

# 基于 SR-Context 的激光雷达点云闭环检测算法

李炯<sup>1</sup>, 邵金菊<sup>2\*</sup>, 王任栋<sup>3</sup>, 赵凯<sup>3</sup>, 梁震<sup>1</sup>

<sup>1</sup>中国人民解放军 95848 部队, 湖北 孝感 432019;

<sup>2</sup>山东理工大学交通与车辆工程学院, 山东 淄博 255000;

<sup>3</sup>陆军军事交通学院军事交通运输研究所, 天津 300161

**摘要** 针对传统激光雷达闭环检测算法受动态障碍物干扰较大、关键帧搜索以及特征匹配耗时较长的问题, 采用 MF-RANSAC 算法并改进 ScanContext, 提出一种鲁棒性更强、耗时更短的 SR-Context 激光雷达闭环检测算法。首先, 利用区域生长算法对扇形栅格化后的点云进行分割; 随后, 不依赖于目标识别和跟踪, 借助动态区域多点选取和多属性查询对应点, 提出一种 MF-RANSAC 算法快速实现动态目标剔除; 最后, 通过简化特征匹配计算和删除闭环历史匹配帧的方式改进 ScanContext 算法, 对去除动态目标后的栅格提取特征实现闭环检测。分别在 KITTI 数据集与实车数据集下进行测试, 实验结果表明, 本文算法在城市动态环境下能够快速准确实现闭环检测进而提高激光雷达建图精度, 且平均耗时仅为 ScanContext 算法耗时的 40%。

**关键词** 传感器; 闭环检测; 智能车; 动态实时定位与建图; 激光雷达

中图分类号 TP242

文献标志码 A

doi: 10.3788/AOS202141.2228002

## A SR-Context Loop-Closure Detection Algorithm of Lidar Point Clouds

Li Jiong<sup>1</sup>, Shao Jinju<sup>2\*</sup>, Wang Rendong<sup>3</sup>, Zhao Kai<sup>3</sup>, Liang Zhen<sup>1</sup>

<sup>1</sup>95848 Army of P.L.A., Xiaogan, Hubei 432019, China;

<sup>2</sup>School of Transportation and Vehicle Engineering, Shandong University of Technology, Zibo, Shandong 255000, China;

<sup>3</sup>Institute of Military Transportation, Army Military Transportation University, Tianjin 300161, China

**Abstract** Traditional light detection and ranging (lidar) loop-closure detection algorithms are greatly interfered with by dynamic obstacles, and key-frame search and feature matching take a long time. In response, this paper proposed a less time-consuming SR-Context lidar loop-closure detection algorithm with stronger robustness based on the multiple-features random sample consensus (MF-RANSAC) algorithm and an improved ScanContext algorithm. Firstly, the region growing algorithm was used to cluster the point clouds that had undergone fan-shaped rasterization. Then, an MF-RANSAC algorithm was proposed to eliminate dynamic targets quickly. This algorithm was based on multi-point selection in a dynamic region and query of corresponding points with multiple attributes rather than target recognition and tracking. Finally, the ScanContext algorithm was improved by simplifying the feature matching calculation and deleting the loop-closure historical matching frames. Loop-closure detection of the point clouds of the current frame after elimination of dynamic targets was thus achieved. Tests were carried out on a KITTI dataset and a real vehicle dataset. The experimental results show that the proposed method delivers quick and accurate loop-closure detection in dynamic urban environments and thereby improves lidar mapping accuracy. The average time it takes is only 40% of that of the ScanContext algorithm.

**Key words** sensors; loop-closure detection; intelligent vehicle; simultaneous localization and mapping; lidar

**OCIS codes** 280.3640; 280.4788

收稿日期: 2021-04-01; 修回日期: 2021-05-01; 录用日期: 2021-06-03

基金项目: 天津市应用基础与前沿技术研究计划(15JCQNJC01000)

通信作者: \*shaojinju@sdut.edu.cn

# 1 引言

动态实时定位与建图(SLAM)是无人驾驶领域的一项关键技术<sup>[1]</sup>。尤其在全球定位系统(GPS)信号较弱或目标环境未知的场景下,该项技术对无人驾驶的安全性至关重要。目前 SLAM 领域常用的传感器分两种:激光雷达<sup>[2-3]</sup>和立体视觉相机<sup>[4-5]</sup>。由于相机受光照影响较大、无法全天候使用,本文选择测量精度高、受环境影响小的激光雷达进行研究。再考虑到 SLAM 系统主要包括前端里程计<sup>[6-8]</sup>和后端闭环优化<sup>[9-10]</sup>两大模块。其中闭环检测作为 SLAM 后端技术的关键一环,在建图后期消除累计误差、提高建图精度方面有着十分重要的现实意义。因此,本文重点介绍城市环境下基于激光雷达的闭环检测算法。

近年来,国内外学者对激光雷达闭环检测方法进行大量研究。Besl 等<sup>[11]</sup>提出一种迭代最近点(ICP)算法,直接对当前帧和历史帧数据逐点匹配,利用重合度判断是否达到闭环。虽然这种方法闭环检测精度较高,但点云数目庞大,因此算法特征匹配耗时较长、实时性较差。张剑华等<sup>[12]</sup>基于点云片段匹配约束,提出一种闭环检测算法。该方法虽然能够提高闭环检测效率,但是受动态障碍物干扰较大且精度较低。He 等<sup>[13]</sup>提出一种 M2DP 闭环检测算法,首先将点云直接投影到指定平面,利用投影向量构建环境特征矩阵,再借助奇异值分解得到全局描述子实现闭环检测。虽然该算法搜索较快、耗时较短,但其易受方向和动态目标影响且精度不高。Kim 等<sup>[14]</sup>基于扇形栅格提出一种 ScanContext 闭环检测算法。该方法分别借助行、列向量分析环境特征矩阵,并依次进行两次搜索实现闭环检测。虽然该方法在一定程度上能够改善 He 等的问题,保证旋转不变性、提高闭环检测准确率,但依旧无法解决动态障碍物干扰问题,且随着历史数据增加,关键帧搜索较慢、耗时较长。

针对以上闭环检测算法普遍存在受动态目标干扰较大、关键帧搜索以及特征匹配耗时较长的问题,本文跳过传统的目标识别、跟踪流程,结合(MF-RANSAC)算法和改进的 ScanContext 算法,提出一种精度更高且耗时更短的(SR-Context)闭环检测算法。首先,利用预处理栅格化和区域生长算法对点云进行分割;随后,通过动态区域多点选取和多属性查询对应点的方式改进传统的 RANSAC 算法<sup>[15]</sup>,实现动态障碍物的剔除;最后,在

ScanContext 算法的基础上,采用范数-欧氏距离优化特征匹配过程,并通过删除闭环历史匹配帧解决关键帧搜索耗时较长的问题,快速准确地实现当前帧点云的闭环检测。

# 2 点云分割

本文提出的 SR-Context 闭环检测算法主要包含点云分割、MF-RANSAC 算法和改进 ScanContext 算法三个模块,各模块之间关系如图 1 所示。考虑到城市环境中障碍物数目较多,因此为了从大量激光雷达点云中快速聚类这些障碍物便于后两个模块处理,本文基于扇形栅格化采用区域生长算法实现点云分割。

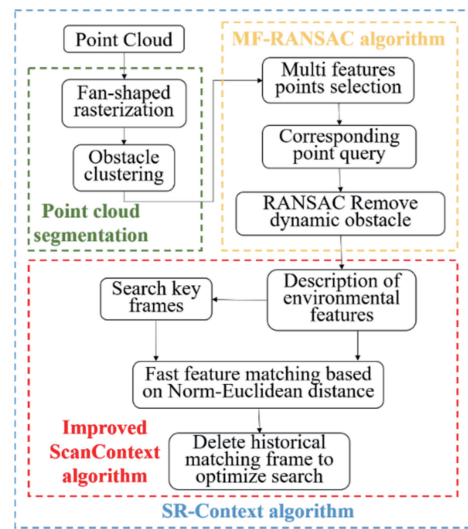


图 1 本文算法模块图

Fig. 1 Module diagram of proposed method

## 2.1 点云数据预处理

本文采用 HDL-64 激光雷达进行闭环检测研究。由于该激光雷达每帧产生将近 13 万个激光点,如果直接处理点云,算法耗时较长。所以本文根据激光雷达自身旋转特性,并借助设备输出的各点水平方位角和距离值,在智能车前后 40 m 范围内将点云划分为扇形栅格用于后续快速分割和闭环检测特征提取,如图 2 所示。点  $i$  在射线和圆周方向的索引值分别为

$$r_i = \frac{d_i}{T_r}, \quad (1)$$

$$s_i = \frac{\theta_i}{T_s}, \quad (2)$$

式中: $d_i$ 、 $\theta_i$  分别为设备输出的点  $i$  到激光雷达坐标原点的直线距离和水平方位角; $T_r$ 、 $T_s$  分别为射线和圆周方向坐标分辨率,本文分别取 0.4 m、 $6^\circ$ 。

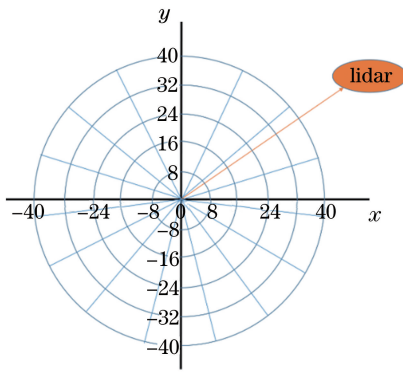


图 2 扇形栅格模型

Fig. 2 Fan-shaped grid model

### 2.2 目标聚类

扇形栅格化后,为了提取环境目标,本文在无需地面分割的基础上,通过栅格高度差采用区域生长算法对扇形栅格实现快速聚类。

1) 计算每个扇形栅格的高度差  $\Delta h$ ,再借助高度差阈值  $T_{\Delta h}$  找出高度差较大的种子栅格,本文  $T_{\Delta h}$  取 0.3 m。

2) 将种子栅格 8 邻域附近  $G_h$  接近的未遍历种子栅格放入同一目标列表,并对列表中未遍历栅格重复 8 邻域搜索,直至目标列表内无未遍历栅格为止,结束单个目标分割。

3) 依次遍历其余未遍历的所有栅格后,便可完成全部目标分割,结果如图 3 所示,图中不同矩形框代表不同目标,依次计算出目标重心  $b_i$ 。

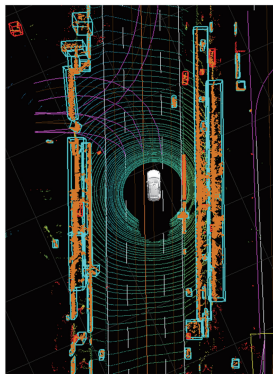


图 3 目标分割效果图

Fig. 3 Target segmentation effect diagram

$$\Delta h = G_h - G_l, \quad (3)$$

$$G_{m_h} - G_{n_h} \leq T_{ch}, \quad (4)$$

$$b_i = \frac{1}{n} \sum_{j=1}^n p_j, \quad (5)$$

式中:  $G_h, G_l$  分别为扇形栅格内点的最大、最小高度;  $G_{m_h}, G_{n_h}$  分别为相邻栅格  $m, n$  内点的最大高度;  $T_{ch}$  为同一目标高度差阈值,本文取 0.4 m;  $p_j$

和  $n$  分别为第  $i$  个障碍物的第  $j$  个点和其包含的总点数。

### 3 基于 MF-RANSAC 动态目标剔除

对于分割完成的目标,现有的技术大多采用识别或跟踪算法对其动静状态进行判断。往往导致运算耗时较长、过程繁琐。而在闭环检测中,我们不必获取目标精准的速度信息,只需清楚其动静即可。因此,本文跳过目标识别、跟踪流程,通过动态区域多点选取和多属性查询对应点的方式,提出一种 MF-RANSAC 算法来快速剔除动态目标。

#### 3.1 动态区域选取多特征点

车辆实际行驶阶段,当静态目标发生遮挡或者距离激光雷达不断变化时,其外形会发生波动导致重心产生偏移,此时单点匹配的 RANSAC 算法精度较差。因此,本文提出一种鲁棒性更强的动态区域选取多特征点的方法,以激光雷达为中心将车身周围的区域划分为四块,如图 4 所示。当目标重心处于不同区域时,分别选择相应区域内分割稳定的红色特征点与历史帧对应位置特征点进行匹配,并将匹配精度最高的结果作为最终结果,从而提升动态目标剔除的鲁棒性与准确性。

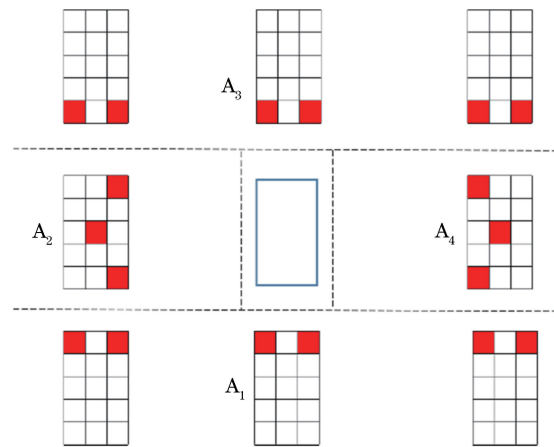


图 4 动态区域选点示意图

Fig. 4 Schematic diagram of point selection in dynamic region

#### 3.2 基于多属性对应点查询

在 RANSAC 匹配中为了得到两帧之间的旋转平移关系,至少需要两组对应目标。为了提高每次随机抽样精度,减少重复抽样次数,提高动态目标剔除效率。本文结合目标的多属性信息,选取与当前帧相差 5 帧的历史帧目标进行对比,提出一种精度更高的对应点查找方法。

1) 从当前帧  $m$  点云中随机抽取两个间距合

适<sup>[16]</sup>的目标  $O_1, O_2$ , 其重心分别为  $p_1, p_2$ 。综合目标的重心、外形和点云数目等属性依次计算目标  $O_1$  与第  $m-5$  帧内各目标的加权距离值  $D_{f_i}$ 。

$$D_b = \min \left[ \frac{|l_n - l_t|}{\max(l_n, l_t)}, \frac{|w_n - w_t|}{\max(w_n, w_t)} \right], \quad (6)$$

$$D_n = \frac{|p_n - p_t|}{\max(p_n, p_t)}, \quad (7)$$

$$D_{f_i} = D_g \times W_g + D_b \times W_b + D_n \times W_n, \quad (8)$$

式中:  $l_n, l_t, w_n, w_t, p_n, p_t$  分别为当前帧障碍物  $O_1$  与第  $m-5$  帧障碍物  $t$  的最小包络矩形长度、宽度、点云数目;  $D_g, D_b, D_n$  分别为当前帧障碍物  $O_1$  与第  $m-5$  帧障碍物  $t$  的重心偏差值、最小包络矩形尺寸偏差值和障碍物点云数目偏差值;  $W_g, W_b, W_n$  分别为上述偏差值对应的权重值, 本文分别取 0.5、0.2、0.2。

2) 从第  $m-5$  帧中提取出加权距离最小的  $N_{\min}$  个目标, 将其包含的重心点  $q_1$  作为目标  $O_1$  的候选点集  $M_1$ , 本文  $N_{\min}$  取 5。

3) 依次遍历点集  $M_1$  中的点  $q_1$ 。如图 5 所示, 先在  $M_1$  中第  $i$  个点  $q_1$  的  $K$  近邻范围内获取满足条件的点集  $M_{2_i}$ , 得到其与目标  $O_1, O_2$  的若干组对应关系。再将  $M_1$  中所有点  $q_1$  对应的点集  $M_{2_i}$  累加, 便可得到目标  $O_2$  的候选点集  $M_2$ , 且  $M_2$  包含的点数  $N_{\text{sum}}$  即为所有对应关系的组数。

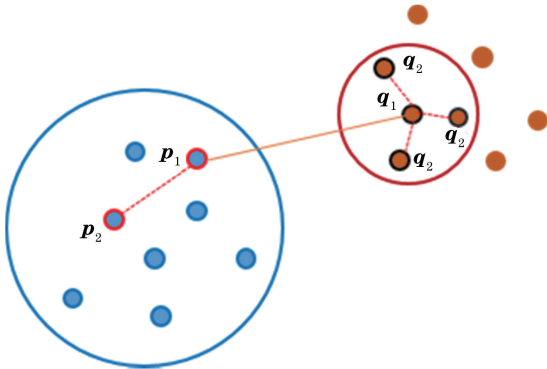


图 5 对应点查询示意图

Fig. 5 Schematic diagram of corresponding point query

$$M_{2_i} = \{q_2 \mid ||p_1 - p_2| - |q_1 - q_2|| < \epsilon\}, \quad (9)$$

$$M_2 = \sum_{i=1}^{N_{\min}} M_{2_i}. \quad (10)$$

### 3.3 MF-RANSAC 匹配剔除动态目标

基于上述内容, 本文提出的 MF-RANSAC 算法流程如下:

1) 利用 3.1 节动态区域多点选取法获取两帧点云中目标的多个稳定特征点。

2) 通过 3.2 节多属性查询对应点法快速抽取得到两帧点云对应目标集合。

3) 采用 RANSAC 算法依次将目标集合中两帧对应目标相应位置的特征点进行匹配。匹配过程中, 因为一帧点云中静态障碍物的总数通常大于动态障碍物, 所以将当前帧  $m$  与第  $m-5$  帧目标进行匹配时(图 6), 采用动态区域鲁棒特征点配准静态目标得到的内点数要比动态目标匹配对应的内点数多, 且即便静态目标重心偏移依旧可以借助有效特征点实现较好匹配效果。最终从目标集合中选取内点数目最多的一组结果作为本次 RANSAC 算法配准结果。

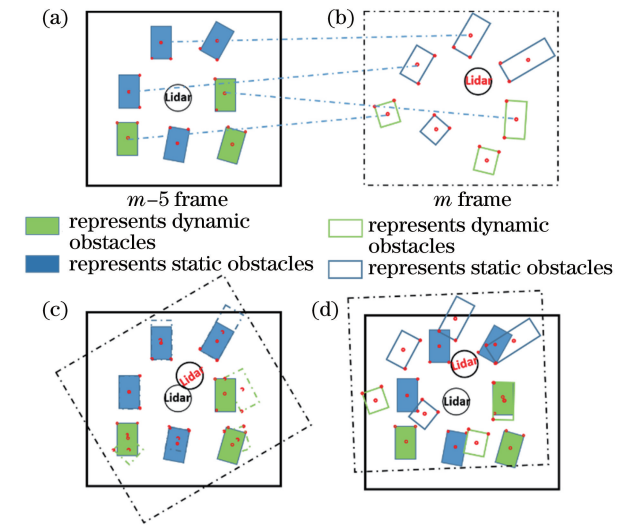


图 6 MF-RANSAC 配准示意图。(a)  $m-5$  帧; (b)  $m$  帧;

(c)(d) 基于静态和动态障碍物的多特征点最优匹配结果

Fig. 6 MF-RANSAC registration diagram. (a)  $m-5$

frame; (b)  $m$  frame; (c)(d) optimal matching

results of multiple feature point using static

obstacle and dynamic obstacle

4) 多次进行步骤 2 采样、步骤 3 配准操作, 保留内点最多的那次旋转、平移参数, 便可将动态目标作为外点准确剔除。

## 4 改进 ScanContext 闭环检测算法

ScanContext 闭环检测算法虽然能够准确实现闭环检测, 但随着历史帧增加搜索关键帧速度较慢, 且特征匹配耗时较长。因此, 本文改进 ScanContext 算法, 利用范数-欧氏距离优化 ScanContext 算法中平移特征矩阵再利用余弦判断相似的特征匹配过程, 并通过删除闭环历史匹配帧提高关键帧搜索速度, 进而快速准确实现去除动态目标后场景的闭环检测。

### 4.1 提取环境特征并获取关键帧

#### 4.1.1 扇形区域特征提取

上一节实现动态目标剔除后,为了在保留环境特征的同时减少栅格数目,尽可能避免扇形栅格重复划分,本文将第 2 节用于聚类的  $60 \times 100$  栅格图,直接按照射线方向合并,快速构建出新的  $N_s \times N_r$  特征提取栅格模型。因为本文  $N_s$ 、 $N_r$  分别取 60、20,所以只需按照射线方向将 5 个蓝色原始扇形栅格合并为一个新的橘色扇形栅格即可,如图 7 所示。随后,依次比较最新栅格内原始非动态目标栅格的  $G_h$ ,将其内部最大的  $G_h$  作为最新栅格的特征值。最终得到当前环境的特征矩阵  $P$ 。

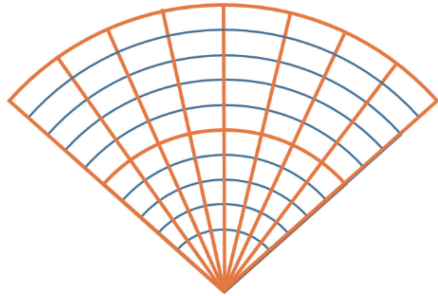


图 7 扇形栅格合并示意图

Fig. 7 Schematic diagram of fan-shaped grid merge

$$P_{ij} = \max_{k \in \{1,2,3,4,5\} \& k \in M} (G_{h-k}), \quad (11)$$

式中: $P_{ij}$  为最新栅格内第  $i$  行、第  $j$  列栅格对应的特征值; $G_{h-k}$  为最新栅格包含的第  $k$  个原始非动态目标栅格最大高度。

#### 4.1.2 Ring-key 选取关键帧

得到环境特征矩阵后,为了快速选取与当前帧最接近的历史关键帧,采用文献[14]的方法:借助范数的概念依次计算特征矩阵每个行向量  $n_r$  的范数值  $\varphi_r$ ,得到旋转不敏感环境描述向量  $k$ 。随后,将当前帧  $k$  与历史帧  $k$  建立 kd 树索引关系。再通过最近邻算法查找距离当前帧 20 帧以上且累计行驶超过 10 m 的历史帧,并计算其与当前帧的向量偏差值  $L_{lossij}$ 。最终按照由小到大的顺序挑选出偏差值小于阈值  $T_{ring}$  的  $N_{ring}$  帧数据作为关键帧,本文  $T_{ring}$  取 0.3。

$$\varphi_r = \frac{\|n_r\|_0}{N_s}, \quad (12)$$

$$k = (\varphi_1, \varphi_2, \dots, \varphi_{20}), \quad (13)$$

$$L_{lossij} = 1 - \frac{k_i \cdot k_j}{|k_i| |k_j|}. \quad (14)$$

### 4.2 基于范数-欧氏距离快速实现特征匹配

获取关键帧后,为了进一步确认车辆行驶是否

达到闭环,本文将当前环境特征矩阵看作列向量的组合,继续与关键帧数据进行特征匹配。考虑到车辆实际行驶至同一区域后,因为航向角偏差等原因,当前帧与关键帧各自特征矩阵列向量之间往往存在平移偏差。所以为了快速、准确获得特征矩阵相似值之和,减少特征匹配阶段向量相似性计算的复杂度、缩短耗时。本文改进了文献[14]中平移特征矩阵利用列向量余弦值判断相似性的方式,使用范数简化列向量并借助计算欧氏距离直接进行相似性判断。具体步骤如下:

1) 计算特征矩阵列向量的范数值  $l$ 、 $m$ ,将待匹配的特征矩阵简化为特征向量,如图 8 中  $p$ 、 $q$  所示。

2) 按照图 8 的方式,依次平移关键帧的特征向量,并将多出的最后一位列向量范数值往前补。再与当前帧特征向量快速计算相似值。直至首尾互换,挑选出最大的相似值作为两帧数据特征矩阵的最终相似值。

3) 按此方式,依次计算所有关键帧与当前帧特征矩阵的相似值。最终从中选取最大的相似值与相似阈值  $T_{sim}$  进行对比,从而完成闭环检测。

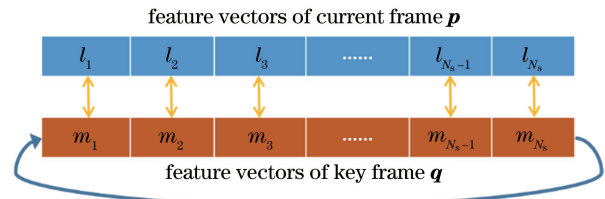


图 8 特征矩阵相似值计算示意图

Fig. 8 Similarity calculation of feature matrix

$$D(l, m) = \sqrt{\sum (l_i - m_i)^2}, \quad (15)$$

$$\text{sim}(l, m) = \frac{1}{1 + D(l, m)}, \quad (16)$$

$$\text{Dis}(k, n) = \max_{w \in \{0,1,\dots,N_s-1\}} \text{sim}(k_w, n), \quad (17)$$

式中: $D(l, m)$  为向量  $l$  与  $m$  之间的欧氏距离值; $\text{sim}(l, m)$  为向量  $l$  与  $m$  之间的相似性数值,范围在  $[0, 1]$ ;  $\text{Dis}(k, n)$  为第  $k$  帧与第  $n$  帧特征矩阵的相似值; $k_w$  为第  $k$  帧特征矩阵对应的特征向量平移  $w$  以后的特征向量; $n$  为第  $n$  帧特征矩阵对应的特征向量。

### 4.3 删除闭环历史匹配帧优化关键帧搜索

为了避免历史数据不断累加,造成关键帧搜索耗时倍数增长的问题。本文进一步改进文献[14]的方法,通过判断车辆是否达到闭环来不断清除重复场景的历史帧数据,优化关键帧搜索。

具体步骤如下:

1) 如果当前车辆实现闭环检测,此时将当前帧对应的最佳关键帧旋转不敏感环境描述向量  $k$  从历史帧数据中移除,重新构建 kd 树索引关系,继续下一帧点云处理。保证随着点云数据不断累加,历史帧数目维持稳定,从而避免 4.1.2 节选取历史关键帧时重复搜索,节省算法耗时。

2) 如果当前车辆并未实现闭环检测,此时保留历史帧数据,继续下一帧点云处理。

## 5 实验结果与分析

为了验证本文所提出闭环检测算法的准确性与实时性,本文分别在 KITTI 数据集与实车采集的城市数据集下进行测试。整个测试过程中,实验设备及环境配置如下:车载工控机采用英特尔 i7-8700 CPU、GTX1060 显卡与 16 GB 内存,系统则采用 Ubuntu16.04。全部程序基于 C++ 编写,配合 PCL 点云库,利用 Opencv 与 Matlab 进行显示与作图。

### 5.1 实验数据集介绍

#### 5.1.1 KITTI 数据集

KITTI 数据集是由德国卡尔斯鲁厄理工学院和丰田美国技术研究院共同组织采集的数据<sup>[17]</sup>。采集平台如图 9 所示:该车配备了一台 HDL-64 激光雷达,一套 RTK-GPS/IMU 定位装置以及一个双目相机。由于 KITTI Odometry 数据集中只有第

00~10 序列的数据包含真实轨迹,其中闭环重复率最高的数据依次为:00,02,05,08(08 包含反向闭环,00,02,05 则只包含同向闭环),所以本文选择以上四个序列的数据进行闭环检测实验。

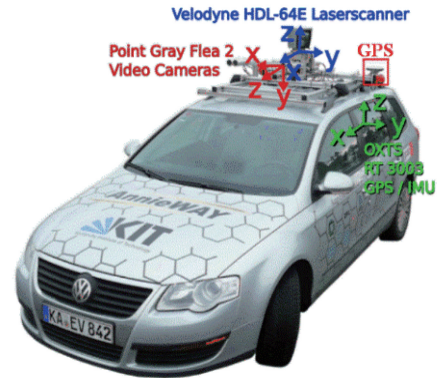


图 9 KITTI 数据集的采集平台

Fig. 9 Collection platform for KITTI dataset

#### 5.1.2 实车数据集

本文采用安装在实验平台顶部的 Velodyne 64 线激光雷达采集实车数据,如图 10(a)所示。该数据主要在城市动态环境下实验获取,其轨迹实况如图 10(b)所示,从 A 开始到 D 结束全程约 40 km, Lidar 输出频率为 10 Hz,平均车速约为 30 km/h。再通过图 10(b)的局部放大图进一步可知,车辆往返驶过同一道路时,动态障碍物的变化会导致场景特征产生差异。



图 10 实车数据集。(a)采集平台;(b)实验路段的卫星地图

Fig. 10 Real vehicle dataset. (a) Collection platform; (b) satellite map of the experimental section

### 5.2 准确率-召回率曲线分析

在 KITTI 数据集与实车数据集下,将本文提出的 SR-Context 算法分别与 ScanContext 算法及 M2DP 算法进行对比。计算出准确率与召回率<sup>[18]</sup>后,分别绘制不同环境下三种算法的准确率-召回率曲线进行分析。其中 SR-Context 算法和 ScanContext 算法根据第一次搜索选取关键帧数目的不同,又可分别划分为 SR-Context-10、SR-Context-50、ScanContext-10、ScanContext-50 进行

测试。

$$R_{\text{Precision}} = \frac{T_P}{T_P + F_P}, \quad (18)$$

$$R_{\text{Recall}} = \frac{T_P}{T_P + F_N}, \quad (19)$$

式中:  $T_P$  为检测闭环正确的数量;  $F_P$  为检测闭环错误的数目;  $F_N$  为检测非闭环的错误数量。

由此可知,准确率与召回率是一对相互矛盾的数值。随着一方增大,另一方通常会缩小。因此,闭

环检测算法性能越好,其对应的准确率与召回率值越大,曲线面积越接近于 1。

### 5.2.1 KITTI 数据集曲线分析

KITTI 数据集下 3 种算法对应 5 种不同形式的准确率-召回率曲线如图 11 所示,图中横轴 Precision 表示检测出闭环中正确的百分比,纵轴 Recall 表示所有闭环能够被正确检测出的百分比,5 条曲线分别表示不同算法在 KITTI 数据集下多次测试得到的准确率-召回率曲线图,由于准确率与召回率相互矛盾的特性,全部曲线均呈现逐渐下降的趋势。由图 11 可知,本文算法对应的曲线面积明显更大,因此在 KITTI 数据集下闭环检测效果要优于另外两种算法。因为 M2DP 算法利用平面投影、二维展开获取匹配特征,缺少空间信息,所以在反向经过同一地点时,受角度影响较大,如图 11(d)所示,

曲线面积几乎为 0,闭环检测效果较差。而 ScanContext 算法利用高度值保留空间信息,并借助行、列向量两次搜索获取关键帧,进而实现闭环检测。该算法受角度影响较小,因此由图 11 可知,其曲线包含面积大于 M2DP 算法,故闭环检测效果比 M2DP 较好。同时该算法选取的关键帧越多曲线面积越大,闭环检测效果越好。而本文算法跳过传统的目标识别、跟踪流程,基于分割结果使用 MF-RANSAC 算法快速滤除动态障碍物,因此获取的环境特征比 ScanContext 算法更加准确,曲线面积也更大,闭环检测精度更高。并且通过删除闭环历史匹配帧,使用较少的关键帧便可快速实现较为准确的闭环检测效果,所以在 4 张图中 SR-Context-10 与 SR-Context-50 对应的准确率-召回率曲线面积接近,闭环检测效果近似。

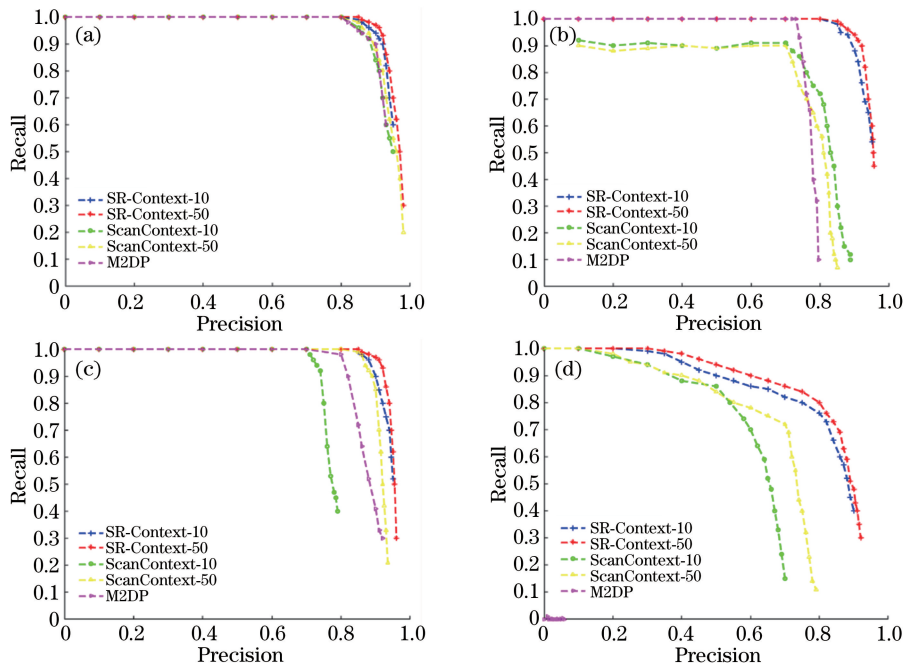


图 11 KITTI 数据集下准确率-召回率曲线图。(a) KITTI 00;(b) KITTI 02;(c) KITTI 05;(d) KITTI 08

Fig. 11 Precision-recall curve under KITTI dataset. (a) KITTI 00; (b) KITTI 02; (c) KITTI 05; (d) KITTI 08

### 5.2.2 实车数据集曲线分析

实车数据集下 3 种算法对应 5 种不同形式的准确率-召回率曲线如图 12 所示,图中 5 条曲线分别表示不同算法在实车数据集下多次测试得到的准确率-召回率曲线图。因为实际城市场景环境较为复杂且动态障碍物较多,所以与 KITTI 数据集下的曲线相比,实车场景下这 5 条曲线的面积均有缩小,故而闭环检测整体准确率均出现下降。由图 12 可知,虽然 ScanContext 算法曲线面积大于 M2DP 算法,但是受限于动态目标干扰,其准确率相对较低。而本文算法

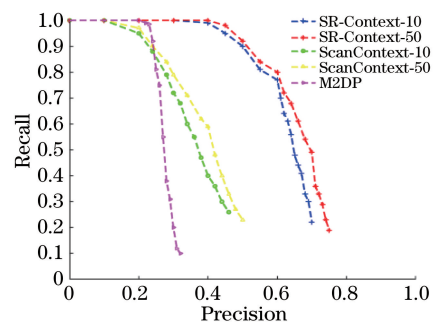


图 12 实车数据集下准确率-召回率曲线图

Fig. 12 Precision-recall curves under real vehicle dataset

利用 MF-RANSAC 配准技术,通过动态区域多点选取和多属性查询对应点的方式快速将动态障碍物作为外点剔除。与 ScanContext 算法相比,本文算法解决了动态目标干扰问题、曲线面积更大,进而提高了实际城市环境下闭环检测的准确率。虽然当车辆行驶至实车数据集包含的狭窄路段或者高度类似的建筑群时,准确率有所下滑,但与另外两种算法相比,本文算法依旧曲线面积最大、闭环检测精度最优。

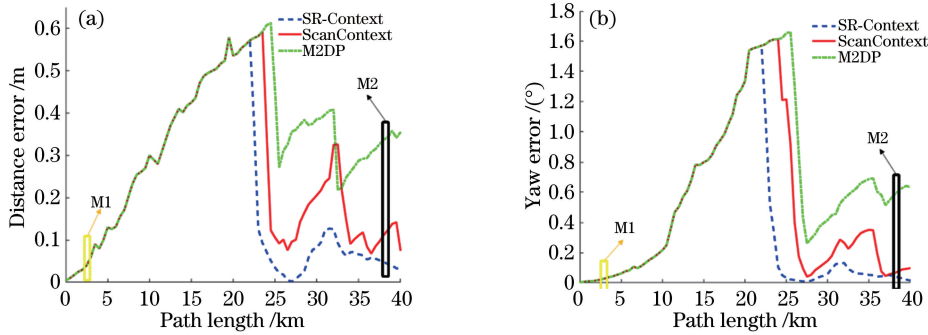


图 13 各算法误差对比图。(a)距离误差;(b)航向角误差

Fig. 13 Error comparison chart of each algorithm. (a) Distance error; (b) yaw error

由图 13 可知,刚开始由于行驶尚未达到闭环,随着逐帧匹配,三种算法建图误差均会累计增大。当行驶至闭环处时,M2DP 算法受角度影响最大,因此反向行驶导致该算法刚开始难以检测到闭环、建图误差持续增大。第一次回环优化后,该算法距离与航向角平均误差分别可达 0.315 m、0.562°,且闭环后整个过程稳定性较差(表 1)。ScanContext 虽然受角度影响较小,但受动态目标干扰在闭环区域

### 5.3 城市动态场景下闭环检测分析

为了更好地分析实际城市场景下闭环检测效果,本文结合建图精度对其展开讨论。分别将 SR-Context、ScanContext 以及 M2DP 算法,与 NDT 算法、非线性优化技术结合实现带有闭环优化的激光雷达里程计。在建图过程中,三种算法的航向角误差、距离误差变化分别如图 13 所示,且在第一次回环优化后,相应误差统计结果如表 1 所示。

内检测依旧不稳定,因此误差会呈现出波动。第一次回环优化后,距离与航向角平均误差分别可达 0.103 m、0.142°。而本文算法通过 MF-RANSAC 算法去除动态目标干扰,率先检测到闭环且检测相对稳定(图 13)。随后通过非线性优化技术,建图误差能够快速收敛且精度较高。与 ScanContext 相比,不仅回环检测标准差更低,且回环后距离与航向角平均误差分别减少 27% 和 56%,如表 1 所示。

表 1 第一次回环优化后各算法误差统计结果

Table 1 Statistical results of errors of each algorithm after the first loop optimization

Algorithm	Distance			Yaw		
	Average error / m	Maximum error / m	Mean standard deviation / m	Average error / (°)	Maximum error / (°)	Mean standard deviation / (°)
SR-Context	0.075	0.127	0.012	0.058	0.132	0.0053
ScanContext	0.103	0.326	0.032	0.142	0.368	0.0182
M2DP	0.315	0.413	0.024	0.562	0.657	0.0175

为了进一步说明本文算法对动态障碍物的抗干扰能力,从图 13 中挑选往返经过包含动态障碍物、具有较大航向角偏差的典型场景 M1、M2 进行分析。车辆往返经过该场景时,MF-RANSAC 算法能够快速准确剔除动态障碍物如图 14(a)、15(a)所示,其中蓝色点代表场景中静态特征点,红色点代表场景中动态目标点,A、B、C 三个框为场景中的典型动态目标区域。ScanContext 算法对往返场景中的全部点云提取特征,并构建特征矩阵如图 14(b)、15(b)所示。通过对比图中 A、B、C 三块区域可以看

出,动态障碍物导致了特征矩阵局部产生较大差异,此时如果对往返数据特征矩阵直接求取相似值,同一场景相似值较小无法准确检测闭环,导致该算法在图 13 中 M2 处产生累计误差。而本文算法采用 MF-RANSAC 算法准确剔除往返场景中的动态点,仅对环境静态点提取特征,构建特征矩阵如图 14(c)、15(c)所示。此时对比图中 A、B、C 三块区域可以看出,剔除动态目标后,往返数据特征矩阵基本一致能够准确检测闭环,故而本文算法在图 13 中 M2 处误差能够通过闭环优化逐渐减小。



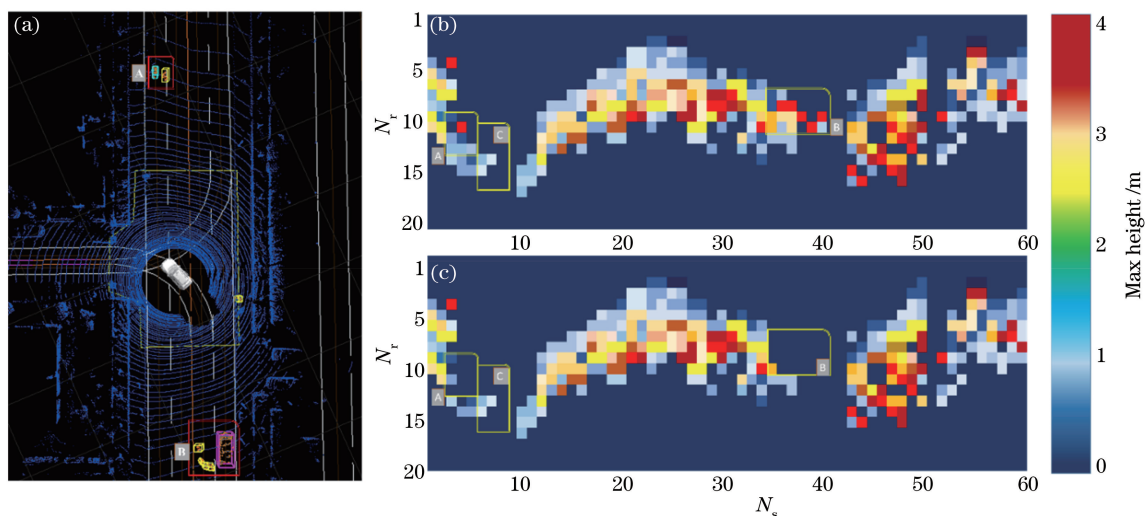


图 14 去的场景图 M1。(a)动态目标区分结果;(b) ScanContext 特征图;(c) SR-Context 特征图

Fig. 14 Leaving scene M1. (a) Dynamic target tracking result; (b) ScanContext feature map; (c) SR-Context feature map

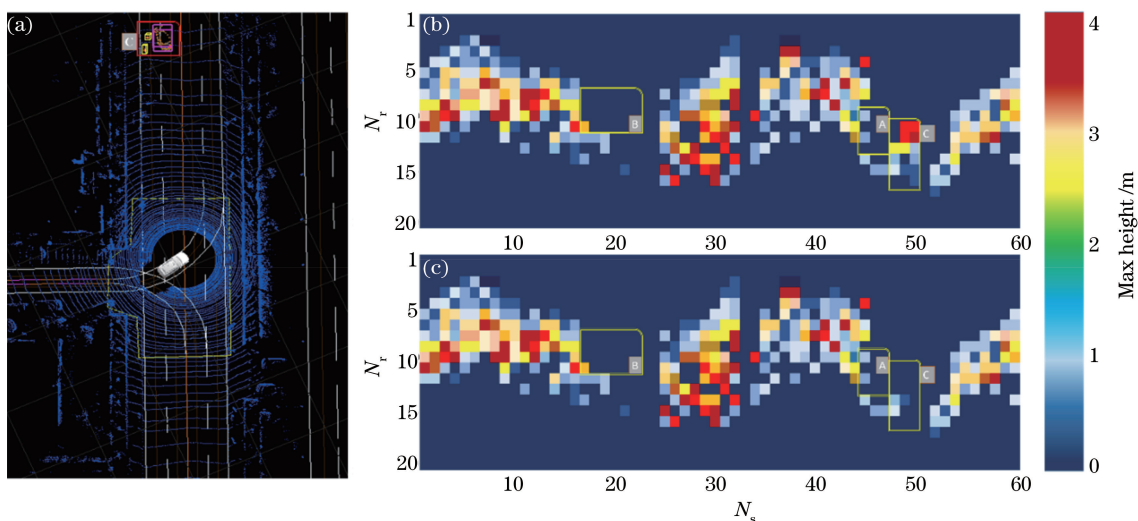


图 15 回的场景图 M2。(a)动态目标区分结果;(b) ScanContext 特征图;(c) SR-Context 特征图

Fig. 15 Returned scene M2. (a) Dynamic target tracking result; (b) ScanContext feature map; (c) SR-Context feature map

### 5.4 耗时对比

分别统计本文算法与 ScanContext 算法、M2DP 算法在两种数据集下的平均耗时,如表 2 所示。由于实际城市场景环境相对复杂,因此各算法在实车数据集下整体要比 KITTI 数据集耗时长。其中本文算法直接利用点云解析时设备输出的水平方位角与距离值进行扇形栅格化,与 ScanContext 算法借助直角坐标计算水平方位角和距离相比,运算量大大减小。随后,本文算法在统计栅格高度用于目标分割的同时,又直接将高度值和扇形栅格用于后续特征提取。所以,虽然本文添加了分割、配准流程,但整体特征提取部分的耗时与 ScanContext

算法相比仅增加 6% 左右,与 M2DP 算法耗时接近。在关键帧搜索阶段,本文通过删除闭环历史匹配帧加速关键帧的搜索,使得相同关键帧下本文算法搜索耗时远低于 ScanContext 算法,且随着关键帧数量的增加算法耗时增长较慢。对于特征匹配阶段,本文使用范数值简化特征矩阵列向量,再借助欧氏距离依次匹配特征矩阵简化后的特征向量获取相似值,与 ScanContext 算法借助夹角余弦平移处理特征矩阵列向量相比,简化了大量的特征匹配运算,极大地提高了匹配速度。由表 2 可知,与 ScanContext 算法相比,本文算法整体平均耗时减少 60%,运行速度可保持在 10 Hz 以上。

表 2 各算法的平均耗时

Table 2 Average running time of different algorithms

Algorithm	KITTI dataset			Real vehicle dataset		
	Getting	Searching candidate	Feature	Getting	Searching candidate	Feature
	descriptor /ms	frames /ms	matching /ms	descriptor /ms	frames /ms	matching /ms
SR-Context-10	23.9	13.2	4.8	27.1	15.4	5.7
SR-Context-50	23.9	30.3	19.5	27.1	37.1	24.2
ScanContext-10	22.5	23.3	35.5	25.2	26.2	38.1
ScanContext-50	22.5	107.4	148.6	25.2	115.7	151.5
M2DP	23.4	2.2	2.6	24.7	2.7	3.1

## 6 结 论

本文通过添加轻量化的分割、配准流程并改进 ScanContext 算法,提出一种新的 SR-Context 激光雷达闭环检测算法。该算法创新点主要体现在两个方面:1)跳过传统的目标识别、跟踪流程,在点云分割的基础上,通过动态区域多点选取和多属性查询对应点的方式提出一种新的 MF-RANSAC 配准算法,将动态目标作为外点快速剔除;2)在 ScanContext 算法 Ring-key 获取关键帧的基础上,通过删除闭环历史匹配帧提高关键帧搜索效率,并采用范数-欧氏距离简化特征匹配运算,最终极大地提高闭环检测效率。

在 KITTI 数据集和实车数据集下,分别对本文提出的算法进行验证。实验结果表明:本文算法利用 MF-RANSAC 算法能够有效克服动态障碍物的干扰,在两种数据集下其闭环检测平均准确率均明显优于 ScanContext 算法和 M2DP 算法,且在实际城市场景中建图误差收敛更快、精度更高。耗时部分,本文算法通过改进 ScanContext 算法的关键帧搜索、特征匹配过程,最终使得平均耗时减少 60%,且采用较少的关键帧参与匹配便可达到较好的效果,提高了实际场景下算法的实时性。

但本文算法依旧存在一些不足,例如当处在结构形状类似、目标特征单一的环境场景时,闭环检测精度较差容易误判。因此,下一步可以尝试引入激光雷达反射率或者融合相机 RGB 信息来实现这类场景的区分,以便于得到更好的闭环检测效果。

## 参 考 文 献

[1] Lu S D, Tu M Y, Luo X Y, et al. Laser SLAM pose optimization algorithm based on graph optimization theory and GNSS [J]. *Laser & Optoelectronics Progress*, 2020, 57(8): 081024.  
陆世东, 涂美义, 罗小勇, 等. 基于图优化理论和 GNSS 激光 SLAM 位姿优化算法[J]. *激光与光电子*

学进展, 2020, 57(8): 081024.

- [2] Jaud M, Rouveure R, Faure P, et al. Method for orthorectification of terrestrial radar maps[J]. *ISPRS Journal of Photogrammetry and Remote Sensing*, 2014, 97: 185-194.
- [3] Wang R D, Li H, Zhao K, et al. Robust localization based on kernel density estimation in dynamic diverse city scenes using lidar[J]. *Acta Optica Sinica*, 2019, 39(5): 0528003.  
王任栋, 李华, 赵凯, 等. 基于核密度估计的城市动态密集场景激光雷达定位[J]. *光学学报*, 2019, 39(5): 0528003.
- [4] Ma J, Bajracharya M, Susca S, et al. Real-time pose estimation of a dynamic quadruped in GPS-denied environments for 24-hour operation [J]. *The International Journal of Robotics Research*, 2016, 35(6): 631-653.
- [5] Zou B, Lin S Y, Yin Z S. Semantic mapping based on YOLOv3 and visual SLAM [J]. *Laser & Optoelectronics Progress*, 2020, 57(20): 201012.  
邹斌, 林思阳, 尹智帅. 基于 YOLOv3 和视觉 SLAM 的语义地图构建[J]. *激光与光电子学进展*, 2020, 57(20): 201012.
- [6] Wang R D, Xu Y C, Qi Y, et al. A robust point cloud registration method in urban dynamic environment[J]. *Robot*, 2018, 40(3): 257-265.  
王任栋, 徐友春, 齐尧, 等. 一种鲁棒的城市复杂动态场景点云配准方法[J]. *机器人*, 2018, 40(3): 257-265.
- [7] Im J H, Im S H, Jee G I. Vertical corner feature based precise vehicle localization using 3D LIDAR in urban area[J]. *Sensors*, 2016, 16(8): 1268.
- [8] Tang Z R, Liu M Z, Jiang Y, et al. Point cloud registration algorithm based on canonical correlation analysis[J]. *Chinese Journal of Lasers*, 2019, 46(4): 0404006.  
唐志荣, 刘明哲, 蒋悦, 等. 基于典型相关分析的点云配准算法[J]. *中国激光*, 2019, 46(4): 0404006.
- [9] Leutenegger S, Lynen S, Bosse M, et al. Keyframe-based visual - inertial odometry using nonlinear optimization [J]. *The International Journal of*

- Robotics Research, 2015, 34(3): 314-334.
- [10] Qiu C L, Huang D Z, Liu H W, et al. Loop closure detection algorithm based on convolutional autoencoder fused with Gist feature [J]. Laser & Optoelectronics Progress, 2019, 56(18): 181501.  
邱晨力, 黄东振, 刘华巍, 等. 融合 Gist 特征与卷积自编码的闭环检测算法 [J]. 激光与光电子学进展, 2019, 56(18): 181501.
- [11] 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.
- [12] Zhang J H, Wu J X, Feng Y T, et al. Segment based loop detection with point cloud matching constraint and trajectory drift optimization [J]. Control Theory & Applications, 2018, 35(12): 1750-1758.  
张剑华, 吴佳鑫, 冯宇婷, 等. 点云片段匹配约束和轨迹漂移优化的回环检测方法 [J]. 控制理论与应用, 2018, 35(12): 1750-1758.
- [13] He L, Wang X L, Zhang H. M2DP: A novel 3D point cloud descriptor and its application in loop closure detection [C] // 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), October 9-14, 2016, Daejeon, Korea (South). New York: IEEE Press, 2016: 231-237.
- [14] 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), October 1-5, 2018, Madrid, Spain. New York: IEEE Press, 2018: 4802-4809.
- [15] Chum O, Matas J, Kittler J. Locally optimized RANSAC [M] // Michaelis B, Krell G. DAGM 2003: pattern recognition. Lecture notes in computer science. Heidelberg: Springer, 2003, 2781: 236-243.
- [16] Aiger D, Mitra N J, Cohen-Or D. 4-points congruent sets for robust pairwise surface registration [J]. ACM Transactions on Graphics, 2008, 27(3): 1-10.
- [17] Geiger A, Lenz P, Stiller C, et al. Vision meets robotics: the KITTI dataset [J]. The International Journal of Robotics Research, 2013, 32(11): 1231-1237.
- [18] Liu Q, Duan F H, Sang Y, et al. A survey of loop-closure detection method of visual SLAM in complex environments [J]. Robot, 2019, 41(1): 112-123, 136.  
刘强, 段富海, 桑勇, 等. 复杂环境下视觉 SLAM 闭环检测方法综述 [J]. 机器人, 2019, 41(1): 112-123, 136.