

基于位姿阈值的 Cartographer SLAM 算法实验案例设计

安翠娟¹, 王旭², 丁大伟³

(1. 北京科技大学 自动化学院, 北京 100083; 2. 北京航发 北京航空材料研究院, 北京 100095; 3. 北京科技大学 自动化学院, 北京 100083)

摘要: 针对 Cartographer 算法回环检测耗时、实时性差的问题, 采用软硬件结合的方式, 设计了一种基于位姿阈值的 Cartographer SLAM 算法实验案例。该案例依托 Clearpath Jackal 无人车和 ROS 软件操作系统, 搭建了硬件平台, 采用 Cartographer SLAM 和位姿阈值优化算法设计了软件操作流程。结果表明, Cartographer SLAM 算法可获取无人车实时位姿, 构建清晰的实验室地图; 位姿阈值优化算法有效的剔除了回环检测时的冗余信息, 提升了计算效率和实时性。实践教学表明, 该实验加深了学生对 Cartographer SLAM 算法的认知, 激发了学生对无人车及人工智能等技术的学习兴趣, 提高了学生的工程实践能力和创新能力。

关键词: 无人车; Cartographer SLAM 算法; 位姿阈值; 实验设计

Experiment design of Cartographer SLAM algorithm based on pose threshold

An Cuijuan¹, Wang Xu², Ding Dawei³

(1. School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing 100083, China; 2. AECC Beijing Institute of Aeronautical Materials, Beijing 100095, China; 3. School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing 100083, China)

Abstract: Cartographer algorithm suffers from time-consuming problem of loop closure detection as well as poor real-time property. Aiming at these problems, an experiment of Cartographer SLAM algorithm based on pose threshold is designed by combining software and hardware. Rely on Clearpath Jackal unmanned vehicle and Robot Operating System (ROS), the experiment hardware platform is built, Cartographer SLAM algorithm and pose threshold

¹ **作者简介:** 安翠娟, 硕士研究生, 高级工程师, 主要研究方向为工业控制、机器人控制, E-mail: cuier1125@126.com。手机 13910403662

³ **通讯作者:** 丁大伟, 博士研究生, 教授, 主要研究方向为信息物理系统安全性与智能控制、多智能体系统优化与控制、无人车定位与导航, E-mail: ddaweiauto@163.com。手机 15011248835

optimization algorithm are adopted to design the software operation process. Experiment results show that the Cartographer SLAM algorithm can obtain the positional posture of the unmanned vehicle in real-time, and build a clear real map of laboratory. The pose threshold optimization algorithm efficiently solves the time-consuming problem of loop closure detection, and improves the computational efficiency and the real-time performance compared with the original Cartographer algorithm. The practical teaching shows that students' cognition of the Cartographer SLAM algorithm is deepened, their study interest in the unmanned vehicle and artificial intelligence is inspired, and their abilities of engineering practice and innovation are also improved with the help of the designed experiment.

Key words: unmanned vehicle; Cartographer SLAM algorithm; pose threshold; experiment

高精度定位是无人车安全移动的关键技术，也是无人车在未知环境中实现自主导航的前提和基础^[1,2]。在众多定位方法中，最常用的是同时定位与地图构建（Simultaneous Localization and Mapping）即 SLAM 技术^[3]。SLAM 技术突破了传统定位方法的局限，能够实现可靠的、高精度的定位并构建动态更新地图，提高了无人车移动的轨迹精度与安全性^[2]，被广泛应用于无人车的研究和开发。根据无人车搭载传感器和应用场景的不同，SLAM 算法也不尽相同。常用的激光雷达 SLAM 算法有 Gmapping 和 Cartographer。其中 Cartographer 算法可进行闭环检测，算法结构更加完整且精度较高^[1]，因此被广泛应用于机场、学校、仓储等较大场景对移动无人车建图定位。Cartographer 算法因增加了回环检测环节，存在耗时多、优化时计算量大^[4]、数据实时性差等问题，不利于及时去除冗余信息，降低了计算效率，影响了定位和建图的质量。

针对上述问题，本文以 Cartographer SLAM 算法为基础，以 Clearpath Jackal 无人车和 ROS 操作软件为依托平台，采用软硬件结合的方式，设计了一种基于位姿阈值的 Cartographer SLAM 优化算法实验案例，以提高计算效率、增加数据实时性、提升定位和建图效果。

1 Cartographer SLAM 算法框架

Cartographer SLAM 算法^[5]框架如图 1 所示。该算法主要由三大模块构成：传感器输入模块，局部 SLAM 模块和全局 SLAM 模块^[6]。传感器输入模块负责接收传感器的输入信息，

并对原始数据进行预处理。传感器数据主要指激光雷达的激光点和 IMU 的加速度、角速度信息，也可以选择性的使用里程计位姿信息^[7]。局部 SLAM 模块主要进行实时位姿更新和局部子图(submap)的建立。当激光帧和 IMU 等信息到达后，IMU 首先辅助激光雷达去除运动畸变，之后进行积分。IMU 的积分值作为激光点进行 ICP 配准的初值，之后进行 ICP^[8]迭代求解，得到载体的位姿和局部子图的位姿，激光帧经过叠加，形成子图。全局 SLAM 模块主要实现回环检测功能^[9]，相比于局部 SLAM，由于考虑了全局信息，因此计算效率较低。局部 SLAM 的目的是生成一系列良好的局部子图，而全局 SLAM 的工作则将这些局部子图优化形成一个全局地图。

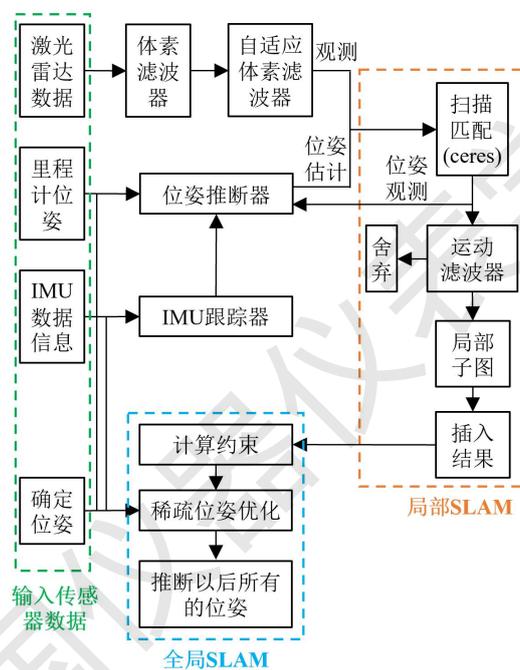


图 1 Cartographer SLAM 算法框架

2 实验案例软硬件系统设计

2.1 硬件系统设计

实验案例硬件系统主要包括车载底盘、传感器及控制操作元件。车载底盘采用 Clearpath Jackal 无人车、传感器采用 Velodyne-16 线激光雷达及 MPU-9250 惯性测量单元，控制操作元件使用索尼蓝牙手柄。

2.1.1 Clearpath Jackal 无人车

Jackal 无人车拥有有机载计算机、GPS 和 IMU，可以拓展搭载激光雷达、相机等其它多种传感器，集成了 ROS 机器人操作系统，可应用于室内、室外等多个场景。无人车系统如图 2 所示：



图2 Clearpath Jackal 无人车系统

2.1.2 Velodyne-16 线激光雷达

为使无人车能更加精确、快速的完成地图构建和路径规划，本实验案例平台选用 Velodyne-16 线激光雷达作为主要传感器。Velodyne-16 线激光雷达采用机械旋转的原理采集激光点，测量距离远，可实现单线和多线的测量。可一次采集多帧激光束，也可一次采集单帧激光束，适用于室内和室外等多个应用场景[10]。Velodyne-16 线激光雷达放于 Jackal 无人车前端，距车底中心位置的垂直距离是 0.308 米，水平距离是 0.12 米，Velodyne-16 线激光雷达实物如图 3 所示：



图3 Velodyne-16 线激光雷达

2.1.3 索尼蓝牙手柄

索尼蓝牙手柄用于操控 Jackal 无人车移动，完成数据采集和场景定位建图等任务，具有低延迟、低功耗、抗干扰能力高、稳定性强和连接蓝牙后不易断连等优点。蓝牙手柄实物如图 4 所示。手柄主要由两部分构成：主要按键区和辅助按键区。主要按键区负责控制无人车前后左右的移动及手柄开关，辅助按键区负责无人车的加速过程。



图 4 索尼蓝牙手柄

2.2 软件系统设计

软件系统设计基于机器人操作平台 ROS (Robot Operating System) [12,13]完成。ROS 是一款用于机器人开发的操作系统，通过进程消息间的通信机制，开发者可以高效的进行机器人的开发研究。

2.2.1 实验室真实地图构建软件设计

采用当前性能最佳的 Cartographer SLAM 算法获取无人车的实时位姿，并构建实验室真实环境的二维栅格地图。软件设计主要包含五部分：数据采集与存储、运动估计、局部地图匹配优化、回环检测、全局一致性地图和轨迹生成。设计流程如图 5 所示：打开无人车电源后，通过蓝牙将操作手柄连接至无人车 PC 端，控制无人车在实验室的移动方向和速度，校验位检验通过后，无人车开始移动。装载在无人车上的激光雷达、惯性测量单元等传感器开始采集数据并将数据存储下来。存储好的数据传送到 PC 端进行 Cartographer SLAM 算法复现以生成轨迹和地图。在点云配准进行运动估计前，得到的数据集会自行进行数据预处理，如利用 IMU 的高频对激光点进行线性插值，完成去除畸变的功能，保证配准时点云是丰富有效的。之后进行局部地图匹配，检测当前帧是否产生回环，如果产生回环则再次优化进行估计，最终生成全局一致性地图和轨迹。Cartographer SLAM 算法构建实验室真实地图软件设计流程如图 5 所示：

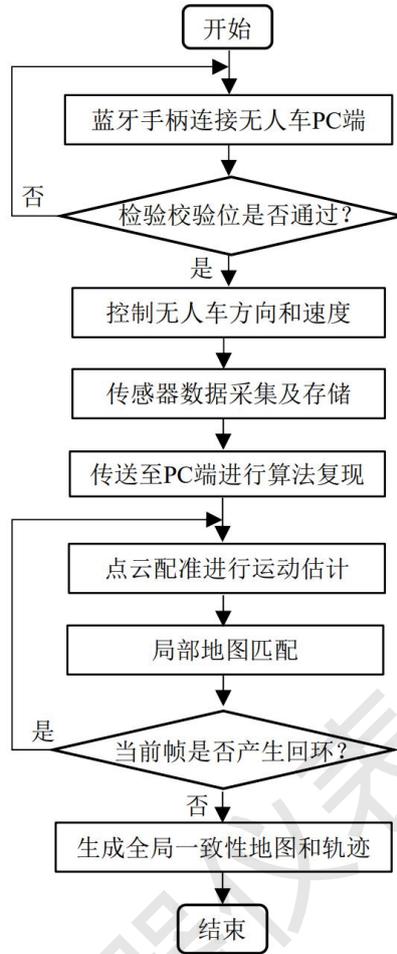


图5 Cartographer SLAM 算法构建实验室真实地图流程图

2.2.2 位姿阈值优化算法设计

随着建图范围逐渐增大，历史帧数量也快速增加。如果一个局部子图对应 100 个历史帧，20 个局部子图就对应 2000 个历史帧[14]。当新的局部子图生成时，新的局部子图会与所有历史帧尝试建立约束。此时会有大量的约束添加到优化队列中，造成优化队列积压，回环检测时间大幅提升，导致数据的实时性变差。为解决上述问题，采取位姿阈值优化算法，对新生成的历史帧进行筛选，将变化较小的历史帧剔除，将变化较大的历史帧加入优化队列，这样构建出来的运动约束节点不仅在数目上有所减少，而且还保留了原始约束节点的空间结构信息，这在减少了计算量的同时，也保证了位姿图优化结果的精度。

图 6 是位姿阈值优化算法示意图，通过该示意图可更好的理解位姿阈值优化算法的实现原理。未优化前，当前生成的局部子图要和所有的点云帧进行回环检测，产生回环后对所有回环检测到的点云帧进行 scan-to-map 的帧图间点云配准。如图(a)所示，小圆圈表示回环检测到的激光帧，其中深色的表示当前帧与上一帧的位姿相比变化较大的激光帧，而浅色的表示当前帧与上一帧的位姿相比变化较小的激光帧。由于要对所有激光点云帧进行

计算，因此随着地图的增大，计算时间会成倍的增长，严重损耗计算效率。而在实际应用中，这些位姿变化较小的激光帧即约束节点值较小的激光帧对真实的位姿估计起到的作用很小，因此可通过位姿阈值优化算法将颜色较浅的激光帧剔除掉，从而提高 Cartographer 算法的整体性能，如图(b)所示。

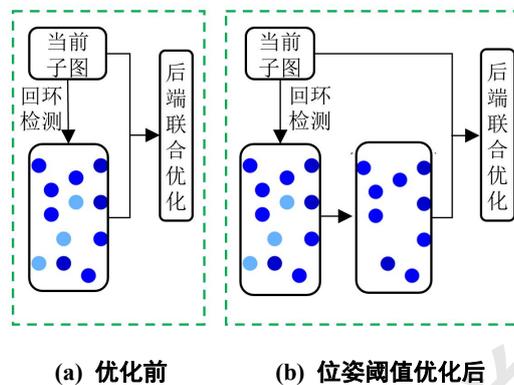


图 6 位姿阈值优化算法示意图

位姿阈值优化算法流程如图 7 所示，首先获取由激光雷达扫描得到的点云数据，通过经典 ICP 配准算法得到当前帧的位姿，同时将当前帧的位姿和上一帧的位姿差作为当前帧的约束节点，所谓约束节点就是相邻两帧的变化量，以便于做点云的筛选，然后设置位姿阈值 w_y 的初始值；基于得到的点云数据，由相邻多帧点云构成局部子图，局部子图用于进一步的点云配准和回环检测；基于满足上述条件的情况下判断当前帧的约束节点是否大于位姿阈值 w_y ，若不大于当前的位姿阈值，则该帧点云位姿不加入优化队列，否则将当前帧加入优化队列，准备和局部子图进行匹配；基于匹配和优化结果得到全局一致性的轨迹和地图。通过位姿阈值的筛选，可以大大降低优化约束节点的数量，提高计算效率。

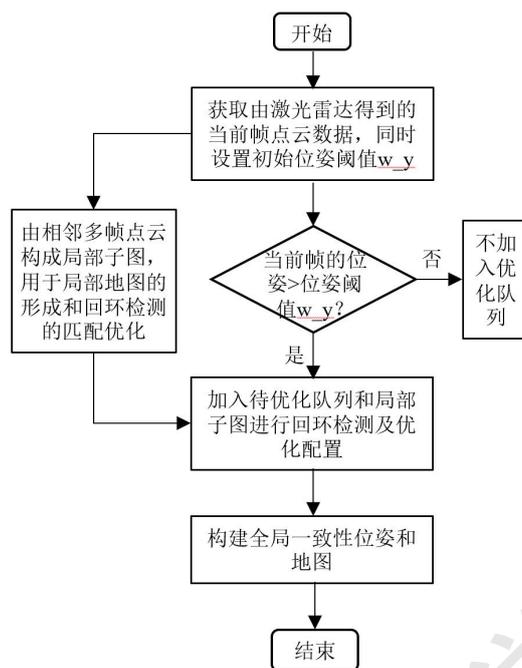


图 7 位姿阈值优化算法流程图

3 实验过程及结果

3.1 实验环境与设备

选取北京科技大学北顺院 2 号楼 311 实验室作为实验环境，实验平台为一台笔记本电脑和一台 Clearpath Jackal 无人车，两台设备均安装 Kinect 版本的 ROS 和 Ubuntu 18.04 操作系统。无人车配置 Velodyne-16 线激光雷达、MPU-9250 惯导测量单元等传感器和索尼蓝牙手柄操作端。手柄通过蓝牙与无人车连接，笔记本电脑 PC 端通过 WIFI 与无人车连接，设置笔记本电脑为主设备。

3.2 实验过程

整个实验分三部分：首先建立笔记本电脑 PC 端与无人车的通信并采集实验室数据，然后采用 Cartographer SLAM 算法处理采集到的实验室数据并构建实验室真实环境地图，最后采用德意志博物馆数据验证位姿阈值优化算法在提升回环检测计算效率、数据实时性等方面的优越性。

3.1.1 PC 端与 Jackal 无人车通信及实验室数据采集

步骤 1：打开无人车电源开关按键，进入无人车的 Ubuntu18.04 操作系统。在操作界面点击右上角“Settings”选项，打开无人车热点。

步骤 2：打开笔记本 PC 端，进入 Ubuntu 操作环境，点击无线连接，查找无人车 Wi-Fi 热点 jackal-nuc，输入密码，建立无人车与笔记本电脑的无线连接。

步骤 3: 在 PC 端打开一个新的终端, 并输入指令“\$ ssh iqr@10.42.0.1”, PC 端与小车开始进行通信。

步骤 4: 在 PC 端打开另一个新的终端, 输入指令“\$ roslaunch iqr_4s_bringup iqr_4s_bring.launch”, 打开无人车内置处理器。

步骤 5: 建立手柄与无人车的蓝牙通讯, 并控制小车运动到合适位置。打开终端, 输入指令“\$ rosbag record -a (记录所有发布的话题信息)”, 开始数据集 bag 包的录制, 采集北京科技大学北顺院 1 号楼 311 实验室数据集。该数据集包含激光雷达和 IMU 的数据信息以及相关 TF 树发布的信息。数据集录制结束后, 按 Ctrl+C 即可退出, 完成录制。将该数据集重命名为 1026.3.bag 新打开三个终端, 分别输入以下指令:

```
$ rosbag info 1026.3.bag
```

```
$ rosbag show sensor_msgs/LaserScan
```

```
$ rosbag show sensor_msgs/Imu
```

查看 bag 包采集的话题信息, 激光雷达和 IMU 所测数据。

3.1.2 Cartographer SLAM 算法构建实验室真实地图

步骤 1: 创建工作空间。

在根目录下打开终端, 输入指令创建工作空间命名 Cartographer2_ws, 创建 src 文件夹, 进入 src, 初始化工作空间。具体指令如下:

```
$ mkdir cartographer2_ws
```

```
$ cd cartographer2_ws
```

```
$ mkdir src
```

```
$ cd src
```

```
$ catkin_init_workspace
```

步骤 2: 编译工作空间。

关闭原有终端, 新开启一个终端, 进入 Cartographer2_ws 进行编译。具体指令如下:

```
$ cd ~/cartographer2_ws/
```

```
$ catkin_make
```

步骤 3: 安装工具库。

在国内的 Gitee 仓库将 Cartographer, Cartographer_ros、Ceres-solver 及 Protobuf 等工具库下载到步骤 1 创建的 src 文件下进行统一编译。具体指令如下:

```
$ cd ~/cartographer2_ws/src
$ git clone https://gitee.com/yltzdhbc/cartographer_ws.git
$ cd ..
$ catkin_make_isolated --install --use-ninja
```

步骤 4: Cartographer 算法构建实验室地图

将 3.2- (1) 步骤 5 所采集的数据 1026.3.bag 从 Jackal 无人车拷贝至 PC 端/Downloads 文件夹下, 利用 Cartographer 算法复现北京科技大学北顺院 1 号楼 311 实验室真实环境地图, 实现 SLAM 功能。新开一个终端, 具体指令如下:

```
$ source devel_isolated/setup.bash
$ roslaunch cartographer_ros demo_backpack_2d.launch
bag_filename:= ${HOME}/Downloads
/1026.3.bag
```

3.1.3 位姿阈值优化算法

采用实验室真实数据和德意志博物馆数据集, 在 Cartographer2_ws/src/cartographer/cartographer/mapping/internal/2d/pose_graph_2d.cc 文件, 在函数 PoseGraph2D::ComputeConstraintsForNode 中修改回环检测判断条件, 增加位姿阈值。查看位姿阈值优化算法对时间效率的提升效果, 验证位姿阈值优化算法的优越性。

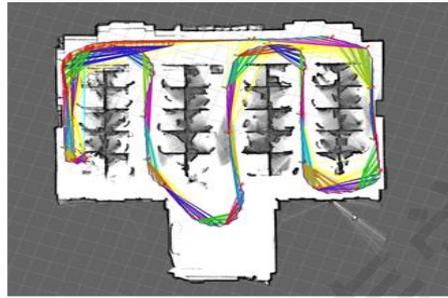
3.2 实验结果

3.2.1 实验室真实场景建图效果

在北顺院 1 号楼 311 实验室, 操纵 Jackal 无人车采集实验室数据集。实验室环境为室内场景, 面积较小, 具有结构化鲜明的特点。真实场景如图 8(a)所示。8(b)是采用 Cartographer SLAM 算法构建的地图。从图 8(b)可以看出, 在短距离、面积较小的场景中, Cartographer 算法利用激光信息和 IMU 信息可以得到良好的轨迹, 且二维栅格地图边界清晰, 畸变和漂移较少。



(a) 实验室真实环境



(b) 实验室真实环境

图8 实验室真实场景建图效果

3.3 位姿阈值优化算法实验结果

3.3.1 位姿阈值优化算法时间效率对比

采用实验室真实数据集，根据无人车所处位置不同，对比不同位姿阈值优化前后的回环检测时间。表1和图9分别在不同位置，位姿阈值分别为无优化、0.5和1时的回环检测所需时间数据。

表1 位姿阈值优化算法时间对比表

优化内容	位置11	位置35	位置36	位置40	位置45	位置46
无优化	51.79	457.83	474.98	571.38	853.44	924.16
位姿阈值 $w_y=0.5$	50.3	219	233.8	295.66	439.29	488.45
位姿阈值 $w_y=1$	49.83	211.22	224.86	278.74	408.83	464.33



优化内容	位置 11	位置 35	位置 36	位置 40	位置 45	位置 46
无优化	51.79	457.83	474.98	571.38	853.44	924.16
位姿阈值 $w_y=0.5$	50.3	219	233.8	295.66	439.29	488.45
位姿阈值 $w_y=1$	49.83	211.22	224.86	278.74	408.83	464.33

图 9 位姿阈值优化算法对比图

从图 9 可以看出，随着检测位置数值越大，回环检测的数据量就越大，因此回环检测算法所用时间就越长。如在位置 40 时，未优化的回环检测算法

耗时为 571.38s；当位姿阈值为 0.5 时，回环检测的时间效率显著提高，耗时为 295.66s；当位姿阈值为 1 时，回环检测的时间效率进一步提升，耗时为 278.74s。因此可以验证位姿阈值优化算法可以提升回环检测的时间效率。从图 9 的对比图中还可以看出，当位置数值较小时，因回环检测的数据量较小，因此位姿阈值优化算法对回环检测的时间效率影响不大，但随着移动距离的逐渐增加，位置数值越来越大，其计算效率会得到显著提高。

3.3.2 位姿阈值优化算法实时性及建图效果对比

为进一步探究基于位姿阈值的 Cartographer 改进算法的优越性，选用德意志博物馆官方数据集构建如图 10 所示的栅格地图。该数据集为室内环境，且环境信息更加丰富、地图面积更大。利用该数据集，可以从时间效率、回环检测数量、实时性等方面验证位姿阈值优化算法的优越性。

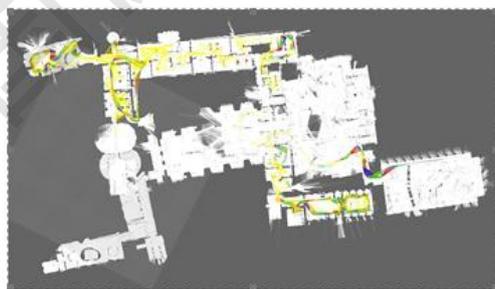
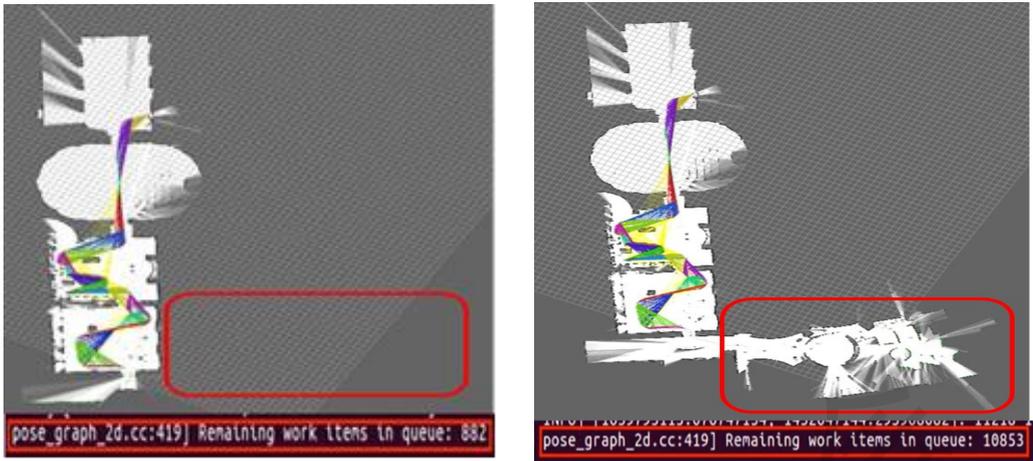


图 10 德意志博物馆栅格地图

操作设备在位置 20 进行回环检测。优化前，回环检测队列的任务数量为 10835，数据量大，回环检测效率低，数据处理有一定滞后，因此设备继续运行一段时间后才开始位置 20 的回环检测，从而导致实际构建的地图比当前回环检测位置的地图多出一部分，如图 11(a) 中红色框区域所示。优化后，位置 20 回环检测的任务数量为 882，较优化前任务数量极大的降低，回环检测计算效率提高，时间缩短，因此位置 20 回环检测时的地图与设备实际建

图一致。通过图 11(a)、 11(b)对比，可以得知位姿阈值可以有效的降低回环检测的任务数量，增强回环检测的实时性，从而提升了建图效果。



(a) 未优化时待优化队列任务数量

(b) 未优化时待优化队列任务数量

图 11 位姿阈值优化算法实时性对比

4 结语

针对 Cartographer SLAM 算法存在耗时多，优化时计算量大、实时性差等问题，设计了一种基于位姿阈值的 Cartographer SLAM 优化算法实验案例。以 Clearpath Jackal 无人车作为底盘，以 Velodyne-16 线激光雷达和 MPU-9250 IMU 作为主要传感器搭建了实验平台硬件系统，依托 ROS 操作系统设计了实验平台软件操作流程。实验结果表明，Cartographer SLAM 算法可构建边界清晰的实验室二维栅格地图；基于位姿阈值优化算法剔除了原有算法回环检测时的冗余信息，降低了回环检测的任务数量，提高了计算效率和数据的实时性。目前该实验平台已应用于本科生实验课程《移动机器人定位与导航》。实践证明，实验有效提升了学生分析问题、解决问题的能力，激发了学生对无人车及人工智能等技术的学习兴趣。该实验平台在自动化专业应用后，学生结合更多专业课理论，拓展了实验内容，丰富了实验手段，完成多项综合性和创新性自主实验，大幅提升了学生工程实践、创新能力。

参考文献：

- [1] 田野, 陈宏魏, 王法胜, 等. 室内移动机器人的 SLAM 算法综述[J]. 计算机科学, 2021, 48(9): 224-234.
- [2] 吴建清, 宋修广. 同步定位与建图技术发展综述[J]. 山东大学学报 (工学版), 2021, 51(05): 16-31.

- [3] 刘涛, 杜宏旺. 基于单目视觉与 IMU 融合的 SLAM 教学实验仪器开发[J]. 实验技术与
管理, 2021, 38(7): 111-119.
- [4] 王鹏. 无人车 SLAM 技术和路径规划算法研究[D]. 哈尔滨: 哈尔滨工业大学, 2021.
- [5] 赵芸, 赵敏. 移动机器人 Cartographer SLAM 建图改进研究[J]. 软件导刊, 2021, 20(4):
69-74.
- [6] 张亮, 刘智宇, 曹晶瑛, 等. 扫地机器人增强位姿融合的 Cartographer 算法及系统实
现[J]. 软件学报, 2020, 31(9): 2679-2690.
- [7] 危双丰, 庞帆, 刘振彬, 等. 基于激光雷达的同时定位与地图构建方法综述[J]. 计算
机应用研究, 2020, 37(02): 327-332.
- [8] Rusinkiewicz S, Levoy M. Efficient variants of the ICP algorithm[C]// Proceedings third
international conference on 3-D digital imaging and modeling. IEEE, 2001: 145-152.
- [9] 佟国峰, 张佳伟, 刘满堂, 等. 基于高效回环检测和重定位的 SLAM 算法[J]. 控制与
决策, 2020, 35(03): 587-592.
- [10] Bula J, Derron M H, Mariethoz G. Dense point cloud acquisition with a low-cost Velodyne
VLP-16[J]. Geosci- entific Instrumentation, Methods and Data Systems, 2020, 9(2):
385-396.
- [11] Bortz, J. E. A New Mathematical Formulation for Strapdown Inertial Navigation[J]. IEEE
Transactions on Aerospace & Electronic Systems, 2007, AES-7(1): 61-66.
- [12] 刘浩. 基于 Cartographer 的校园智能物流机器人设计与实现[D]. 重庆: 重庆大学, 2020.
- [13] Aaron Martinez, Enrique Fernandez, 马丁内斯, 等. ROS 机器人程序设计[M]. 机械工业
出版社, 2014.
- [14] 张华健, 钱钧. 基于 ROS 的远程呈现移动机器人系统设计[J]. 制造业自动化, 2019,
41(09): 111-114.
- [15] 岳伟韬, 苏婧, 谷志珉, 等. 占据栅格地图的最佳栅格大小与地图精度[J]. 机器人,
2020, 42(02): 199-206.