

基于激光雷达的室内机器人的建图方法

李 静, 朱 红

西安电子科技大学机电工程学院, 陕西 西安
Email: 15002207415@163.com

收稿日期: 2021年4月16日; 录用日期: 2021年4月30日; 发布日期: 2021年5月18日

摘 要

近年来, 移动机器人在各个领域的应用日益广泛, 与之相关的技术已在国内外机器人领域掀起一股研究热。对环境的认识和定位从而实现自主导航是移动机器人智能化的重要标志和特征。在完全未知环境下的即时定位与地图构建(SLAM)也一直是移动机器人领域的研究重点。由于激光SLAM具有不易受环境影响、测量距离远且测量精度高、成本低的优点, 因此本文采用改进的Hector_SLAM算法来完成室内环境下地图的创建。本文中系统主要包括由硬件系统和软件系统两部分。首先通过硬件系统对环境进行感知, 然后提取周围环境特征信息, 最后通过软件系统完成二维环境地图的实时创建。在得到环境地图的基础上, 采用改进的卡尔曼滤波法对初始数据进行优化处理, 以获得更为精确的环境地图。实验结果表明, 本文所用系统成本低但性能好, 能较准确的构建二维环境地图, 并能成功使用在小型室内移动机器人的视觉导航中。

关键词

激光雷达, SLAM, ROS, Hector_SLAM

Drawing Method of Indoor Robot Based on Lidar

Jing Li, Hong Zhu

School of Electro-Mechanical Engineering, Xidian University, Xi'an Shaanxi
Email: 15002207415@163.com

Received: Apr. 16th, 2021; accepted: Apr. 30th, 2021; published: May 18th, 2021

Abstract

In recent years, mobile robot has been widely used in various fields, and the related technology has set off a research heat in the field of robot at home and abroad. It is an important sign and feature of intelligent mobile robot to realize autonomous navigation by understanding and lo-

cating the environment. Real time localization and mapping (SLAM) in completely unknown environment has always been the research focus in the field of mobile robot. Because laser slam is not easy to be affected by the environment, the measurement distance is long, the measurement accuracy is high and the cost is low, so the improved Hector is used in this paper SLAM algorithm to complete the indoor environment map creation. The system used in this paper mainly consists of hardware system and software system. Firstly, the environment is perceived by the hardware system, and then the feature information of the surrounding environment is extracted. Finally, the real-time creation of two-dimensional environment map is completed by the software system. Based on the environmental map, the improved Kalman filter is used to optimize the initial data to obtain more accurate environmental map. The experimental results show that the system used in this paper has low cost but good performance, and can accurately construct two-dimensional environment map, and can be successfully used in the vision navigation of small indoor mobile robot.

Keywords

Lidar, SLAM, ROS, Hector_SLAM

Copyright © 2021 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 绪论

在人类科学技术发展史上, 移动机器人的出现无疑是浓墨重彩的一笔。在未知环境中, 移动机器人也可以依靠自身配置实现自主移动, 完成人类既定给它们的任务。移动机器人的研究涉及很多学科, 比如传感器技术、人工智能、控制理论、计算机等, 是目前科学研究最热门的领域之一。移动机器人的应用较之普通机器人更加广泛, 它可以在工业、农业和服务业中一定程度地取代人类, 也可以应用在许多十分危险的环境, 比如救灾抢险、国防军事、空间探测、施工勘探等。许多关键技术都包含在智能移动机器人的研究中, 比如多传感器信息融合、定位和导航、路径规划等, 其中尤以核心技术自主导航最为重要。激光导航具有成本低、精度高、测量范围广等诸多优点, 所以激光导航应用十分广泛[1] [2] [3]。移动机器人自主导航的关键是 SLAM 技术即同时定位与建图。当今学术界的共识是, 只有 SLAM 技术得到充分发展, 智能移动机器人人才会更加全面应用。

移动机器人的同时定位和地图创建技术一直是智能移动机器人领域的重要问题。随着人工智能和计算机技术的发展, 智能自主移动机器人已成为机器人领域的一个重要研究方向和研究热点[4]。对于已知所在环境或者已知机器人所处位置的自主定位和地图创建, 已经有了一些比较成熟和实用的解决方案。然而, 在许多环境中, 机器人的工作环境地图很难甚至是不可能得到, 机器人无法利用全球定位系统进行定位。在这种情况下, 机器人需要在一个完全未知的环境中创建一个地图, 并且它的位置是不确定的。与此同时, 环境地图的建立是移动机器人自主定位和移动首要条件。这就是由 Smith、Self 和 Cheeseman 于 1988 年提出的移动机器人即时定位和地图创建(SLAM)的问题[5] [6] [7]。现在这一问题被认为是实现真正自主的移动机器人的关键一环。

本文共分为五章, 第一章为绪论, 在总结国内外参考文献的基础上提炼出课题的研究背景与意义, 并介绍了本文的研究内容; 第二章为激光雷达及 SLAM 原理简介, 介绍了激光雷达的工作原理和本文采

用的 Hecto_SLAM 建图方法; 第三章为系统设计, 介绍了论文硬件及软件平台的搭建与改进的基于卡尔曼滤波的 Hecto_SLAM 建图算法; 第四章为实验与结论, 介绍了 RPLIDAR 测试过程与改进的 Hecto_SLAM 建图实验与结果分析; 第五章为实验总结与展望, 对本文所做工作进行全面总结, 并对实验未能解决的问题进行展望, 对进一步的研究提供方向。

2. 激光雷达及 SLAM 原理简介

2.1. 激光雷达工作原理

激光雷达是一种通过发射激光束检测目标位置、速度等特征量的雷达系统。它的工作原理是首先将探测信号(激光束)传输到目标, 然后在接收到目标反射的信号(目标回波)后, 将其与发射信号进行比较处理, 最终得到目标的相关信息, 如目标距离、方位、高度、速度、姿态和形状等参数, 从而完成对飞机和导弹等目标的探测、跟踪和识别。因此, 激光雷达是激光技术与现代光电探测技术相结合的成果, 是一种先进的探测手段[8] [9]。

本文所用的 RPLidar A2 激光雷达是上海 SLAMTEC 公司研制的新一代低成本二维激光雷达。它的激光测距频率高达 4000 次每秒, 可以在半径 6 米范围内的二维平面中进行 360°全方位的激光测距扫描, 并能在其所处的环境中生成平面点云的地图信息。在地图测绘、移动机器人导航和物体或环境建模等领域都可以使用这些点云地图信息[10]。

在实际应用中, 激光雷达可分为四种状态[11]。第一种状态称之为空闲状态, 此时激光雷达通电, 当电机开始旋转时, 激光扫描和接收模块不工作; 第二种状态称之为请求状态, 在接收到测距请求后, 激光雷达初始化, 准备开始测距操作, 并在接收停止请求后停止扫描; 第三种状态称之为扫描状态, 当电机旋转时, 目标被连续扫描, 获取数据的其中一种状态; 最后一种状态称之为系统保护状态, 当雷达正常运行时, 系统保护状态不存在, 只有当雷达连接发生错误或内部芯片损坏时, 才会出现系统保护状态。在这种状态下, 激光雷达能保护系统以免再次受到外界破坏。进入此状态后, 只有重新启动才能还原到其他三种状态。因此, 在实际工作中, 激光雷达四种状态之间的转换关系如图 1 所示。

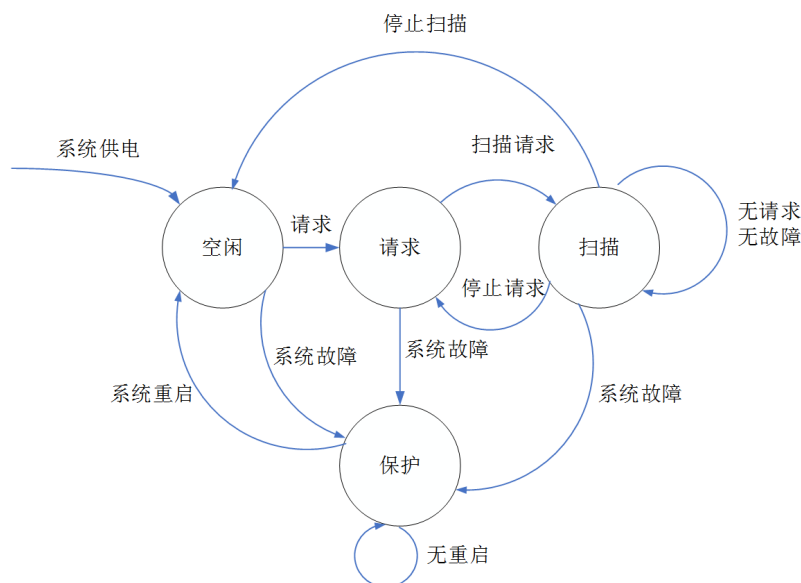


Figure 1. Transition relationship between rplidar working states

图 1. RPLidar 工作状态之间的转换关系

2.2. 激光雷达与外设通信

外接设备主要包括台式机、笔记本以及 Raspberry Pi 等。本文所说的外接设备为笔记本电脑。由于笔记本电脑上没有直接连接到激光雷达所引出引脚的端口, 因此首先需要使用 USB 适配器将接口转换为 Micro USB 端口, 并通过数据线连接到笔记本。通过激光雷达引脚的 TX 端口进行数据传输, RX 端口读取计算机给出的操作, 如启动扫描请求和停止扫描请求等。PWM 的控制信号是 MOTOCTL, 它通过调制脉冲参数, 控制电机的转速。VCC 和 GND 是 5 V 电源接口, 它们提供传输数据和驱动电机的电压。

在外接设备的 TX 到 RPLidar 的数据传输过程中, 两个信号一个接一个地发送到雷达, 第一个信号是测距请求信号, 第二个信号是 PWM 控制信号。激光雷达与外部设备之间的通讯模式如图 2 所示。

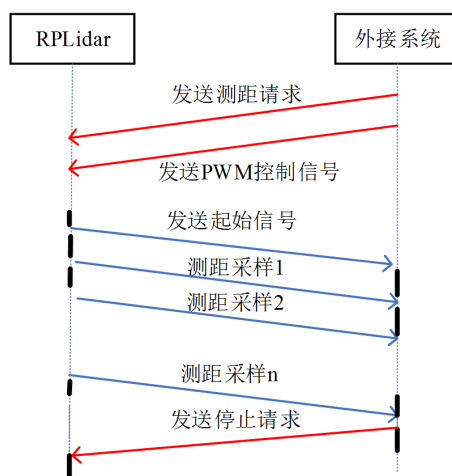


Figure 2. Communication mode between lidar and peripherals

图 2. 激光雷达与外设的通讯方式

2.3. Hector_SLAM 算法

Hector_SLAM 是一种二维的 SLAM 方法。与其他 SLAM 算法相比, Hector_SLAM 的优势是不需要里程计信息, 其姿态估计仅基于输入的激光数据, 并与先验地图匹配。激光雷达的高更新频率和高精度为快速建图和姿态估计提供了强有力的硬件保障。Hector_SLAM 所采用的激光匹配算法是以高斯 - 牛顿 (Gauss-Newton) 迭代公式为基础的。通过寻找移动机器人到先验地图的最佳刚体转换 $\xi = [p_x \ p_y \ \psi]^T$, 使激光数据与地图达到了最优匹配。即求最小的 ξ^* 。

$$\xi^* = \arg_{\xi} \min \sum_{i=1}^n [1 - M(S_i(\xi))]^2 \quad (2-1)$$

式中 $M(S_i(\xi))$ 为地图在 $S_i(\xi)$ 处的值, 如果接着给出一个位姿估计的初始值 ζ , 则 $\Delta\xi$ 的求取就转化为了通过迭代求最优化的问题。

$$\sum_{i=1}^n [1 - M(S_i(\xi + \Delta\xi))]^2 \quad (2-2)$$

用一阶泰勒展开式将 $M(S_i(\xi + \Delta\xi))$ 展开, 并对 $\Delta\xi$ 求偏导数, 得

$$\Delta\xi = H^{-1} \sum_{i=1}^n \left[\nabla M(S_i(\xi)) \frac{\partial S_i(\xi)}{\partial \xi} \right]^T [1 - M(S_i(\xi))] \quad (2-3)$$

式中 H 为黑塞矩阵。

尽管 Hector_SLAM 算法不提供闭环检测, 但它仍然适用于多种场景, 能够构建出完整的环境地图, 并且只需要非常少的计算量, 因此算法的实用性得到了很大的提升。实际生活中, 该算法已经成功地应用于智能移动机器人、交通工具和无人机等领域, 再次验证了该算法的实用性[12]。

3. 系统设计

3.1. ROS 平台

2007 年, 美国斯坦福大学的人工智能实验室与 Willow Garage 公司合作开发了 ROS (Robot Operating System) 系统。ROS 是一个开放源代码, 一个后台操作系统, 或者说可以说是一个辅助机器人性质的操作系统。它不仅能提供类似于操作系统的功能, 包括抽象描述硬件、管理底层驱动程序、执行公共功能、传递程序间消息、管理程序发布包, 它还能提供一些可以被取得的工具和库, 这是构建、编写和运行多处理器集成过程的主要方式[13] [14]。总之, ROS 采用一种名为 Node 的分布式的处理框架, 而这一设计大大提高了机器人开发领域代码的重用性, 因为它在运行时不仅可以单独操作要执行的文件, 还可以对文件进行分层次耦合, 这些过程都可以封装在数据包或者堆栈中, 以便于分发或者共享。除此之外, ROS 还能够实现基于联合系统代码库的方式。在这种方式下, 可将文件系统级别提升至社区级, 然后共同协作以便能够对问题作出独立的决定。以上所有功能都可以通过 ROS 的基本工具来实现。

ROS 系统的主要特点可以概括为: 点对点设计、多语言支持、精简与集成、工具包丰富、免费并且开源[15]。

ROS 的主要功能模块如下: 1) 节点(Node): 一个节点表示 ROS 可执行文件中的一个运行程序, 且节点具有唯一性。例如, 在创建地图的过程中, 节点负责激励激光雷达与执行合成地图的节点一起工作, 并且单个节点提供整个系统。2) 软件包(Package): 是 ROS 中, 一种基本的组织软件形式, 涉及各种各样的文件, 包括配置文件、依赖库、编译工具及其各种形式的文档(例如 Node)。3) 节点管理器(Master): 顾名思义, 节点管理器就是管理所有 Node, 是 ROS 的中心节点。在此基础上, 它还可以完成每个节点的订阅主题通信。节点管理器在程序运行之前将不运行。其他计算的名称注册和查找功也由节点管理器提供。节点管理器是节点间找到彼此、交换信息、调用服务功能的保障。4) 消息(Message): 节点之间需要通过消息进行通信。其中包含由一个节点发送到其他节点的数据信息。除此之外, 消息还具有不同的类型和数据结构。消息充当单个节点的角色, 在各节点之间发布不同类型的订阅数据。5) 主题(Topic): 一种在节点之间相互传输数据的总线。节点首先向相应的主题发送信号来发送不同的消息。标识和区分消息内容就是通过不同名称的主题。如果每个节点在工作中需要某种类型的数据, 那么订阅该主题即可。每个节点可以发布和订阅多个主题, 也即一个主题可以对应于多个节点, 发布者和主题订阅者只有开始通信才知道对方的存在。订阅者只有当发布者在全局工作区发布信息时才会发现和订阅信息, 也就是说, 可以接收消息。6) 服务(Service): 经常在分布式系统中运用于请求和应答交互的方式。作为 ROS 中与上述功能模块不同的其他通信方式, 但发布和订阅模型不适合请求和应答交互。因此, 在一个系统中, 请求和应答是通过定义为一对消息组合结构的服务来实现的: 其中一个用于请求, 另外一个用于应答。所有节点都可以在 ROS 客户端写入代码来与提供节点通信, 并使用它提供的服务。这种交互操作模式类似于对远程助手的调用。7) 发射(Launch): 当机器人运行一个包含多个相互连接且包含多个参数的节点的程序时可以先使用 roslaunch 执行其中一个脚本, 该脚本不仅可以一次启动多个节点, 而且 roslaunch 文件能够被重复使用。

3.2. 软件部分

论文所用移动机器人平台框架如图 3 所示, 采用成本较低的国产 RPLIDAR 激光传感器, 主要通过

调用 RPLIDAR 雷达扫描数据, 处理数据, 完成实时地图的生成, 并采用一种计算难度较小但精度最优的 SLAM 算法即 Hector_SLAM 算法, 该算法将 SLAM 简单划分为计算距离和更新地图两个步骤。第一步是在传感器每一次扫描得到的数据点集中获取直线特征; 第二步是在更新步骤中将扫描线添加到地图中, 即在扫描障碍物时将周围的点绘制成点集, 而不是一个个单独的点。

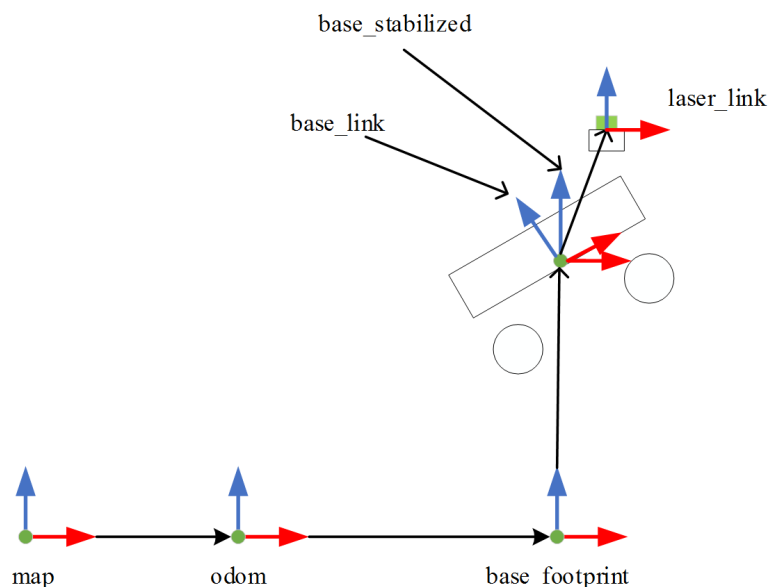


Figure 3. System framework of mobile robot platform

图 3. 移动机器人平台系统框架



Figure 4. Laser processing process

图 4. 激光处理过程

激光处理的过程如图 4 所示, 通常包括: 数据预处理(preprocessing) - 滤波(filtering) - 聚类(clustering) - 拟合(fitting) - 匹配(matching)。激光获得测出的距离后, 进行预处理, 滤掉无关紧要的点, 分割处理区域内的数据, 对同一环境特征区域进行按类分类, 接着拟合参数, 根据环境特征与激光传感器的几何关系, 分析激光传感器的数据规律, 分析特征点质心的位置及其环境特性[5]。最后, 将扫描数据与现有数据相关联, 以确定机器人的相对位置。

本文采用的提取环境特征的方法是 RPLIDAR 扫描数据点的区域分割方法。RPLIDAR 得到扫描数据后, 首先分割扫描数据点集。划分数据主要有两个过程, 一是先把较小的特征分辨出来, 二是从整合的信息中提取出目标特征信息。这样做的目的是使所有子集中都包含特征信息。假设一个扫描周期结束后, RPLIDAR 扫描数据中的离散信息点数为 n , 相邻两点之间的距离为 d , 则根据自适应阈值的规则, 将离散数据点分割划分为彼此相互独立的区域 m 。临界值的选择要具体情况具体分析。当扫描特征点靠近 RPLIDAR 时, 临界值不宜太大, 相反, 当扫描点远离 RPLIDAR 时, 应适当地提高临界值。临界值 D 的设置要根据 RPLIDAR 的最大测量距离和角度分辨率而定, 之后比较临界值 D 和相邻两点间距离 d 之间的大小关系。当 D 小于 d 时, 则把该点作为区域 m 的一个分界点, 将区域 m 划分为两个区域: 区域 m_1 和区域 m_2 , 根据上述方法, 最终将区域 m 划分为 N 个区域即 $m_1, m_2, m_3 \dots m_N$, 这些区域相互独立。在区域分割完成后, 分别统计各个区域内的数据点数量, 当出现区域中仅有小于或者等于

三个数据点的情况时, 就将该区域中的所有数据点均视为噪声, 因此要把该区域中的数据点集中排除, 不参与地图构建[16]。

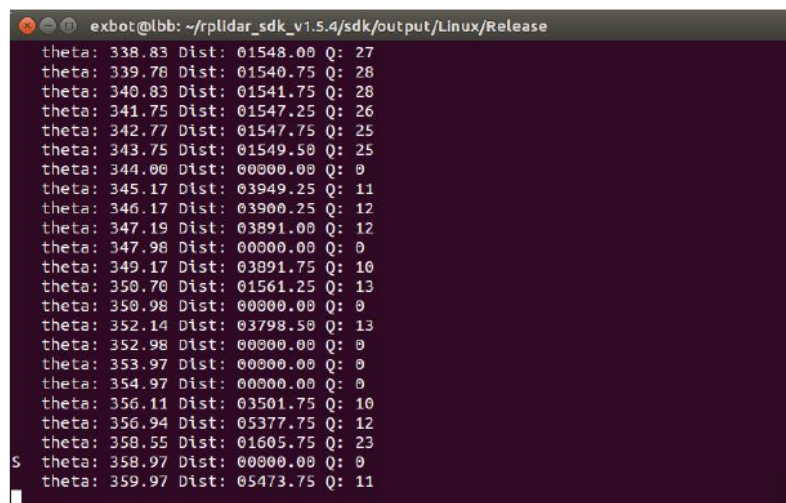
通过仔细分析 RPLIDAR 的工作原理, 就会得出 RPLIDAR 扫描特征点的距离与地图中所反映的点集间连续点间距成正比关系的结论。因此, 划分区域时, 有必要选择一种有根据的方法, 防止把正确的数据错误地当作噪声数据。如果发生这样的误判, 就会给后续点集的特征提取带来误差, 影响合成和定位的准确性。

在扫描的数据点中, 采用聚类方法, 提取直线特征后, 需要拟合多条线段成直线。为了完成最终地图的构建, 本文采用最小二乘法进行拟合。

全局地图的更新需要外部传感器扫描周围环境得到信息, 然后将这些信息与实际特征进行匹配, 最后利用 Kalman 滤波器进行机器人姿势和状态的估计。前面进行完之后才能进入下一个周期, 采集前一阶段的数据, 根据前一周期的位置和方向以及本周期的前段数据, 对数据进行采集和匹配, 并再次使用前一个周期的全局地图和当前周期的数据, 匹配估计当前机器人的姿态, 然后更新全局地图, 随之进入下一个周期。

本文采用的环境表示方法是栅格地图, 推测 RPLIDAR 扫描后占据网格的几率和点集在地图中的位置。因为其所占用网格是割裂的, 所以直接插值和求取梯度(离散度问题)的方法不适用, 因此, 本文采用双精度插值方法, 在连续概率分布下, 把所占栅格地图的值当作采样。

将 RPLIDAR 扫描得到的数据与所获得的地图数据相匹配, 匹配是从前面搜索的位置作为起点进行的, 利用 RPLIDAR 的高频扫描, 会使匹配更加准确。采用高斯牛顿法求解激光扫描匹配问题, 将 RPLIDAR 扫描得到的点集映射到现有的地图上, 非线性方程的二阶泰勒展开可以实现此映射过程, 这种方法虽然比直接快速收敛快, 但初始点要求很高。这张地图采用多分辨率的形式。



```

exbot@lbb: ~/rplidar_sdk_v1.5.4/sdk/output/Linux/Release
theta: 338.83 Dist: 01548.00 Q: 27
theta: 339.78 Dist: 01540.75 Q: 28
theta: 340.83 Dist: 01541.75 Q: 28
theta: 341.75 Dist: 01547.25 Q: 26
theta: 342.77 Dist: 01547.75 Q: 25
theta: 343.75 Dist: 01549.50 Q: 25
theta: 344.00 Dist: 00000.00 Q: 0
theta: 345.17 Dist: 03949.25 Q: 11
theta: 346.17 Dist: 03900.25 Q: 12
theta: 347.19 Dist: 03891.00 Q: 12
theta: 347.98 Dist: 00000.00 Q: 0
theta: 349.17 Dist: 03891.75 Q: 10
theta: 350.70 Dist: 01561.25 Q: 13
theta: 350.98 Dist: 00000.00 Q: 0
theta: 352.14 Dist: 03798.50 Q: 13
theta: 352.98 Dist: 00000.00 Q: 0
theta: 353.97 Dist: 00000.00 Q: 0
theta: 354.97 Dist: 00000.00 Q: 0
theta: 356.11 Dist: 03501.75 Q: 10
theta: 356.94 Dist: 05377.75 Q: 12
theta: 358.55 Dist: 01605.75 Q: 23
s theta: 358.97 Dist: 00000.00 Q: 0
theta: 359.97 Dist: 05473.75 Q: 11

```

Figure 5. Rplidar data
图 5. Rplidar 数据

如图 5 所示, 从激光雷达获得的 PC 打印数据, S 为扫描起始标志位, theta 为扫描点相对于 RPLIDAR 的夹角(角度表示, $[0^{\circ}$ 至 $360^{\circ}]$), Dist 为测距点相对于 RPLIDAR 的距离(mm), Q 为采样点信号的质量。与 RPLIDAR 的通讯需要单个节点完成激光数据采集, 通过运行 Rplidar Node 这个节点, Rplidar 的运行程序就会启动, 等到 Rplidar 进入稳定状态才会开始扫描数据, 扫描到的数据从串口实时输出, ROS 获取该数据后, 再把消息发送到主题/Scan。

4. 实验与结论

4.1. RPLIDAR 测试

二维环境地图的构建所需要软件系统是由 RoboPeak 团队提供的 RPLIDAR 扫描屏幕的可视化软件框架抓取器。目前, 该软件需要在 Windows 平台上运行。

首先, 将 SDK 工具包在 RPLIDAR 网站上进行下载。解压缩后有两个文件夹, 分别是 SDK 和 TOOLS, 其中后者不仅包含雷达串口驱动 CP2102_driver, 也包括 Windows 下的上位机软件。然后通过图 6 所示的接口将 RPLIDAR 连接到 PC。



Figure 6. Rplidar power interface diagram

图 6. Rplidar 电源接口图

解压缩后根据自己版本的 Windows 运行不同的驱动程序, 主要有 X86 和 X64 两种。在前面的步骤中安装串口驱动程序, 然后插入雷达运行 frame_grabber.exe, 并选中 COM4, 如图 7 所示。

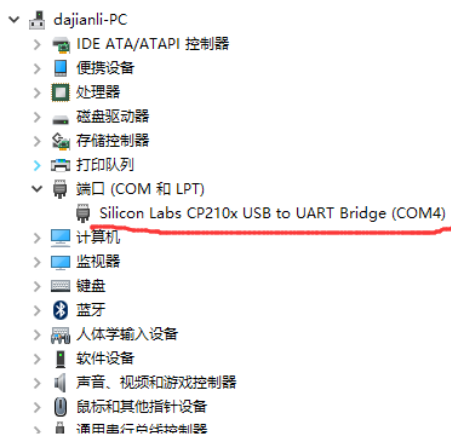


Figure 7. Serial port selection

图 7. 串口选择

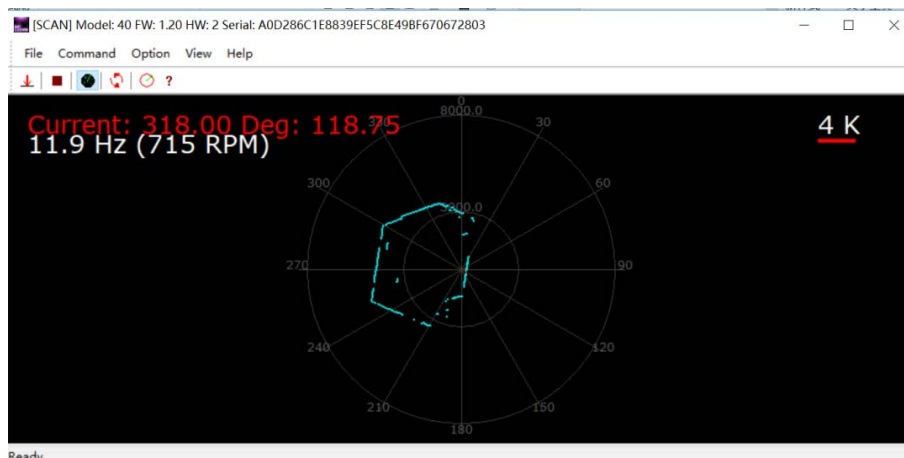


Figure 8. Experimental scan
图 8. 实验扫描图

在实验中, 实验是在 16 号楼 516 实验室, 如图 9 所示。准备好后, 将智能车放在任务开始点并打开开关键。激光雷达通信接口开始通信, 设置扫描频率为 12.0 Hz 然后就可以在 PC 端看到实时扫描画面, 如图 8 所示。通过通信接口输出到主机, 完成进一步的数据处理。这其中的数据主要包括角度和距离。



Figure 9. Real environment diagram of experiment
图 9. 实验真实环境图

4.2. 算法实现

hector_mapping 是 hector_slam 的重要组成部分之一。显然, map 就是在 ROS 的图形化工具 rviz 上的输出地图。hector_mapping 是一种二维的 SLAM 方法, 它相对于其他方法的优点是不管有没有里程计信息都可以使用, 也可以在显示侧倾/俯仰运动的平台上使用, 例如传感器。这种算法的实现得益于现代激

光雷达技术的发展, 因为它进行二维姿态估计只能依靠更新频率很快和精度很高的激光雷达系统。它的缺点是不提供确切的闭环检测, 但对于许多现实世界的场景来说, 它已经足够精密准确。已经成功应用在无人机、无人地面车辆、手持式地图绘制设备和四旋翼无人机的数据记录中[17]。

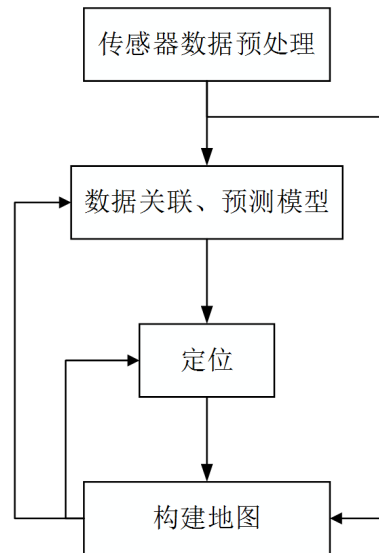


Figure 10. Slam flow chart of mobile robot
图 10. 移动机器人 SLAM 流程图

我们在分析了前文提出的改进算法的基础上, 得到了整体算法的实现流程图如图 10 所示, 本文详细的解释和阐述了 ROS 系统的资源及其算法。作为 SLAM 的结果输出, 本文使用无里程计的 `hector_mapping` 这一 SLAM 方法, 利用 ROS 中的 `hector_mapping` 数据包来完成输出和保存最后的环境地图。系统中使用的高更新频率的激光雷达可以在建图的同时不断进行二维位置估计。最大释然关联法常常应用在地图构建实现算法中, 以栅格地图的形式表示该地图。实验是在宽约两米的实验楼五楼的走廊进行。在笔记本电脑上启动 ROS 系统, 接着运行 `hector_mapping` 软件包启动相应的节点。具体操作步骤如下:

1、硬件设备配置: 用 USB 接口将 RPLIDAR 和计算机连接起来, 使 RPLIDAR 输出的数据能被计算机接受。

2、开启节点管理器: 在使用 ROS 之前先运行程序 `$roscore`, 管理节点信息, 保证节点之间能相互正常的通信。

3、开始构建地图命令: `$roslaunch rplidar_ros view_slam.launch`, 在该命令运行后, 系统可以连续接收从 RPLIDAR 输出的数据, 同时构建地图、更新地图。

本文引入命令集成的形式, 不但可以达到简化启动步骤的目的, 还可以轻松便捷的把需要在终端里输入的命令, 写入同一个启动文件。它可以节省时间, 降低出错率, 同时编写 shell 脚本, 使系统能够在启动时就可以完成快速构建地图。

在 SLAM 过程中, 机器人的位置和姿态以及地图信息共同构成了系统在环境中的状态。一方面 RPLIDAR 扫描得到的是它自身的特征点信息, 而全局环境坐标下的信息才是构建地图所需的, 因此坐标转换是必不可少的。另一方面机器人需要在全球环境坐标系中预估它的位置和姿态, 并通过坐标变换计算出全局环境中特征点的坐标, 从而生成地图, 而机器人在完成自身位置和姿态的计算的同时实现自身在环境中的定位。如图 4~6 所示, 本文是由节点 `link1_broadcaster` 来进行特征点的坐标转换。首先特征点的转换消息要由节点 `link1_broadcaster` 来订阅, 然后该节点通过主题 `/tf` 将转换消息输出出去, 最后

hector_height_mapping 节点只需订阅/tf 节点, 就能够得到特征点的转换信息。

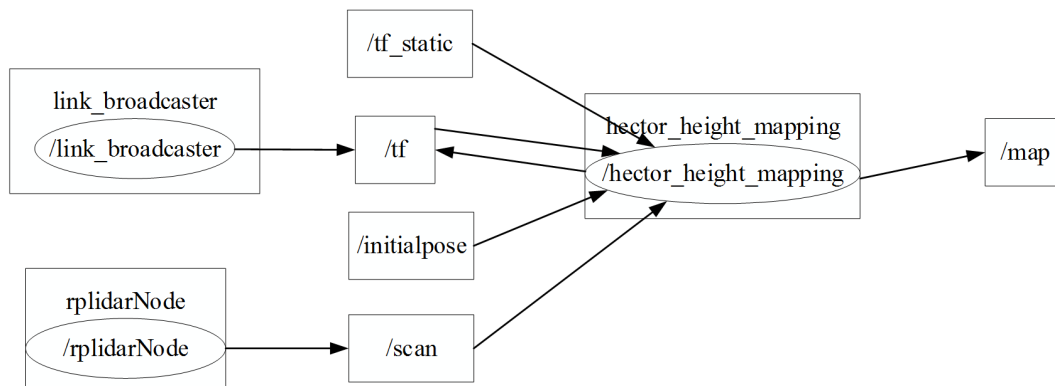


Figure 11. Slam communication network in ROS

图 11. ROS 中的 SLAM 通信网络

从图 11 中我们可以看出输入有四部分, 它们分别为坐标系转换信息 tf, 静态坐标信息 tf_static, 激光雷达数据 scan 和机器人在地图上的初始位置 initialpose; 其中坐标系 tf 和地图 map 作为输出, 进行数据采集的是节点 rplidar Node, 也可用于环境感知, hector_height_mapping 节点完成处理传感器扫描到的数据并进行建图的功能。它们在主题模式下互相交流。最初, 机器人在环境中的姿态是不确定的, 在这种情况下, RPLIDAR 生成的环境地图没有特征信息, 只能利用部分不完整的观测信息来估计机器人在环境中的当前位置和姿态, 然后当机器人继续获取特征观测信息时, 根据环境中确定的位置和姿态以及相对于机器人的位置信息, 在地图中添加特征信息, 当移动机器人第二次观察到同一个特征点的时候, 就可以获得特征点在局部环境中的实时位置信息, 把坐标变换成局部环境坐标的位姿, 利用之前机器人坐标系中的观测值来估计当前机器人的位姿, 这就实现了环境中的定位。以上过程就是 SLAM 过程, 将观测到的特征信息转化为地图, 从而实现了地图的构建。同时, 机器不断更新位置和姿态, 实现了同步定位。

4.3. 实验结果与分析

通过上述步骤, 在走廊实验环境中, 机器人往返一圈建立了以下地图。图 12 显示的地图是直接使用未经处理的数据构建的, 从图中可以看出, 环境特征地图可以大略构建出来, 但也可以很明显看出实验过程中噪音对建图影响较大。

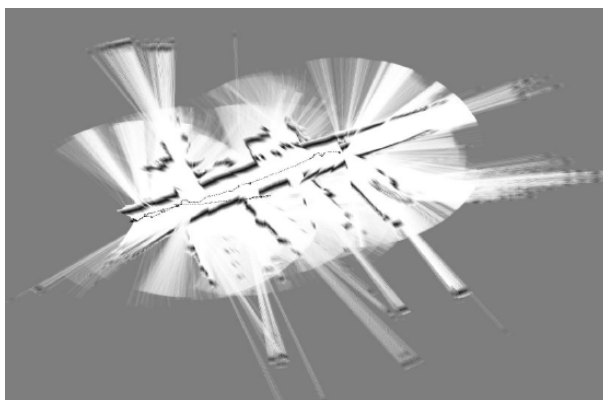


Figure 12. Direct composition of laser raw data

图 12. 激光原始数据直接构图

图 13 是在 ROS 系统中使用改进的 SLAM 算法构造的地图。结果表明, 改进后的总体效果明显优于原始数据的直接构图。

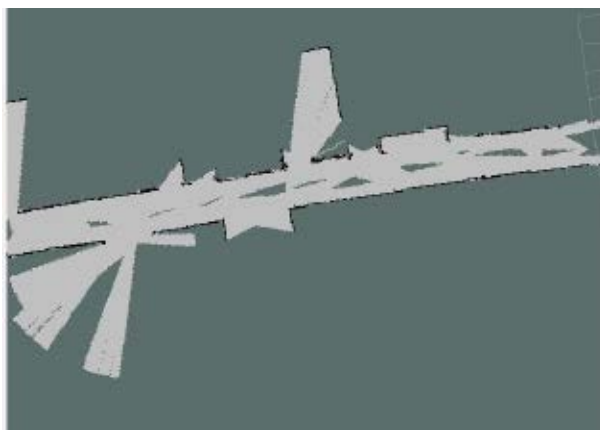


Figure 13. Composition after improved algorithm
图 13. 算法改进后的构图

5. 总结

工业机器人越来越成为衡量一个国家科技创新发展和制造业发展的主要标志, 逐渐受到全世界各个国家的重视。发达国家把机器人提升到国家战略的地位, 促进了机器人产业的发展。我国“十三五”计划也以机器人为重点发展对象, 机器人慢慢得到了国家的有力支持。定位与导航是移动机器人自主行走的其中一个重要方向。目前常用的两种与定位相关的硬件是激光雷达和 RGBD 摄像头, 激光雷达通过光束可以获取 360°的环境距离数据, RGBD 摄像头可以获取图像的深度数据, 从而得到距目标的距离, 同时, 图像可以在帧间匹配, 计算出物体运动的距离。

SLAM 技术是研究机器人自主运动的基础, 而基于激光雷达的 SLAM 技术应用范围很广泛。本文在研究了 SLAM 技术的原理、框架和模型的基础上, 以机器人操作系统 ROS 为软件开发平台, 结合其他机器人研究人员的研究成果, 利用机器人 SLAM 平台进行了实验研究, 得到满意的结果。

1) 本文在查阅国内外大量文献和相关网站的基础上, 总结了本课题的研究背景和意义, 并介绍了国内外移动机器人和 SLAM 技术的研究现状, 深入分析了移动机器人自主定位导航的 SLAM 技术。

2) 简要介绍了 Hector_SLAM 算法、激光雷达测距原理、EKF 滤波算法和 ROS 平台, 建立了移动机器人构图系统的框架, 并设计了测试环境, 测试了该移动机器人系统。

3) 最后, 以机器人操作系统为软件开发平台, 以低成本激光雷达为媒介, 开发了机器人 SLAM 平台。实验结果表明, 该算法可以在低精度机器人底盘、低成本激光雷达上, 实现良好的建图效果。

虽然本文对 SLAM 技术及机器人平台做了一定的研究, 但仍然存在许多的不足, 在后期工作中, 还要进一步对这些不足进行改善。

参考文献

- [1] 吴明轩, 神文文, 姜忠民, 等. 室内自主移动机器人系统设计[J]. 机械科学与技术, 2013, 32(6): 44-48.
- [2] 毛彬彬. 基于激光雷达的移动机器人同时定位与地图创建的研究[J]. 机械设计与制造, 2009(2): 197-199.
- [3] 高雅, 李晓娟, 关永, 等. 运用定理证明器 ACL2 验证机器人操作系统 ROS 节点间通信[J]. 小型微型计算机系统, 2014, 35(9): 2126-2130.
- [4] Kamarudin, K., Mamduh, S.M., Shakaff, A.Y., et al. (2014) Performance Analysis of the Microsoft Kinect Sensor for

-
- 2D Simultaneous Localization and Mapping (SLAM) Techniques. *Sensors*, **14**, 23365-23387. <https://doi.org/10.3390/s141223365>
- [5] Kamarudin, K., Mamduh, S.M., Shakaff, A.Y.M., *et al.* (2013) Method to Convert Kinect's 3D Depth Data to a 2D Map for Indoor SLAM. *IEEE International Colloquium on Signal Processing and Its Applications*, Kuala Lumpur, 8-10 March 2013, 247-251. <https://doi.org/10.1109/CSPA.2013.6530050>
- [6] 张晏龙. 室内定位关键技术研究[D]: [博士学位论文]. 合肥: 中国科学技术大学, 2014.
- [7] Bradford, W. and Parkinson, J.J.S. (1966) *Global Positioning System: Theory and Applications*. American Institute of Aeronautics and Astronautics, Reston.
- [8] Pahlavan, K., *et al.* (2002) Indoor Geolocation Science and Technology. *Communications Magazine, IEEE*, **40**, 112-118. <https://doi.org/10.1109/35.983917>
- [9] 苗斌. GPS 辅助定位技术的现状与比较[J]. 企业导报, 2009(4): 190-191.
- [10] 林传分, 覃锡忠, 贾振红. 基于 EEMD 的全质心 UWB 室内定位算法[J]. 激光杂志, 2014, 35(5): 34-38.
- [11] Werb, J. and Lanzl, C. (1998) Designing a Position System for Finding Things and People Indoors. *Spectrum IEEE*, **35**, 71-78. <https://doi.org/10.1109/6.715187>
- [12] Want, R., Hopper, A., Falcao, V., *et al.* (1992) The Active Badge Location System. *ACM Transactions Information Systems*, **10**, 91-102. <https://doi.org/10.1145/128756.128759>
- [13] Gezici, S., Zhi, T., Ginanakis, G.B., *et al.* (2005) Localization via Ultra-Wideband Radios: A Look at Positioning Aspects for Future Sensor Networks. *Signal Processing Magazine, IEEE*, **22**, 70-84. <https://doi.org/10.1109/MSP.2005.1458289>
- [14] Casas, R., Cuartielles, D., Marco, A., *et al.* (2007) Hidden Issues in Deploying an Indoor Location System. *Pervasive Computing, IEEE*, **6**, 62-69. <https://doi.org/10.1109/MPRV.2007.33>
- [15] Dissanayake, M.W.M.G., Newman, P., Clark, S., *et al.* (2001) A Solution to the Simultaneous Localization and Map Building (SLAM) Problem. *IEEE Transactions on Robotics and Automation*, **17**, 229-241. <https://doi.org/10.1109/70.938381>
- [16] 汪苑, 林锦国. 几种常用室内定位技术的探讨[J]. 中国仪器仪表, 2011(2): 54-57.
- [17] Kaemarungsi, K. and Krishnamurthy, P. (2004) Modeling of Indoor Positioning Systems Based on Location Fingerprinting. *INFOCOM 2004. Twenty-Third Annual Joint Conference of the IEEE Computer and Communications Societies*, Vol. 2, 1012-1022.