm and the performance of each algorithm on the  $z$ -axis

6 结 论

本文提出了一种基于增量扩展卡尔曼滤波的激光惯导紧耦合 SLAM 算法,旨在解决现有 SLAM 算法在户外长距离复杂环境中定位精度不足和建图漂移等问题。算法的前端里程计模块通过构建 IMU 运动模型和测量模型,并采用迭代扩展卡尔曼滤波对 IMU 数据与激光雷达数据进行紧耦合融合,为后端优化提供初始估计。回环检测模块通过关键帧提取三角描述子,建立回环检测机制以有效消除长时间运行中积累的误差。在后端优化部分,采用 GTSAM 因子图优化框架,结合 IMU 预测因子、里程计因子以及回环检测因子进行全局优化,进一步消除系统中的累积误差,并生成一致性的全局点云地图。实验结果表明,相较于其他激光 SLAM 算法,本文提出的算法在长距离复杂环境下表现出更强的鲁棒性,能够实现更高的定位精度,并且在建图效果上也取得了显著改进。

参考文献

[1] 余联想,郑明魁,欧文君,等. 多传感器融合的移动机器人室外激光 SLAM 算法优化与系统实现[J]. 电子测

量与仪器学报,2023,37(2):48-55.  
YU L X, ZHENG M K, OU W J, et al. Optimization and system implement of outdoor lidar SLAM algorithm for mobile robots based on multi-sensor fusion[J]. Journal of Electronic Measurement and Instrumentation, 2023, 37(2): 48-55.  
[2] 诸葛晶昌,高宏,罗其俊,等. 激光雷达 IMU 紧耦合 SLAM 算法研究[J]. 仪器仪表学报,2024,45(11): 243-251.  
ZHUGE J CH, GAO H, LUO Q J, et al. Research on tightly coupled SLAM algorithm for LiDAR IMU[J]. Chinese Journal of Scientific Instrument, 2024, 45(11): 243-251.  
[3] CAMPOS C, ELVIRA R, RODRIGUEZ J J, et al. ORB-SLAM3: An accurate open-source library for visual, visual-inertial, and multimap SLAM[J]. IEEE Transactions on Robotics: A Publication of the IEEE Robotics and Automation Society, 2021, 37(6): 1874-1890.  
[4] QIN T, LI P L, SHEN SH J. Vins-mono: A robust and



- versatile monocular visual-inertial state estimator [J]. IEEE Transactions on Robotics, 2018, 34 (4): 1004-1020.
- [5] XU X B, ZHANG L, YANG J, et al. A review of multi-sensor fusion slam systems based on 3D LIDAR [J]. Remote Sensing, 2022, 14(12): 2835-2862.
- [6] ZHANG J, SINGH S. LOAM: Lidar odometry and mapping in real-time [C]. Robotics: Science and Systems, 2014, 2(9): 1-9.
- [7] SHAN T X, ENGLLOT B. Lego-loam: Lightweight and ground-optimized lidar odometry and mapping on variable terrain [C]. 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS). IEEE, 2018: 4758-4765.
- [8] QIN C, YE H Y, PRANATA C E, et al. Lins: A lidar-inertial state estimator for robust and efficient navigation[C]. 2020 IEEE International Conference on Robotics and Automation(ICRA). IEEE, 2020: 8899-8906.
- [9] SHAN T X, ENGLLOT B, MEYERS D, et al. Lio-sam: Tightly-coupled lidar inertial odometry via smoothing and mapping [C]. 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS). IEEE, 2020: 5135-5142.
- [10] XU W, ZHANG F. Fast-lío: A fast, robust lidar-inertial odometry package by tightly-coupled iterated kalman filter [J]. IEEE Robotics and Automation Letters, 2021, 6(2): 3317-3324.
- [11] XU W, CAI Y X, HE D J, et al. Fast-lío2: Fast direct lidar-inertial odometry[J]. IEEE Transactions on Robotics, 2022, 38(4): 2053-2073.
- [12] 李倩,陈付龙,郑亮,等. IMU 紧耦合的多激光雷达定位与建图方法[J]. 电子测量技术, 2024, 47(9): 26-32.
- LI Q, CHEN F L, ZHENG L, et al. IMU tightly coupled multi-lidar positioning and mapping method[J]. Electronic Measurement Technology, 2024, 47(9): 26-32.
- [13] YUAN CH J, LIN J R, ZOU Z H, et al. Std: Stable triangle descriptor for 3d place recognition[C]. 2023 IEEE International Conference on Robotics and Automation(ICRA). IEEE, 2023: 1897-1903.
- [14] 张冰战,尹晨晨,李志远,等. 基于点云特征全局搜索的回环检测算法[J]. 电子测量与仪器学报, 2024, 38(3): 176-186.
- ZHANG B ZH, YIN CH CH, LI ZH Y, et al. Loop closure detection algorithm based on global search of point cloud features[J]. Journal of Electronic Measurement and Instrumentation, 2024, 38(3): 176-186.
- [15] BAI CH G, XIAO T, CHEN Y J, et al. Faster-LIO: Lightweight tightly coupled LiDAR-inertial odometry using parallel sparse incremental voxels [J]. IEEE Robotics and Automation Letters, 2022, 7 (2): 4861-4868.

## 作者简介

**孙渊**, 博士, 教授, 主要研究方向为自动化设备的过程控制、机器人技术与应用。

E-mail: suny@sdju.edu.cn

**陈志金**(通信作者), 硕士研究生, 主要研究方向为移动机器人定位导航。

E-mail: jin11208@163.com

**董昊轩**, 硕士研究生, 主要研究方向为移动机器人多传感器融合定位导航。

E-mail: 1034158928@qq.com