

DOI:10.19651/j.cnki.emt.2415655

IMU 紧耦合的多激光雷达定位与建图方法^{*}

李倩^{1,2} 陈付龙² 郑亮¹ 赵法龙¹ 陈智君¹

(1. 长三角哈特机器人产业技术研究院 芜湖 241006; 2. 安徽师范大学计算机与信息学院 芜湖 241000)

摘要: 在许多移动机器人的应用场景下,如自动化仓储物流场景,由于激光雷达安装位置的限制,采用单一激光雷达的 SLAM 解决方案存在视场受限以及难以闭环的问题。为此基于 FAST-LIO2 算法提出了一种 IMU 紧耦合的多激光雷达定位与建图方法,该方法在扩展了机器人的感知范围的同时提高了定位精度和建图效果。通过公开数据集的离线测试以及自建实验平台的在线测试,相较于 M-LOAM、FAST-LIO2 和 Faster-LIO 算法,所提出的算法在定位精度和建图效果上取得了显著提升,并具有更低的回环漂移。

关键词: 移动机器人;同时定位与建图;多激光雷达;紧耦合;惯性导航

中图分类号: TN958.98; TP391.9 文献标识码: A 国家标准学科分类代码: 510.99

IMU tightly coupled multi-lidar positioning and mapping method

Li Qian^{1,2} Chen Fulong² Zheng Liang¹ Zhao Falong¹ Chen Zhijun¹

(1. Hart Robot Industry Technology Research Institute of Yangtze River Delt, Wuhu 241006, China;

2. College of Computer and Information, Anhui Normal University, Wuhu 241000, China)

Abstract: In various applications of mobile robotics, such as automated warehousing logistics scenarios, due to the limitation of lidar installation location. The adoption of a single LiDAR for Simultaneous Localization and Mapping (SLAM) introduces challenges pertaining to restricted field of view and complexities in achieving loop closure. In response, we propose a multi-LiDAR localization and mapping methodology incorporating tight coupling with an Inertial Measurement Unit (IMU), building upon the FAST-LIO2 algorithm. This approach not only expands the perceptual range of the robot but also enhances localization precision and mapping effect. Through rigorous evaluation via offline tests utilizing public datasets and online experiments conducted on the experimental platform, the proposed algorithm demonstrates marked enhancements in localization accuracy and mapping effect compared to the M-LOAM and FAST-LIO2 algorithms, concurrently exhibiting reduced loop closure drift.

Keywords: mobile robot; simultaneous localization and mapping; multi lidar; tightly coupling; inertial navigation

0 引言

得益于传感器技术的快速发展,自主感知移动机器人技术正在不断迭代和进步。在此背景下,同时定位与建图(simultaneous localization and mapping, SLAM)成为移动机器人在未知环境中实现自主运行的关键技术之一。

基于相机的视觉 SLAM^[1]能够提供丰富的 RGB 信息,但对光照环境、初始化和测量范围比较敏感。相比于视觉 SLAM,激光 SLAM 不受光照条件的影响,并且具有更远的感知距离,尤其是随着 Velodyne VLS-128 和 Ouster OS1-128 等高精度激光雷达的出现,基于激光雷达的 SLAM 算法更加适用于 3D 空间^[2],目前 3D 激光 SLAM 技术已经广泛应用于各种工业场景,如煤矿作业场景^[3]、自

动化仓储物流场景^[4]和无人巡检场景^[5]等。在过去的二十年里,人们提出大量基于激光雷达的 SLAM 算法,其中,Zhang 等^[6]提出了一种松耦合的雷达里程计与建图算法(lidar odometry and mapping, LOAM),LOAM 将同步定位与建图划分为高频里程计问题与低频建图问题,通过迭代最近点(iterative closest point, ICP)实现扫描配准,但 LOAM 容易受无特征场景和小视场激光雷达的影响,并且耗时较多^[7]。Jiao 等^[8]提出了一种多雷达激光 SLAM 算法(multi-lidar odometry and mapping, M-LOAM),该算法通过融合多个激光雷达数据来提升机器人对环境的感知能力,但其基于滑动窗口的数据融合过程较为复杂,并且其采用的惯性测量单元(inertial measurement unit, IMU)松耦

收稿日期:2024-03-19

* 基金项目:国家自然科学基金(61972438)、芜湖市科技计划项目(2022yf50)资助

合方案,相较于 IMU 紧耦合方案的定位性能较差。2021 年,香港大学的 MaRS 团队提出了一种具有较高效率和鲁棒性的激光雷达惯性里程计框架(a fast and robust lidar-inertial odometry, FAST-LIO)^[9],FAST-LIO 通过反向传播补偿雷达运动失真,并采用紧耦合的迭代卡尔曼滤波融合雷达特征点和 IMU 测量值。为了保留更多的环境细节并保证定位与建图效率,该团队基于 FAST-LIO 算法,提出了 FAST-LIO2 框架^[10],FAST-LIO2 将原始点直接注册至点云地图,而不提取特征点,并利用增量式 KD 树(intrinsic-kd tree,ikd-Tree)动态数据结构提高整体性能。同年,Bai 等^[11]基于 FAST-LIO2 提出一种并行稀疏增量体素的轻量级紧耦合激光雷达-惯性里程计算法(lightweight tightly coupled lidar-inertial odometry, Faster-LIO)。然而,基于单一雷达的 FAST-LIO2 在一些特殊的应用场景下并不适用,如自动化仓储物流场景,货物装载于移动搬运机器人(automated guided vehicle, AGV)上方,激光雷达一般安装于 AGV 前方、后方或者侧面,从而导致 FOV 受限的问题,本文研究了一种基于 IMU 紧耦合的多雷达同时定位与建图方法(tightly coupled multi lidar-inertial

odometry,M-LIO),相较于 FAST-LIO2 算法,该方法能够有效地融合多个视野(field of view,FOV)的点云数据。本研究在公开数据集以及自行采集的数据集同 M-LOAM、FAST-LIO2 和 Faster-LIO 算法进行了对比实验,结果表明,所提出的 M-LIO 算法在定位精度和建图效果方面均表现出更高的水平。

1 算法框架

本文算法的研究目标是根据接收的多激光雷达数据以及 IMU 数据,实现实时的六自由度(six degrees of freedom tracking,6DOF)状态估计,并构建全局一致的点云地图。如图 1 所示,该算法由数据融合和前端里程计两部分组成,其中,数据融合是融合不同 FOV 雷达的点云数据,具体包括时间同步和空间同步。前端里程计通过 IMU 积分预测 IMU 点的状态量,再根据预测的状态量实现点云畸变校正,从而将扫描点云映射到同一时刻,然后,使用紧耦合的迭代卡尔曼滤波器迭代更新点云帧的状态量,最后,将激光点注册到全局地图,全局地图由 ikd-Tree 结构维护,并基于 ikd-Tree 进行降采样^[12]。

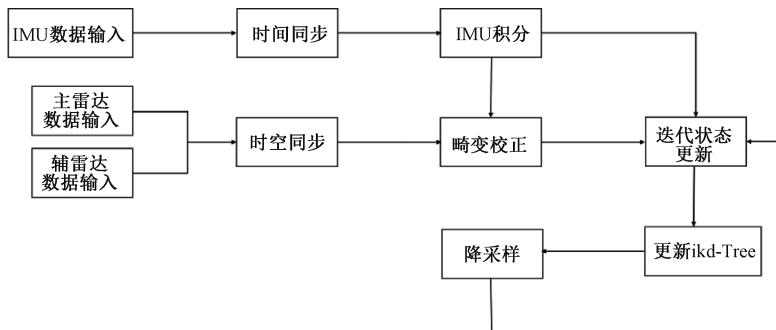


图 1 算法框架

2 系统模型描述

使用封装算子田、曰描述广义加法和广义减法,田和曰的具体定义如下:

$$\text{田}: \mathbf{M} \times \mathbf{R}^n \rightarrow \mathbf{M}; \quad \square: \mathbf{M} \times \mathbf{M} \rightarrow \mathbf{R}^n \quad (1)$$

其中, \mathbf{M} 表示维度为 n 的流形空间 ($\mathbf{M} = \text{SO}(3)$), \mathbf{R}^n 表示维度为 n 的线性空间。假设 I 为 IMU 坐标系, L 为雷达坐标系, ${}^I\mathbf{T}_L = ({}^I\mathbf{R}_L, {}^I\mathbf{P}_L)$ 表示雷达坐标系到 IMU 坐标系的变换矩阵,IMU 运动模型^[8]如下:

$$\left\{ \begin{array}{l} {}^G\dot{\mathbf{P}}_I = {}^G\mathbf{V}_I, \\ {}^G\dot{\mathbf{V}}_I = {}^G\mathbf{R}_I(\mathbf{a}_m - \mathbf{b}_a - \mathbf{n}_a) + {}^G\mathbf{g}, \\ {}^G\dot{\mathbf{R}}_I = {}^G\mathbf{R}_I[\omega_m - \mathbf{b}_\omega - \mathbf{n}_{\omega}]_\perp, \\ {}^G\dot{\mathbf{g}} = 0, \\ \dot{\mathbf{b}}_\omega = \mathbf{n}_{ba}, \\ \dot{\mathbf{b}}_a = \mathbf{n}_{ba}. \end{array} \right. \quad (2)$$

其中, ${}^G\mathbf{R}_I$ 、 ${}^G\mathbf{P}_I$ 和 ${}^G\mathbf{V}_I$ 分别表示 IMU 在世界坐标系的

姿态、位置和速度, ${}^G\mathbf{g}$ 表示世界坐标系下的重力加速度, \mathbf{a}_m 、 ω_m 分别表示 IMU 的加速度测量值和角速度测量值, \mathbf{n}_a 、 \mathbf{n}_ω 表示 IMU 的测量噪声, \mathbf{b}_a 表示 IMU 关于加速度的零偏, \mathbf{b}_ω 表示 IMU 关于角速度的零偏, \mathbf{n}_{ba} 表示 \mathbf{b}_a 的高斯噪声, \mathbf{n}_{ba} 表示 \mathbf{b}_ω 的高斯噪声, 符号 $[\mathbf{a}]_\perp$ 表示向量 \mathbf{a} 的反对称矩阵。

将上述模型离散化,可以获得以下模型:

$$\mathbf{x}_{i+1} = \mathbf{x}_i \text{田} (\Delta t f(\mathbf{x}_i, \mathbf{u}_i, \mathbf{w}_i)) \quad (3)$$

其中, Δt 表示 IMU 的测量间隔, f 表示变化率函数, 变化率函数的具体定义如式(4)~(6)所示,其中 \mathbf{u} 表示 IMU 输入, \mathbf{w} 表示噪声。

$$f(\mathbf{x}_i, \mathbf{u}_i, \mathbf{w}_i) = \begin{bmatrix} \omega_{m_i} - \mathbf{b}_{\omega_i} - \mathbf{n}_{\omega_i} \\ {}^G\mathbf{V}_{I_i} \\ {}^G\mathbf{R}_{I_i}(\mathbf{a}_{m_i} - \mathbf{b}_{a_i} - \mathbf{n}_{a_i}) + {}^G\mathbf{g} \\ \mathbf{n}_{ba_i} \\ \mathbf{n}_{ba_i} \\ \mathbf{0}_{3 \times 1} \end{bmatrix} \quad (4)$$

$$\left\{ \begin{array}{l} \dot{x} = [{}^G\mathbf{R}_t^T \ {}^G\mathbf{P}_t^T \ {}^G\mathbf{V}_t^T \ \mathbf{b}_m^T \ \mathbf{n}_a^T \ {}^G\mathbf{g}^T]^T \in \mathbf{M}, \\ u = [\omega_m^T \ \mathbf{a}_m^T], \\ w = [\mathbf{n}_\omega^T \ \mathbf{n}_a^T \ \mathbf{n}_{ba}^T \ \mathbf{n}_{ba}^T]^T \end{array} \right. \quad (5)$$

$$\left\{ \begin{array}{l} \mathbf{M} = SO(3) \times \mathbf{R}^{15}, \\ \dim(\mathbf{M}) = 18 \end{array} \right. \quad (6)$$

3 数据融合

3.1 雷达标定与空间同步

为了将多个雷达点云映射到同一个坐标系下,需要预先进行雷达间的外参标定。对于多雷达系统,可以选定一个雷达作为主雷达,其他雷达作为辅助雷达,雷达之间的标定是各个辅助雷达与主雷达之间的外参标定。雷达与雷达之间的外参标定分为两类:一种是存在公共 FOV 的双雷达外参标定,另一种是不存在公共 FOV 的双雷达外参标定^[13]。

当主雷达与辅助雷达之间存在公共 FOV 时,在静止状态下,采集激光雷达的点云数据,使用点云配准算法(generalized iterative closest point, GICP)^[14]对主雷达点云帧和辅助雷达点云帧进行配准,获取主雷达和辅助雷达之间的变换矩阵 T 。

当主雷达与辅助雷达之间不存在公共 FOV 或者 FOV 重叠率较小时,可以在主雷达和辅助雷达之间增设一个激光雷达作为参考雷达。同时获取参考雷达与主雷达,以及参考雷达与辅助雷达之间的转换矩阵 $\mathbf{T}_1, \mathbf{T}_2$,即可计算主雷达和辅助雷达之间的转换矩阵 T 为:

$$\mathbf{T} = \mathbf{T}_1^{-1} \times \mathbf{T}_2 \quad (7)$$

如式(8)所示,空间同步是利用变换矩阵 T ,将辅助雷达的点云帧 \mathbf{P}_a 映射到主雷达坐标系下为:

$$\mathbf{P} = \mathbf{T}\mathbf{P}_a \quad (8)$$

3.2 雷达时间同步

雷达数据的时间同步是实现多雷达系统的关键问题之一,若将时间不同步的点云数据用于定位与建图,将会导致定位不准确以及点云地图的重影现象。时间同步分为时间硬同步和时间软同步^[15],硬同步要求所有激光雷达共用同一个时钟源,软同步是对同一时间段内起始时间戳或终止时间戳不一致的点云帧进行数据处理。

本文主要研究了一种时间软同步的方法。如图 2 所示,对于激光雷达,可以获取一系列点云帧,每帧点云为激光雷达在单个周期内对环境的 3D 扫描。假设雷达 1 表示主雷达,其他雷达 $s(s \in \{2, \dots, n\})$ 表示辅助雷达。各雷达的扫描频率保持一致,所有辅助雷达以主雷达为基准进行时间对齐。

对于接收的主雷达和辅助雷达点云帧,先进行空间同步处理,再基于主雷达点云的时间戳,根据式(9)确定是否存在满足时间同步条件的辅助雷达点云帧为:

$$fabs(t_i - t_j) \leqslant 0.5T \quad (9)$$

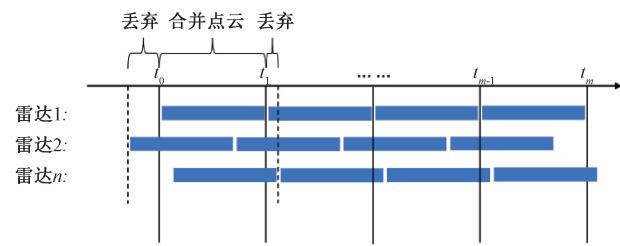


图 2 点云帧时间同步

其中, t_i 表示主雷达第 i 帧点云的起始时间戳, t_j 表示辅助雷达第 j 帧点云的起始时间戳, T 表示雷达扫描周期, $fabs$ 表示绝对值函数。对于所有满足时间同步条件的主雷达和辅助雷达点云帧,按照式(10)进行合并为:

$$\mathbf{P} = \mathbf{P}_i + \sum_{t_i - t_m}^T \mathbf{P}_m + \sum_0^{T+t_i-t_n} \mathbf{P}_n \quad (10)$$

其中, \mathbf{P}_i 表示主雷达第 i 帧点云, \mathbf{P}_m 表示满足时间同步条件并且起始时间戳小于 \mathbf{P}_i 起始时间戳的辅助雷达点云帧, t_m 表示 \mathbf{P}_m 的起始时间戳, $\frac{T+t_i-t_n}{t_i-t_m} \mathbf{P}_m$ 表示 \mathbf{P}_m 中局部时间戳介于 $t_i - t_m$ 与 T 之间的点云, \mathbf{P}_n 表示满足时间同步条件并且起始时间戳大于 P_i 起始时间戳的辅助雷达点云帧, t_n 表示 \mathbf{P}_n 的起始时间戳, $\frac{T+t_i-t_n}{0} \mathbf{P}_n$ 表示 \mathbf{P}_n 中局部时间戳介于 0 与 $T + t_i - t_n$ 之间的点云。对于 $\frac{T+t_i-t_n}{t_i-t_m} \mathbf{P}_m$ 和 $\frac{T+t_i-t_n}{0} \mathbf{P}_n$ 中的激光点,可以按照式(11)更新局部时间戳 t_s 为:

$$t_s = t_s + \Delta s \quad (11)$$

其中, Δs 表示主雷达点云帧与辅助雷达点云帧之间的起始时间间隔,对于 $\frac{T+t_i-t_n}{t_i-t_m} \mathbf{P}_m$ 中的激光点, $\Delta s = t_m - t_i$, 对于 $\frac{T+t_i-t_n}{0} \mathbf{P}_n$ 中的激光点, $\Delta s = t_n - t_i$ 。

4 状态估计

基于 FAST-LIO2 算法,通过前向传播预测 IMU 状态量,再通过反向传播对点云帧进行运动补偿,从而将每一帧点云映射到同一时刻,并且获取初始状态量,用于之后的迭代卡尔曼滤波过程。

4.1 前向传播

前向传播通过 IMU 积分,传播状态量 \hat{x} 和相应的协方差矩阵。根据式(3),将过程噪声设置为 0,可估计 t_{i+1} 时刻的状态量 \hat{x}_{i+1} ,计算公式如下:

$$\hat{x}_{i+1} = \hat{x}_i \boxplus (\Delta t f(\hat{x}_i, \mathbf{u}_i, 0)) \quad (12)$$

令 \tilde{x}_{i+1} 表示 t_{i+1} 时刻的状态量误差, \tilde{x}_{i+1} 具体定义如下^[16]:

$$\tilde{x}_{i+1} = \mathbf{x}_{i+1} \boxminus \hat{x}_{i+1} = (\mathbf{x}_i \boxplus \Delta t f(\mathbf{x}_i, \mathbf{u}_i, \mathbf{w}_i)) \boxminus (\hat{x}_i \boxplus \Delta t f(\hat{x}_i, \mathbf{u}_i, 0)) \simeq \mathbf{F}_{\tilde{x}} \tilde{x}_i + \mathbf{F}_w \mathbf{w}_i \quad (13)$$

其中, $\mathbf{F}_{\tilde{x}}$ 和 \mathbf{F}_w 分别表示 t_{i+1} 时刻状态量误差 \tilde{x}_{i+1} 关于 t_i 时刻状态量误差 \tilde{x}_i 和噪声项 \mathbf{w}_i 的一阶偏导。令 $\hat{\mathbf{P}}_i$ 表

示 t_i 时刻 \tilde{x}_i 的协方差矩阵,协方差矩阵 \hat{P}_{i+1} 可表示为:

$$\hat{P}_{i+1} = \mathbf{F}_{\tilde{x}} \hat{P}_i \mathbf{F}_{\tilde{x}}^T + \mathbf{F}_w Q \mathbf{F}_w^T \quad (14)$$

$$\mathbf{F}_{\tilde{x}} = \begin{bmatrix} \text{Exp}(-\hat{\omega}_i \Delta t) & \mathbf{0} & \mathbf{0} & -A(-\hat{\omega}_i \Delta t)^T & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_3 & \mathbf{I}_3 \Delta t & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ -{}^g\hat{\mathbf{R}}_{I_i} \hat{\boldsymbol{\omega}}_{m_i} - \hat{\mathbf{b}}_{\omega A} \Delta t & \mathbf{0} & \mathbf{I}_3 & \mathbf{0} & -{}^g\hat{\mathbf{R}}_{I_i} \Delta t & \mathbf{I}_3 \Delta t \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}_3 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}_3 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}_3 \end{bmatrix} \quad (15)$$

$$\mathbf{F}_w = \begin{bmatrix} -A(-\hat{\omega}_i \Delta t)^T & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -{}^g\hat{\mathbf{R}}_{I_i} \Delta t & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_3 \Delta t & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}_3 \Delta t \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \quad (16)$$

4.2 运动补偿

由于雷达不是在同一时刻采集激光点数据,机器人运动会造成雷达点云帧的畸变,可以基于4.1节中IMU积分所估计的位姿,将同一帧点云映射到同一时刻。运动补偿利用前向传播所预测的IMU位姿,计算当前激光点和帧尾激光点之间位姿变换 ${}^{l_k}\check{\mathbf{T}}_{l_j}$,其中 k 表示第 k 帧, j 表示第 k 帧中的第 j 个激光点,再通过位姿变换 ${}^{l_k}\check{\mathbf{T}}_{l_j}$ 更新激光点,计算公式如下:

$${}^{l_k}\mathbf{P}_{f_j} = {}^l\mathbf{T}_L^{-1} {}^{l_k}\check{\mathbf{T}}_{l_j} {}^l\mathbf{T}_L {}^{l_k}\mathbf{P}_{f_j} \quad (17)$$

4.3 模型构建与迭代更新

通过迭代卡尔曼滤波实现状态量的迭代更新,状态量误差是由运动模型的误差和观测模型的误差两部分组成,因此,迭代更新可以视为运动模型和观测模型的联合优化问题。

1) 观测模型线性化

在局部地图中,对雷达点云进行匹配,可构造观测方程。将残差定义为世界坐标系下的雷达点到全局地图中最近平面之间的距离为:

$$h(\mathbf{x}_k, {}^l_n_{f_j}) = G_j({}^g\mathbf{T}_{l_k} {}^{l_k}\check{\mathbf{T}}_{l_j} {}^l\mathbf{T}_L ({}^{l_j}\mathbf{P}_{f_j} - {}^l_n_{f_j}), {}^g\mathbf{q}_j) \quad (18)$$

其中, ${}^l_n_{f_j}$ 表示雷达的测量噪声, ${}^g\mathbf{q}_j$ 表示当前雷达点在全局地图中的匹配点, G_j 表示点面残差函数,具体定义如下^[7]:

$$G_j = \frac{|({}^g\mathbf{p}_j^k - {}^g\mathbf{q}_a)^T [\mathbf{r} \times \mathbf{s}]|}{|({}^g\mathbf{q}_a - {}^g\mathbf{q}_b) \times ({}^g\mathbf{q}_a - {}^g\mathbf{q}_c)|}$$

$$\mathbf{r} = {}^g\mathbf{q}_a - {}^g\mathbf{q}_b$$

$$\mathbf{s} = {}^g\mathbf{q}_a - {}^g\mathbf{q}_c \quad (19)$$

其中, ${}^g\mathbf{q}_a, {}^g\mathbf{q}_b$ 与 ${}^g\mathbf{q}_c$ 表示平面点, ${}^g\mathbf{p}_j^k$ 表示世界坐标

其中, Q 表示 t_i 时刻噪声 w_i 的协方差矩阵, $\mathbf{F}_{\tilde{x}}$ 与 \mathbf{F}_w 的具体定义如式(15)和(16),其中, $\hat{\boldsymbol{\omega}}_i = \mathbf{a}_{m_i} - \mathbf{b}_{a_i}$ ^[7]。

系下的雷达点坐标。由于 ${}^{l_j}\mathbf{P}_{f_j} - {}^{l_j}\mathbf{n}_{f_j}$ 表示原始雷达点的测量真值, \mathbf{x}_k 表示第 k 帧点云的状态量真值,点面距离残差为0。如式(20)所示,线性化观测方程,即在 $\mathbf{x}_k = \hat{\mathbf{x}}_k^\kappa$ 处进行一阶展开, $\hat{\mathbf{x}}_k^\kappa$ 表示对第 k 帧点云帧进行第 κ 次迭代时的状态量:

$$\mathbf{0} = \mathbf{z}_j^\kappa + \mathbf{H}_j^\kappa \hat{\mathbf{x}}_k^\kappa + \mathbf{v}_j \simeq \mathbf{z}_j^\kappa (\hat{\mathbf{x}}_k^\kappa, \mathbf{0}) + \mathbf{H}_j^\kappa \hat{\mathbf{x}}_k^\kappa + \mathbf{v}_j = \mathbf{z}_j^\kappa + \mathbf{H}_j^\kappa \hat{\mathbf{x}}_k^\kappa + \mathbf{v}_j \quad (20)$$

其中, \mathbf{H}_j^κ 表示观测方程关于 $\hat{\mathbf{x}}_k^\kappa$ 的雅可比矩阵, \mathbf{v}_j 来自于激光雷达的测量噪声。

2) 运动模型线性化

前向传播为迭代更新提供状态量先验,为了构造迭代优化过程中的目标函数,线性化状态量误差 $\tilde{\mathbf{x}}_k$ 为:

$$\mathbf{x}_k - \hat{\mathbf{x}}_k = (\hat{\mathbf{x}}_k^\kappa \boxplus \hat{\mathbf{x}}_k^\kappa) \boxminus \hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^\kappa \boxminus \hat{\mathbf{x}}_k + \mathbf{J}^\kappa \tilde{\mathbf{x}}\kappa_k \quad (21)$$

其中, \mathbf{J}^κ 表示状态量误差 $\tilde{\mathbf{x}}_k$ 关于 $\hat{\mathbf{x}}_k^\kappa$ 的雅可比矩阵。

3) 迭代优化

同时考虑先验分布和后验分布,基于最大后验估计,构造迭代优化的目标函数为:

$$\min_{\mathbf{x}_k} (\|\mathbf{x}_k - \hat{\mathbf{x}}_k\|_{\hat{\mathbf{P}}_k^{-1}}^2 + \sum_{j=1}^m \|\mathbf{z}_j^\kappa + \mathbf{H}_j^\kappa \hat{\mathbf{x}}_k^\kappa\|_{\mathbf{R}_j^{-1}}^2) \quad (22)$$

其中, $\|\mathbf{x}\|_M^2 = \mathbf{x}^T \mathbf{M} \mathbf{x}$, $\hat{\mathbf{P}}_k$ 表示状态量误差的协方差矩阵, \mathbf{R}_j 表示原始测量噪声的协方差矩阵。

通过高斯牛顿法对状态量进行迭代优化,迭代公式如下^[7]:

$$\tilde{\mathbf{x}}\kappa + 1_k = \hat{\mathbf{x}}_k^\kappa \boxplus (-K \mathbf{z}_k^\kappa - (I - K H) (J^\kappa)^{-1} (\hat{\mathbf{x}}\kappa_k \boxminus \hat{\mathbf{x}}_k)) \quad (23)$$

其中, K 表示卡尔曼滤波增益,具体定义如下:

$$\mathbf{K} = \mathbf{P} \mathbf{H}^T (\mathbf{H} \mathbf{P} \mathbf{H}^T + \mathbf{R})^{-1} \quad (24)$$

迭代后的 $\tilde{\mathbf{x}}\kappa + 1_k$ 用于目标函数的计算,重复迭代更新过程,直到目标函数收敛。最后,获得最优的状态量和对应状态误差的协方差:

$$\bar{\mathbf{x}}_k = \hat{\mathbf{x}}_k^{\kappa+1}, \bar{\mathbf{P}}_k = (\mathbf{I} - \mathbf{K} \mathbf{H}) \mathbf{P} \quad (25)$$

5 实验与讨论

为了验证本文所提出的算法的性能,分别在公开数据集上和移动实验平台上进行了实验验证。开源数据集选

用 NTU VIRAL, 该数据集使用自主飞行器在南洋理工大学校园内采集, 包含视觉惯性测距激光雷达数据, 使用了其中位于南洋理工大学中央停车场、生物学院广场以及研究科技广场停车场 3 个场景的数据集。此外, 本文搭建了用于在线测试的移动机器人平台, 如图 3 所示, 车辆的左右两侧分别搭载 RS-LIDAR-16 型激光雷达。其采样频率为 10 Hz, 水平分辨率为 0.2°, 垂直分辨率为 2°, 左侧激光雷达作为作为主雷达。主雷达的正上方安装 YIS512 型号 IMU, 采样频率为 200 Hz。此外, 采用 Ubuntu20.04 系统, ROS 版本为 noetic。



图 3 实验数据采集平台

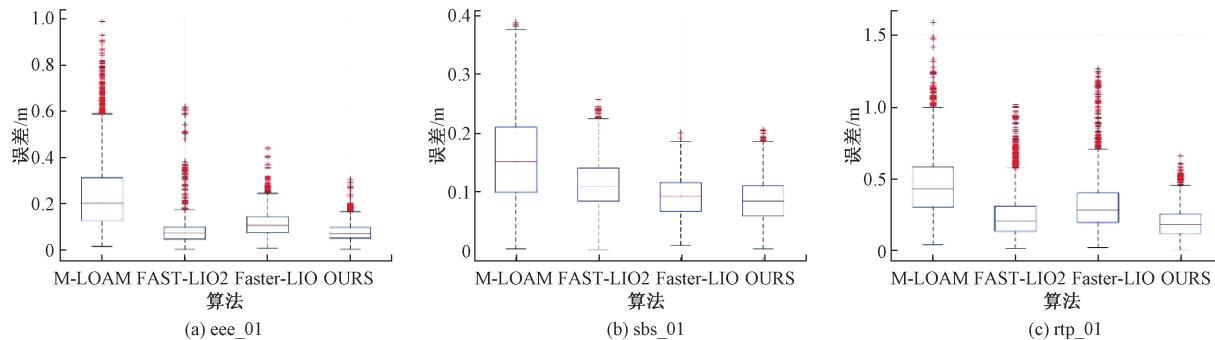


图 4 误差分布对比图

表 1 开源数据集绝对轨迹误差结果对比表 m

| 数据包 | 算法 | 最大 误差 | 最小 误差 | 平均 误差 | 均方根 误差 |
|--------|------------|----------|----------|----------|-----------|
| eee_01 | M-LOAM | 0.988 | 0.016 | 0.205 | 0.286 |
| | FAST-LIO2 | 0.616 | 0.003 | 0.067 | 0.094 |
| | Faster-LIO | 0.443 | 0.009 | 0.103 | 0.125 |
| | Ours | 0.289 | 0.006 | 0.067 | 0.084 |
| sbs_01 | M-LOAM | 0.388 | 0.004 | 0.142 | 0.177 |
| | FAST-LIO2 | 0.257 | 0.003 | 0.097 | 0.118 |
| | Faster-LIO | 0.200 | 0.009 | 0.082 | 0.097 |
| | Ours | 0.194 | 0.001 | 0.075 | 0.092 |
| rtp_01 | M-LOAM | 1.590 | 0.041 | 0.391 | 0.506 |
| | FAST-LIO2 | 1.017 | 0.013 | 0.220 | 0.299 |
| | Faster-LIO | 1.262 | 0.020 | 0.276 | 0.364 |
| | Ours | 0.658 | 0.004 | 0.170 | 0.219 |

5.1 开源数据集实验

选取了 3 种不同场景下的数据集 eee_01、sbs_01、rtp_01, 用于测试 M-LOAM、FAST-LIO2、Faster-LIO 与 M-LIO 的定位性能, 在所有算法测试过程中, 公共参数均保持一致。如图 4 所示, 为各算法关于位置误差的箱线图, 纵轴表示预测位置与激光跟踪仪所测算的目标位置之间的偏差, 横轴为 0~1 之间的随机数, 箱框上边缘线和下边缘线分别表示误差的上四分位和下四分位, 箱框上的横线表示误差的中位数, 十字标识符表示离群误差, 上横线与下横线分别表示非离群点的最大与最小值。图 4(a)中各算法分别包含 2 961 组误差, 图 4(b)中各算法分别包含 2 720 组误差, 图 4(c)中各算法分别包含 2 919 组误差。由图 4(a)可知, 相较于 M-LOAM 和 Faster-LIO, M-LIO 的框线和上四分位向下移动, 并且离群误差明显变小, 相较于 FAST-LIO2, M-LIO 的离群误差总体变小, 而其框线变化不明显。由图 4(b)可知, 相比于 M-LOAM、FAST-LIO2 和 Faster-LIO, M-LIO 的框线、上四分位以及离群点均向下移动。由图 4(c)可知, 相比于 M-LOAM 和 Faster-LIO, M-LIO 的框线、上四分位以及离群点向下移动。相较于 FAST-LIO2, M-LIO 的箱框上边界和离群点向下移

动。因此, M-LIO 的整体误差低于 FAST-LIO2、M-LOAM 和 Faster-LIO。

通过最小误差、最大误差、平均误差和均方根误差对各个算法进行定量分析, 由表 1 可知, M-LIO 的定位误差明显小于 M-LOAM 算法, 由此可验证基于 IMU 紧耦合多雷达 SLAM 方案的定位精度优于 IMU 松耦合多雷达 SLAM 方案。此外, 在 3 个不同的场景下, M-LIO 算法的定位误差均小于 Faster-LIO 算法, 并且 M-LIO 算法的最大误差和均方根误差均小于 FAST-LIO2 算法, 由此可验证, 相较于 FAST-LIO2 算法和 Faster-LIO 算法, M-LIO 具有更强的鲁棒性和更高的定位精度。

如图 5(a)~(c)所示, 分别为 M-LIO 算法在 eee_01、sbs_01 和 rtp_01 数据集下的所构建的点云图。

5.2 自测数据集测试

为了进一步验证 M-LIO 的定位精度和建图效果, 将 M-LIO 分别与 FAST-LIO2 算法、M-LOAM 算法和

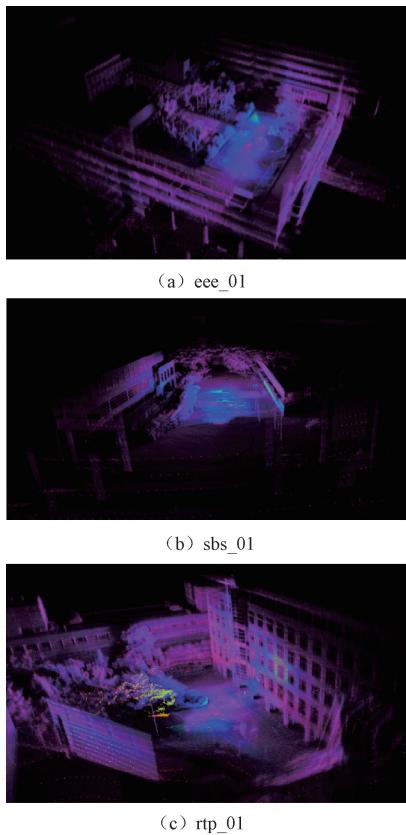


图 5 使用 M-LIO 对公开数据进行建图的结果

Faster-LIO 算法进行在线测试,实验场景选取芜湖鸠江区机器人成果转化中心 3 号楼、5 号楼以及 7 号楼周围车道,3 次测试的路径全长分别约为 400 m、400 m 和 200 m。

1) 定位精度对比

小车回环前后的真实位置通过 A7960 型激光跟踪仪采集,如图 6 所示,为 M-LIO 关于 FAST-LIO2、M-LOAM 和 Faster-LIO 算法的均方根误差对比图,图中前 17 组数据为回环之前的定位误差,后 20 组数据为回环之后的定位误差。定位误差的计算过程包括:首先根据前 17 组激光跟踪仪坐标系下的位置与 IMU 全局坐标系下的位置,计算 IMU 全局坐标系到激光跟踪仪坐标系的刚体变换,再将 38 组 SLAM 算法的估计位置坐标通过变换矩阵转换到激光跟踪仪坐标系下,转换后的位置坐标与激光跟踪仪所测算的位置坐标之间的绝对差值即为位置估计的绝对轨迹误差。由图 7 可知,相比于 FAST-LIO2 算法、M-LOAM 算法与 Faster-LIO 算法, M-LIO 明显降低了回环误差。M-LIO 的定位误差与 FAST-LIO2 算法、M-LOAM 和 Faster-LIO 算法在回环之前基本一致,但 M-LIO 的定位误差在回环之后明显低于 FAST-LIO2 算法 M-LOAM 算法以及 Faster-LIO 算法。回环之后,由于移动小车回到出发点附近,匹配的初始点云增多,因此各算法的定位误差局部降低。

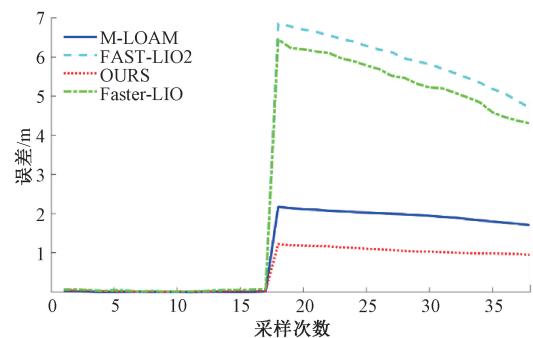


图 6 均方根误差对比图

自测室外场景下,各算法误差如表 2 所示,由表 2 可知,M-LIO 的定位误差明显低于 M-LOAM 算法、FAST-LIO2 算法与 Faster-LIO 算法,由此可验证,将 IMU 紧耦合与多激光雷达相结合的 SLAM 方案可以在一定程度上减少漂移现象,从而提高定位精度。

表 2 在线测试结果

| 数据包 | 算法 | 最大误差 | 平均误差 | 均方根误差 |
|-----|------------|---------|---------|---------|
| 测试一 | M-LOAM | 2.174 0 | 1.088 7 | 1.461 8 |
| | FAST-LIO2 | 6.846 1 | 3.294 5 | 4.432 8 |
| | Faster-LIO | 6.446 1 | 3.011 5 | 4.055 2 |
| | OURS | 0.950 7 | 0.602 0 | 0.799 9 |
| 测试二 | M-LOAM | 2.374 0 | 1.168 2 | 1.575 2 |
| | FAST-LIO2 | 5.652 4 | 2.826 4 | 3.788 6 |
| | Faster-LIO | 5.946 1 | 3.056 2 | 4.099 6 |
| | OURS | 1.331 8 | 0.671 6 | 0.892 9 |
| 测试三 | M-LOAM | 1.590 0 | 0.664 2 | 0.896 1 |
| | FAST-LIO2 | 2.162 0 | 0.982 0 | 1.314 3 |
| | Faster-LIO | 1.911 8 | 0.943 8 | 1.257 3 |
| | OURS | 1.347 1 | 0.612 2 | 0.812 7 |

2) 建图效果对比

由图 7 可知,M-LIO 算法所构建的地图中车辆的位置偏移小于 FAST-LIO2 和 M-LIO 算法所构建地图中车辆

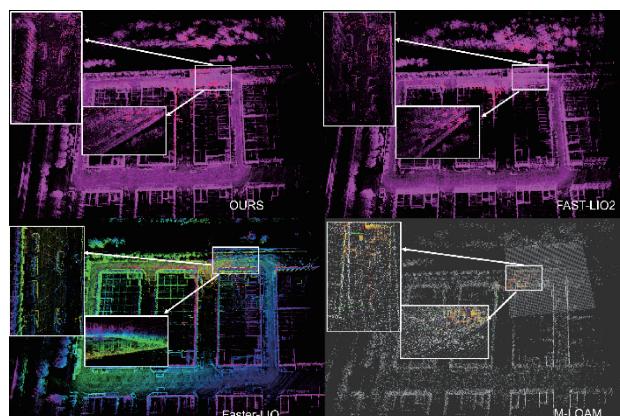


图 7 建图效果对比图

的位置偏移，并且 M-LIO 地图中的围墙在 Z 轴方向上无明显偏移，而 FAST-LIO2 和 Faster-LIO 算法所构建的地图中的围墙在 Z 轴方向上存在明显偏移，因此相较于 FAST-LIO2、Faster-LIO 和 M-LIO 算法，所提出的算法显著提升了建图效果。

6 结 论

本文提出了一种基于 IMU 紧耦合的多激光雷达定位与建图方法。该方法融合了来自多个不同安装位置激光雷达的点云数据，并采用紧耦合的迭代卡尔曼滤波器将 IMU 和雷达数据进行融合定位算法研究。在拓展机器人感知范围的同时也在一定程度上改善了机器人长时间移动引起的漂移现象。本文在开源数据集以及自主搭建的数据采集平台上对所提出的 M-LIO 与 FAST-LIO2、M-LOAM 以及 Faster-LIO 算法进行了对比离线实验和在线实验。结果表明，所提出的 M-LIO 在定位精度和建图效果优于 FAST-LIO2、M-LOAM 与 Faster-LIO 算法。

参考文献

- [1] 张继贤, 刘飞. 视觉 SLAM 环境感知技术现状与智能化测绘应用展望 [J]. 测绘学报, 2023, 52(10): 1617-1630.
- [2] 周治国, 曹江微, 邸顺帆. 3D 激光雷达 SLAM 算法综述 [J]. 仪器仪表学报, 2021, 42(9): 13-27.
- [3] 马宝良, 崔丽珍, 李敏超, 等. 露天煤矿环境基于 LiDAR/IMU 的紧耦合 SLAM 算法 [J]. 煤炭科学技术, 2024, 52(3): 236-244.
- [4] 许志华, 彭远航, 林佳轩, 等. 单线激光雷达与测距仪耦合的移动式仓储存量盘点装备研发与应用 [J/OL]. 金属矿山, 2023, 1-12. <http://kns.cnki.net/kcms/detail/34.1055.TD.20231110.1459.004.html>.
- [5] 鲍雪. 基于三维激光雷达的巡检机器人定位与建图技术研究与应用 [D]. 南京: 东南大学, 2021.
- [6] ZHANG J, SINGH S. LOAM: Lidar odometry and mapping in real-time [C]. Robotics: Science and Systems Conference(RSS), 2014.
- [7] SHAN T, ENGLOT B. LeGO-LOAM: Lightweight and ground-optimized lidar odometry and mapping on variable terrain [C]. IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS), 2018.
- [8] JIAO J, YE H, ZHU Y, et al. Robust odometry and mapping for Multi-LiDAR systems with online extrinsic calibration [J]. IEEE Transactions on Robotics, 2022, 38(1): 351-371.
- [9] XU W, ZHANG F. FAST-LIO: A fast, robust LiDAR-Inertial odometry package by tightly-coupled iterated kalman filter [J]. IEEE Robotics and Automation Letters, 2021, 6(2): 3317-3324.
- [10] XU W, CAI Y, HE D, et al. FAST-LIO2: Fast direct LiDAR-Inertial odometry [J]. IEEE Transactions on Robotics, 2022, 38(4): 2053-2073.
- [11] BAI C, ZHANG F, GAO X, 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.
- [12] CAI Y, XU W, ZHANG F. Ikd-tree: An incremental K-D tree for robotic applications [J]. ArXiv preprint arXiv:2211.15444, 2022.
- [13] 万乐民. 融合双激光雷达的 SLAM 方法 [D]. 长沙: 中南大学, 2022.
- [14] HEIDE N, EMTER T, PETEREIT J. Calibration of multiple 3D LiDAR sensors to a common vehicle frame [C]. 50th International Symposium on Robotics(ISR), 2018.
- [15] 张普, 刘金清, 肖金超, 等. 基于相机与激光雷达融合的目标定位与跟踪 [J]. 激光与光电子学进展, 61(8): 0828004.
- [16] YANG X, LIN X, YAO W, et al. A robust LiDAR SLAM method for underground coal mine robot with degenerated scene compensation [J]. Remote Sensing, 2023, 15(1): 186.

作者简介

李倩, 硕士研究生, 主要研究方向为机器人定位导航技术及应用。

E-mail: liqian_@ahnu.edu.cn

陈付龙, 博士, 教授, 主要研究方向为嵌入式与普适计算、信息物理融合系统、高性能计算机体系结构、物联网安全。
E-mail: long005@ahnu.edu.cn

郑亮, 硕士, 工程师, 主要研究方向为机器人建模规划与控制。

E-mail: zhengliang_happy@126.com

赵法龙, 硕士, 工程师, 主要研究方向为视觉及激光多传感器融合 SLAM 定位建图技术及应用。

E-mail: zhaofalong4612@gmail.com

陈智君(通信作者), 硕士, 工程师, 主要研究方向为机器视觉及机器人定位导航技术及应用。

E-mail: cvkaka@foxmail.com