

文章编号:1001-5078(2025)07-1012-10

· 激光应用技术 ·

基于 BA/NDT 与 LOAM 融合的激光 SLAM 方法

杨 奎,梁冬泰,胡盛辉
(宁波大学 机械工程与力学学院,浙江 宁波 315211)

摘要:针对 3D 激光雷达 SLAM 算法缺乏回环检测及雷达移动过快易导致里程计漂移的问题,提出了一种 BA(光束法平差)/NDT(正态分布变换)与 3D 激光雷达 LOAM 融合的 SLAM 方法。首先,改进激光 BA 算法,最小化特征点与边缘和平面之间的距离;其次,采用 LOAM 算法,前端通过帧到帧的里程计估计输出粗位姿,后端采用 BA 输出精位姿;最后,通过 NDT 估计线面特征矩阵信息,进行旋转不变性转化并执行回环检测,直至最新帧与初始地图对齐,完成全局位姿修复。在公开的数据集 KITTI 测试表面,本文总的位置均方根误差为 16 cm,相比与 LOAM 算法减小了约 33 % 误差,整体运行速度是 LOAM 的 2.23 倍。为了进一步在实际环境中验证,设计了手持式建模设备,并以 cartographer 算法的位姿数据为基准,测试结果表明本文方法总的位置均方根误差为 20.23 cm,相比与 LOAM 算法减小了约 75 %,有效提高了定位精度。

关键词:同步定位与地图构建;光束法;正态直方图;单元格存储

中图分类号:TN958.98;TP391 文献标识码:A DOI:10.3969/j.issn.1001-5078.2025.07.002

LiDAR SLAM method based on BA/NDT and LOAM fusion

YANG Kui, LIANG Dong-tai, HU Sheng-hui
(Faculty of Mechanical Engineering and Mechanics, Ningbo University, Ningbo 315211, China)

Abstract: Aiming at the problem that the 3D LiDAR SLAM (Simultaneous Localization and Mapping) algorithm lacks loop-closure detection and are prone to odometry drift when the radar moves too fast, a SLAM method that integrates BA (Bundle Adjustment)/NDT (Normal Distribution Transformation) and 3D LiDAR LOAM is proposed. Firstly, the lidar BA algorithm is improved by minimizing the distance between feature points and edges or planes. Then, the LOAM algorithm is employed, where the front-end estimates coarse poses through frame-to-frame odometry, and the back-end refines the poses using BA. Finally, NDT is utilized to estimate the line and plane feature matrix, performing rotation-invariant transformations and loop closure detection until the latest frame aligns with the initial map, completing the global pose correction. The test is carried out on the public data set KITTI show that the proposed method achieves a root mean square error (RMSE) of 16 cm, reducing the error by about 33 % compared to the LOAM algorithm, and the overall running speed is 2.23 times faster than LOAM. To further validate the method in real-world environments, a handheld modeling device is designed, using the pose data from the cartographer algorithm as a reference. The test results demonstrate that the proposed method achieves an RMSE of 20.23 cm, reducing the error by about 75 % compared to LOAM, effectively improving positioning accuracy.

Keywords: SLAM; bundle adjustment; normal distribution transformation; voxel storage

1 引言

SLAM 是指在没有环境和先验信息的情况下,机器人搭载特定传感器在运动过程中建图并同时估

计自身的位姿。当搭载的传感器为激光雷达时,被

基金项目:宁波市公益性科技计划项目(No. 2022S004);宁波市重点研发计划项目(No. 2023Z131);宁波市“科创甬江 2035”科创生态育成计划项目(No. 2024Z062)资助。

作者简介:杨 奎(1999-),男,硕士研究生,主要从事机器人技术,多传感器融合 SLAM 研究。E-mail:yangkui0108@163.com

通讯作者:梁冬泰(1981-),男,副教授,主要从事视觉检测,机器人感知及操控技术研究。E-mail:liangdongtai@nbu.edu.cn

收稿日期:2024-10-08

称为激光 SLAM。SLAM 技术是机器人自主导航的关键与核心,在自动驾驶和 AR 领域有着重要应用。经过多年的发展,以激光雷达传感器为核心的激光 SLAM 技术成为研究的热点。

Grisetti^[1]于 2007 年提出第一个具有里程碑意义的粒子滤波激光 SLAM 系统——Gmapping,但其应用场景有限,随着粒子数量增加,构建大地图需要更多的内存和计算资源。2011 年,Stefan^[2]等人提出了 Hector-SLAM,实现了帧到地图的匹配,可以在不平坦区域进行二维建图。Hess^[3]等人于 2016 年提出了 Cartographer,采取了图优化框架,适用于二维激光雷达,在主流的 CPU 上能够实时建图与定位,成为当前最为完善的激光 SLAM 系统之一,包含经典激光 SLAM 框架里的所有流程:特征提取、闭环检测、后端优化等。由于单线激光雷达的局限性,生成的地图通常缺少三维信息,而无人驾驶的发展使得多线激光雷达的需求激增,三维 SLAM 的研究随之兴起。张楫^[4]等人于 2014 年提出 LOAM,将特征点的概念引入激光点云领域,成为激光特征点 SLAM 中最优秀的系统之一,尽管该系统在主流 CPU 上运行良好,但缺乏后端优化和闭环检测,导致在大场景建模中里程计容易漂移。2018 年,Tixiao Shan^[5]等人提出了 LEGO-LOAM,引入地面优化,实现了轻量级稠密建图,但对场景有较高要求。2021 年 Tixiao Shan^[6]提出了 LVI-SAM,集成 LIS 和 VIS,两个系统只要有一个正常就可以运行整个系统,展现了较好的鲁棒性,但增加了硬件成本。

在激光 SLAM 的研究中,人们逐渐认识到其在实际应用中的局限性^[7-8]。当缺乏回环检测时,里程计定位容易漂移,导致地图失效。因此,回环检测与 SLAM 技术的结合成为研究热点。传统回环检测

方法,如 Biber^[9]的关键帧方法和 Magnusson^[10]的 NDT 方法已被提出,Kim^[11]提出了 Scan Context,通过全局描述子将点云深度和角度信息投影到二维平面,用于回环检测,但处理稀疏数据效果不佳。BE-HLEY 等^[12]的 SuMa 方法则首次在激光雷达上利用 Surfel 地图高效生成投影数据关联,成功建立了大尺度环境的全局一致性地图。近年来,基于深度学习的回环检测成为热点。OverlapNet^[13]通过图像重叠率检测回环,避免依赖两帧点云的相对位姿。然而,在环境变化时,图像差异大,重叠率降低,影响检测精度。Ma 等^[14]提出的 Overlap 方法通过 IMU 传感器分析 Z 轴角速度和加速度来检测转弯和减速带,但在复杂城市环境中,语义信息不足,限制了检测效果。文献[15]提出了一种自监督激光里程计算法,使用 SuperPoint^[16]提取投影强度图像上的特征点,并通过 ICP 估算位姿进行在线训练。尽管该算法能够高频率输出里程计信息,但由于特征点稀疏,无法构建可导航地图。此外,深度学习依赖大规模数据集和高 GPU 算力,在实际应用中普适性较差。

综上所述,激光 SLAM 因其高精度和不受光照影响的特点,成为研究重点,但存在回环检测缺失及算法在移动过快时失效等问题。针对以上问题,本文提出一种基于 BA/NDT 与 3D 激光雷达松耦合的 SLAM 方法,即 BA/NDT 与 3D 激光雷达分别估计自身的位姿再进行融合。3D 激光雷达采用线面特征提取法的 LOAM 算法,具有 LOAM 算法特征提取,里程计解算等所有功能,可以在标准的 CPU 上实时运行。NDT 在后端优化时为激光雷达 LOAM 提供回环检测,BA 在雷达移动过快激光雷达跟踪丢失或里程计出现大幅度漂移时,提供短时间的位姿里程计修复,具体的算法框架如图 1 所示。

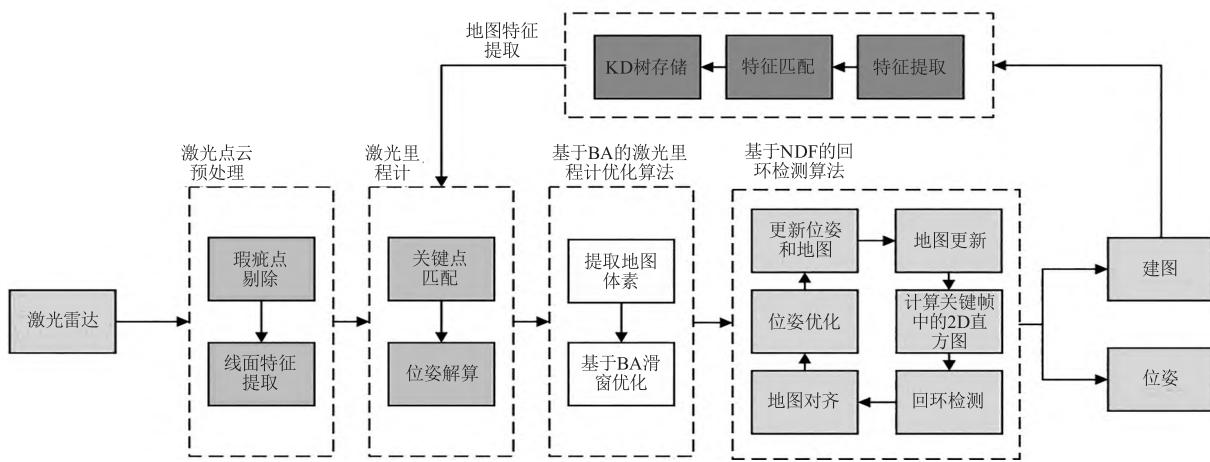


图 1 算法框架
Fig. 1 Algorithm framework

2 BA 位姿修复

视觉 BA (Bundle Adjustment) 光束平差法^[17], 是对不同视角中提取的视觉特征点进行最优位姿求解。本文将视觉 BA 方法运用到激光 BA 中。首先, 通过 ICP 算法初步估计机器人位姿;然后, 利用激光 BA 算法进一步优化机器人位姿和地图;最后, 构建残差函数求解最优位姿和相应地图的更新。激光 BA 算法的核心假设:每一帧激光点云中的特征点与周围特征点存在关联, 令一帧点云地图中的特征点为 p , 地图上有 N 个点。如图 2 所示, 该地图包含了平面特征和边缘特征, 面特征和边缘特征的最小距离点即为点 q 。

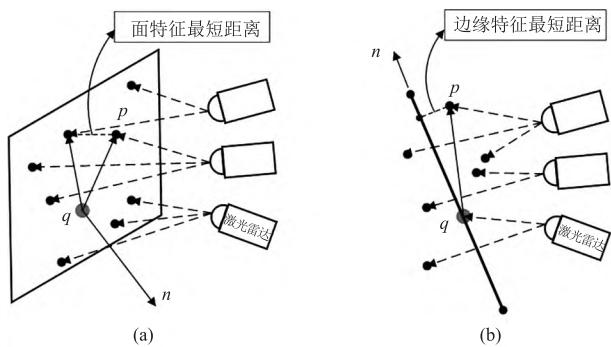


图 2 雷达多帧之间的特征关联

Fig. 2 Feature association between multiple radar frames

为了将特征点转换到世界坐标系下, 假设点云地图由 M 帧构成, 每一帧的位姿 $T_i = \{R_i, t_i\}$ 包含旋转矩阵 R_i 和平移矩阵 t_i , 对应的世界坐标下的特征点为:

$$P_i = R_i P_i + t_i \quad (1)$$

其中, P_i 是为了将一帧点云上的点坐标进行统一管理。通过最小化点云中点到平面和边缘的最小距离平方和, 优化出最优的变换矩阵 T 。

设 q 为点云中的某个已知存在的点, n 为平面和边缘的单位法向量, 假设有 N 个特征点, 可得变换矩阵 T :

$$T = \frac{1}{N} \sum_{i=1}^N ((P_i - q) (P_i - q)^T) n \quad (2)$$

其中, T 是两帧点云之间的配准矩阵, 可以通过求解 T 的最小特征值 $\lambda_{\min}(T)$, 该值越小, 地图中点到边缘和平面的最小距离的收敛性越好。得到的变换矩阵 T 的位姿更加精确。

$$T = \lambda_{\min}(T) \quad (3)$$

2.1 地图体素二层提取

为了统一管理同一平面或边缘的特征点, 减少计算量并提高位姿求解精度, 本节对图 3 中的三维

结构进行点云体素化处理^[18-19]。若某个体素中包含特征点, 则将该体素四分化, 并检查四个子体素是否存在非特征点, 若有则继续四分化。经过两次四分化后, 能够有效过滤杂点, 保证计算效率。原点云可见图 3(a)所示。

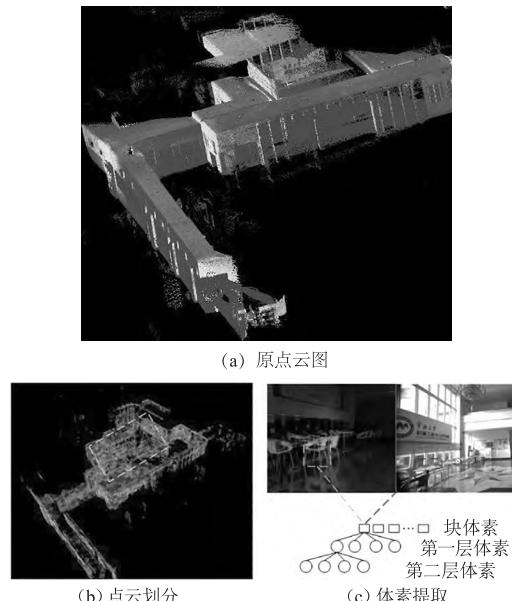


图 3 点云体素划分

Fig. 3 Point cloud voxel division

由图 3 可知, 大厅到实验室的点云场景图 3(a)被有效的进行了体素划分, 划分效果如图 3(b)所示。首先将激光雷达获得点云地图进行体素化分;然后, 利用体素二层提取的方法进行地图杂点的过滤;最后, 采用滑窗优化进行机器人位姿的精确估计。体素二层提取的模型为:

$$\lambda(p + \Delta p) = \lambda_k(p) + J(p)\Delta p + \frac{1}{2}\Delta p^T H(p)\Delta p \quad (4)$$

其中, $H(p)$ 为雅可比矩阵 J 的二阶求导; $J(p)$ 为雅可比矩阵 J 的一阶求导, 若 λ 的值超过设定阈值, 则过滤并移除该体素点。

2.2 基于 BA 的激光滑窗优化

激光点云滑窗优化借鉴了视觉 BA 的方法, 有效地保留了关键帧的信息, 对点云线面特征的体素化进行合理管理。图 4 展示了对真实场景中的椅子进行特征提取匹配, 从而进行二层体素的划分。激光 BA 是一种非线性优化方法, 将激光的位姿和点云的位置作为优化变量, 通过求解最小化误差函数公式(3)得到机器人的最优位姿和及对应的地图。

利用 BA 优化的位姿进行整体点云的滑窗优化, 滑窗优化的模型为:

$$T = \{A, B\} \quad (5)$$

其中, A 为新的激光关键帧, B 为旧的激光关键帧。新关键帧经过体素提取后存储在关键帧点云中, 并与旧关键帧的有用信息一起用于机器人位姿计算。将新的关键帧与已有点云结合, 进行滑窗优化, 最终得到精确的位姿估计和相应的环境地图。

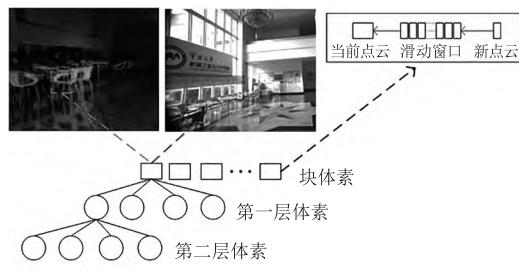


图 4 体素滑窗优化

Fig. 4 Voxel sliding window optimization

3 NDT 回环检测

在三维点云的统计模型中, 使用 NDT 方法^[20]的概率分布来计算点云间的最优匹配, 其对应的高维协方差矩阵可表示点云局部信息。首先, 采用体素化方法将点云划分成网格; 然后根据网格局部的形状和方向对点云进行分类; 最后, 将点云转为多维正态分布模型, 若两帧点云的正态分布参数匹配度较高, 即对应的概率密度较大, 则判定该点云与当前局部地图高度相似, 进而实现回环检测。本文基于 NDT 提出了一种结合特征向量与特征值的模糊 MC 方法进行后端回环检测, 能够提升定位精度并减少里程计位姿修复时间。 M 为三维激光点数据的平均值, C 为其对应的协方差矩阵。本节将讨论如何结合关键帧管理、位姿和地图优化更新来进一步改进算法。

3.1 关键帧直方图转换

关键帧管理是激光 SLAM 算法中的关键环节, 能够有效提升算法的效率与鲁棒性。通过优化关键帧的选择, 可以最大限度地减少计算量和存储空间的需求, 同时提升定位精度和地图构建的准确性。如图 5 所示, 激光点云的关键帧管理过程包括用三角形表示关键帧, 单正方形表示体素点云, 双正方形表示二维直方图。关键帧用于表达体素点云之间的关系, 并将其转换为二维直方图。最终, 优化得到的位姿和地图被集成到整个地图管理系统中, 进行地图更新。

本文设计了一种模糊 MC 算法, 提取点云中特征点的信息, 并计算均值 M 和协方差矩阵 C_d 的:

$$\begin{cases} M = \frac{1}{n} \sum_i N_i \\ C_d = \frac{1}{n} \sum_i (N_i - M)(N_i - M)^T \end{cases} \quad (6)$$

其中, N_i 是关键帧中第 i 个点云数据; n 是关键帧中的点云个数; M 是平均值; C_d 是协方差矩阵; M 和 C_d 反映了点云的数据信息。为了提高协方差矩阵的稳定性, 需要乘以旋转矩阵 R , 该矩阵包含线面特征, 其中面特征较为稳定, 因此本文采用面特征:

$$R = [V_{d_1}, V_{d_2}, V_{d_1} \times V_{d_2}] \quad (7)$$

其中, 特征向量 V_{d_1} 和 V_{d_2} 是协方差矩阵 C_d 对应的特征向量。通过对特征向量矩阵进行旋转变换, 得到:

$$\begin{cases} C_d = R \times C_d \\ C_d = [C_{d_x}, C_{d_y}, C_{d_z}] \end{cases} \quad (8)$$

其中, C_d 是由 $C_{d_x}, C_{d_y}, C_{d_z}$ 组成的, 将其放在图 2 所示的笛卡尔坐标系下。

接着, 将线面特征矩阵组成的 2D 直方图信息存入 H 矩阵, 该矩阵用于地图回环检测。由于线面特征存在边缘误检, 需使用噪声滤除。开源视觉库 OpenCV 提供了多种模糊滤波器, 用于图像降噪和平滑处理。

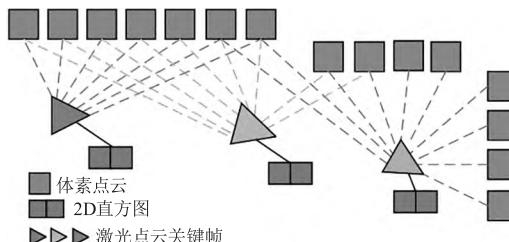


图 5 关键帧管理

Fig. 5 Key frame management

由于高斯模糊具有较快的运行速度和显著的去噪效果, 本文选择使用高斯模糊对 H 矩阵进行噪声滤除。如图 6 所示, α, β 是笛卡尔坐标系下特征点的空间位姿参数, 采集到的空间点以 2D 直方图的形式表示, 并经过放大处理以增强线面特征信息。

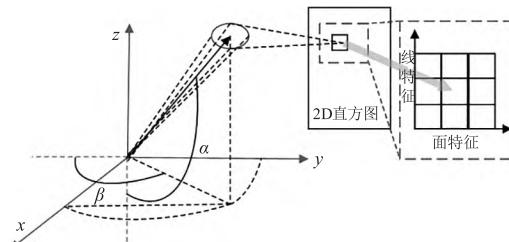


图 6 关键帧 2D 直方图

Fig. 6 Key frame 2D histogram

$$H(\alpha, \beta) \xrightarrow{\text{模糊滤波}} H'(\alpha, \beta) \quad (9)$$

其中, $H(\alpha, \beta)$ 经过 OpenCV 的模糊滤波得到包含

位置参数的直方图 $H'(\alpha, \beta)$, 用来模糊滤除周围不必要的噪点信息, 从而提升线面特征信息在回环检测中的精度, 并增强系统的环境鲁棒性。

线面特征信息中特征向量可用其对应的特征值来简化系统计算。特征向量对应的特征值大小代表线面特征, 假设 $\lambda_1 > \lambda_2 > \lambda_3$:

$$\begin{cases} \lambda_2/\lambda_1 < t, t \in [0, 1] \\ \lambda_3/\lambda_2 < t, t \in [0, 1] \end{cases} \quad (10)$$

其中, $\lambda_1, \lambda_2, \lambda_3$ 为特征向量对应的特征值; t 为阈值, 满足第一种情况是线特征, 满足第二种情况是面特征。得到的线面特征可用于后续的位姿估计和地图管理。

3.2 位姿和地图更新

在获得回环信息后, 检测到的位姿、地图与历史地图进行配准, 利用线面特征信息进行全局优化。位姿优化采用非线性优化方法, 通过当前帧和历史关键帧的位姿进行基于因子图的优化。主要步骤如下: 首先, 建立因子图, 将机器人位姿和地图特征点作为图的节点, 并将约束关系作为图的边, 如图 7 所示。然后, 为每条约束边定义一个代价函数, 表示在该约束下机器人位姿和特征点的误差。最后, 使用非线性优化的 Levenberg-Marquardt 算法最小化代价函数, 得到机器人的最优位姿估计, 进而拼接对应的点云地图, 实现全局地图的优化。

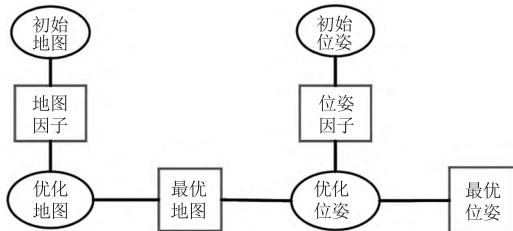


图 7 位姿和地图优化因子图

Fig. 7 Pose and map optimization factor graph

4 BA/NDT-LOAM 融合

4.1 LOAM 算法

LOAM 是一个基于线面特征点的激光雷达实时定位算法。它能够在标准的 CPU 上运行, 进行帧到帧以及帧到地图匹配, 同步进行纯雷达定位和周围环境地图构建。LOAM 算法主要有两个串行的线程, 第一个是通过特征提取来使用特征点代替完整的数据帧, 使用点的曲率来计算平面光滑度; 第二个是里程计解算, 通过高频率的里程计实现粗定位和低频率的里程计实现精定位。

LOAM 算法将当前点和前后 5 个点, 用来计算局部表面的平滑度:

$$c = \frac{1}{2 \times n} \sum_j^{(-n, n)} \| X_{(k,i)}^L - X_{(k,i-j)}^L \| \quad (11)$$

其中, c 是曲率计算系数; $X_{k,j}^L$ 是雷达坐标系下第 k 帧第 i 个点。曲率越大表示边缘点, 越小表示平面点。

构建误差模型, 点到线的误差函数:

$$d_e = \frac{| (\bar{X}_{(k+1,i)}^L - \bar{X}_{(k,j)}^L) \times (\bar{X}_{(k+1,i)}^L - \bar{X}_{(k,l)}^L) |}{| \bar{X}_{(k,j)}^L - \bar{X}_{(k,l)}^L |} \quad (12)$$

其中, $\bar{X}_{(k+1,i)}^L$ 、 $\bar{X}_{(k,j)}^L$ 和 $\bar{X}_{(k,l)}^L$ 分别表示雷达坐标系下 i, j, l 坐标点。

点到面的误差函数:

$$d_H = \frac{\left[(\bar{X}_{k+1,i}^L - \bar{X}_{k,j}^L) \times (\bar{X}_{k,j}^L - \bar{X}_{k,m}^L) \right]}{| (\bar{X}_{k,j}^L - \bar{X}_{k,l}^L) \times (\bar{X}_{k,j}^L - \bar{X}_{k,m}^L) |} \quad (13)$$

其中, $\bar{X}_{(k,m)}^L$ 表示雷达坐标系下第 $k+1$ 帧的第 m 个点的坐标。

理论误差模型:

$$loss = \sum_{i=1}^{N_e} d_{ei} + \sum_{i=1}^{N_H} d_{Hi} = D(\bar{X}_{(k+1,i)}^L) \quad (14)$$

$D(\bar{X}_{(k+1,i)}^L)$ 表示第 $k+1$ 帧中的第 i 个点的变化点到参考点云的所有的点到线和点到面的距离累加。

考虑到雷达频率较低并包含运动补偿:

$$\bar{X}_{(k+1,i)}^L = RX_{(k+1,i)}^L + T_{(k+1,i)}^L \quad (15)$$

最终误差模型变为:

$$loss = D(G(X_{(k+1,i)}^L, T_{k+1}^L)) \quad (16)$$

$$T_{k+1}^L = [t_x, t_y, t_z, \theta_x, \theta_y, \theta_z]^T \quad (17)$$

通过对向量求导, 解雅可比矩阵 J , 得到 T_{k+1}^L :

$$T_{k+1}^L \leftarrow T_k^L - (J^T J + \lambda diag(J^T J))^{-1} J^T d \quad (18)$$

通过帧间匹配得到 T_{k+1}^L 后, 仍需通过线性陀螺法对地图间的匹配进行修正, 以输出精确的里程计信息。为了解决大场景较易出现机器人位姿漂移的问题, 本文加入 BA 和 NDT 方法进行位姿修复, 其中 DNT 利用线面特征矩阵信息获得 C_Σ , 进行位姿初步优化。

4.2 线面特征矩阵信息

在 LOAM 算法中, 没有回环检测会导致里程计出现漂移, 但是可以通过点云特征特供相似环境信息。把每一帧点云划分成八叉树体素网格, 每一个网格存储点云特征信息。假设激光雷达的坐标点 $P_i = [P_{ix}, P_{iy}, P_{iz}]^T$, 代表第 i 个在雷达坐标系下的点云坐标。

每个体素中的点可表示为:

$$C_c = \begin{bmatrix} \text{round}(P_{ix}/D_x) \cdot D_x + 0.5D_x \\ \text{round}(P_{iy}/D_y) \cdot D_y + 0.5D_y \\ \text{round}(P_{iz}/D_z) \cdot D_z + 0.5D_z \end{bmatrix} \quad (19)$$

其中, D_x, D_y, D_z 分别表示一帧点云在空间中的大小; C_c 为体素中心的坐标。

假设在 C_c 中有 M 个点云数据, 为了求得这块点云簇数据的平均值, 用 \bar{P} 来表示。

$$\bar{P} = \frac{1}{M} \left(\sum_{i=1}^M P_i \right) \quad (20)$$

当前帧的协方差可表示为

$$C_{\Sigma} = \frac{1}{M-1} \left(\sum_{i=1}^M (P_i - \bar{P})(P_i - \bar{P})^T \right) \quad (21)$$

下一帧点云均值可表示为:

$$\bar{P}' = \frac{1}{M+1} (M \bar{P}' + P_{M+1}) \quad (22)$$

下一帧的协方差矩阵可表示为:

$$C_{\Sigma}' = \frac{1}{M} ((M-1)(C_{\Sigma}') + (M+1) \cdot (\bar{P}' - \bar{P})(\bar{P}' - \bar{P})^T + 2(P_{M+1} - \bar{P}') \cdot (\bar{P}' - \bar{P})^T + (P_i - \bar{P}')(P_i - \bar{P}')^T) \quad (23)$$

其中, C_{Σ}' 表示上一帧的协方差矩阵, \bar{P}' 表示上一帧的平均值, 协方差矩阵代表了当前点云簇里面的数据和前后帧之间的波动情况, 当前帧协方差矩阵由前后帧均值、前帧协方差组成。

4.3 BA/NDT-LOAM 融合算法

利用得到的协方差矩阵进行融合, 针对初始化阶段 LOAM 算法输出的粗略位姿精度较低的问题, 本文采用 BA 优化 LOAM 的位姿信息。随后, 基于 NDT 算法提取的线面特征作为回环检测的评判标准, 最终通过 Ceres 库实现整体位姿的修复和地图更新。改进后的激光 BA 通过多帧之间点-线、点-面之间的对应关系进行优化, 将边缘点集中到同一个边缘, 将平面点对齐到同一平面上, 便于特征管理, 如图 8 所示。

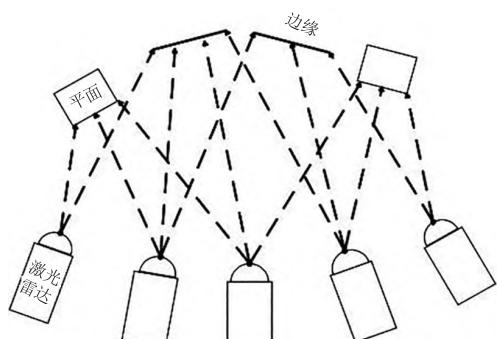


图 8 激光 BA 图
Fig. 8 LiDAR BA diagram

5 实验分析

5.1 实验硬件平台搭建

为验证本文方法在实际环境中的性能, 依据图 9 的硬件系统搭建了实验平台。底层为手持式移动平台, 中间通信层负责 PC 端与激光雷达的串口通信, 并通过 ROS(机器人操作系统)实现数据传输, 决策层则在 PC 上运行 SLAM 算法。

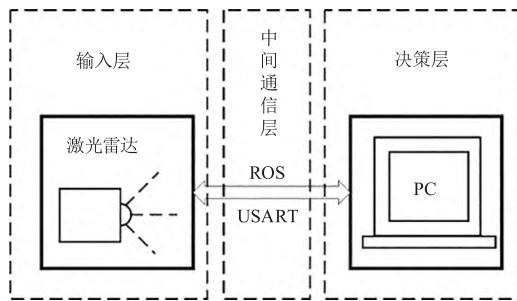


图 9 硬件系统图

Fig. 9 Hardware system

实验设备包括联想 Y7000 电脑, 配置为 Intel i5-10300H 和 NVIDIA GeForce GTX 1650 显卡; 激光雷达为大疆 LIVOX AVIA, 频率 10 Hz, 水平视场角为 70.4°, 垂直视场角为 77.2°。实验平台如图 10 所示, 右上角是手持模式实验图。

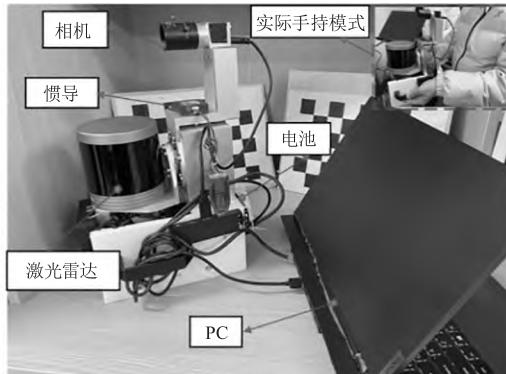


图 10 实验平台

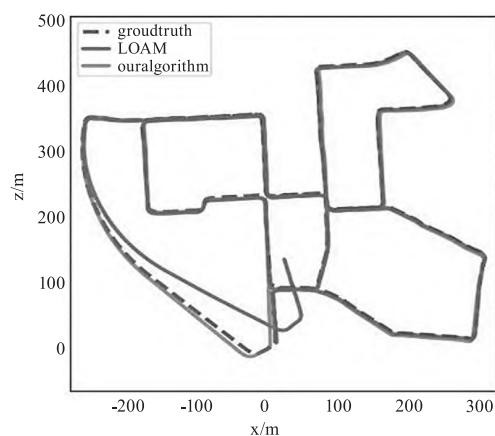
Fig. 10 Experimental platform

5.2 公开数据集验证与分析

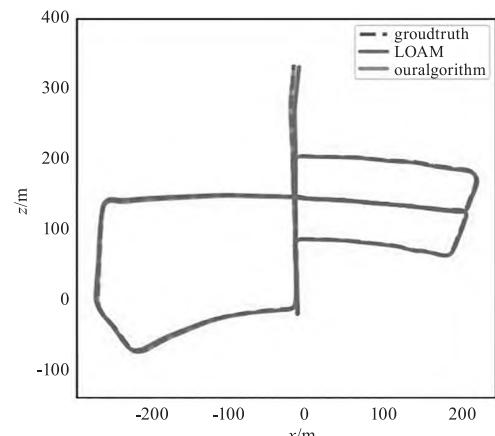
为验证算法的有效性, 本文利用文献[21]提供的 KITTI 数据集进行实验。该数据集在德国卡尔斯鲁厄的农村和高速公路区域拍摄, 包含最多 15 辆汽车和 30 名行人的场景。数据由装有激光雷达的车辆以 10 Hz 频率采集, 涵盖市区、乡村和高速公路等多种环境, 提供图片、视频、雷达数据以及真实的位姿估计和传感器与载体坐标系的转换关系。

本文选取四个不同路况的室外场景 KITTI00、KITTI05、KITTI08、KITTI09, 以及大型工业场景 KITTI10 分别与 LOAM 以及测量真值 groundtruth 进行轨

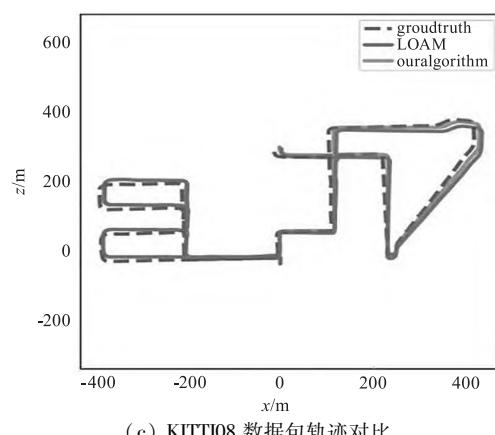
迹的对比,如图 11 所示,对于 KITTI00、KITTI05、KITTI08、KITTI09 数据包,由图 11(a)、(b)、(c)、(d)可以看出,本文算法能够精确定位和跟踪,与真实轨迹基本重合,显示出良好的定位效果。对于 KITTI 02 数据包,由于后半段雷达运动过快导致里程计数据长时间缺失,从图 11(e)可以看出,本文算法与 LOAM 均未能完全跟踪和定位。LOAM 在大部分情况下跟踪失败,而本文算法仅有少部分失效,表现出显著优于 LOAM 的跟踪与定位能力,说明本文方法在里程计漂移时能够提供短时间内的有效位姿估计。



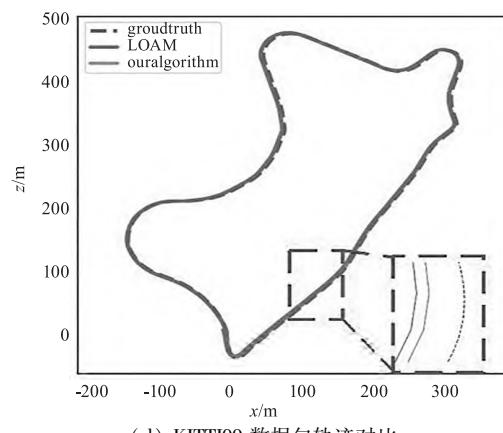
(a) KITTI00 数据包轨迹对比



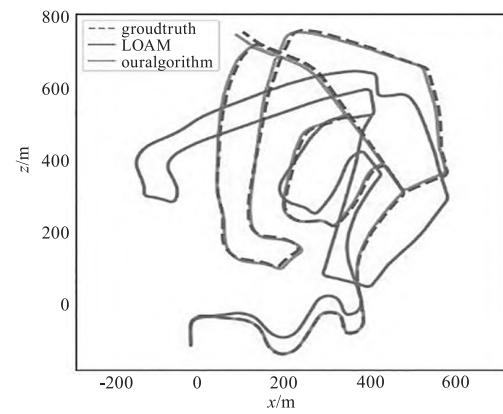
(b) KITTI05 数据包轨迹对比



(c) KITTI08 数据包轨迹对比



(d) KITTI09 数据包轨迹对比



(e) KITTI02 数据包轨迹对比

图 11 算法估计轨迹和真实轨迹比较

Fig. 11 Comparison between ground truth and estimated trajectories of the two algorithms

本文对 11 个数据包进行了实验,与 LOAM 算法进行了对比。表 1 是 11 个数据包的位姿均方根误差(RMSE)对比,表 2 是在 CPU i5-10300H 上运行的平均速度,速度定义为处理一个数据包时每帧的平均耗时。

表 1 在 KITTI 数据集中位置均方根误差的对比
Tab. 1 Comparison of translation RMSE in KITTI dataset

数据集	本文方法/m	LOAM/m
KITTI 00	0.036	0.035
KITTI 01	0.017	0.019
KITTI 02	0.015	X
KITTI 03	0.011	0.027
KITTI 04	0.012	0.010
KITTI 05	0.025	0.027
KITTI 06	0.014	0.024
KITTI 07	0.003	0.015
KITTI 08	0.004	0.002
KITTI 09	0.005	0.019
KITTI 10	0.018	0.060

表 2 在 KITTI 数据集中运行速度的对比

Tab. 2 Comparison of operating speed in KITTI dataset

数据集	本文方法/s	LOAM/s
KITTI 00	0.027	0.062
KITTI 01	0.030	0.088
KITTI 02	0.033	0.045
KITTI 03	0.032	0.053
KITTI 04	0.021	0.069
KITTI 05	0.031	0.029
KITTI 06	0.040	0.035
KITTI 07	0.031	0.081
KITTI 08	0.028	0.077
KITTI 09	0.032	0.074
KITTI 10	0.040	0.087

从表 1 可看出,本文方法的总体精度略优于 LOAM。针对整个 KITTI 数据集,LOAM 的总 RMSE 为 23.8 cm,而本文方法为 16 cm,减小了约 33%。对于 LOAM 无法完全跟踪的 KITTI02 数据包,本文方法依然能够有效跟踪。

表 2 显示,在 CPU i5-10300H 上,本文方法的平均每帧耗时为 0.031 s,而 LOAM 为 0.069 s,说明本文方法的运行速度略快于 LOAM。LOAM 的帧间配准依赖于 ICP 算法,导致配准时间较长,ICP 耗时约为 11 s,且运行中需要不断调整地图对齐度,进一步增加了时间开销。相比之下,本文方法采用 BA/NDT 进行配准和回环检测,初始化时间约为 8 s,因此整体运行速度是 LOAM 的 2.23 倍。

5.3 关键帧信息融合效果

在实际应用 NDT 与 LOAM 进行点云存储融合时,本文算法在初始化阶段对点云进行了存储管理,图 12(c)中通过虚线方格进行点云线面特征信息的管理,如图 12(d)所示。

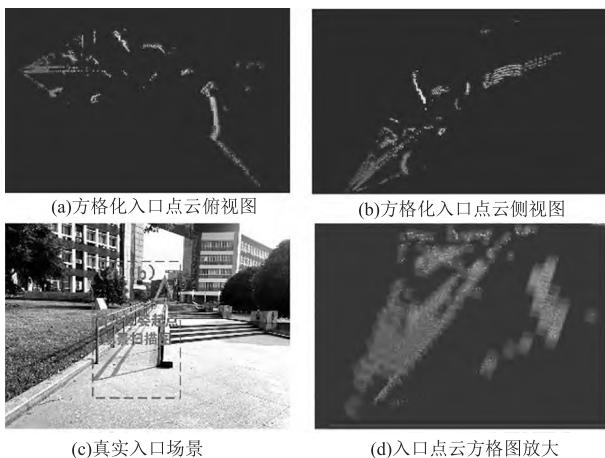
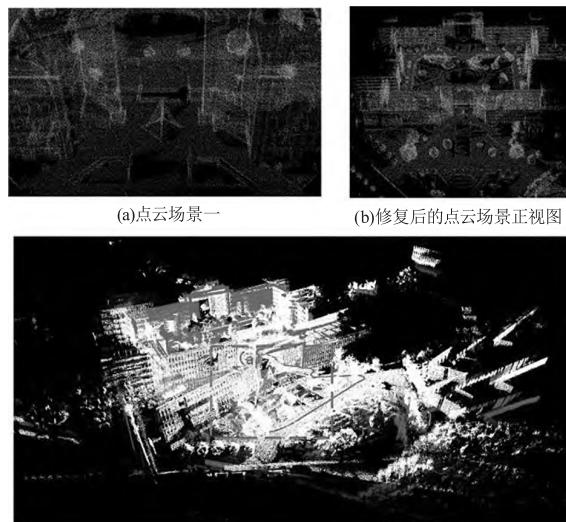


图 12 点云存储管理

Fig. 12 Point cloud storage management

5.4 实际环境验证与分析

为了验证本文方法在较大运动范围内的定位效果,选取教学楼作为场景进行测试,如图 13 所示,起点到终点的位置重合,采用 cartographer 结果作为真值。本文方法得到的轨迹和使用 LOAM 测得的轨迹和地图对比如图 13(c)所示,贴住地面的点云和轨迹代表修复后的地图和里程计。结果表明,图 13(a)和 13(b)中大场景下的地图漂移得到了很好的修复,场景还原度较高。

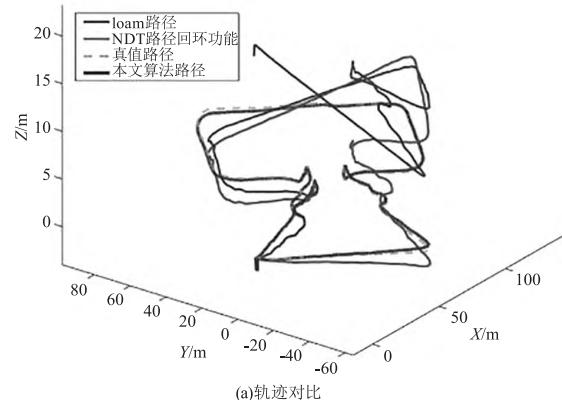


(c)不加回环检测和加了回环检测的地图场景

图 13 实际测试轨迹和地图对比

Fig. 13 Actual test track and map comparison

在测试中,手持设备沿 554 m 路径运行,如图 14(c)所示。图 14(a)展示 LOAM、NDT 路径回环、真值与本文算法 BA/NDT + LOAM 的对比。结果表明,本文方法实现了闭环,地图和轨迹均与真实轨迹高度贴合。图 14(b)展示了修复后的地图和轨迹,表 3 是算法的均方根误差对比,本文方法的总均方根误差最小,为 20.23 cm。由于设备在大场景模式下进行扫描建图,微小波动可忽略不计,只要检测到回环,地图中的里程计误差即可得到修复,场景大小与真实地图基本一致。



(a)轨迹对比

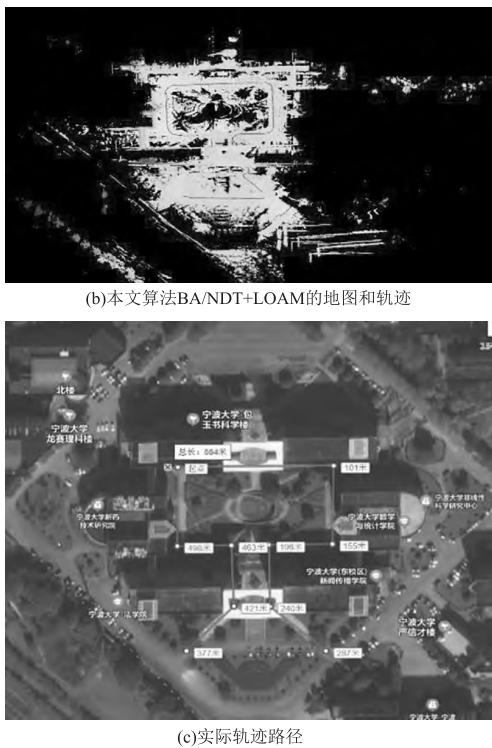


图 14 回环检测和修复效果

Fig. 14 Loop closure detection and repair

表 3 不同算法的均方根误差对比

Tab. 3 Comparison of RMSE of different algorithms

方法	RMSE/m
LOAM	0.8023
NDT + LOAM	0.8569
本文方法	0.2023

6 结论

本文提出一种 BA/NDT 与 3D 激光雷达松耦合融合的方法。激光雷达采用 LOAM 算法, 具有 LOAM 的特征提取、里程计解算等所有功能。在里程计最后输出阶段利用 NDT 检测回环为激光雷达 LOAM 系统提供位姿修复。在雷达移动场景过大激光雷达跟踪漂移时, BA 能提供短时间内的位姿估计。通过公开数据集和实际环境的测试验证了本文方法的可行性, 实验结果表明本文方法具有较高的定位精度, 在公开数据集测试中总的位置均方根误差为 16 cm, 在实际环境中测试总的位置均方根误差为 20.23 cm, 总的平均运行速度是 LOAM 的 2.23 倍, 可以基本满足实时性的要求, 且有效提高了定位精度。

参考文献:

- [1] Grisetti G, Stachniss C, Burgard W, et al. Improved techniques for grid mapping with Rao-Blackwellized particle

- filters[J]. IEEE Transactions on Robotics, 2007, 23(1): 34–46.
- [2] Kohlbrecher S, Von Stryk O, Meyer J, et al. A flexible and scalable SLAM system with full 3D motion estimation[C]//2011 IEEE International Symposium on Safety, Security, and Rescue Robotics. Kyoto: IEEE, 2011: 155–160.
- [3] Hess W, Kohler D, Rapp H, et al. Real-time loop closure in 2D LIDAR SLAM[C]//IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2018: 16–21.
- [4] Zhang J, Singh S. Lidar odometry and mapping in real-time[J]. Robotics: Science and Systems, 2014, 2(9): 1–9.
- [5] 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: IEEE, 2018: 4758–4765.
- [6] Shan T X, Englot B, Ratti C. LVI-SAM: tightly-coupled Li-dar-visual-inertial odometry via smoothing and mapping[C]//2021 IEEE International Conference on Robotics and Automation (ICRA). Xi'an: IEEE, 2017: 5692–5698.
- [7] Cattaneo D, Vaghi M, Valada A. LCDNet: deep loop closure detection and point cloud registration for LiDAR SLAM[J]. IEEE Transactions on Robotics, 2022, 38(4): 2074–2093.
- [8] Tsintotas K A, Bampis L, Gasteratos A. The revisiting problem in simultaneous localization and mapping: a survey on visual loop closure detection[J]. IEEE Transactions on Intelligent Transportation Systems, 2022, 23(11): 19929–19953.
- [9] Biber P, Strasser W. The normal distributions transform: a new approach to laser scan matching[C]//Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003). Las Vegas: IEEE, 2003: 2743–2748.
- [10] Magnusson M, Andreasson H, Nuchter A, et al. Appearance-based loop detection from 3D laser data using the normal distributions transform[C]//2009 IEEE International Conference on Robotics and Automation. Kobe: IEEE, 2009: 23–28.
- [11] 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). 2018: 4802–4809.
- [12] Behley J, Stachniss C. Efficient surfel-based SLAM using 3D laser range data in urban environments[EB/OL]. <https://www.roboticsproceedings.org/rss14/p16.pdf>.
- [13] Chen X, Läbe T, Milioto A, et al. OverlapNet: loop closing

- for LiDAR-based SLAM[DB/OL]. (2021-05-24). <https://arxiv.org/abs/2105.11344>.
- [14] Ma J Y, Zhang J, Xu J T, et al. Overlap Transformer: an efficient and yaw-angle-invariant transformer network for LiDAR-Based place recognition [J]. IEEE Robotics and Automation Letters, 2022, 7(3):6958-6965.
- [15] Guadagnino T, Chen X, Sodano M, et al. Fast sparse LiDAR odometry using self-supervised feature selection on intensity images[J]. IEEE Robotics and Automation Letters, 2022, 7(3):7597-7604.
- [16] DeTone D, Malisiewicz T, Rabinovich A. SuperPoint: self-supervised interest point detection and description[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops. Salt Lake City: CVF, 2018:224-236.
- [17] Liu Z, Liu X Y, Zhang F. Efficient and consistent bundle adjustment on lidar point clouds[J]. IEEE Transactions on Robotics, 2023, 39(6):4366-4386.
- [18] Chen Y K, Liu J H, Zhang X Y, et al. Voxelnext: fully sparse voxelnet for 3D object detection and tracking [C]//2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Vancouver: IEEE, 2023:21674-21683.
- [19] Kang Z X, Wang S Z, Cui Y Y, et al. Transformer-based algorithm for voxelized laser point cloud target detection[J]. Laser & Infrared, 2023, 53(2):202-207. (in Chinese)
康自祥, 王升哲, 崔雨勇, 等. 基于 Transformer 的体素化激光点云目标检测算法[J]. 激光与红外, 2023, 53(2):202-207.
- [20] Chai M N, Liu Y S, Ren L J. Two-step loop closure detection based on laser point cloud NDT features[J]. Laser & Infrared, 2020, 50(1):17-24. (in Chinese)
柴梦娜, 刘元盛, 任丽军. 基于激光点云 NDT 特征的两步回环检测[J]. 激光与红外, 2020, 50(1):17-24.
- [21] Anon. KITTI datasets[DB/OL]. <http://www.cvlibs.net/datasets/kitti> .