文章编号:1674-2974(2025)08-0111-11 DOI:10.16339/j.cnki.hdxbzkb.2025288

基于关键帧的双模式车辆激光SLAM

秦洪懋[1,2],杨龙安[1],周云水[1†],张润邦[1],高铭[1],边有钢[1,2]
(1. 湖南大学 机械与运载工程学院,湖南 长沙 410082;
2. 湖南大学无锡智能控制研究院,江苏 无锡 214115)

摘 要:同时定位与建图(simultaneous localization and mapping, SLAM)技术在自动驾驶领域有着广泛的应用,其中精度和计算效率是SLAM最重要的两个指标.然而,传统的激光雷达里程计难以准确高效地提取关键帧,导致构建地图时包含了过多的冗余帧.此外,大多数激光里程计需要对每一帧执行帧到地图的对齐,这带来了高额的计算负担.本文提出了一种基于关键帧的双模式激光里程计与建图方法,通过计算两帧点云之间的特征相似度并将其与运动自适应阈值进行比较,提取关键帧.随后,对关键帧和非关键帧执行不同的配准算法,旨在最小化计算资源的消耗.此外,利用点的水平距离信息计算权重函数,并将其整合到加权位姿约束中.本文提出的SLAM系统在KITTI数据集和实车上进行了大量试验,KITTI序列00-10的结果显示,平移误差仅有0.56%,在旋转上的误差为0.002 1(°)/m.实时性方面,与F-LOAM相比,本文算法将平均速度提高了26.5%,比轻量级系统LeGO-LOAM更快.

关键词:自动驾驶;关键帧;同时定位与建图(SLAM);双模式
中图分类号:TP399 **文献标志码**:A

Keyframe-based Dual-mode Vehicle LiDAR SLAM

QIN Hongmao[1,2],YANG Long'an[1],ZHOU Yunshui[1†],ZHANG Runbang[1],GAO Ming[1],BIAN Yougang[1,2]
(1. College of Mechanical and Vehicle Engineering, Hunan University, Changsha 410082, China;
2. Wuxi Intelligent Control Research Institute of Hunan University, Wuxi 214115, China)

Abstract: Simultaneous localization and mapping (SLAM) technology is widely used in the field of autonomous driving, where accuracy and computational efficiency are the two most important indicators. However, traditional LiDAR odometry faces challenges in accurately and efficiently extracting keyframes, resulting in an excess of redundant frames during map construction. Additionally, the majority of LiDAR odometry systems require aligning each frame to the map, which imposes a substantial computational burden. This paper proposes a dual-mode LiDAR odometry and mapping method based on keyframes. By computing the feature similarity between two point clouds and comparing it with a motion-adaptive threshold, keyframes are extracted. Subsequently, different registration algorithms are applied to keyframes and non-keyframes to minimize computational resource consumption. Furthermore, a weight function is calculated using point horizontal distance information and integrated into the

* **收稿日期**:2024-03-12
基金项目:国家重点研发计划项目(2021YFF0501102),National Key R&D Program of China (2021YFF0501102);国家自然科学基金资助项目(52272415,52102456),National Natural Science Foundation of China(52272415,52102456)
作者简介:秦洪懋(1984—),男,江苏盐城人,湖南大学副教授
† 通信联系人,E-mail:zhouys@hnu.edu.cn

weighted pose constraints. The SLAM system proposed undergoes extensive testing on the KITTI dataset and real vehicles. The results from KITTI sequences 00-10 demonstrate a translational error of only 0.56% and a rotational error of 0.002 1 degree/m. In terms of real-time performance, compared with F-LOAM, our algorithm improves average speed by 26.5%, and even outperforms lightweight system LeGO-LOAM.

Key words: autonomous driving; keyframe; simultaneous localization and mapping (SLAM); dual-mode

随着自动驾驶的兴起,SLAM技术受到了广泛关注,SLAM依靠多种传感器感知外部环境,解算位姿实现精确的实时定位,是自动驾驶技术的重要一环.现阶段,SLAM在各种定位环境中得到越来越广泛的应用,特别是在城市、峡谷等GNSS信号差的场景.与基于视觉的SLAM相比,激光雷达具有高精度和不受光线明暗变化影响等优点,在自动驾驶领域越来越受青睐,已经成为自动驾驶车辆的标配传感器.近年来,出现了许多优秀的激光雷达SLAM算法[1-3].

自动驾驶车辆的计算平台需要同时支持感知、规控、定位、控制等多方面算力需求,其计算资源有限,能够支持SLAM的算力很少,因此实时性是SLAM的关键指标之一.最初,广泛使用的经典迭代最近点(iterative closest point,ICP)算法[4]及其变体被用于直接对齐原始点云,这些方法统称为基于ICP的方法[5-7].ICP算法最小化对齐误差,迭代优化相对位姿使得点到点的误差和最小.然而,ICP算法处理的是原始点云,计算代价大且对噪声敏感.此外,传统的ICP方法及其变体通过构建k维(k-d)树进行最近邻搜索,其计算成本随地图规模的扩大而不断增加.FAST-LIO2[8]直接进行点云配准,不依赖特征提取,并引入了一种ikd-Tree结构,确保计算资源需求不会随着地图大小的增加而呈线性增长.

LOAM[9]的提出开辟了基于特征的激光SLAM方法.在基于特征的方法[10-12]中,LOAM基于点的局部平滑度选择边缘和表面特征,通过特征点对构造优化问题,显著减少了待匹配点的数量.LOAM将高频率的帧间匹配和低频率的帧到地图配准相结合,并引入了点到线和点到面残差建立优化模型,显著提高了特征法SLAM的准确性和效率.Lio-mapping[13]对滑动窗口内的所有帧执行帧到地图的配准,实时性低下.F-LOAM[14]简化了LOAM方法,消除高频率的帧间匹配,直接采用单帧到局部地图的配准来进行位姿优化.与LOAM相比,F-LOAM在保持精度的同时,平均处理速度提高了3倍.MULLS[15]提出了一种基于点、线和面特征的多尺度最小二乘优化方法,这是一种高效且低漂移的3D激光SLAM系统.

值得注意的是,上述所有方法都忽视了关键帧的筛选,地图规模增长较快,这会影响回环检测和后端优化的计算效率.关键帧策略旨在从一系列普通帧中选择一个帧作为局部帧的代表.一个良好的关键帧策略可以评估点云和特征的质量,筛选特征稳定丰富的帧作为关键帧.通过关键帧策略可以过滤掉冗余和不稳定帧,这可以防止不相关或错误的数据渗入优化过程,从而影响定位和建图的准确性.此外,关键帧策略可以减少待优化的帧的数量,从而提高闭环检测的准确率和效率.在后端优化过程中,关键帧充当了局部普通帧的代表,大幅压缩了后端优化的规模,在保证后端优化精度的同时,也大大减轻了计算压力.

然而,激光点云是无序的,并且缺乏环境纹理信息,故提取关键帧是一个艰巨的挑战,而现有的相关工作明显不足.因此,一些激光SLAM系统不会检测关键帧,或者主要依赖粗糙的帧间角度和平移变化来确定关键帧,这种方法缺乏精度和稳定性,并且难以适应不同工况.LeGO-LOAM[16]是一种轻量级的LiDAR SLAM,它采用聚类算法进行点云分割,并引入两步列文伯格-马夸尔特优化方法.此外,它还采用了粗略的关键帧策略,提高了其实时性能.然而,LeGO-LOAM将所有距离超过0.3 m的帧都标记为关键帧,关键帧提取效果较差,对LeGO-LOAM的定位精度具有负面影响.在LIO-SAM[17]中,当空间距离和旋转变化超过固定阈值时,当前帧被指定为新的关键帧,只有关键帧才被纳入后端因子地图的优化中.尽管该算法引入了关键帧策略以提升计算效率和优化效果,但在不同工况下,连续帧之间的平移与旋转变化幅度存在较大差异,因此难以确定一个在不同工况下均适用的统一阈值.在LILI-OM[18]中,考虑了当前帧与局部特征地图之间的特征重叠率.如果重叠率低于60%,或者当前帧与最新关键帧之间

的时间差超过了定义的阈值,那么当前帧就被指定为新的关键帧.如上所述,现有的激光SLAM缺乏稳定有效的关键帧提取方法,这是影响SLAM定位精度和计算效率的关键因素.Shen等[19]提出了一种基于帧之间语义差异的自适应选择策略,其效果依赖于语义分割精度,泛化性较差,在不同场景难以自适应应用,并且语义分割对算力要求更高.Zu等[20]提出了一种基于视觉的关键帧动态阈值设置方法,利用统计理论分析各帧之间的姿态差来表征摄像机的运动特性,采用多层模糊推理机制进行联合判断,得出关键帧筛选的动态阈值,应用在ORB-SALM3系统自适应选择关键帧.InfoLa-SLAM[21]提出了一种使用费舍尔信息矩阵去度量当前帧和上一关键帧配准误差,平均信息值越大,表示配准误差越小,以此筛选关键帧.该方法需要先进行配准,需要容忍一定的配准误差才能保证稀疏有效的关键帧,精度和实时性难以平衡.

另外,位姿优化模型会给所有点对分配相同的权重,尽管并非所有的几何约束对位姿优化的影响都相同[22].F-LOAM优先对具有较高局部平滑度的边缘特征和具有较低局部平滑度的平面特征进行匹配,相应地分配更高的权重进行加权优化.KFS-LIO[23]提出了一个特征评估函数来衡量各个点的可信度,权衡它们对残差的贡献.WiCRF2[24]基于运动可观测性提出了一种加权方法,提高了系统的鲁棒性和准确性.ROI-cloud[25]提出了一种基于点云的体素化方法,并通过粒子滤波选择感兴趣的区域进行注册,将处理限制在感兴趣区域内的点上.

针对上述问题,我们提出了一种基于关键帧的双模式车辆激光SLAM,衡量帧间特征差异以筛选关键帧,并基于关键帧检测结果执行双模式的激光里程计配准算法.受到Scan-Context[26]和Iris[27]的启发,引入了一种利用点云中的边缘和平面特征的新型关键帧提取方法.首先,我们引入二维特征矩阵(feature matrix,FM)的概念,FM封装了单帧点云的几何结构特征信息,FM之间的差异量化了两帧之间特征变化的程度.因此,通过计算当前帧和最新关键帧的FM之间的相似度,即可量化帧间的特征差异以进行关键帧筛选.此外,因为不稳定工况下的特征变化显著,所以引入运动自适应阈值与特征相似度比较,过滤不稳定工况下的帧.依据关键帧的检测结果,相邻的非关键帧和关键帧之间存在大量特征重叠,依靠最近的关键帧进行配准足以提供足够的约束.因此,我们设计了一种双模式配准方法,对非关键帧仅执行帧与帧的配准,对关键帧仅执行帧与局部地图的配准.在点云配准的优化模型中,远距离点对旋转具有更加显著的约束,故融合点的水平距离作为残差权重,实现加权优化.本文的主要贡献如下:

1)提出了一种新颖的激光点云关键帧提取方法,该方法综合考虑了几何特征相似性和运动自适应阈值,以提取稳定且合适的关键帧.

2)提出了基于关键帧的双模式激光里程计算法.在非关键帧时,仅执行帧间匹配,以获得更高的计算效率;而在关键帧时,则执行帧到局部地图的配准,以确保系统的定位精度.

3)利用点云中得出的水平距离数据,为远距离点赋予更高的权重,这些点对旋转具有更强的约束.

1　系统概述

本文提出了一种基于关键帧的双模式激光SLAM框架,如图1所示,主要包括特征提取、关键帧检测、运动自适应阈值、双模式配准和加权约束模块.

系统每获取一帧激光点云,就先提取边缘和平面特征.将所提取的三维特征投影到二维平面上,由于激光雷达的扫描特性是从中心向外360°扫描,故投影结果是一个圆形.将圆形展开构造得到特征矩阵(FM),矩阵的每个单元存储该位置所有点的高度信息并升序排列.当前帧的FM构造完成后,将与最新关键帧的FM进行匹配,计算两个FM的相似度,并与运动自适应阈值比较以筛选关键帧.基于关键帧检测结果,执行双模式的激光里程计配准算法,并融入点的水平信息进行加权优化位姿.

首先进行特征提取.激光雷达是多线束的,每一线激光都会扫描周围环境,最终所有线束的扫描结果汇聚得到一帧完整的激光点云.此外,激光雷达的水平分辨率远高于竖直分辨率,因此针对激光雷达每一线点云都提取特征点,计算每个点的局部平滑度以筛选线特征点和面特征点,如式(1)所示.

$$\sigma = \frac{1}{|S|}\sum_{P_i,P_j\in S,i\neq j}\left(\left\|\boldsymbol{P}_i-\boldsymbol{P}_j\right\|\right) \tag{1}$$

式中:S是$\boldsymbol{P}_i$在同一行中的邻近点集合,试验中集合的大小设置为10;$\boldsymbol{P}_j$是$\boldsymbol{P}_i$的邻近点;σ是$\boldsymbol{P}_i$的局部平滑度.

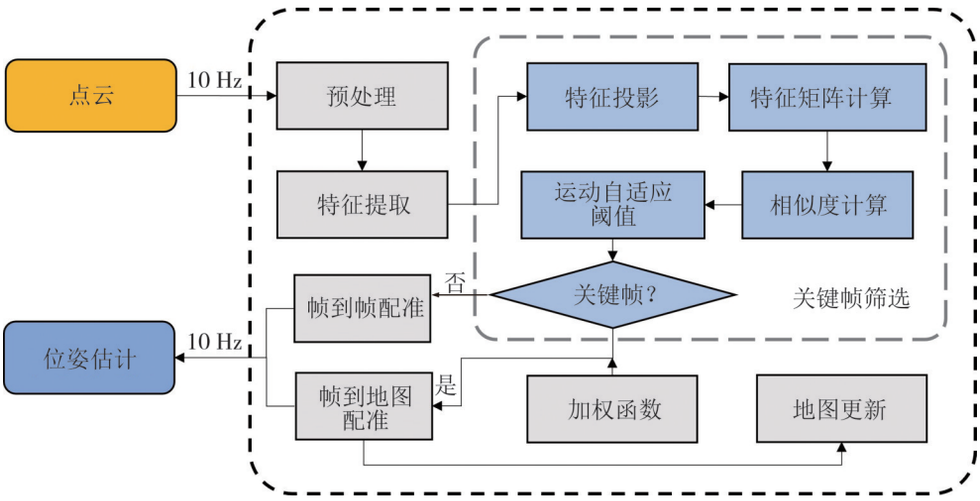


图 1 系统框架

Fig.1 System framework

根据所计算的点的局部平滑度,从每个激光线束中选择具有最高局部平滑度的前20个点作为线特征点.为了提升特征提取的速度,我们丢弃了LOAM中的弱特征点,将除线特征点外的所有点均视为面特征点.

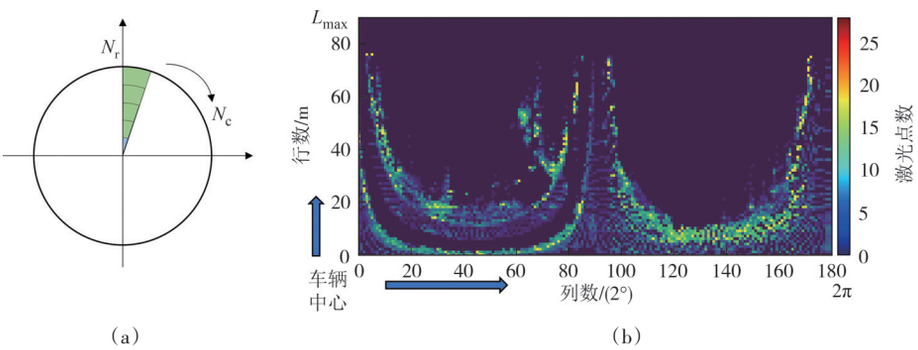
2 关键帧检测

一个合理的关键帧策略可以消除冗余和不稳定的特征,同时保留丰富的环境信息.然而,现有的激光SLAM中的关键帧检测方法十分粗糙,大都依据空间位置和旋转约束判定,这对定位精度产生不利影响.与传统方法相比,我们利用提取的边缘和平面特征进行关键帧选择,直接衡量两帧的特征差异,这将包含更加稳定和丰富的特征.

首先,对提取的线面特征进行下采样,以减少激光点数量.再将线面特征点投影到二维水平面,将投影得到的圆形平面展开为特征矩阵(FM),FM中的每个单元格容纳一个特征向量(FV),存储投影到该单元格的点的高度信息.每当订阅新的点云,计算FM并将其与最新关键帧的FM进行匹配,计算它们的特征相似度.随后,将特征相似度与运动自适应阈值进行比较以筛选关键帧.自适应阈值基于运动信息解算,以避免将不稳定工况下的点云纳入关键帧.在本文的后续讨论中,当前帧的FM表示为 C,而最新关键帧的FM表示为 L.

2.1 点云投影

投影沿着特征点云的高度方向进行,得到一个圆形平面,将其沿径向和切向方向等间隔地分割成单元格,如图2所示.在径向和切向方向上,分割的每个单元格分别对应于FM的行索引和列索引.图2(a)为特征点云的俯视图,特征在径向和切向均匀地划分为 N_r 和 N_c 个单元,分别对应于图2(b)矩阵的行数和列数.点云的最大水平感知距离 L_{max} 设置为90 m.在试验中,特征矩阵的径向分辨率为1 m,切向分辨率为2°.图2(b)中矩阵单元颜色代表该位置存储的激光点数量.



(a) (b)

图 2 特征点云投影

Fig.2 The projection of features

需要注意的是,投影是基于车辆当前位姿的局部坐标系进行的,可避免引入累积定位误差.然而,激光雷达的机械旋转特性使得每帧点云存在运动畸变,单帧内的点不在同一位置采集.因此,引入匀速运动模型将所有点与当前帧的起始位置对齐.

Iris 将点沿高度方向划分为八个部分,并以汉明距离的形式存储点的高度,而本文则将每个点分配到其相应的特征向量中,以保留一帧中的所有特征.此外,鉴于特征相似性计算仅涉及最新的关键帧,计算成本保持在较低水平.

2.2 特征相似度计算

如图2(b)所示,每个FM包含 $N_c \times N_r$ 个FV,每个FV存储该位置对应的点的高度,因此FM反映了环境的几何结构.而 C 和 L 具有相同索引的FV之间的相似度反映了当前帧和最新关键帧之间几何特征变化的程度.最终,通过对 C 和 L 的所有对应FV的特征相似度进行加权计算,即可求出两帧之间特征差异度.

FM的维度是固定的,取决于单元格分辨率,但FV的维度并不固定,这取决于单元格中点的数量.为了衡量两个FV之间的距离,将维度较低的FV补充为高度为0的点,以确保它们具有相同的维度.此外,将每个FV按降序排列可以确保高度较大的点成对出现,高处的点所蕴含的环境信息更丰富.对所有订阅到的点云都执行上述算法,以获取其FM和FV.

遍历当前帧的FM,将每个FV与最新关键帧的FM相同索引处的FV进行匹配.根据FV的维度,将计算两个FV之间差异的方法分为三种情况:第一种,参与匹配的FV都为空,表示该位置没有特征点,相似度设为默认值1;第二种,仅有一个FV为空,表明该位置的特征差异巨大,相似度设置为0;第三种,两个FV都存储了点,计算两者余弦距离作为特征相似度,这反映了特征向量间的差异.最终,FV的特征相似度计算如下:

$$s(i,j)=\begin{cases}1, \boldsymbol{C}_{ij}=\boldsymbol{0} \wedge \boldsymbol{L}_{ij}=\boldsymbol{0}\\ \dfrac{\boldsymbol{C}_{ij}\cdot\boldsymbol{L}_{ij}}{\left\|\boldsymbol{C}_{ij}\right\|\cdot\left\|\boldsymbol{L}_{ij}\right\|}, \boldsymbol{C}_{ij}\neq\boldsymbol{0} \wedge \boldsymbol{L}_{ij}\neq\boldsymbol{0}\\ 0, \text{其他}\end{cases} \tag{2}$$

式中:i 和 j 分别是FM的行和列索引;$\boldsymbol{C}_{ij}$ 和 $\boldsymbol{L}_{ij}$ 分别是 C 和 L 在 (i,j) 处的FV;$s(i,j)$ 是 C 和 L 在 (i,j) 处的特征相似度.两个FV之间的相似度越高,$s(i,j)$ 越大.

单个位置的FV之间的差异计算如上所示.综合考虑所有FV之间的特征相似度 $s(i,j)$,即可得到全局特征距离,从而量化两帧之间的差异,如式(3)所示.

$$d=\frac{1}{n}\sum_{i}^{N_r}\sum_{j}^{N_c}\left(1+\frac{i-N_r/2}{2N_r}\right)\left[1-s(i,j)\right] \tag{3}$$

式中:d 为加权得到的特征距离;n 是 C 和 L 中相同索引处的FV不同时为空的计数.C 和 L 之间的相似度越高,d 越小.

随着车辆的移动,新的特征将更多地出现在远处,故更高的权重被分配给更远的点,以更灵敏地感知特征变化.

当车辆保持静止,环境特征基本保持不变,各FV中的特征维持不变,所解算出的特征距离 d 很小.在动态场景下,动态物体可能会影响局部FV的特征分布,但我们的方法考虑了 C 和 L 中的全局FV差异,占比更大的静态物体起决定作用,对局部动态物体具有鲁棒性.随着车辆移动,不断感知到新的环境特征,d 逐渐增大.一旦 d 超过特征距离阈值,就表明当前帧与最新关键帧之间存在显著差异.传统的关键帧提取方法通常为监测空间距离和旋转变化,然而该方法很难建立一个能够有效适应不同车速和工况下的变化阈值.相反,我们所提出的方法直接聚集于帧间的特征差异,这是两帧能否配准良好的根本.只要环境特征发生变化,就应当被视为关键帧,受车速和工况影响小.

2.3 运动自适应阈值

关键帧的质量直接影响着激光里程计和建图的精度,因此必须保证所筛选关键帧特征的稳定和健壮.如前所述,本文采用的关键帧选取策略是通过计算两帧之间的特征距离,并与阈值比较来选取的.然而,在不稳定的工作条件下,例如车辆遇到减速带时,受到大幅冲击,其提取的特征与上一关键帧会出现显著不同,计算的特征差异会很大.因为该帧实际包含的环境特征是不稳定的,所以不应将其纳为关键帧,否则会引入错误的地图信息,而且错误是不可逆转的,这对建图和随后的定位都会产生永久影响.因此,我们结合运动信息构建自适应特征阈值,避免将不稳定工况下捕获的帧包含到地图中.

旋转变化对运动强度的感知更敏感,因此我们通过计算两个最新帧之间的欧拉角来评估运动工况,如式(4)所示.

$$t=\begin{cases}t_1, \Delta R \leqslant t_2\\ t_1+\lg\dfrac{\Delta R}{t_2}, \text{其他}\end{cases} \tag{4}$$

式中:t是计算得到的自适应特征距离阈值;t_1是稳定情况下的固定阈值;t_2是角度变化阈值;ΔR是最新两帧之间欧拉角的模长.在所有试验中,t_1为0.6,t_2为1.5°.

对于具有大幅旋转变化的工作条件,特征距离的阈值会相应提升,以避免将剧烈运动时刻包括为关键帧.另外,这种自适应调节不会影响正常工况下的关键帧选择.

3 基于关键帧的配准算法

基于关键帧检测结果,我们将激光里程计分为帧与帧配准和帧与地图配准两种模式.对关键帧进行帧与地图配准,对非关键帧进行帧与帧配准.对于线特征点,采用点线ICP计算残差;对于面特征点,采用点面ICP计算残差,如式(5)和式(6)所示.

$$f_e(\boldsymbol{P}_i)=\frac{\left|(\boldsymbol{TP}_i-\boldsymbol{P}_1)\times(\boldsymbol{TP}_i-\boldsymbol{P}_2)\right|}{\left|\boldsymbol{P}_1-\boldsymbol{P}_2\right|} \tag{5}$$

$$f_s(\boldsymbol{P}_i)=\frac{(\boldsymbol{P}_1-\boldsymbol{P}_2)\cdot(\boldsymbol{P}_1-\boldsymbol{P}_3)}{\left|(\boldsymbol{P}_1-\boldsymbol{P}_2)\times(\boldsymbol{P}_1-\boldsymbol{P}_3)\right|}\cdot(\boldsymbol{TP}_i-\boldsymbol{P}_1) \tag{6}$$

式中:$f_e(\boldsymbol{P}_i)$和$f_s(\boldsymbol{P}_i)$分别表示点到线的残差和点到平面的残差;$\boldsymbol{P}_i$是在局部坐标系中表示的边缘或平面特征点;$\boldsymbol{T}$是当前帧的全局位姿,通过该位姿可将点$\boldsymbol{P}_i$变换至世界坐标系下;$\boldsymbol{P}_1,\boldsymbol{P}_2,\boldsymbol{P}_3$是全局坐标系下与$\boldsymbol{TP}_i$的最近点.

3.1 双模式配准

我们依据帧间的特征差异程度筛选关键帧.对于非关键帧,这意味着当前帧与最新关键帧之间的特征差异较小,即存在大量的特征重叠,因此通过帧到帧的配准可以提供足够的几何约束.将当前帧与最新关键帧特征点云进行配准,在速度上远远快于与地图的配准.另外,我们是将当前帧与最新关键帧对齐,而不是与普通帧对齐.因为普通帧仅通过帧到帧的配准进行对齐,并未通过帧到地图的配准进行进一步优化,因此比关键帧包含更多的不确定性.

对于关键帧,这意味着当前帧观察到的特征与最新关键帧中的特征显著不同,即车辆在当前时刻检测到了很多新的环境特征.在这种情况下,仅依靠帧与帧的配准难以提供足够的几何约束.因此,当前帧必须与信息更加丰富的局部地图进行对齐,以构建更有效的几何约束.子图的大小是车辆在全局地图中以当前位置为中心的100 m区域.

在这两种配准模式之间切换,可以在保证足够精度的同时显著减少计算成本并提升实时性.

3.2 加权约束

激光雷达在捕获环境信息时受到真实世界物体的水平距离的影响,因此远处特征表现出稀疏性,近处特征表现出稠密性,如图2(b)所示.特征的非均匀分布导致近处点相对远处点的权重更高.然而,远处点对旋转的约束更强,因此应该增强远处特征构建的约束.针对这一挑战,我们利用水平距离信息构建加权约束,为远处特征分配更高的权重,如式(7)所示.每个残差被分配一个适当的权重,通过最小化这些残差的总和来优化位姿,如式(8)所示.

$$w=\mathrm{e}^{\frac{r-r_{\min}}{r_{\max}-r_{\min}}} \tag{7}$$

$$\boldsymbol{T}=\arg\min_{\boldsymbol{T}}\sum_{\boldsymbol{P}_i\in\boldsymbol{P}_e}\boldsymbol{w}_i\left|f_e(\boldsymbol{P}_i)\right|+\sum_{\boldsymbol{P}_j\in\boldsymbol{P}_s}\boldsymbol{w}_j\left|f_s(\boldsymbol{P}_j)\right| \tag{8}$$

式中:r是特征点的水平距离;$r_{\min}$是激光雷达的最小有效使用距离;$r_{\max}$是激光雷达感知的最大距离;$\boldsymbol{T}$是要优化的全局位姿;$f_e(\boldsymbol{P}_i)$表示点到线的残差;$f_s(\boldsymbol{P}_j)$表示点到平面的残差;$\boldsymbol{w}_i$和$\boldsymbol{w}_j$是点的权重;$\boldsymbol{P}_e$和$\boldsymbol{P}_s$分别是当前帧的边缘特征和平面特征的集合.

3.3 地图更新

位姿求解完成后,将进行地图的更新,我们会在世界坐标系内维护一个全局地图.具体来说,只会使用从关键帧中提取的特征来更新这个全局地图,而与非关键帧相关的特征在完成位姿估计后被丢弃.这种策略明显减小了地图的规模,从而提高了SLAM的计算效率.此外,在内存资源有限的计算平台上,这种方法还显著降低了内存消耗.

我们依据关键帧的全局位姿将其点云转换到世界坐标系中,注册到一个体素化的全局地图上.此外,关键帧的特征点云和位姿是分开存储的,这样可以方便地与回环检测和后端优化模块集成.另外,由于关键帧的数量明显少于普通帧,回环检测和后端优化的效率都会显著提升.本文采用了Scan Context的方法执行回环检测.

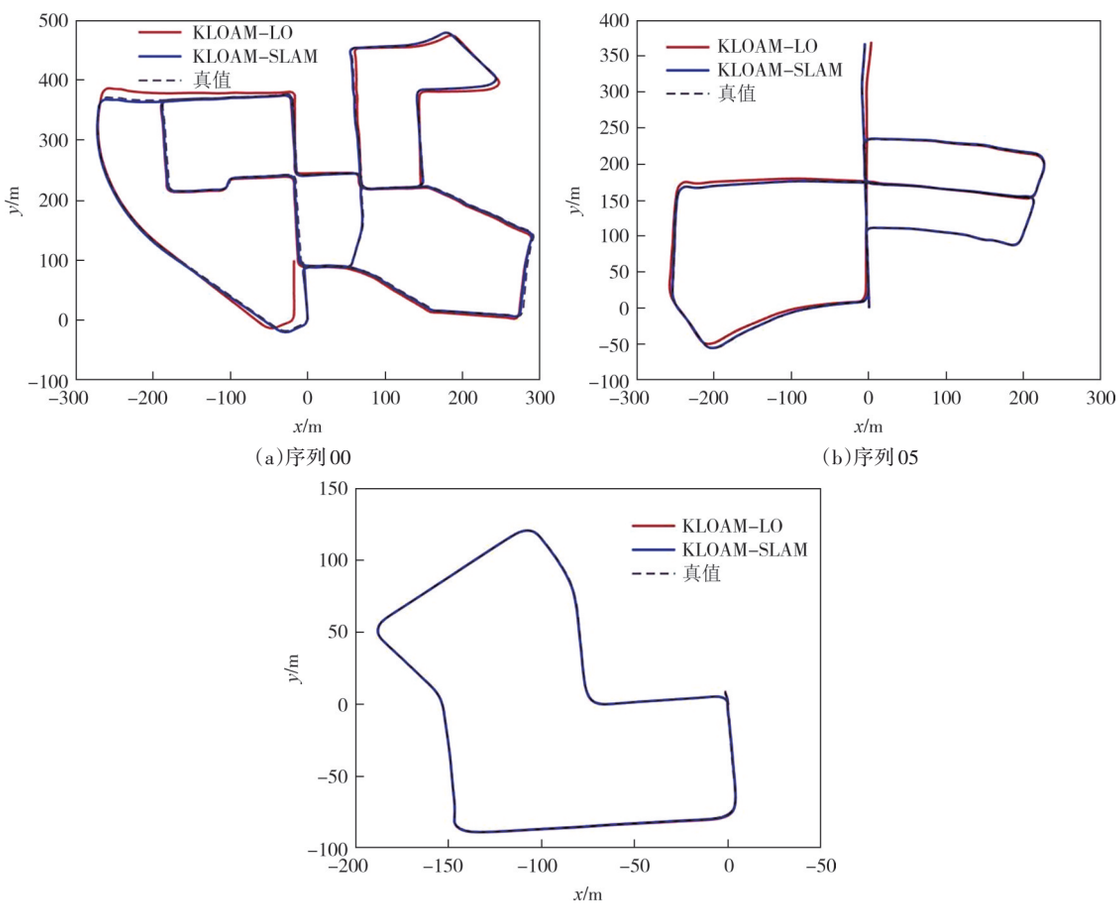
4 试验测试

本文通过计算定位的精度和耗时来验证系统的综合性能.定位精度采用平均平移误差ATE(average translation error,单位是%)和平均旋转误差ARE(average rotational error,单位是°/100 m)评估,计算方法由KITTI数据集定义.本文在公开数据集KITTI

和实车上都做了测试,并与当前杰出的激光SLAM系统进行比较.鉴于本文是基于LOAM框架实现的,故选择了一些基于LOAM的方案进行比较分析,这包括LOAM[9]、A-LOAM、F-LOAM[14]和LeGO-LOAM[16].为更全面地展示本文所提出方法的有效性,我们还与LOAM系列之外的杰出方法进行了比较,包括SuMa++[28]和MULLS[15].我们的代码完全使用C++编写,并在机器人操作系统ROS Noetic和Ubuntu 20.04 LTS上运行.为了准确对比计算成本,所有试验均在一台装有Intel i5-12500H、2.50 GHz处理器的笔记本电脑上进行.

4.1 KITTI数据集测试

KITTI数据集是SLAM领域中最具权威性的数据集之一,被广泛用于测试各种SLAM算法,包括ORB-SLAM[29]、LOAM、F-LOAM、MULLS等.KITTI平台配备了两个灰度相机、两个彩色相机、一台Velodyne 64线激光雷达和一套GPS导航系统.我们在KITTI的00~10数据集序列上进行了试验验证,涵盖了超过23 000帧的数据,车辆行驶距离达22 000 m,最高速度达96 km/h.此外,还建立了一组试验包括回环检测和后端优化模块,作为完整的SLAM系统.我们将本系统在部分序列的轨迹绘制出来,并与KITTI的真实轨迹进行比较,如图3所示.KLOAM-LO代表无回环检测的系统,KLOAM-SLAM代表具备回环检测功能的系统.图3(a)~(c)分别表示KITTI的00、05、07序列.



(a)序列00　　(b)序列05

(c)序列07

图3　车辆定位轨迹

Fig.3　The trajectory of vehicle localization

所有试验结果如表1所示,K-LOAM的平均精度排名第二,ATE仅为0.56%,ARE为0.002 1(°)/m.此外,在序列02中,K-LOAM具有最高精度.

为验证本系统的实时性,我们计算了所有方法在KITTI的00—10序列上每帧的平均计算时间,如表2所示.本系统展现出最高的处理速度,每帧处理时间为38.5 ms.而LeGO-LOAM是一种轻量级激光SLAM,其速度与我们的速度相当.然而,我们仅考

表 1 KITTI 数据集定位误差 ATE/ARE

Tab.1 The localization error ATE/ARE on KITTI dataset

方法	序列片段											
	00	01	02	03	04	05	06	07	08	09	10	平均
F-LOAM	0.70/0.29	1.95/0.49	1.04/0.37	0.95/0.47	0.68/0.39	0.49/0.29	0.54/0.27	0.42/0.22	0.93/0.32	0.70/0.30	1.02/0.40	0.86/0.35
LeGO-LOAM$^{\triangle}$	2.17/1.05	13.4/1.02	2.17/1.01	2.34/1.18	1.27/1.01	1.28/0.74	1.06/0.63	1.12/0.81	1.99/0.94	1.97/0.98	2.21/0.92	2.49/1.00
LOAM	0.78/–	1.43/–	0.92/–	0.86/–	0.71/–	0.57/–	0.65/–	0.63/–	1.12/–	0.77/–	0.79/–	0.84/–
A-LOAM	0.69/0.32	1.96/0.49	4.53/1.46	0.94/0.48	0.72/0.40	0.49/0.24	0.61/0.27	0.42/0.25	1.04/0.33	0.73/0.31	1.00/0.39	1.19/0.45
MULLS*	0.54/0.13	0.62/0.09	0.69/0.13	0.61/0.22	0.35/0.08	0.29/0.07	0.29/0.08	0.27/0.11	0.83/0.17	0.51/0.12	0.61/0.19	0.52/0.13
SuMa++*	0.64/0.22	1.60/0.46	1.00/0.37	0.67/0.46	0.37/0.26	0.40/0.20	0.46/0.21	0.34/0.19	1.10/0.35	0.47/0.23	0.66/0.28	0.70/0.29
K-LOAM/nc$^{\triangle}$	0.63/0.26	0.75/0.17	0.61/0.21	0.82/0.22	0.49/0.27	0.41/0.24	0.35/0.16	0.34/0.30	0.99/0.28	0.67/0.18	0.74/0.29	0.62/0.23
K-LOAM/nk$^{\triangle}$	0.85/0.39	2.10/0.54	1.16/0.43	0.91/0.51	0.79/0.54	0.64/0.33	0.59/0.30	0.53/0.37	0.96/0.37	0.85/0.40	1.40/0.54	0.98/0.43
K-LOAM/ns$^{\triangle}$	0.59/0.23	0.79/0.17	0.59/0.20	0.80/0.20	0.46/0.28	0.38/0.22	0.32/0.14	0.33/0.27	0.95/0.26	0.61/0.17	0.73/0.28	0.60/0.24
K-LOAM$^{\triangle}$	0.61/0.25	0.71/0.13	0.55/0.19	0.73/0.22	0.42/0.21	0.36/0.21	0.30/0.13	0.32/0.25	0.93/0.26	0.56/0.15	0.68/0.27	0.56/0.21
K-LOAM*$^{\triangle}$	0.60/0.20	0.71/0.13	0.55/0.16	0.73/0.22	0.42/0.21	0.34/0.16	0.36/0.11	0.32/0.22	0.92/0.22	0.65/0.13	0.68/0.27	0.57/0.18

注:1)ATE 表示平均平移误差,单位是%; ARE 表示平均旋转误差, 单位是(°)/(100 m).2)nc 表示没有加权约束的试验组;nk 表示使用传统关键帧策略的试验组;ns 表示没有运动自适应阈值的试验组,详细信息在 4.3 节消融试验中展示.表格中的定位精度,有下划线的数据表示最高精度,蓝色数据表示第二位精度.试验组中,*表示该组包含回环检测,△表示该组采用了关键帧检测.

虑了其前端计算,因为其后端优化在额外的线程中运行.虽然这不影响实时性能,但会消耗额外的计算资源.LeGO-LOAM 的后端较慢,在我们的测试中,该线程的平均执行时间在 200~300 ms 之间.此外,它每跳过固定数量的帧才执行一次后端优化,这对定位精度产生了不利影响,如表 1 所示.我们将本文中的关键帧检测方法替换为 LIO-SAM[17]的传统关键帧检测方法,用 K-LOAM/nk 表示.该组试验结果的实时性大幅下降,平均每帧处理时间增加为 53.5 ms,这是因为传统的关键帧检测引入了大量冗余帧,增加了计算负担.另外,在所测试的方法中,从是否使用关键帧的角度来看,基于关键帧的方法表现出更高的计算效率.

4.2 实车测试

我们的数据是配备了 VLP-32c 激光雷达和 NPOS220 GNSS/IMU 系统的车辆收集的,如图 4 所示.图 4(a)为试验车辆,图 4(b)和图 4(c)为试验环境,图 4(d)为试验场景中存在回环的建图效果.试验场景涵盖了郊区、繁华的城市区域和公园,包括大量动态障碍物.值得注意的是,序列 00 和 01 是在城市道路上以平均时速 50 km/h 采集的,而序列 02 是在狭窄街道上收集的,序列 03 是在公园内获取的,平均时速为 25 km/h.此外,激光点云和 GNSS/IMU 数据的采集速率均为 10 Hz,使用 RTK-GNSS/IMU 作为定位真值.

表 2 平均计算时间

Tab.2 Average calculation time

方法	MULLS	A-LOAM	LeGO-LOAM$^{\triangle}$	F-LOAM	SuMa++	K-LOAM/nk$^{\triangle}$	K-LOAM$^{\triangle}$
耗时/ms	49.2	114.9	39.6	52.4	96.4	53.5	38.5

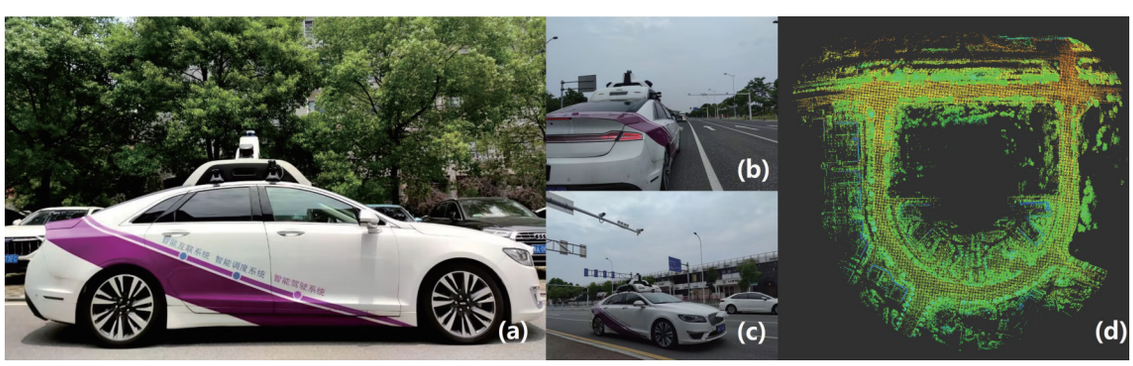


图 4 实车试验场景与效果

Fig.4 Actual vehicle test scenarios and effects

我们对K-LOAM、F-LOAM、A-LOAM和MULLS进行了测试.如表3所示,K-LOAM表现出最高的定位精度,ATE为0.56%.此外,K-LOAM具有最快的平均每帧计算时间,达到25 Hz.图4(d)提供了K-LOAM实时构建的序列03的地图的可视化表示.如该图所示,树木、道路和建筑物清晰和明确的轮廓印证了本文方法在复杂的户外环境中实现高精度定位和地图绘制的能力.

表3　实车试验结果的ATE/平均计算时间

Tab.3　ATE /average time consumption from actual vehicle test results

方法	序列片段				
	00	01	02	03	平均
F-LOAM	0.99/61.2	0.84/57.6	0.38/43.4	0.84/56.6	0.76/54.7
MULLS	1.06/46.9	1.31/42.1	0.69/50.2	1.17/48.5	1.06/46.9
A-LOAM	1.21/274.7	1.00/212.5	0.49/158.1	0.89/202.2	0.90/211.9
K-LOAM	0.74/45.7	0.53/45.9	0.33/32.1	0.64/35.7	0.56/39.9

注:1)ATE表示平均平移误差,单位是%;2)平均计算时间的单位是ms.

4.3　消融试验

为了验证本文算法的有效性,我们设置了三组消融试验,分别评估关键帧检测、动态自适应阈值和加权约束算法的有效性.每组试验中仅调整一个方法,其余方法保持不变,独立评估该方法的影响.

4.3.1　关键帧检测方法

为了评估本文提出的关键帧检测方法与传统方法之间的优劣,我们引入了一组消融试验,标记为nk.在这组试验中,关键帧检测方法被替换为LIO-SAM所采用的传统方法,该方法基于空间距离和角度变化过滤关键帧.而KITTI数据集中不同序列的速度和工况差异很大,仅使用固定的距离和角度阈值很难精确地选择关键帧.试验结果如表1和表2所示,与K-LOAM相比,nk的精度和效率都显著降低,证实了本文基于特征变化筛选关键帧的方法的有效性和稳健性.值得注意的是,01序列的精度下降最为显著,因为它是在高速公路场景中录制的,最高速度为96 km/h.

4.3.2　运动自适应阈值

本组消融试验旨在评估本文提出的自适应阈值方法应对剧烈运动场景的效果.在这组试验中,使用固定阈值而不考虑运动信息,用ns表示.试验结果如表1所示,与KLOAM相比,ns未过滤不稳定的帧,几乎所有序列的精度均有所下降,其中在运动最为剧烈的高速公路场景(01序列)下降最明显.此外,过滤掉不稳定关键帧,可以减少关键帧数量,从而提高了计算效率.尽管如此,仍会受到路面颠簸的影响,因此本文算法适用于城市平整路面.

4.3.3　加权约束

为验证本文提出的加权约束算法的有效性,设置消融试验取消加权函数,这组试验被标记为nc.如表1所示,试验结果明确表明,缺乏加权约束的方法表现出较低的定位精度,证实了本文的加权约束算法对精度提升的帮助.

表4　算法移植到其他系统的ATE/ARE

Tab.4　ATE/ARE when transplanting algorithms to another system

指标	方法	序列片段											
		00	01	02	03	04	05	06	07	08	09	10	平均
ATE/ARE	A-LOAM	0.80/0.32	1.96/0.49	4.53/1.46	0.95/0.48	0.72/0.40	0.49/0.24	0.61/0.27	0.44/0.25	1.08/0.33	0.73/0.31	1.00/0.39	1.21/0.45
	A-LOAM/k	0.79/0.32	1.75/0.45	3.2/1.20	0.93/0.51	0.87/0.54	0.51/0.25	0.61/0.28	0.43/0.29	1.05/0.31	0.77/0.33	1.17/0.46	1.10/0.45
耗时/ms	A-LOAM	127.1	126.6	114.2	110.5	110.8	118.3	127.5	99.4	127.3	109.4	92.8	114.9
	A-LOAM/k	94.4	72.6	65.7	69.7	65.9	85.1	82.1	66.1	98.7	76.1	63.2	76.3

4.4　在其他SLAM系统的表现

为进一步评估本文提出的关键帧策略的健壮性,我们将本文算法应用于经典的A-LOAM框架.框架的差异导致无法完整地将本文的工作整合到A-LOAM中.这是因为:将关键帧检测方法移植到A-LOAM中,需要对关键帧同时执行帧间匹配和帧与

地图配准,对非关键帧仅进行帧间匹配.另外,A-LOAM的点云存储在一个体素化的大地图中,而不是保留单个帧的点云.因此,非关键帧只能与最新的普通帧匹配,而不是关键帧,这对帧间匹配的精度有不利影响.

如表4所示,试验仍然在KITTI数据集上进行,该组试验标记为 A-LOAM/k.与A-LOAM的结果相比,A-LOAM/k 的平均精度提升了 9.1%.其中,A-LOAM 在 KITTI 的 02 序列上表现出退化现象[30],定位精度显著下降.相反,本文的关键帧系统在该系统退化时仍表现出强大的健壮性和出色的性能.在 02 序列上,A-LOAM/k 的定位精度显著提高,达到了29.8%.更重要的是,我们的关键帧系统显著提高了A-LOAM在所有序列上的速度,平均提升了33.6%.

5 结 论

本文提出了一种基于关键帧的双模式实时激光SLAM,创新性地直接衡量两帧之间的特征差异,以筛选关键帧.通过提取每帧的边缘和平面特征以构建特征矩阵(FM)和特征向量(FV),计算FM之间的相似度,并与运动自适应阈值比较完成关键帧检测.该关键帧策略显著减小了地图的规模并提升了系统实时性.另外,本文设计了双模式激光里程计,基于关键帧检测的结果执行不同的配准算法.对于关键帧,需要与局部地图对齐以获得更丰富的约束,而非关键帧只需与关键帧对齐以减少系统计算资源的消耗.此外,本文结合点的水平距离信息构造加权约束,提高了定位的精度.K-LOAM在KITTI公共数据集和实车数据集上进行了全面测试,与F-LOAM、MULLS、SuMa++和其他最新方法相比,表现出高精度和优越的实时性能.

参考文献

[1] CHEN X, MILIOTO A, PALAZZOLO E, et al. SuMa: efficient LiDAR-based semantic SLAM[C]//2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Macau, China. IEEE,2019:4530-4537.

[2] WANG H, WANG C, XIE L H. Intensity-SLAM: intensity assisted localization and mapping for large scale environment[J]. IEEE Robotics and Automation Letters,2021,6(2):1715-1721.

[3] ZHENG X, ZHU J K. Traj-LO: in defense of LiDAR-only odometry using an effective continuous-time trajectory[J]. IEEE Robotics and Automation Letters,2024,9(2):1961-1968.

[4] BESL P J,MCKAY N D. A method for registration of 3-D shapes [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence,1992,14(2): 239-256.

[5] VIZZO I, GUADAGNINO T, MERSCH B, et al. KISS-ICP: in defense of point-to-point ICP-simple, accurate, and robust registration if done the right way [J]. IEEE Robotics and Automation Letters, 2023, 8(2) :1029-1036.

[6] CHEN K,LOPEZ B T,AGHA-MOHAMMADI A A,et al. Direct LiDAR odometry: fast localization with dense point clouds[J]. IEEE Robotics and Automation Letters,2022,7(2):2000-2007.

[7] DELLENBACH P, DESCHAUD J E, JACQUET B, et al. CT-ICP: real-time elastic LiDAR odometry with loop closure[C]// 2022 International Conference on Robotics and Automation (ICRA). Philadelphia,PA,USA. IEEE,2022:5580-5586.

[8] XU W,CAI Y X,HE D J,et al. FAST-LIO2:fast direct LiDAR-inertial odometry[J]. IEEE Transactions on Robotics, 2022, 38(4): 2053-2073.

[9] ZHANG J, SINGH S. LOAM: lidar odometry and mapping in real-time [C]//Robotics: Science and Systems X. Robotics: Science and Systems Foundation, 2014, 2(9): 1-9.

[10] CHEN S B,MA H,JIANG C H,et al. NDT-LOAM:a real-time lidar odometry and mapping with weighted NDT and LFA[J]. IEEE Sensors Journal,2022,22(4):3660-3671.

[11] ALI W,LIU P L,YING R D,et al. A feature based laser SLAM using rasterized images of 3D point cloud[J]. IEEE Sensors Journal,2021,21(21):24422-24430.

[12] ZHENG X,ZHU J K. Efficient LiDAR odometry for autonomous driving[J]. IEEE Robotics and Automation Letters,2021,6(4): 8458-8465.

[13] YE H Y,CHEN Y Y,LIU M. Tightly coupled 3D lidar inertial odometry and mapping[C]//2019 International Conference on Robotics and Automation (ICRA). Montreal,QC,Canada. IEEE, 2019:3144-3150.

[14] WANG H,WANG C,CHEN C L,et al. F-LOAM: fast LiDAR odometry and mapping [C]//2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Prague, Czech Republic. IEEE,2021:4390-4396.

[15] PAN Y,XIAO P C,HE Y J,et al. MULLS: versatile LiDAR SLAM *via* multi-metric linear least square [C]//2021 IEEE International Conference on Robotics and Automation (ICRA). Xi'an,China. IEEE, 2021: 11633-11640.

[16] SHAN T X,ENGLOT 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). Madrid, Spain. IEEE,2018:4758-4765.

[17] SHAN T X,ENGLOT 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). Las Vegas,NV,USA. IEEE,2020:5135-

5142.

[18] LI K L, LI M, HANEBECK U D. Towards high-performance solid-state-LiDAR-inertial odometry and mapping[J]. IEEE Robotics and Automation Letters, 2021, 6(3): 5167-5174.

[19] SHEN B K, XIE W M, PENG X D, et al. LIO-SAM++: a lidar-inertial semantic SLAM with association optimization and keyframe selection[J]. Sensors, 2024, 24(23): 7546.

[20] ZU L N, WEI C R, SUN Q Q, et al. Adaptive keyframe selection strategy of visual SLAM in complex poses[J]. IEEE Sensors Journal, 2025, 25(1): 1756-1767.

[21] LIN Y, DONG H Q, YE W T, et al. InfoLa-SLAM: efficient lidar-based lightweight simultaneous localization and mapping with information-based keyframe selection and landmarks assisted relocalization[J]. Remote Sensing, 2023, 15(18): 4627.

[22] DUAN Y F, PENG J, ZHANG Y, et al. PFilter: building persistent maps through feature filtering for fast and accurate LiDAR-based SLAM [C]//2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Kyoto, Japan. IEEE, 2022: 11087-11093.

[23] LI W, HU Y, HAN Y H, et al. KFS-LIO: key-feature selection for lightweight lidar inertial odometry [C]//2021 IEEE International Conference on Robotics and Automation (ICRA). Xi'an, China. IEEE, 2021: 5042-5048.

[24] CHANG D X, HUANG S J, ZHANG R B, et al. WiCRF2: multi-weighted LiDAR odometry and mapping with motion observability features[J]. IEEE Sensors Journal, 2023, 23(17): 20236-20246.

[25] ZHOU Z B, YANG M, WANG C X, et al. ROI-cloud: a key region extraction method for LiDAR odometry and localization [C]//2020 IEEE International Conference on Robotics and Automation (ICRA). Paris, France. IEEE, 2020: 3312-3318.

[26] KIM G, KIM A. Scan context: egocentric spatial descriptor for place recognition within 3D point cloud map[C]//2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Madrid, Spain. IEEE, 2018: 4802-4809.

[27] WANG Y, SUN Z Z, XU C Z, et al. LiDAR iris for loop-closure detection [C]//2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Las Vegas, NV, USA. IEEE, 2020: 5769-5775.

[28] CHEN X, MILIOTO A, PALAZZOLO E, et al. SuMa++: efficient LiDAR-based semantic SLAM[C]//2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Macau, China. IEEE, 2019: 4530-4537.

[29] MUR-ARTAL R, MONTIEL J M M, TARDÓS J D. ORB-SLAM: a versatile and accurate monocular SLAM system[J]. IEEE Transactions on Robotics, 2015, 31(5): 1147-1163.

[30] ZHANG J, KAESS M, SINGH S. On degeneracy of optimization-based state estimation problems[C]//2016 IEEE International Conference on Robotics and Automation (ICRA). Stockholm, Sweden. IEEE, 2016: 809-816.