

AF-RLIO: 雷达-激光雷达-惯性信息自适应融合用于在挑战性环境下实现鲁棒里程计

Chenglong Qian^{1,†}, Yang Xu^{2,†}, Xiufang Shi¹, Jiming Chen², and Liang Li^{2,*}

Abstract—在机器人导航中,保持精确的姿态估计和在复杂、动态环境中的导航是至关重要的。然而,环境挑战,如烟雾、隧道和恶劣天气,可能显著降低单一传感器系统(如 LiDAR 或 GPS)的性能,从而影响自主机器人的整体稳定性和安全性。为了解决这些问题,我们提出了一种自适应融合方法:AF-RLIO。该方法整合了 4D 毫米波雷达、LiDAR、惯性测量单元(IMU)和 GPS,利用这些传感器的互补优势实现复杂环境中的鲁棒里程计估计。我们的方法由三个关键模块组成。首先,预处理模块利用雷达数据协助 LiDAR 移除动态点,并在环境条件对 LiDAR 不利时作出判断。其次,动态感知多模态里程计模块选择合适的点云数据进行扫描到地图匹配,并使用迭代误差状态卡尔曼滤波器与 IMU 紧密耦合。最后,因子图优化模块在里程计和 GPS 数据之间平衡权重,构建一个姿态图进行优化。该方法已经在数据集上进行了评估,并在实际的机器人环境中进行了测试,证明了其在诸如烟雾和隧道等挑战性条件下的有效性和相对于现有方法的优势。此外,我们在 <https://github.com/NeSC-IV/AF-RLIO.git> 上开源了我们的代码,以惠及研究社区。

I. 介绍

对于自主机器人来说,当进入具有挑战性的环境时,稳健可靠的里程计至关重要。在如烟雾弥漫的区域或狭窄隧道等复杂环境中,基于激光雷达的融合算法在同时定位与地图构建(SLAM)方面面临显著挑战[1]–[3]。烟雾和粉尘等颗粒物降低了激光雷达扫描的质量,严重影响了点云配准[3]。在这些环境中,传感器数据的噪声和不确定性增加,使得仅依赖激光雷达和惯性测量单元(IMU)的里程计系统准确性下降。此外,在狭窄或封闭的环境中,例如隧道、密集的城市区域和峡谷,保持可靠的全球定位系统(GPS)信号变得困难,GPS 异常值可能导致系统故障[4]–[6]。因此,为了在极端天气和具有挑战性的条件下实现准确的里程计估计,选择更稳健的传感器和实施适当的多模态融合是至关重要的。

与 LiDAR 相比,4D 毫米波雷达具有更长的波长,并且不易受到如雨、雪和雾等恶劣天气条件的影响[7]。

此外,其感知更远距离和捕获多普勒信息的能力可以增强在具有相似几何特征的环境中里程计估计的鲁棒性。然而,毫米波雷达点云比激光雷达生成的点云更稀疏且不太稳定。

为此,我们提出了一种以 IMU 为中心的多传感器融合方法。我们的多模态融合策略利用雷达点云帮助 LiDAR 在 LiDAR 点云有效时去除动态障碍物,同时紧密融合

¹College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023, China. ²College of Control Science and Engineering, Zhejiang University, Hangzhou 310027, China. liang.li@zju.edu.cn

This work was supported by National Natural Science Foundation of China (NSFC) under Grants 62203383, U23A20326, 62372414, and 62403420, the National Postdoctoral Program for Innovative Talents under Grant BX20240317, and the China Postdoctoral Science Foundation under Grant 2024M752792. (* : Corresponding author, †: equal contribution.)

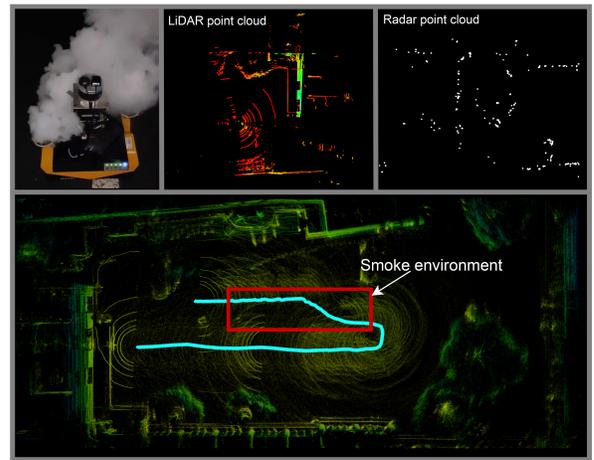
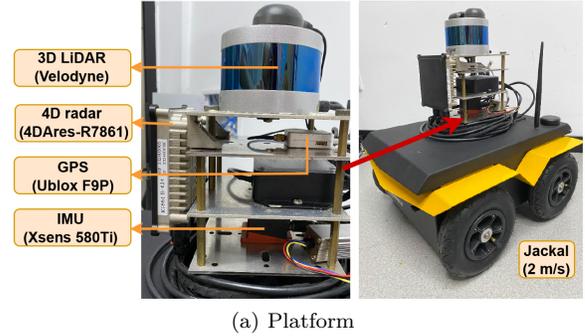


Fig. 1. (a) 所提系统的测试平台:一辆速度约为 2m/s 的 Jackal 车。(b) 在烟雾环境中显示激光雷达和雷达点云,并验证 AF-RLIO 对雷达数据的自适应融合以完成测距和建图。

LiDAR 和 IMU 数据以确保可靠的里程计。在 LiDAR 性能降低的环境中,系统通过切换至毫米波雷达与 IMU 信息的紧密融合来防止里程计漂移。

总的来说,我们的主要贡献是:

- 我们提出了一种自适应多模态测距系统,该系统可以在 IESKF 框架内选择性地将雷达和 LiDAR 与 IMU 紧密结合,然后使用姿态图优化松散融合加权 GPS 信息。该系统充分利用每个传感器在不同环境条件下的优势,实现稳健且准确的定位,即使在 LiDAR 或 GPS 性能下降的挑战条件下也能稳定运行。
- 我们设计了一种鲁棒的 GPS 异常值检测方法,该方法利用雷达估算的自车速度和动态感知的多模态里程计来识别 GPS 异常值。系统能够平滑调整 GPS 权重,以减轻异常值的影响,从而提高整体系统的可靠性。
- 在多个数据集和真实场景中进行了广泛的实验,证

明了所提算法的准确性和稳健性。

II. 相关工作

A. 激光雷达-惯性测距法

激光雷达惯性测程 (LIO) 主要可以分为基于滤波的 [8]–[12] 和基于优化的方法 [13]–[16]。

基于滤波的方法，例如扩展卡尔曼滤波器，通常实时递归和平滑地更新系统状态向量和协方差，以融合新的 LiDAR 和 IMU 测量。例如，FAST-LIO2 [10] 采用迭代误差状态卡尔曼滤波器 (IESKF)，无需通过优化器来精细化地图点或机器人状态。它直接将点注册到地图中，无需特征提取，充分利用环境信息并适应新兴的 LiDAR 技术。另一方面，基于优化的方法将 LIO 问题表述为非线性优化，旨在最小化滑动窗口或整个轨迹上预测测量值与观测测量值之间的残差。

然而，LiDAR 易受环境因素影响，在烟雾或极端天气条件下，可能无法生成注册帧 [17]。即使受到 IMU 的约束，也可能无法维持长期稳定可靠的里程计。

近些年来，基于雷达的 SLAM 方法在机器人技术中，特别是在自动驾驶领域获得了越来越多的关注。这些方法利用了雷达传感器的独特优势，例如在恶劣天气条件（如雾天、下雨和扬尘）下的鲁棒性，这通常会降低基于 LiDAR 和摄像头系统的性能。这也促进了雷达惯性里程计 (RIO) 的发展，其中涉及雷达扫描匹配和多传感器融合的研究。在基于 EKF 的 RIO 方法中，惯性数据、基于雷达的自速度估计和雷达点云配准紧密结合以估计机器人的三维位置和姿态。此外，受到基于优化的 LIO 方法的启发，研究人员将 RIO 问题表述为因子图，从而实现滑动窗口或整个轨迹的姿态估计的联合优化。

B. 雷达-激光雷达-惯性测距法

为了充分利用激光雷达和雷达的优势，最近的研究集中于融合这两种传感器进行定位和制图 [18]–[20]。在雾、尘或光线过低导致激光雷达性能下降的场景中，雷达仍能提供可靠的测量。反之，在良好条件下，激光雷达的高分辨率数据可以精细化地图并提高定位精度。例如，DR-LRIO [21] 通过图优化紧密结合激光雷达、雷达和 IMU。它将雷达估计的线速度和位姿差异估计整合到因子图中，从而在环境恶化时实现稳健的里程计。然而，雷达数据的长期融合可能由于雷达误差较大而降低整体定位精度。[22] 的方法通过识别激光雷达点云中的激光雷达易退化点来选择作为里程计估计的点。然而，仅依赖雷达或激光雷达，而没有固有感知传感器，如 IMU 和 GPS，仍然在实现稳健定位方面面临重大挑战。

因此，本文提出了一种选择性融合方法，该方法根据环境条件自适应地利用不同的外部感知传感器，以应对不利的环境变化，确保机器人里程计的可靠性和准确性，特别是在感知退化的环境中，如烟雾、隧道和高度动态的场景。

III. 方法

A. 概述

我们的 AF-RLIO 系统的概览如图 2 所示，系统由三个主要模块组成：点云预处理、动态感知多模态里程计和自适应因子图优化。点云预处理模块利用 4D 毫米波雷达、LiDAR 和 IMU 从雷达点云中提取动态和静态点。对齐后的雷达点云随后用于辅助 LiDAR 去除动态

障碍物。动态感知多模态里程计模块通过判断预处理后的 LiDAR 点云是否退化，选择合适的点云数据。当机器人进入 LiDAR 失效的环境时，系统自动切换到雷达惯性系统以维持里程计。当环境恢复正常时，系统切换回 LiDAR 惯性系统进行 SLAM。因子图优化模块结合了 GPS 数据，其中 GPS 因子根据环境变化自动调整其权重以处理异常，最终生成平滑且可靠的里程计输出。

B. 点云预处理

1) 雷达预处理：这款 4D 雷达能够提供 3D 位置和多普勒速度的测量。然而，原始 4D 雷达数据往往受到噪声和杂波的污染。首先，雷达的多普勒速度用于初步估计自行车速度，这有助于在单帧中区分动态和静态点云。需要注意的是，这些动态点并不一定代表动态障碍物。它们通常由移动的机器人或行人生成，并且具有高点云密度和一致的多普勒信息。为了进一步提取动态点云，我们使用密度聚类算法 DBSCAN [23]。该方法根据周围数据点的密度识别聚类，能够有效地过滤掉噪声点，并去除具有不一致多普勒信息的低密度点云聚类，从而确保准确提取动态机器人和行人，如图 3 所示。雷达自行车速度通过文献 [24] 中提出的线性最小二乘法进一步优化，剔除地面下的错误点并获得可用的静态点云。然后，将点云分割成三个互斥的子集：动态点集 $(\cdot)^D$ ，静态点集 $(\cdot)^S$ 和噪声点集 $(\cdot)^N$ 。形式上，这种分割表示为：

$$R = R^S \cup R^D \cup R^N, \quad (1)$$

其中 $(\cdot)^S \cap (\cdot)^D \cap (\cdot)^N = \emptyset$ 。

2) 激光雷达预处理：由于 LiDAR 本身不提供速度信息，因此 LiDAR 点云中的动态障碍物会对点云配准的准确性产生负面影响。因此，仅依靠 LiDAR 来快速准确地去除动态障碍物具有挑战性。为了解决这个问题，我们的系统使用雷达来辅助实时移除每个 LiDAR 帧中的动态点。在这个过程中，输入的是雷达动态点云 R^D 和 LiDAR 点云 L 。进行坐标对齐以确保来自不同传感器的数据在一个帧内的准确整合，包括空间校准和时间同步。高频 IMU 数据用于通过反向传播将雷达帧与 LiDAR 帧对齐。经过精确处理后，从雷达中提取的动态点有效地表示动态障碍物的位置。然而，由于两个传感器之间点云稀疏性的差异，实现 LiDAR 和雷达点之间的精确对应具有挑战性。为了解决这一问题，我们使用 kd-tree 结构来组织两个点云，并采用基于欧几里得距离的计算来判断 LiDAR 点云中的点是否对应于动态点。具体来说，如果 LiDAR 点 L_i 和动态雷达点 R_j^D 之间的欧几里得距离低于某个阈值，则该 LiDAR 点被分类为动态点，并从 LiDAR 点云中移除。如图 3 所示，可以获得排除动态点的过滤 LiDAR 点云集 L^{filtered} ：

$$L^{\text{filtered}} = \left\{ L_i^S \in L \mid \min_{j \in \{1, 2, \dots, n\}} d(L_i, R_j^D) \leq \epsilon \right\}, \quad (2)$$

其中 $d(\cdot)$ 表示欧几里得距离。 i 和 j 分别代表 LiDAR 和雷达点云序列。

C. 动态感知多模态里程计

我们用 \mathbf{x} 表示机器人状态，在第 i 个 IMU 采样时间的状态表示为：

$$\mathbf{x} = [\mathbf{R}_i^T \mathbf{p}_i^T \mathbf{v}_i^T \mathbf{b}_w^T \mathbf{b}_a^T \mathbf{g}^T]^T, \quad (3)$$

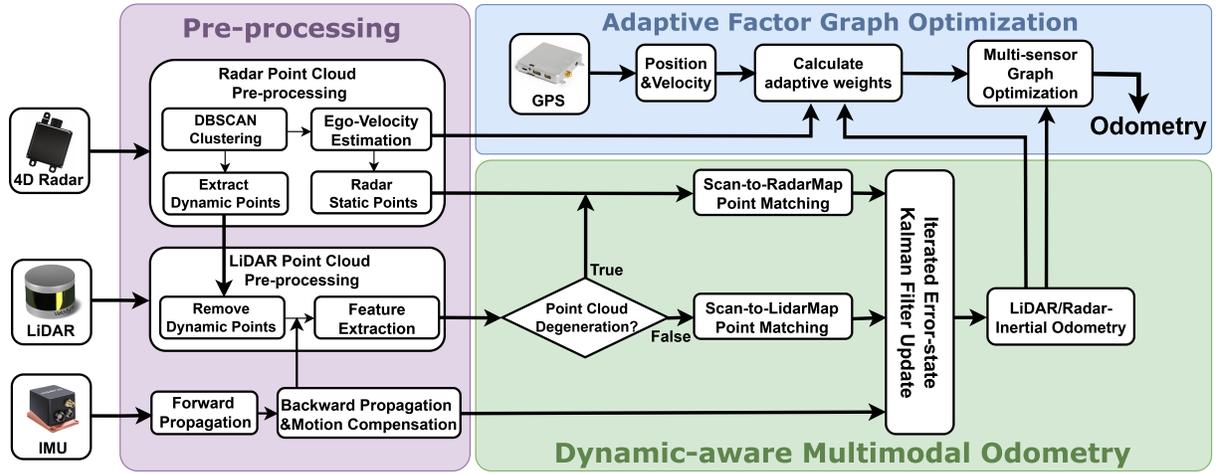


Fig. 2. AF-RLIO 的框架概述。该方法可以自适应地切换激光雷达和雷达的融合。多模态融合策略利用雷达点云协助激光雷达消除动态障碍物。在激光雷达点云不可用的环境中，系统切换到紧密集成雷达和 IMU 信息以获得里程计，该里程计在后端优化中进一步使用 GPS 进行精化。

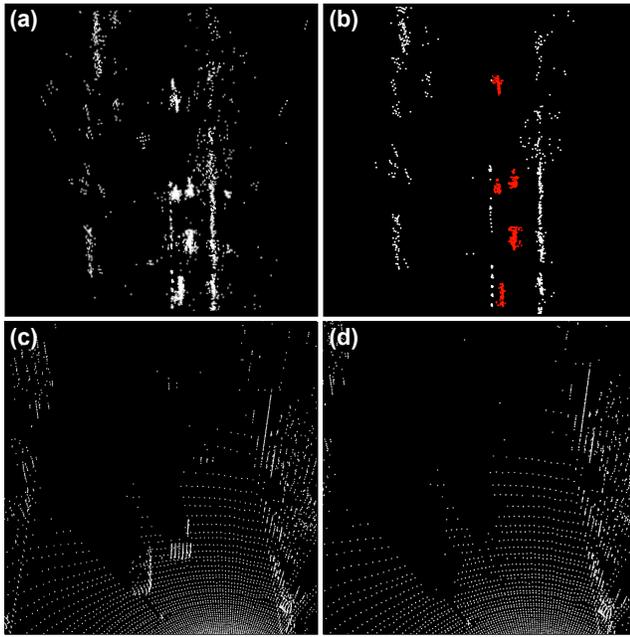


Fig. 3. 预处理前后的 LiDAR 和 4D 雷达成点云。(a) 和 (c): 原始雷达和 LiDAR 点云; (b): 雷达成点云经过动态-静态分离和噪声移除后的结果; (d): 去除动态障碍物后的 LiDAR 点云。

其中 \mathbf{R}_i 表示在 $SO(3)$ 上的旋转矩阵, \mathbf{p}_i 和 \mathbf{v}_i 分别代表位置和速度。项 \mathbf{b}_ω 和 \mathbf{b}_a 分别对应于来自 IMU 的陀螺仪和加速度计的偏差。此外, \mathbf{g} 表示世界坐标系中的重力向量, 需要对其进行估计。误差状态可以在工作点附近的切空间中表示为 $\delta\mathbf{x} \in \mathbb{R}^{18}$ 。收到高频 IMU 数据后, 通过前向传播首先估计机器人的当前状态, 以获得误差状态 $\tilde{\mathbf{x}}_{i+1}$ 和协方差 $\hat{\mathbf{P}}_{i+1}$ 。

与逐帧匹配相比, 扫描到地图匹配提供了更高的稳定性和精度 [10]。因此, 我们采用一种基于 ikd-tree 的 [25] 扫描到地图匹配方法, 这有助于实时计算。在典型环境中, LiDAR 生成密度相对稳定且高精度的点云, 使其非常适合与地图进行匹配。然而, 在如隧道或烟雾弥漫的环境中, LiDAR 点云的密度会降低, 并且从点云中提取

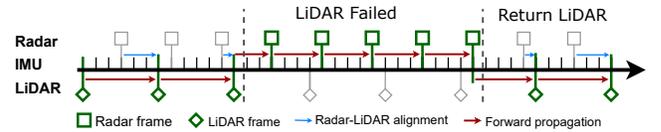


Fig. 4. 本文提出了动态感知多模态里程计。如图所示, IMU 测量作为架构中的驱动力, 通过与 LiDAR 或雷达的紧密耦合实现稳健的里程计。

明显的形状特征变得具有挑战性, 使得成功匹配变得困难。因此, 在与 IMU 紧密结合之前, 评估当前帧中点云的质量是至关重要的。

在去除动态点后, 我们从激光雷达数据中提取特征点。当特征点的比例持续低于总点云的百分之一时, 认为机器人进入了退化环境。在这种情况下, 如图 4 所示, 雷达成点云可以替代激光雷达成点云进行扫描到地图的匹配。然而, 由于雷达和激光雷达之间的点云分辨率和特征特性的差异, 直接匹配可能因雷达成点云相对稀疏而导致较差的鲁棒性和错误匹配造成的潜在系统故障。为了解决这个问题, 在从激光雷达-惯性子系统切换到雷达-惯性子系统之前, 我们预先构建雷达成子地图, 并在切换后执行雷达扫描到雷达成子地图的匹配。同样, 当激光雷达成点云密度和特征点数量恢复正常时, 我们预先构建激光雷达成子地图并从雷达-惯性子系统返回到激光雷达-惯性子系统。将激光雷达扫描匹配到激光雷达成子地图, 允许在迭代卡尔曼滤波器中进行状态更新, 从而减轻子系统切换期间由于初始匹配错误引起的跳变误差。

我们采用 IESKF 方法来迭代地更新观测模型。在每次迭代中, 计算当前变量 $\delta\mathbf{x}$ 并用于推导下一次迭代的状态向量 \mathbf{x} 和协方差矩阵 \mathbf{P} , 直到达到收敛。基本上, 每次迭代涉及求解一个带有先验的最小二乘问题:

$$\min_{\tilde{\mathbf{x}}_k} \left(\|\mathbf{x}_k \ominus \hat{\mathbf{x}}_k\|_{\hat{\mathbf{P}}_k}^2 + \sum_{j=1}^m \|r(\tilde{\mathbf{x}}_k^j)\|_{\mathbf{R}_j}^2 \right), \quad (4)$$

其中 k 表示第 j 次 LiDAR 扫描的结束时间, $\mathbf{x}_k \ominus \hat{\mathbf{x}}_k$ 表示状态预测误差, $\hat{\mathbf{P}}_k$ 表示传播协方差, $r(\tilde{\mathbf{x}}_k^j)$ 是观测残差, 表示当前状态估计与观测数据之间的误差。 \mathbf{R}_j 是第 j 个点的噪声协方差。

D. 自适应因子图优化

GPS 能够在开放环境中提供准确的位置信息，但在隧道或密集的城市区域等场景中可能会受到干扰或阻塞。GPS 异常值还可能严重扭曲基于融合的里程计。为了快速有效地检测 GPS 异常值，我们提出了一种基于残差 χ^2 检验的方法，该方法通过检查里程计姿态数据和速度估计之间的一致性来评估 GPS 测量的可靠性。通过在因子图优化中调整 GPS 权重，可以减轻异常值的影响。我们将两帧之间的姿态变化 $\Delta \mathbf{p}$ 和速度 \mathbf{v} 定义为 χ^2 测试的一致性标准。姿态变化来源于 LiDAR/雷达-惯性系统，而速度则使用过滤后的雷达静态点云 R^S 估算。因此，我们将残差 \mathbf{e}_k 定义为：

$$\mathbf{e}_k = \begin{bmatrix} \mathbf{e}_{\Delta P} \\ \mathbf{e}_V \end{bmatrix} = \begin{bmatrix} \Delta \mathbf{p}_{RLIO} - \Delta \mathbf{p}_{GPS} \\ \mathbf{v}_{Radar} - \mathbf{v}_{GPS} \end{bmatrix}. \quad (5)$$

如果 GPS 正常运行，残差 \mathbf{e}_k 应遵循零均值高斯分布；否则，它将有偏差。判决变量 λ_k 被定义为加权平方残差的和，其中权重由相关协方差矩阵 \mathbf{A} 提供：

$$\lambda_k = \mathbf{e}_k^T \mathbf{A}^{-1} \mathbf{e}_k. \quad (6)$$

由于 GPS 表示一个六维状态，离群值检验函数 λ_k 对应于一个 6 自由度 χ^2 分布。在计算了第 k 帧的检验统计量 λ_k 之后，我们对 λ_k 进行如下处理：

$$\begin{cases} \text{Normal,} & \text{if } \lambda_k < T_{min} \\ \text{Uncertain,} & \text{if } T_{min} \leq \lambda_k \leq T_{max} \\ \text{Abnormal,} & \text{if } \lambda_k > T_{max} \end{cases}. \quad (7)$$

当异常值测试函数 λ_k 小于 T_{min} 时，表示可以完全信任 GPS 数据。在这种情况下，因子图优化可以更多地依赖 GPS 数据来减少累计误差。如果测试函数超过 T_{max} ，则表示 GPS 数据不可靠，优化将直接移除 GPS 因子以消除异常值的影响。当 λ_k 处于中间阈值范围内时，GPS 数据被视为存疑。在长时间未整合 GPS 数据的场景中，GPS 数据与里程计位姿之间可能会出现差异。在这种情况下直接加入 GPS 因子可能导致里程计跳跃甚至系统故障。因此，处于中间阈值的 GPS 数据可作为优化过程中的过渡因子，确保里程计的平滑融合，同时逐渐与 GPS 数据对齐或适当偏离。当 GPS 数据处于不确定区域时，会应用平滑函数以促进这种过渡：

$$\rho_k = \alpha \mathbf{e}_k + (1 - \alpha) \rho_{k-1}. \quad (8)$$

调节该过渡使用的平滑系数为 α （其中 $0 < \alpha < 1$ ）。当 GPS 测量被视为位姿优化过程中的一个约束时，所得到的 ρ_k 用于代价函数中，而不仅仅是简单地最小化平方残差之和：

$$\arg \min_{\mathbf{x}} \sum \|r(\mathbf{z}_{RLIO}, \mathbf{x})\| + \sum \rho_k \|r(\mathbf{z}_{GPS}, \mathbf{x})\|, \quad (9)$$

其中 $r(\cdot)$ 表示传感器测量因子的残差。此方法允许逐步整合 GPS 数据，使优化过程对 GPS 和里程计数据之间的潜在差异更加鲁棒，从而保持整个系统的稳定性。

IV. 实验

我们使用 MSC 数据集 [26]、Snail 数据集 [27] 以及从真实环境中收集的烟雾数据评估了我们的 AF-RLIO 系统。我们计算了绝对位姿误差 (APE) 和相对位姿误差 (RPE) 的均方根误差 (RMSE) 作为指标 [28]。

A. MSC 数据集

为了比较，我们选择了广泛认可的 LiDAR 惯性系统 FAST-LIO2 [10]、雷达惯性 SLAM 方法 RIO 和 LiDAR 雷达惯性融合方法 LRIO 作为基准。其中，RIO 是从我们的方法中衍生出来的，禁用了 LiDAR，而 LRIO 同时使用 LiDAR 和雷达。

选定的场景包括 U_C0、U_D0 和 U_F0，这些场景的特点是具有高度动态障碍物。此外，R_A2 代表一个雪景，而 U_B0 对应于一个隧道穿越场景。U_A0 是一个静态环境。我们的评估主要集中在 RPE 和 APE 的指标上，使用 RTK 作为真值。

TABLE I
APE (m) / RPE (m) of SLAM systems on MSC Dataset

Seq ID	FAST-LIO2 [10]	RIO	LRIO	AF-RLIO
U_A0	1.80/0.256	9.62/0.357	1.88/0.268	1.80/0.256
U_B0	21.67/*	9.10/*	5.05/*	2.59/*
U_C0	1.30/0.676	1.25/0.683	1.28/0.688	1.28/0.672
U_D0	1.31/1.495	2.64/6.701	3.70/1.565	1.04/1.490
U_F0	1.91/1.202	4.42/3.139	2.20/1.201	1.48/1.202
R_A2	2.81/1.360	10.95/4.741	11.46/1.378	2.76/1.359

不同算法在 MSC 数据集上的位姿误差结果显示在表 I 中。在这些场景中，所提出的 AF-RLIO 系统通过去除动态障碍物来增强点云配准的精度，从而在高度动态的环境中提高定位的准确性。AF-RLIO 算法的有效性在隧道环境中特别明显，在这些环境中，由于几何相似性，LiDAR 可能无法进行配准，从而导致 LiDAR 惯性里程计失效。在这种情况下，雷达可以接收更全面的点云数据以维持里程计。与单独使用 RIO 相比，我们的系统在隧道外具有丰富几何特征的环境中利用 LIO，实现了更高的精度。尽管 LRIO 能够缓解隧道内效果的下降，其准确性因雷达点云中相对较高的误差而受到影响，导致精度降低。在主要是静态的环境中，我们的算法保持了与 FAST-LIO2 相当的性能。

B. 蜗牛数据集

我们进一步将我们的方法与其他算法在蜗牛数据集上进行了比较。与 MSC 数据集相比，蜗牛数据集的序列较少，但序列距离较长，并且包含动态性强和隧道的环境，这更有效地突出了我们算法的优势。

TABLE II
APE (m) / RPE (m) of SLAM systems on Snail Dataset

Seq ID	FAST-LIO2 [10]	RIO	LRIO	AF-RLIO
81R	203.3/0.555	216.7/0.828	168.0/0.536	24.9/0.387
IAF	41.2/0.561	60.0/1.130	61.1/0.582	38.7/0.562
IF	15.6/0.566	23.7/0.838	17.6/0.567	14.6/0.564
IFEA	16.5/0.374	34.3/0.949	22.9/0.371	16.4/0.373

使用不同算法的蜗牛数据集的轨迹误差结果如表 II 所示。81R 序列包括隧道和高速公路环境，尤其具有挑战性。即使在显著退化的情况下，我们的算法在该序列中表现出了显著的优势，保持了稳定的里程计输出。虽然 LRIO 算法没有出现退化，但在优化过程中持续结合雷达数据，在具有明显几何特征的环境中产生了负面影响。在 IAF 序列中，由于具有动态障碍物，我们的系统在 LIO 框架内有效利用了其优势。IF 和 IFEA 是没有高动态的校园序列。

C. 烟雾环境

我们还在烟雾环境中进行了现实世界的实验，以验证所提算法的可行性和通用性。Jackal 地面机器人配备了 Velodyne VLP-16 LiDAR 和 4DAres-R7861 前向雷达，两者都以 10Hz 的扫描频率获取点云数据。雷达的仰角为 $\pm 15^\circ$ ，方位角为 $\pm 75^\circ$ ，方位分辨率为 0.25° ，仰角分辨率为 0.5° ，检测范围约为 200 米。地面真实轨迹是通过 RTK-GPS 获得的。所有计算均在配有 Ubuntu 20.04 和 ROS Noetic 的笔记本电脑上进行。所有测试均在 24 核的 Intel i7-14650HX CPU 上进行。机器人的配置和传感器如图 1 所示。

TABLE III

Comparison of APE (m) / RPE (m) in the smoke environment

	FAST-LIO2 [10]	RIO	LRIO	AF-RLIO
SMOKE	12.8/0.141	0.67/0.116	0.63/0.115	0.55/0.114

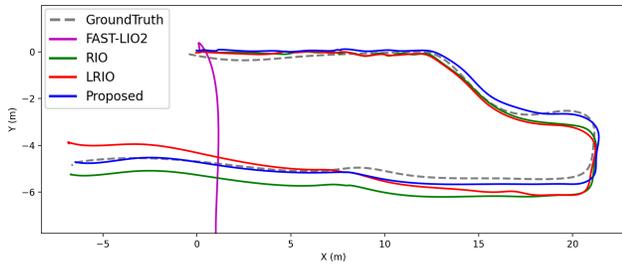


Fig. 5. 现实世界烟雾环境中的比较轨迹。

我们收集了一个数据集，记录了机器人在一个部署了烟雾发生器的室外区域导航的情况，此时浓烟显著减少了激光雷达的有效范围，并遮挡了其视野的大片区域。即使在未被烟雾完全阻挡的区域，激光雷达仍然难以捕捉到准确的点云数据。随着烟雾的消散，激光雷达逐渐恢复获得可靠的点云数据。图 5 展示了所提方法与其他 SLAM 算法在烟雾环境中的轨迹比较，而表 III 展示了位姿误差。可以看出，FAST-LIO2 在烟雾出现时立即失效，因为在这种条件下激光雷达噪声高且范围受限。虽然 RIO 和 LRIO 能相对较好地保持位姿，但其准确性不如 AF-RLIO，因为雷达数据中的误差干扰了无烟环境中的最终位姿估计。

D. GPS-挑战环境测试

MSC 数据集中 URBAN_B0 序列涉及通过一个 GPS 信号被遮挡的隧道，这使得准确定位变得困难。当机器人进入隧道时，数据不确定性随着深度增加，但在进入一个更开放的环境后，GPS 数据逐渐变得更准确。

TABLE IV

Comparison of APE (m) in the GPS-challenge environment

	Constant-GPS	Threshold-GPS	Adaptive-GPS
URBAN_B0	failed	22.82	2.26

我们比较了三种算法：不带异常检测的 Proposed-Constant，基于阈值判断的 Proposed-Threshold，以及自适应算法。其结果轨迹如图 6 所示，绝对位姿误差 (APE) 结果如表 IV 所示。在没有异常检测的算法中，由

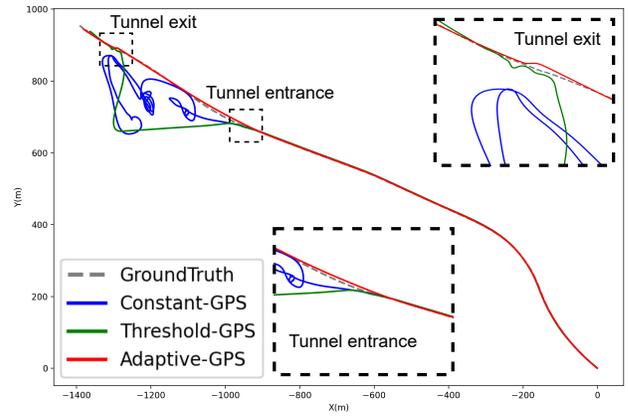


Fig. 6. 隧道场景中无 GPS 异常值检测 (常规-GPS)、基于阈值的异常值检测方法 (阈值-GPS) 和所提出的异常值检测方法 (自适应-GPS) 的轨迹比较。

于包含错误的 GPS 数据，系统失败。基于阈值的 GPS 判断因不能最佳地确定何时接受或拒绝 GPS 数据而失败，导致优化中的发散。然而，自适应方法能够快速而准确地评估 GPS 数据的可靠性，在不确定性增加时降低其在优化中的权重，确保里程计不会受到错误数据的影响。此外，当机器人离开隧道时，自适应方法平滑地增加 GPS 权重，减轻优化过程中数据跳跃的影响。结果表明，我们的方法能够有效检测并去除 GPS 异常，抑制误差累积，避免 GPS 异常的影响。

这篇文章提出了一种基于多模态融合的里程计方法，AF-RLIO，它集成了 IMU 和 GPS，并可以自适应地切换 LiDAR 和雷达传感器。它确保了机器人在诸如隧道、充满烟雾的区域以及室内外过渡等具有挑战性的环境中能够实现稳健的里程计。通过利用雷达和 LiDAR 点云的互补优势，AF-RLIO 采用了一种基于 IESKF 的方法，将预选点与 IMU 数据紧密结合，从而实现精确的里程计。在后端，算法将自适应加权的 GPS 数据与里程计联合优化，有效减少长期映射过程中累积的误差，并在 GPS 信号间歇性错误出现时促进平滑过渡。该算法已在多个数据集和现实场景中测试，显示出在具有相似几何特征和烟雾环境中的强大鲁棒性。

References

- [1] K. Ebadi, L. Bernreiter, H. Biggie, G. Catt, Y. Chang, A. Chatterjee, C. E. Denniston, S.-P. Deschênes, K. Harlow, S. Khattak, et al., "Present and future of slam in extreme underground environments," arXiv preprint arXiv:2208.01787, 2022.
- [2] D. Lee, M. Jung, W. Yang, and A. Kim, "Lidar odometry survey: recent advancements and remaining challenges," Intelligent Service Robotics, vol. 17, no. 2, pp. 95–118, 2024.
- [3] M. Bijelic, T. Gruber, and W. Ritter, "A benchmark for lidar sensors in fog: Is detection breaking down?," in 2018 IEEE intelligent vehicles symposium (IV), pp. 760–767, IEEE, 2018.
- [4] J. Liu, W. Gao, and Z. Hu, "Optimization-based visual-inertial slam tightly coupled with raw gnss measurements," in 2021 IEEE International Conference on Robotics and Automation (ICRA), pp. 11612–11618, IEEE, 2021.
- [5] J. Nubert, S. Khattak, and M. Hutter, "Graph-based multi-sensor fusion for consistent localization of autonomous construction robots," in 2022 International Conference on Robotics and Automation (ICRA), pp. 10048–10054, IEEE, 2022.
- [6] R. Wang, Y. Jing, C. Gu, S. He, and J. Chen, "End-to-end multi-target flexible job shop scheduling with deep reinforcement learning," IEEE Internet of Things Journal, 2024.

- [7] K. Harlow, H. Jang, T. D. Barfoot, A. Kim, and C. Heckman, "A new wave in robotics: Survey on recent mmwave radar applications in robotics," arXiv preprint arXiv:2305.01135, 2023.
- [8] C. Qin, H. Ye, C. E. Pranata, J. Han, S. Zhang, and M. Liu, "Lins: A lidar-inertial state estimator for robust and efficient navigation," in 2020 IEEE international conference on robotics and automation (ICRA), pp. 8899–8906, IEEE, 2020.
- [9] W. Xu and F. Zhang, "Fast-lio: A fast, robust lidar-inertial odometry package by tightly-coupled iterated kalman filter," IEEE Robotics and Automation Letters, vol. 6, no. 2, pp. 3317–3324, 2021.
- [10] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, "Fast-lio2: Fast direct lidar-inertial odometry," IEEE Transactions on Robotics, vol. 38, no. 4, pp. 2053–2073, 2022.
- [11] Y. Wu, T. Guadagnino, L. Wiesmann, L. Klingbeil, C. Stachniss, and H. Kuhlmann, "Lio-ekf: High frequency lidar-inertial odometry using extended kalman filters," in 2024 IEEE International Conference on Robotics and Automation (ICRA), pp. 13741–13747, IEEE, 2024.
- [12] C. Bai, T. Xiao, Y. Chen, H. Wang, F. Zhang, and X. Gao, "Faster-lio: Lightweight tightly coupled lidar-inertial odometry using parallel sparse incremental voxels," IEEE Robotics and Automation Letters, vol. 7, no. 2, pp. 4861–4868, 2022.
- [13] H. Ye, Y. Chen, and M. Liu, "Tightly coupled 3d lidar inertial odometry and mapping," in 2019 International Conference on Robotics and Automation (ICRA), pp. 3144–3150, IEEE, 2019.
- [14] T. Shan, B. Englot, D. Meyers, W. Wang, C. Ratti, and D. Rus, "Lio-sam: Tightly-coupled lidar inertial odometry via smoothing and mapping," in 2020 IEEE/RSJ international conference on intelligent robots and systems (IROS), pp. 5135–5142, IEEE, 2020.
- [15] K. Li, M. Li, and U. D. Hanebeck, "Towards high-performance solid-state-lidar-inertial odometry and mapping," IEEE Robotics and Automation Letters, vol. 6, no. 3, pp. 5167–5174, 2021.
- [16] S. Zhao, H. Zhang, P. Wang, L. Nogueira, and S. Scherer, "Super odometry: Imu-centric lidar-visual-inertial estimator for challenging environments," in 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 8729–8736, IEEE, 2021.
- [17] K. Ebadi, L. Bernreiter, H. Biggie, G. Catt, Y. Chang, A. Chatterjee, C. E. Denniston, S.-P. Deschênes, K. Harlow, S. Khattak, et al., "Present and future of slam in extreme environments: The darpa sub challenge," IEEE Transactions on Robotics, 2023.
- [18] P. Fritsche, S. Kueppers, G. Briese, and B. Wagner, "Fusing lidar and radar data to perform slam in harsh environments," in Informatics in Control, Automation and Robotics: 13th International Conference, ICINCO 2016 Lisbon, Portugal, 29-31 July, 2016, pp. 175–189, Springer, 2018.
- [19] H. Yin, R. Chen, Y. Wang, and R. Xiong, "Rall: end-to-end radar localization on lidar map using differentiable measurement model," IEEE Transactions on Intelligent Transportation Systems, vol. 23, no. 7, pp. 6737–6750, 2021.
- [20] M. Mielle, M. Magnusson, and A. J. Lilienthal, "A comparative analysis of radar and lidar sensing for localization and mapping," in 2019 European Conference on Mobile Robots (ECMR), pp. 1–6, IEEE, 2019.
- [21] M. Nissov, N. Khedekar, and K. Alexis, "Degradation resilient lidar-radar-inertial odometry," arXiv preprint arXiv:2403.05332, 2024.
- [22] C. Noh and A. Kim, "Adaptive lidar-radar fusion for outdoor odometry across dense smoke conditions," 2024.
- [23] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, et al., "A density-based algorithm for discovering clusters in large spatial databases with noise," in kdd, vol. 96, pp. 226–231, 1996.
- [24] C. Doer and G. F. Trommer, "An ekf based approach to radar inertial odometry," in 2020 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), pp. 152–159, IEEE, 2020.
- [25] Y. Cai, W. Xu, and F. Zhang, "ikd-tree: An incremental kd tree for robotic applications," arXiv preprint arXiv:2102.10808, 2021.
- [26] M. Choi, S. Yang, S. Han, Y. Lee, M. Lee, K. H. Choi, and K.-S. Kim, "Msc-rad4r: Ros-based automotive dataset with 4d radar," IEEE Robotics and Automation Letters, 2023.
- [27] J. Huai, B. Wang, Y. Zhuang, Y. Chen, Q. Li, Y. Han, and C. Toth, "Snail-radar: A large-scale diverse dataset for the evaluation of 4d-radar-based slam systems," arXiv preprint arXiv:2407.11705, 2024.
- [28] M. Grupp, "evo: Python package for the evaluation of odometry and slam.." <https://github.com/MichaelGrupp/evo>, 2017.