



汕 頭 大 學

SHANTOU UNIVERSITY

硕 士 学 位 论 文

题 目：基于 Rao-Blackwellized 粒子滤波器的同
时构建二维和三维地图的 SLAM 方法

英文题目：A SLAM with Simultaneous Construction of
2D and 3D maps Based on Rao-Blackwellized
Particle Filters

姓 名 姚 利 学 号 11509028

所在学院 工学院 导师姓名 范 衡

专 业 信息与通信工程

入学日期 2015 年 9 月 答辩日期 2018 年 5 月

学位论文原创性声明

本论文是我个人在导师指导下进行的工作研究及取得的研究成果。论文中除了特别加以标注和致谢的地方外，不包含其他人或其他机构已经发表或撰写过的研究成果。对本文的研究做出贡献的个人和集体，均已在论文中以明确方式标明。本人完全意识到本声明的法律责任由本人承担。

作者签名：_____ 日期：_____年____月____日

学位论文使用授权声明

本人授权汕头大学保存本学位论文的电子和纸质文档，允许论文被查阅和借阅；学校可将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或其他复制手段保存和汇编论文；学校可以向国家有关部门或机构送交论文并授权其保存、借阅或上网公布本学位论文的全部或部分内容。对于保密的论文，按照保密的有关规定和程序处理。

作者签名：_____ 导师签名：_____

日期：_____年____月____日 日期：_____年____月____日

摘 要

近年来,随着机器人应用领域不断的深入人类的生活,以及无人驾驶汽车的迅速发展,计算机科学、图像处理技术、人工智能等领域技术的快速更新,多学科交叉的智能机器人技术开始逐步应用在人们的日常生活之中,例如家庭扫地机器人、仓库运输智能移动机器人、外卖机器人等等[1]。智能机器人通过传感器感知现实世界实现可靠定位技术是移动机器人最基本且关键的技术。SLAM 的出现就是为了解决智能移动机器人的这项关键技术,当机器人在未知的环境中,能够依赖携带的传感器获得环境中的特征信息和自身位姿,进行实时的定位和环境建模。比较常用的 SLAM 传感器有激光雷达和相机,两者各有优势,基于激光雷达的测量值准确,定位高效,但无法获取更丰富的三维信息。而相机获取图片计算量比较大,且定位实时性较差,在特征不明显场景下,容易导致定位失败。本课题是一种基于 ROS 操作系统,融合了激光雷达数据和深度相机两种传感器同时构建二维和三维地图的方案,使用激光雷达和蒙特卡罗定位算法来解决地图构建过程中的定位问题,充分利用激光雷达能够做到更精确的定位的优势,结合深度相机获取三维点云辅助构建三维地图。其中激光雷达定位重点研究结合了非线性滤波算法在移动机器人中的地图构建的问题,一种基于 Rao-Blackwellized 粒子滤波的 SLAM 算法,可以提供可靠的定位信息。而深度相机的优势获得环境中更为具体的三维环境信息,配合激光获取的定位信息实时的构建三维八叉树地图。两种地图结合应用于机器人在二维环境中的路径规划以及装配机械臂的机器人在三维空间中机械臂的碰撞检测和路径规划等问题。

关键词: RB 粒子滤波 同时定位与地图构建 数据关联 激光雷达 深度相机

Abstract

In recent years, the rapidly update of computer science, image processing technology, artificial intelligence and other fields of technology, multi-disciplinary intelligent robot technology began to apply in people's daily life, such as home sweeping robots, warehouse transport intelligent mobile robots, takeaway robots and so on. Intelligent robots use sensors to sense the real world to achieve reliable positioning technology is the most basic and key technology of mobile robots. The emergence of SLAM is to solve this key technology of the intelligent mobile robot. When the robot is explored in unknown environment, it relied on the carried sensor to obtain the characteristic information and its position in the environment, real-time positioning and mapping. The more commonly used sensors are laser and cameras in the SLAM field. The measurement based on laser radar is accurate and the positioning is efficient, but it is impossible to obtain more abundant 3D information. On the other hand, the camera acquires a lot of information about environment, but has poor real-time positioning. This project is based on ROS that integrates both lidar and depth camera sensors to construct two-dimensional and three-dimensional maps simultaneously. The combination of laser and Monte-Carlo algorithm to solve the positioning problem, combined with depth camera acquisition of point cloud to build three-dimensional map. Based on the focus of Lidar positioning research, the problem of mapping is nonlinear filtering algorithms, This SLAM algorithm is based on Rao-Blackwellized particle filtering can provide reliable positioning information. The advantage of the depth camera is to obtain more specific three-dimensional environmental information in the environment, and to construct a three-dimensional octree map in real time with the laser-derived positioning information. The combination of the two maps can be applied to the path planning of the robot in a two-dimensional environment, and the collision detection and path planning of the robot arm in the three-dimensional space in the three-dimensional space.

Key words: RBPF, SLAM, data association, laser, depth camera.

目 录

摘 要	1
Abstract	II
目 录	3
第 1 章 绪 论	1
1.1 引言	1
1.2 SLAM 研究的背景	3
1.3 SLAM 技术研究现状	7
1.4 主要研究内容及结构安排	9
第二章 滤波理论在 SLAM 领域的应用	10
2.1 滤波理论在 SLAM 领域研究概述	10
2.2 基于粒子的 SLAM 系统一般模型	12
第三章 基于 ROS 移动平台的搭建	14
3.1 移动平台与传感器的介绍	14
3.1.1 激光雷达	14
3.1.2 kinect 1.0	15
3.2 ROS 的基础以及优点	19
3.2.1 ROS 简介	19
第四章 SLAM 地图创建与系统建模	21
4.1 2D 环境表示的方法	21
4.2 机器人系统的建模	24
4.2.1 运动学构型以及概率运动学	24
4.2.2 里程计运动模型	25
4.2.3 传感器测量模型	27
第五章 RBPF 定位与构图的理论	31
5.1 粒子滤波基本算法	31
5.2 从运动模型中采样	32
5.3 重采样技术	33

5.4 更新观测到的特征估计.....	34
5.5 未知数据关联.....	35
5.6 占用栅格地图构建算法.....	35
5.7 本章小结.....	36
第六章 实验验证与分析.....	40
6.1 实验平台.....	40
6.2 Gmapping 算法定位.....	41
6.3 三维栅格地图的构建.....	43
6.4 不同场景实验结果的对比.....	44
6.5 回环检测.....	52
6.6 小结.....	53
第七章 总结与展望.....	54
致 谢	58
攻读硕士学位期间主要的工作成果.....	59

第 1 章 绪 论

1.1 引言

在科技发展与信息爆炸的时代,人工智能技术的研究成为各国科研机构、各高新科技公司的热门研究课题。机器人是人工智能高度发展和融合的体现,是集计算机科学、控制科学,人工智能科学,导航科学为一体的前沿技术,其背后凝结了众多科学技术发展的成果。移动机器人的分类有多种多样,从工作环境来分,有室内移动机器人和室外移动机器人,对于不同的工作环境,移动机器人需要选择不同的传感器;从移动方式上分类有轮式移动机器人、步行移动机器人、以及爬行机器人等等;按照机器人的作业空间来分,又可以分为陆地机器人,水下机器人,以及无人飞机。但无论何种机器人都需要具备感知、处理、决策、执行等能,而 SLAM 的提出就是为了解决自主移动机器人的全自动判断与决策等行为。

同时定位与地图构建(Simulation Localization and Mapping,SLAM)是实现移动机器人自主移动的一个基础且关键的核心的问题[2][3]。SLAM 问题主要包括两方面:第一,移动机器人在未知的环境中,从未知地点出发,在运动的过程中能够完成对自身位置与姿态的识别,称之为定位。第二,通过不断的探索环境中的特征达到定位自身的位姿并实时的构建环境地图。所以定位与地图构建是自主机器人导航中必须解决的两大关键问题[4]。

移动机器人要实现真正的智能化,大致可以归纳为三个一般性问题[5][6]:“我在那儿?”,“我要去哪儿?”和“我如何到达目的地?”,前两个问题涉及到机器人的定位与地图构建的问题,该问题是可以理解为当机器人进入陌生的空间环境中,需要 SLAM 技术解决机器人与周围环境的相对空间的描述,即机器人的定位问题,并根据定位信息对机器人周围环境建模。第三个问题是移动机器人的路径规划问题,机器人根据之前构建的环境模型,从当前的位置规划出一条合理的路径,使得机器人能够安全的到达目的地,以完成设定的任务,这虽然不是 SLAM 问题,但路径规划是 SLAM 研究的目的。

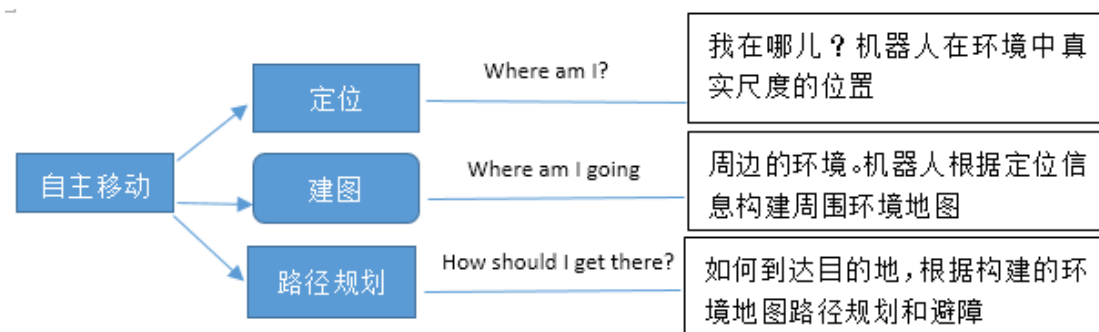


图 1-1 移动机器人导航归纳的三个问题

目前，SLAM 技术主要被运用于智能家居、AR、无人驾驶等领域。智能家居领域扫地机器人是最早用到 SLAM 技术商业应用，这些扫地机是通过 SLAM 算法结合激光雷达或者摄像头的方法，让扫地机器人能够高效绘制室内地图，并根据地图信息智能分析和规划扫地环境。在 AR 产品中基于 SLAM 技术开发的代表性产品有微软的 Hololens[7]，谷歌的 Project Tango[8]，这些 AR 产品使用摄像头实现 VR 头盔能够做到实时的扫描着周围的环境，构建周围环境的三维信息，同时可以完成虚拟信息和真实世界间的无缝连接。当然 SLAM 技的应用中目前比较火热的自动驾驶技术，比如谷歌公司基于激光雷达以及各种传感器融合的的无人驾驶技术，该无人驾驶技术是指利用激光雷达等传感器构建高精度的环境地图。



图 1-2 SLAM 的应用领域

以上应用领域无一不是以 SLAM 技术为核心，与其说 SLAM 是一种算法，不如说是一个综合的概念[9]。因为 SLAM 技术是一门交叉学科研究，一般包括传感器的选择和使用、地图构建、以及回环检测等等方法，虽然 SLAM 技术一直在不断的完善和发展，并逐渐的与目前比较火热的深度学习技术相结合。但仍然有许多细节需要进一步的完善，比如传感器噪声模型的建立，回环检测，场景语义分割等问题。

1.2 SLAM 研究的背景

自从 1980 年代起始，SLAM 概念的提出到现在，一直以来都是机器人领域研究的热点问题。各个领域对移动机器人的需求也在不断的增长，SLAM 技术首先从军事领域发展起来，“玉兔号”月球探测机器人的成功登月完成数据采集任务，军用移动机器人在伊拉克战场上的稳定表现。随着科技的不断进步，民用技术需求的扩大，SLAM 技术也从军用技术逐渐转向民用，服务机器人完成室内清洁任务，防爆机器人完成危险任务等等，这些实例说明了机器人正在逐步的进入更多人们的视野，改善人们的生活质量。

SLAM 技术是集成了信息技术和人工智能的最新理论研究成果，拥有十分广泛的应用领域，十分典型的 SLAM 应用领域就是自动驾驶，其中以谷歌公司和特斯拉公司研发的无人驾驶技术为代表，SLAM 技术是最为关键的实现之一。国内无人车技术也在迅速发展，比如 2017 年 12 月 5 号深圳无人驾驶公交车首次道路运行成功，这充分的说明 SLAM 技术在自动驾驶领域已经取得了阶段性胜利。



图 1-3 深圳无人驾驶公交车

对于 SLAM 技术的研究具有极高的应用价值，同时也推动了模式识别与人工智能、自动导航等众多领域学科的发展，为智能汽车，航空航天，军事。航海等多领域的自主导航与决策提供了理论研究。SLAM 系统使用的传感器从开始阶段的声呐传感器，到目前比较流行的单线和多线激光雷达，再从测距传感器到视觉 SLAM 用到的到单目、双目、RGBD、ToF 相机等各种视觉传感器不断的发展，再到与惯性测量单元 IMU 等惯性导航传感器的融合，SLAM 的算法也在不断的更新迭

代,也从开始的基于滤波器的方法向基于优化的方法转变,技术框架也从开始的单一线程向多线程演进。无论从学术界还是工业界正在不断的刷新 SLAM 技术,目前已经有很多商业产品应用 SLAM 技术走进千家万户,最令人期待便是无人驾驶技术的成熟。

最先发展并成熟应用起来的是基于激光雷达的 SLAM 技术,采用单线或多线的激光雷达,它的优点是能够获取比较精确的角度和测距信息,并且传感器的扫描范围广,主要用到的 SLAM 算法主要是滤波器方法,比如 EKF, PF 等算法。相比于激光雷达,视觉 SLAM 传感器一般使用相机,常用的相机传感器主要有单目相机、双目相机以及 RGBD 相机。相机则更加轻便且便宜,并且图像能够为 SLAM 系统提供更加丰富的环境信息,当然视觉 SLAM 的缺点是图像信息的实时处理需要很高的计算能力。



图 1-4 激光雷达产品



图 1-5 视觉传感器

那么对于以激光雷达为传感器的 SLAM 技术,常常用来建立栅格地图用于机器人的路径规划问题,每个栅格以概率的形式表示被障碍物占据的概率,适用于二维平面上的移动机器人的路径规划。在文章[10]综述中对比了关于激光雷达开源的 SLAM 算法[11-17]。其中 hectorSLAM[11][12]精度要求相对比较高,要求激光雷达具有高更新率和比较小测量噪声,使用高斯牛顿方法对获取扫描点云的匹配算法, Gmapping[13][14]算法是最为广泛基于激光雷达应用的 2D-SLAM 方法,采用 RBPF 粒子滤波算法,需要大量粒子更新迭代来获取更好的结果,推导过程十分复杂,但是能够实现更加稳健的机器人的定位,该方法也是本课题中使用的定位方法。KartoSLAM[15]算法的核心思想是一种基于图优化的方法,这种方法相比较以上的几种方法来说,在大环境下构建地图会更有优势。lagoSLAM[16]算法是利用最小化非线性非凸代价函数解决局部凸近似的问题,以便于用来更新图配置的图优化 SLAM 的方法,而最近更为高效的是 cartographer[17]算法,该算法是 2016 年谷歌开源的算法库改进了 GMapping 计算复杂度以及没有有效处理

闭环问题等缺点。具有低计算资源消耗，该项目的工程依赖项少，可移植到小型设备中实现实时优化顺利运行等优点。

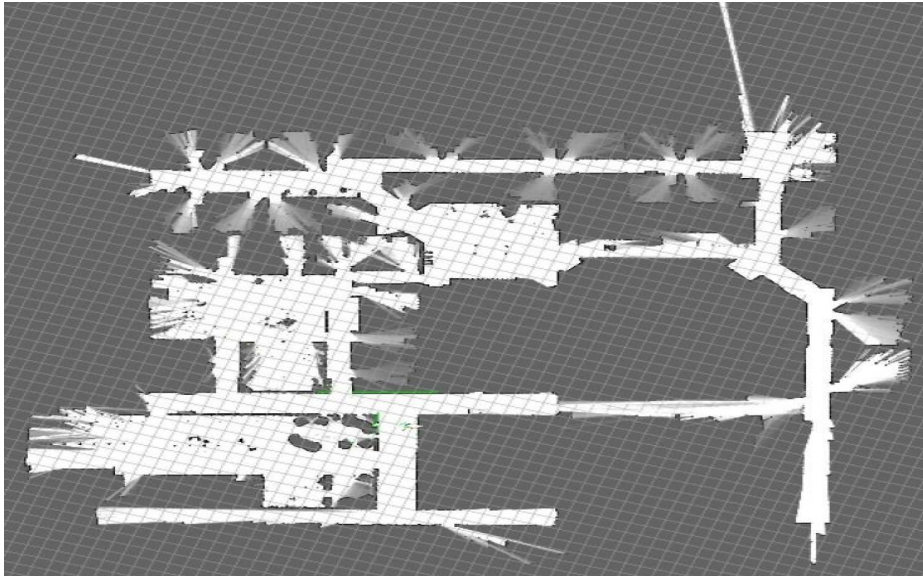


图 1-5 谷歌 Cartographer 激光 SLAM 的地图构建

随着计算机的硬件计算能力的提升，以及 GPU 各种加速计算能力的芯片的发展，基于成熟的视觉 SLAM 框架下结合深度学习近几年成为国内外学术研究者的热点，并且，对于视觉 SLAM 在小型 PC 上实时运行已经成为现实。基于视觉的 SLAM 系统的发展历程中，比较具有代表性的 SLAM 系统有很多：MonoSLAM[19]是由 Davison 等研究者在 2007 年提出的第一个基于单目的视觉 SLAM 系统，被业界认为是众多 SLAM 相关工作的起源。该方法以扩展卡尔曼滤波为后端处理算法，实时追踪前端相对稀疏的特征点，实现构建环境的特征点地图，但是该方法也有很多缺点诸如应用场景很窄小，路标特征点的数量有限，需要克服追踪的稀疏特征点非常容易丢失的情况。

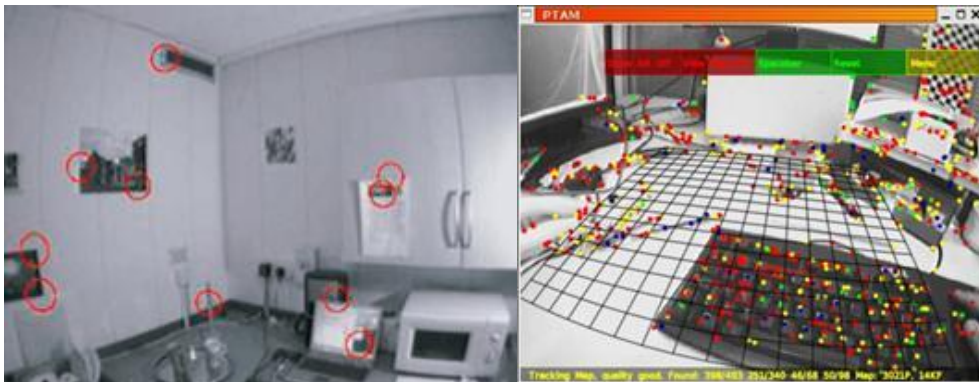


图 1-6 左图是 monoSLAM 第一个单目纯视觉的 SLAM，右图是实时 SLAM 系统 PTAM

2007 年, Klein 等人提出了 PTAM (Parallel Tracking and Mapping) [20], 此方案实现了跟踪与建图过程的并行化处理, 在后端使用非线性优化的方法, 该方法是视觉 SLAM 中首次区分出前后端的概念, 具有十分重要的意义, 为后期的 SLAM 的框架奠定了基础, 但仍然没有解决应用的场景小, 跟踪容易丢失问题。2015 年提出了继承了 PTAM 的更加完善易用的 ORB-SLAM[21] 系统, 该 SLAM 系统支持单目、双目、RGB-D 三种模式, 整个系统具有良好的泛用性, 设计出围绕 ORB 特征的里程计以及回环检测的字典, 具有良好的稳定性, 当然也存在不足之处便是计算特征点是非常耗时的。对于 RGBD 传感器上的 SLAM 方案比较经典的就是 RTAB-MAP (Real Time Appearance-Based Mapping), 该方案给出了一套完整的 RGB-D SLAM 方案, 实现基于特征点的视觉里程计、基于词袋的闭环检测、后端的位姿图优化, 点云以及三角网格地图。

近年来, SLAM 算法引起了越来越多的研究者的关注, 针对移动机器人的 SLAM 问题拥有两个关键特点, 分别是连续性和离散型, 用于估计机器人位姿的定位问题是连续的, 但是在基于特征的表示方式中, 物体可能是地标, 或者由测距传感器探测带的某些部分, 需要顾及各个测量与地图中各地标的一致性, 该推理过程是离散问题。针对此问题研究者们将 EKF 算法用于并解决 SLAM 问题, 可以说最早的且最有影响力的 SLAM 算法便是基于 EKF 算法[22][23], EKF-SLAM 方法中, 构建地图的部分是基于特征路标点组成的。对机器人的运动和预测是采用高斯噪声假设限制。在 SLAM 发展的历史上在 EKF 的基础演生出各种变种 SLAM 的方法, 比如 UKF (Unscented Kalman Filter) [24] 等, 但是 EKF 的弱点是: 首先在计算量方面比较大, 资源消耗相对严重。其次就是在测量数据的融合方面, EKF 算法是基于已知测量值与路标特征点之间对应的关系这个前提条件, 所以对周围环境的特征检测技术要求比较高, 有时甚至采用人工标记的特征点的方法来满足要求。如果 EKF-SLAM 在运行过程中由于机器人携带传感器的测量值与特征路标点之间发生错误关联, 这将会导致算法的发散, 致使系统定位构图失败, 所以该算法对不准确的数据关联问题相对比较敏感, 鲁棒性比较差。

最近这些年 SLAM 的算法逐渐的开始基于滤波器的方法转向基于图优化的方法的方向转变, 比如基于优化的 Graph-SLAM[25] 方法, 该算法是利用图模型对 SLAM 具体问题进行系统建模, 建模过程中的节点对应不同时刻的机器人位姿以

及环境组成系统的状态,其中图中的边表述了系统状态之间的一种约束关系。对比基于滤波的方法和基于图优化的方法,不同之处在于信息的表达之上。比如 EKF-SLAM 是通过协方差矩阵和均值表达信息,而基于图优化的 SLAM 方法通过约束图来表达信息,其中更新 EKF 协方差矩阵需要较大的计算量,而图优化的方法却比较低廉,但是图优化的算法在恢复地图和路径时,要求进行附加的推理计算过程,相比较 EKF 的方法是随时维护最优地图估计和机器人的位姿。

1.3 SLAM 技术研究现状

随着传感器硬件的发展,无论是基于激光雷达和视觉的 SLAM 的研究都有长足的发展,但两者又各有优势,激光雷达测距准确,误差较小,在除了强光照射以外的环境中能够稳定的运行,且激光雷达采集到一系列分散的具有角度和距离信息的点云,处理起来比较简单,且点云是包含了环境中几何关系的,这更加有益于机器人路径规划和导航功能的实现。基于激光雷达的 SLAM 的研究也比较成熟,比如谷歌 Cartographer。近几年 SLAM 的发展更多的在视觉 SLAM 以及激光雷达与视觉结合的 SLAM 的研究,基于视觉的 SLAM 可以从环境中获取更为丰富的环境信息,物体的纹理信息,具有辨别场景的能力。

在更多的学术研究者眼里,未来的 SLAM 发展将会呈现更多的可能性。首先 SLAM 技术结合具体应用,将会向小型化,轻量化的方向发展,算法能够在嵌入式或者手机等小型设备上顺利运行,走进人们的生活中,比较具有代表性的设备如 AR/VR 等,这些都是 SLAM 技术用于教学、娱乐等生活中常见的应用领域,并将在众多场景中为用户带来更加真实的体验。但是目前来说这些小型化的应用设备对运行 SLAM 占用的计算资源比较大,所以对于能够在小型化设备上的运行还需要一段时间的发展。再比如利用高性能的计算设备,实现精密的三维重建,以及场景理解等功能,这是无人驾驶以及使用高精度地图导航的最重要的步骤,这也是目前该领域与深度学习结合最为密切的研究点。当然更重要的方向便是多传感器的融合,无论是一个机器人,一辆无人车还是 AR/VR 设备等硬件设备,都不可能只携带一种传感器就能达到很高的可靠性,大多数都是多种传感器的融合,比如视觉与惯性导航的融合是 SLAM 领域中数据融合研究方向关注的热点[26-28]。由于 IMU 虽然可以测得角速度和加速度,但都存在明显的漂移,相比于 IMU,相机数据基本不会有漂移,所以 IMU 为快速运动提供了较好的解决方式,而相机

能在慢速运动下解决 IMU 的漂移问题。所以两者是互补的。在 SLAM 中传感器融合的商业应用领域，大疆公司推出的集成视觉里程计、避障 RGBD 相机三大功能与一体的 Guidance[29]产品是一个典范，Guidance 便是用双目视觉和超声波的数据融合的科技产品，十分具有代表性。



图 1-7 大疆 Guidance 视觉传感导航系统

语义 SLAM 是 SLAM 与深度学习结合的一个大方向，随着深度学习持续不断的火热发展，越来越多的研究者们试图将机器人环境中的物体的信息结合到 SLAM 中，在文献[30-32]就提出把物体的识别与视觉 SLAM 结合起来，构建带有标签的地图，也称之为高精度地图，高精度地图无论是对移动机器人还是无人驾驶车技术的发展都是十分重要的环节，目前为止都没有很好的解决对无人驾驶或者机器人周围的环境实时的生成高精度地图。让机器人知道周围物体的标签，将会为机器人的回环检测以及位姿的优化等问题提供更多的有利条件。越来越多的研究者们开始研究使用神经网络对图像识别、检测、分割，为语义 SLAM 做出贡献。所以语义 SLAM 将会是结合深度学习很有前景的研究方向。

基于深度相机传感器构建的三维地图不仅能够提供详细的环境描述，还可以实现机器人的定位、制作语义地图等，还可以进行场景分割与识别，以及 SLAM 的回环检测等功能，这是一般激光传感器无法实现的功能。利用深度相机构建三维地图的研究有：

F. Endres[33]等人提出 RGB-D SLAM 算法，采用 g2o[34]进行 3D 图优化，能够实时的构建三维空间环境的八叉树地图，创建的地图可直接用于定位与导航。

Stuckler[35]等人提出一种基于点云匹配的方法,这种方法将深度相机获取的RGB信息和深度信息转换为八叉树模型,再利用特征描述子寻找两个八叉树模型的关联,再进行ICP匹配完成三维点云地图的构建。Felix Endre[36]等人提出了一种构建三维室内地图的方法,SLAM前端完成传感器获取的图像的特征提取和匹配工作,采用经典的ICP算法的估计相机的位姿变换矩阵,SLAM后端是利用g2o进行优化,最后得到三维体素栅格地图,同样可用于三维环境中的机器人的运动规划与碰撞检测。

1.4 主要研究内容及结构安排

主要研究结合激光雷达和深度相机的两种不同优势的传感器融合的基于Rao-Blackwellized粒子滤波SLAM方法,解决移动机器人的定位与构建地图的问题。对移动机器人的SLAM问题中系统的模型进行详细的阐述,并在SLAM方案中选择使用激光雷达作为定位传感器,发挥其定位准确的优势,同时结合深度相机kinect能够获取更加丰富的三维环境同时构建地图,从理论与实践上实现两种传感器的有效融合,并降低系统的计算复杂度。从而实现同时构建二维与三维地图,重现更加真实的场景地图,同时两种地图都是占用栅格地图可用于各种移动机器人以及装配机械臂的机器人的路径规划和三维空间中的障碍检测与规划。二维地图用于平面移动机器人的导航,三维地图用于机器人在空间中的障碍物碰撞检测。本文各章安排如下:

第一章为绪论,对SLAM问题的进行综述以及应用背景,并总结SLAM的发展历史和当前的研究现状和未来SLAM发展的可能性。第二章详细介绍了本文所研究的粒子滤波理论在SLAM应用中的发展和应用,并说明基于粒子滤波SLAM的一般模型。第三章是关于SLAM的研究的一些基本理论知识以及工具的介绍,对ROS系统的基本介绍,简要的介绍了两种传感器rplidar和Kinect的理论基础。第四章介绍SLAM中的系统模型以及地图构建。具体对二维和三维栅格地图算法以及移动机器人在各种模型的介绍。第五章介绍文章的重点Rao-Blackwellized滤波算法的流程以及占用栅格地图的概率表示的方法。第六章具体介绍基于ROS的SLAM算法的实现过程,以及对激光雷达定位参数对构建准确的二维和三维地图的重要性以及实验结果分析。第七章是对整篇文章的总结与展望。

第二章 滤波理论在 SLAM 领域的应用

2.1 滤波理论在 SLAM 领域研究概述

滤波理论的研究自高斯 (K. Gauss) 提出最小二乘估计法开始, 接着 1960 年卡尔曼 (R. E. Kalman) [37] 提出了卡尔曼滤波 (Kalman Filtering) 方法, 标志着滤波理论的建立, 这种基于滤波的概率方法致力于研究机器人的感知和行为的不确定性, 但是当时提出的卡尔曼滤波只适用于线性系统, 之后在经过 Bucy, Sunahara [38] 等人提出了扩展卡尔曼滤波的方法, 用来处理非线性系统, 接下来的时间里, 随着计算机的计算能力的不断增长, 基于粒子滤波的方法逐渐的成为主流的滤波方法, 并在此基础上涌现出很多改进的变种算法, 以适应不同系统, 提高粒子滤波的处理效率。

在滤波理论的 SLAM 研究当中, 主要包括: 基于 EKF-SLAM 的算法、基于粒子滤波的 SLAM 算法, 基于 EKF 滤波的各种变种算法, 比如 UKF (Unscented Kalman Filter) [39], IEKF (Iterated Extended Kalman Filter) [40] [41]。但是由于基于 Kalman 滤波 SLAM 存在一些缺点: 首先基于 EKF 计算复杂度高, 且基于 Kalman 滤波的算法是使用一个协方差矩阵来维持系统状态估计的误差, 这样就会随着地图中的路标数目的增加, 时间复杂度会迅速增大, 这就不可避免的使得计算时间复杂度升为 $O(N^3)$, 其中 N 为地图中环境路标的个数, 导致系统的运算量逐渐的增大, 容易引起定位失败。其次, 基于 Kalman 滤波的 SLAM 算法是对机器人的运动模型和观测模型要求必须基于高斯模型的, 但是在实际环境条件下, 建立合理且满足要求的模型是十分困难的, 这就容易导致算法的不确定性, 降低了系统的可靠性。最后, 基于 EKF 滤波 SLAM 算法的数据关联问题, 该算法要求机器人通过传感器观测到的路标特征和环境中的实际路标之间数据关联性是已知的, 所以当观测值和实际路标之间有错误的数据关联时, 将会引起基于 EKF 滤波的 SLAM 算法发散, 从而使算法失效。

粒子滤波 [42-44] 算法相比卡尔曼滤波能够解决数据关联性, 并具有一定的鲁棒性。所以粒子滤波的方法一直以来都是 SLAM 领域的研究热点, 本文旨在对移动机器人同时定位与地图创建通过粒子滤波算法进行研究, 并且主要针对利用激光雷达的粒子滤波的定位和基于深度相机的构建地图的研究。基于粒子滤波器的 SLAM 方法也是基于 EKF 的方法上衍生出来的, 其中应用最为广泛的粒子滤波

算法为 Rao-Blackwellized 粒子滤波算法，该算法的核心是在 SLAM 的表达形式上因式分解为移动机器人的位姿估计和地图估计两部分，其中移动机器人的位姿状态递归的估计使用粒子滤波方法，环境特征路标估计使用 EKF 方法，每个粒子有多个环境特征，每一个环境特征对应一个 EKF，此方法融合了粒子滤波和 EKF 算法的优点，不仅降低了计算复杂度，还使得系统具有良好的鲁棒性。

在本文的第五章中将重点研究粒子滤波在移动机器人中的应用，使用 Rao-Blackwellized 粒子滤波[45][46]算法结合激光雷达与深度相机两种传感器，同时构建二维栅格地图和三维八叉树地图。流程图如 2-1 所示：

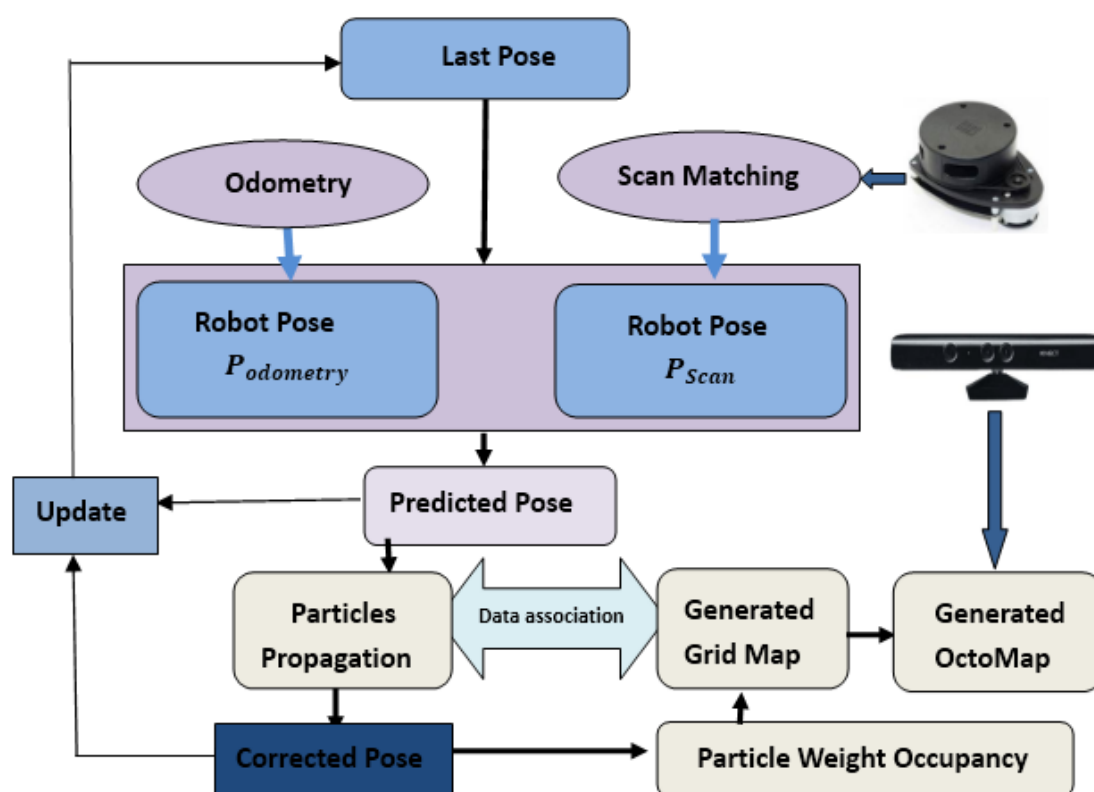


图 2-1 本课题中移动机器人算法的整体框架

该算法的框架定位和二维地图的构建是利用激光雷达，而构建三维地图的部分是采用 kinect1.0，首先通过激光雷达的数据通过点云帧之间的匹配计算传感器估计的机器人的运动轨迹，同时结合里程计信息，这可以统称为粒子滤波的预测部分。融合机器人的里程计信息和传感器匹配算法计算的机器人位姿信息，同时随着时间的推移，激光雷达会更新观测值，对粒子进行迭代更新，同时更新上一时刻的粒子，并由观测值之间的数据关联进行粒子权重的更新，对于机器人位姿的预测以及更新是一个循环的过程，同时根据粒子的重要性权重会将最有可能

的代表机器人位置的粒子留下，而那些权重较低的粒子则被替代，与此同时，根据估计的权重最可能的粒子的位姿，也就是机器人的定位信息。对地图进行更新和计算每个栅格的概率，此时，Kinect1.0 采集到的点云数据也将根据粒子的定位信息，同时进行点云的拼接，构建一个三维的八叉树地图。

2.2 基于粒子的 SLAM 系统一般模型

移动机器人的 SLAM 问题包含对自身位姿信息的确定和外部环境信息的获取和处理，当在位置的环境中移动时，机器人使用自身配备的各种传感器计算自身位置信息以及探测周围位置路标信息，如图 2-2，图中就描述了移动机器人的 SLAM 的系统状态， x_t 表示 t 时刻移动机器人的位姿向量，其中机器人的初始位置为 x_0 ， m_i 表示第 i 个路标的位置， u_t 为机器人从 $t-1$ 时刻到 t 时刻的输入控制向量， z_t 为 t 时刻的观测值，在一系列的输入控制 u_0, u_1, \dots, u_k 的作用下，机器人运动到 x_{k+1} ，在运动过程中，机器人观测其附近的路标，以此类推。

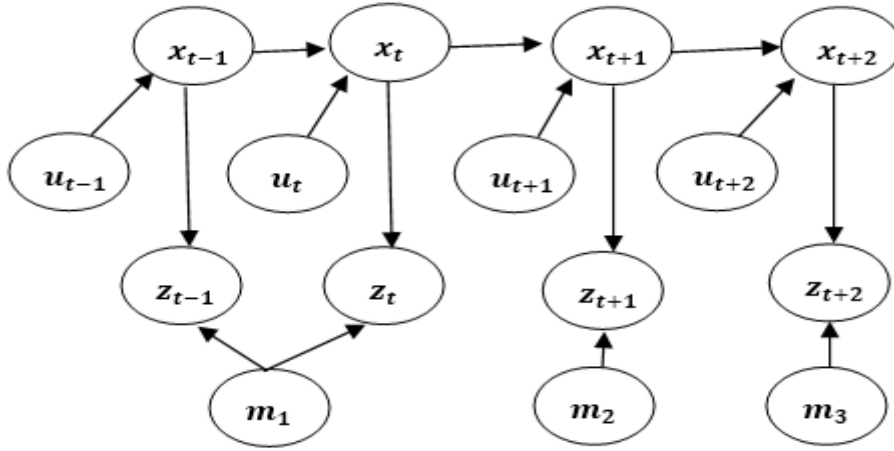


图 2-2 移动机器人 SLAM 系统状态图

把 $k+1$ 时刻移动机器人 SLAM 系统的状态记为 $[x_{k+1}, m]$ ，其中 m 表示地图 $m = [m_1, m_2, \dots, m_i]$ 是环境中的路标特征点的集合，机器人的概率运动模型 $p(x_{k+1} | x_k, u_k)$ 是表示在已知 $k+1$ 时刻机器人控制输入 u_k 和 k 时刻的位姿 x_k 的条件下求 $k+1$ 时刻位姿 x_{k+1} 的条件概率。对于观测模型 $p(z_{k+1} | x_{k+1}, m)$ 在已知 $k+1$ 时刻机器人位姿 x_{k+1} 和路标集合 m 的条件下求 $k+1$ 时刻测量值 z_{k+1} 的条件概率上式中的运动模型通过机器人的运动学和动力学分析而得，一般在采用机器人的

里程计模型作为运动模型, 而对于观测模型可以通过传感器反复测量的数据得到, 在本课题中使用激光雷达的数据作为测量模型。所以 SLAM 问题就转化为求取状态向量 x_{k+1} 的后验概率密度 $p(x_{k+1}, m | z_{1:k+1}, u_{0:k}, x_0)$ 的问题, 即在已知所有输入 u_k 和观测值 z_k 的条件下求取机器人位姿 x_{k+1} 的地图 m 的联合条件概率, 由贝叶斯公式得到:

$$p(x_{k+1}, m | z_{1:k+1}, u_{0:k}, x_0) \propto p(z_{k+1} | x_{k+1}, m, z_{1:k}, u_{0:k}, x_0) * p(x_{k+1}, m | z_{1:k}, u_{0:k}, x_0) \quad (2-1)$$

基于移动机器人运动是马尔可夫过程, 即系统 $k+1$ 时刻的状态完全有 k 时刻的信息确定, 所以 2-1 变为

$$p(x_{k+1}, m | z_{1:k+1}, u_{0:k}, x_0) \propto p(z_{k+1} | x_{k+1}, m) p(x_{k+1}, m | z_{1:k}, u_{0:k}, x_0) \quad (2-2)$$

其中 $p(x_{k+1}, m | z_{1:k}, u_{0:k}, x_0)$ 的计算公式为:

$$\begin{aligned} p(x_{k+1}, m | z_{1:k}, u_{0:k}, x_0) &= \int p(x_{k+1}, m | x_k, z_{1:k}, u_{0:k}, x_0) p(x_k | z_{1:k}, u_{0:k}, x_0) dx_k \\ &= \int p(x_{k+1}, m | x_k, z_{1:k}, u_{0:k}, x_0) p(m | x_k, z_{1:k}, u_{0:k}, x_0) p(x_k | z_{1:k}, u_{0:k}, x_0) dx_k \\ &\xrightarrow{\text{Markov}} \int p(x_{k+1} | x_k, u_k) p(m | x_k, z_{1:k}, u_{0:k}, x_0) p(x_k | z_{1:k}, u_{0:k}, x_0) dx_k \\ &= \int p(x_{k+1} | x_k, u_k) p(x_k, m | z_{1:k}, u_{0:k-1}, x_0) dx_k \end{aligned} \quad (2-3)$$

根据 Markov 假设以及贝叶斯的公式, $k+1$ 时刻位姿向量 x_{k+1} 和地图 m 后验概率分布 $p(x_{k+1}, m | z_{1:k+1}, u_{0:k}, x_0)$ 可以表示为如下的贝叶斯模型:

$$p(x_{k+1}, m | z_{1:k+1}, u_{0:k}, x_0) \propto \underbrace{p(z_{k+1} | x_{k+1}, m)}_{\text{观测模型}} \underbrace{p(x_{k+1} | x_k, u_k)}_{\text{运动模型}} \underbrace{p(x_k, m | z_{1:k}, u_{0:k-1}, x_0)}_{\text{K 时刻的后验分布}}$$

K+1 时刻的后验分布 观测模型 运动模型 K 时刻的后验分布

第三章 基于 ROS 移动平台的搭建

3.1 移动平台与传感器的介绍

3.1.1 激光雷达

本课题使用的是 360 度全角度测量的单线低成本激光雷达 RPLIDAR，实现了 360 度 6m 范围内的全方位测距检测，激光所扫描到点云信息可用于环境轮廓扫描与 3D 重建，地图构建，以及机器人定位与导航等多个应用领域。



图 3-1 激光雷达 RPLIDAR

如图 3-1 所示，展示了激光雷达的工作原理，激光雷达运行时发射器会发送经过调制的红外激光束波，激光束遇到障碍物后将激光反射回来被视觉采集系统所接受，应用三角解算原理实时计算出被照射的物体与雷达中心的距离以及当前的夹角信息。

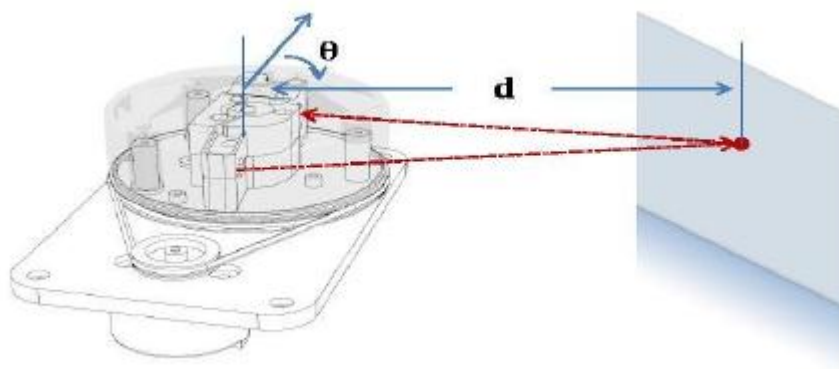


图 3-2 RPLIDAR 的工作原理图

图 3-2 所示了单个激光束照射点测距的过程，其中 θ 是扫描点相对于雷达正前方的夹角， d 是待测被照射物体与雷达中心的距离。以雷达中心为极坐标的

原点，将得到各个扫描点的坐标为： $z_i = (\rho_i, \theta_i) = (d_i, \theta_i), i = 1, \dots, N$ 。其中 N 为激光雷达顺时针扫描 360 度得到的测量数据的个数，该数据个数是随着扫描频率设定的不同而不同。在一般情况下，激光雷达对周围环境进行实时扫描时能够获得比较精确和可靠的测量数据，但是在某些特殊环境下，例如强烈光照的户外，桌椅的腿部，高透过率的玻璃，高反射率的镜子等等物体时，会出现较大的误差，这对机器人的定位将会产生严重的影响。如图 3-3 所示为该激光雷达工作时在可视化工具下显示的点云。

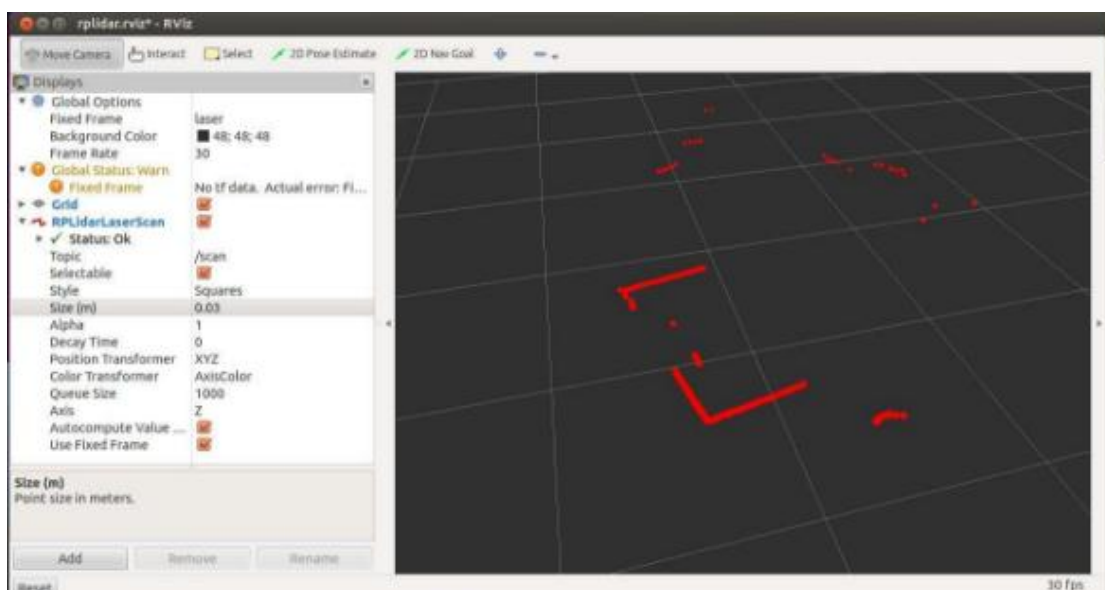


图 3-3 RPlidar 点云数据可视化

3.1.2 kinect 1.0

图3-4给出了Kinect1.0的整体结构,kinect1.0是微软在2010年对XBOX360体感周边外设正式发布的名称，主要是针对家庭娱乐的游戏传感器，人们只需要通过在传感器前展示动作，传感器可以获取人体的骨骼形状，实时的捕获动作，且具有麦克风语音信号输入功能，以达到人机互动的概念[47]。该深度相机的主要结构：RGB摄像头，红外发射器，红外摄像头，下方底座内置马达，可根据实际情况以及具体应用场景调整 Kinect 的俯仰角，两侧分布着阵列式的麦克风，可用于语音控制。对于其中的 RGB 摄像头用于拍摄视角范围内的彩色图像得到就是 RGB 普通的图像，红外线发射器是 Kinect 的光源，称之为主动光，通过发射该光束，主动投射红外光谱，在物体表面产生随机的散斑反射回来，这时红外线摄像接受红外发射器产生的随机散斑，创建相机可视范围内的深度值，该方法可

以获取的深度精度较高，可以精确到厘米以内，测量范围大。



图 3-4 kinect 1.0 实物

目前微软公司已经发布 kinect 2.0 版本，该版本的深度相机相比较 kinect1.0 具备更好的测量精度，不同于 kinect 1.0 的工作方式，kinect 1.0 是基于结构光测量法，通过红外发射器发射一束光线，而红外接收器通过返回的结构光信息，计算物体与传感器之间的距离，能够捕捉物体表面的深度与纹理信息，但是 Kinect 1.0 只适合在室内使用，在室外容易受光照等因素的影响无法工作。而 kinect 2.0 是一种以及飞行时间（Time of Flight, ToF）的测距方法，通过发射连续脉冲信号，接收器根据发送到返回之间的光束飞行时间，计算物体的深度信息，一般可以在室外使用。该类型深度相机价格便宜且稳定性较好，可以实时的捕捉到真实环境的彩色图像和深度图像，且能够计算出彩色图像每个像素点的深度信息，且精度误差在可接受的范围之内，因此对于它广泛的应用在真实环境中的 SLAM 三维重建。

由 Kinect 传感器获得的深度图像的有效距离大约为 0.7m 到 6m，其中在较近或者较远出精度不准确，在 0.7m 到 4.0m 的测量结果有较高的精度。利用 Kinect 获得的 RGB 图像和 depth 图像如图 3-5 所示。

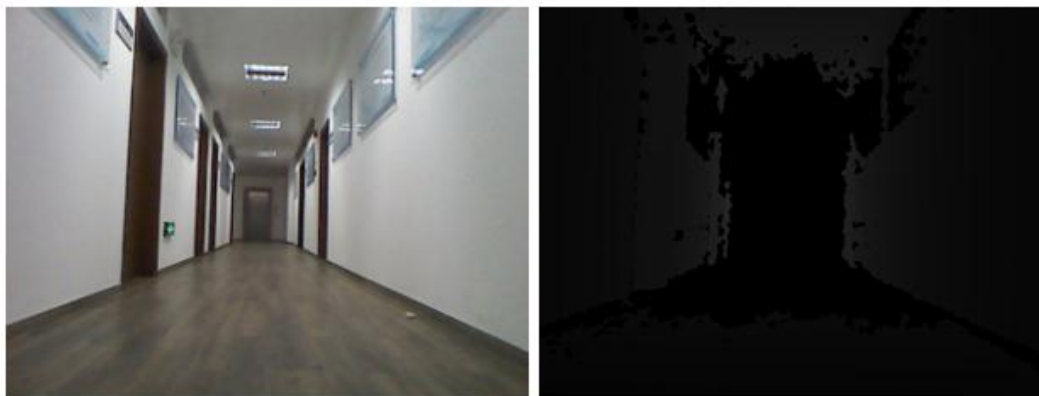


图 3-5 Kinect1.0 获取的彩色图像和深度图像

利用 Kinect 的红外摄像机能够获得周围物体的深度信息，测得深度信息即可以获取某一位置的三维空间下的坐标。具体工作原理如图 3-6，由通常以 Kinect 摄像机的透视中心为三维空间坐标的原点，以红外摄像机指向红外发射器的方向为 x 轴的正方向，按照右手定则的规则确定 y 轴的正方向，那么 z 轴是垂直于图像平面的方向，Z 轴正方向为从图像平面指向环境中的物体。

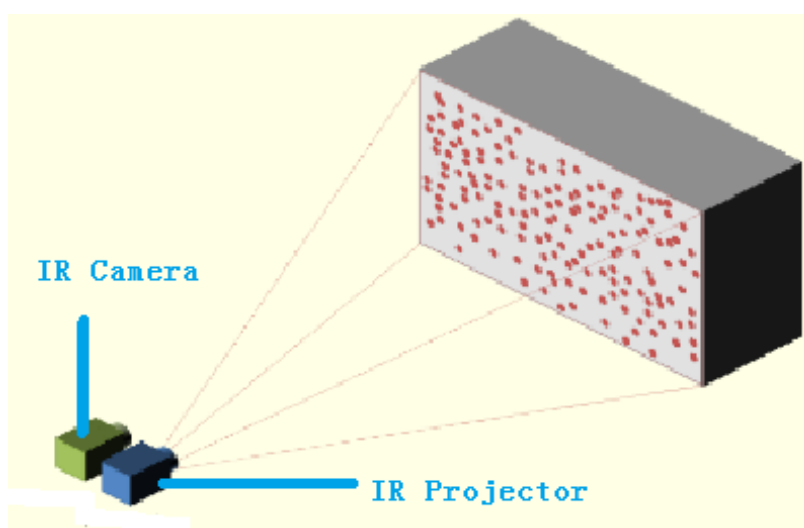


图 3-5 Kinect v1 结构光工作时发射的散斑图案

在图 3-6 为结构光深度相机的测量深度的数学模型，对于三维坐标测量是有两个平面，一个参考平面和目标平面。当物体位置在参考平面时，记录下由红外发射到物体表面的散斑图案，当物体位于目标平面时，散斑图案会在 x 轴上会产生一个视差。那么由三角形相似可得：

$$x \frac{D}{b} = \frac{Z_0 - Z_k}{Z_0} . \quad (3-1)$$

$$\frac{d}{f} = \frac{D}{Z_k} \quad (3-2)$$

在 3-1 和 3-2 中， D 为物体在目标平面的移动的距离， b 为基线的长度，基线是指红外发射与接受传感器之间的距离。其中 f 为红外摄像机的焦距， d 是散斑图案在图像平面的视差，根据上面两式子，消去 D ，可得：

$$Z_k = \frac{Z_0}{1 + \frac{Z_0}{fb}d} \quad (3-3)$$

在 (3-3) 中常量 Z_0 、 f 和 b 可通过校准得到。相机的焦距 f 和环境中某点的深度值决定了该点的成像尺寸。由该点的图像坐标和尺度能够计算出 X、Y 轴的坐标：

$$X_k = -\frac{Z_k}{f}(x_k - x_0 + \delta_x) \quad (3-4)$$

$$Y_k = -\frac{Z_k}{f}(y_k - y_0 + \delta_y) \quad (3-5)$$

其中， x_0 和 y_0 是二维成像平面坐标系的原点坐标，原点的坐标是相机的内参决定的， x_k 和 y_k 是空间中的点在二维成像平面坐标系上投影的坐标， δ_x 和 δ_y 是镜头形变的校正项也是相机内参决定。所以通过式 (3-3) (3-4) (3-5) 可以得到空间中某点的坐标位置 (X_k, Y_k, Z_k) 。

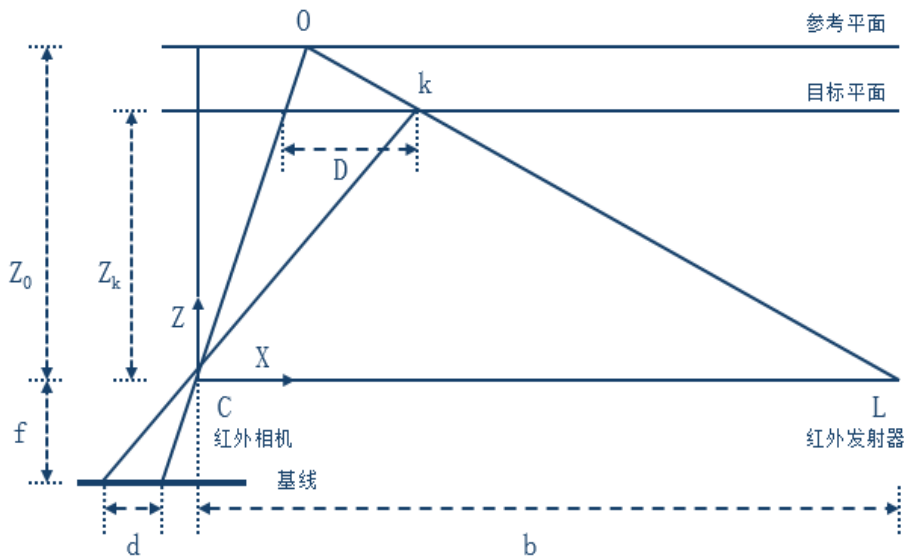


图 3-6 深度测量的数学模型

3.2 ROS 的基础以及优点

3.2.1 ROS 简介

2010 年 Willow Garage 公司发布了开源机器人操作系统 ROS (Robot Operating System) [48], 该系统是一个基于 Linux 的开源系统, 很快就再机器人领域引起了很大的影响。该操作系统的目的在于构建一个能够整合不同研究成果, 实现算法, 代码重用的用于机器人仿真与控制的软件平台 [49]。

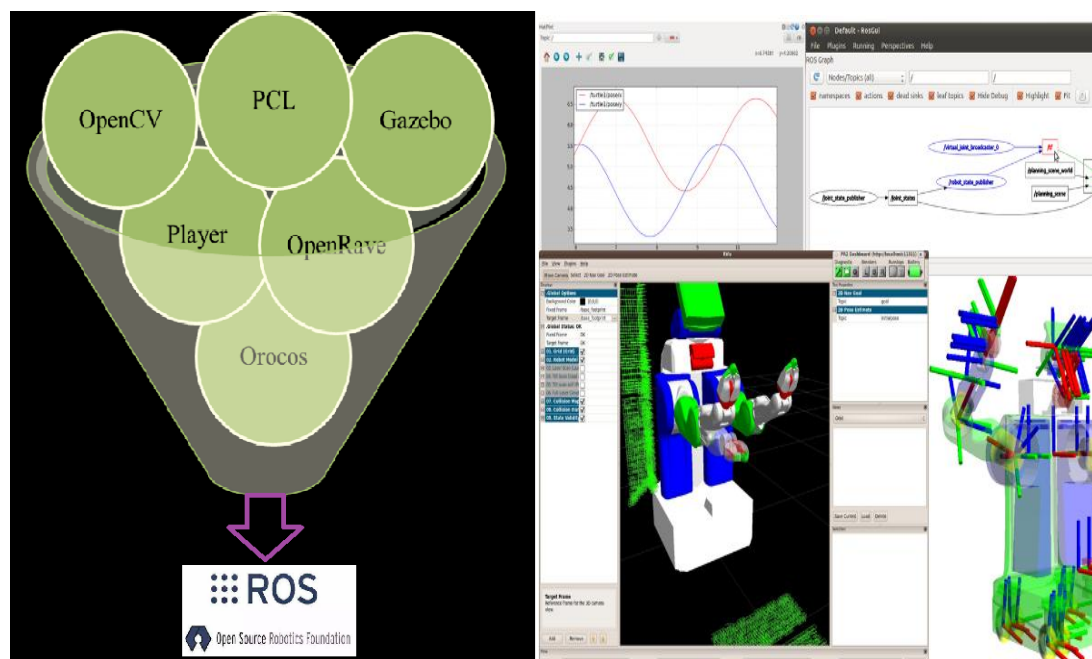
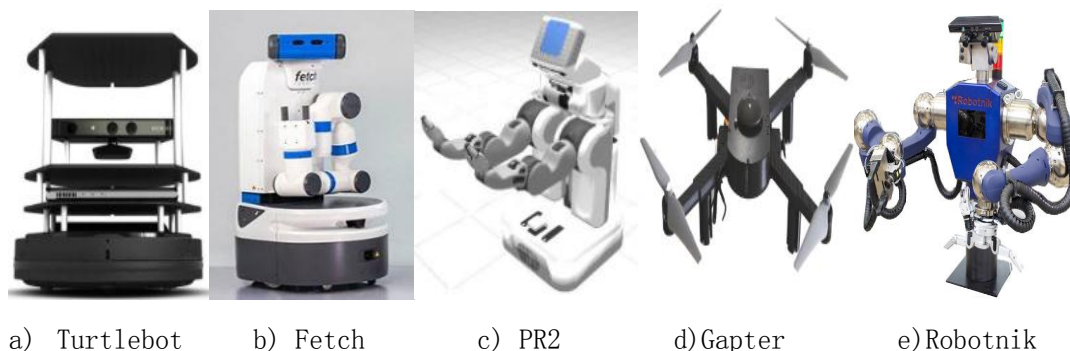


图 3-7 左图 ROS 集成了很多基础库, 右图 ROS 提供的各种工具

ROS 通常采用 Python, C++, Lisp 编程语言, 整合了许多函数库, 包括 OpenCV、PCL、OpenGL 等, ROS 提供了包括硬件抽象描述、底层设备控制、以及一些可视化数据程序和软件库, 让研究者们更专注算法层面的研究, 极大的提高了开发效率。



a) Turtlebot

b) Fetch

c) PR2

d) Gapter

e) Robotnik

图 3-8 支持 ROS 的平台机器人

ROS 的完整版本有很多模块工具，大大提高研究者开发的效率，比如 `rqt_plot` 模块用来绘制实时话题携带数据的曲线图, `RViz` 为 ROS 提供可视化显示的工具, 可以显示机器人模型，文字图标等二次开发。`2D/3D simulators` 模块为机器人提供仿真, `navigation and 2D/3D perception` 模块为机器人提供视觉及导航算法。

ROS 是一种跨平台模块化软件通讯机制, 这些模块是有有软件包(`packages`) 组成, 软件包一般包括依赖项、配置文件和可执行文件等, 是 ROS 操作系统中最重要的组织单元。ROS 中节点是执行计算的过程单元, 并可以接受和发送消息与其他节点通信, 这种通信机制可以把系统模块化, 所以在一个机器人系统中通常是由多个节点组成, 比如本课题中的 SLAM 方法的构图的过程中, 通常一个节点控制激光测距仪, 一个节点获取深度相机的点云数据, 一个节点控制驱动的马达, 一个节点执行定位, 一个节点用于构建地图, 而且这些节点可以同时运行。

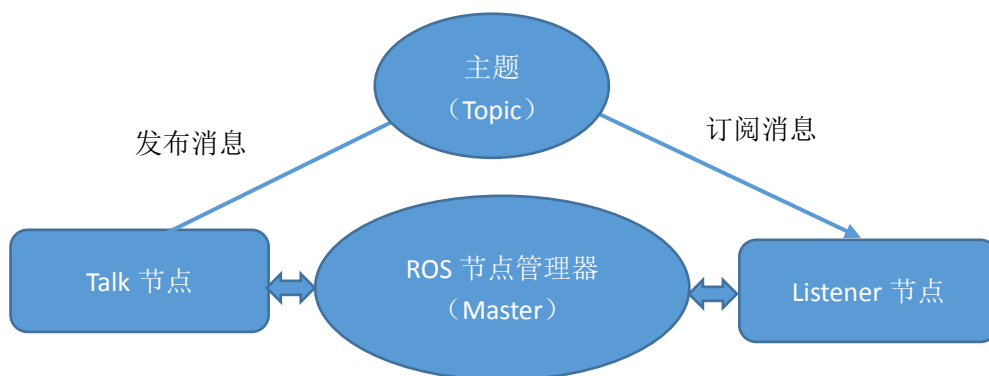


图 3-9 ROS 中节点管理器，节点，主题，消息之间的关系

节点之间的通信有四个模块部分如图 3-9 所示：节点管理器 (Master) 该管理器提供登记节点名称和查找其他节点的功能, 对节点之间的通信提供信息交换和调用服务, 是统筹整个系统的节点的作用。消息 (Message) 是节点之间相互通信的工具, 一般是有类型域构成的简单的数据结构。主题 (Topic) 是消息通过一个带有发布和订阅功能传输系统来传送。服务 (Service) 是请求和回复交互方式经常被用于分布式系统中。

第四章 SLAM 地图创建与系统建模

4.1 2D 环境表示的方法

对于如何描述环境地图是移动机器人地图创建过程中的很重要的环节,一般意义上理解的地图是所有路标点的集合,也就是机器人通过携带的传感器获取环境信息对现实世界进行建模,构建出的地图是环境中综合因素的表示结果。其中比较占有决定性因素的主要是实验环境的规模大小,如果环境范围越大对机器人来说构建地图的计算量和负荷越大,导致获得地图就会越困难。其次是传感器的精度,如果机器人携带的传感器是没有噪声甚至是可以忽略不计的,那么地图的构建将会显得十分简单,但是现实中是传感器没有噪声是不存在的,而且我们知道传感器的噪声越大,构建地图的难度就越大。如果环境中有很多类似的物体将会很容易导致机器人定位失败,比如在在不同的时刻经历相似的地方,这是非常不容易建立地图与环境的联系。如果环境中存在环形回路,那么机器人在这个环形中构建地图将会十分困难,这时就需要闭环检测来提高构图的准确性和效率。

目前,移动机器人的 SLAM 中比较常用的地图可分为栅格地图、特征地图和拓扑地图[50-52]。这些地图从某个角度来说都是具有一定的共性,比如地图要便于计算机的处理,这样才能满足地图用于路径规划等其他用途,机器人可以根据生成的地图完成后续的任务,同时这些地图都可以很方便加入新的路标特征更新地图。当然从不同的角度就会有不同分类的地图,按照地图表示环境的维度来分,3D 地图主要用于无人机,装有机械手的移动机器人,水下工作机器人以及空间机器人,而 2D 地图则主要用于地面机器人,多用于平面上移动的机器人的路径规划方面,如果按照地图的用处来分类,有以下几个作用:

(1) 带有路标特征地图可以用于机器人在现实场景中的定位作用。

(2) 机器人可以利用给定的地图,规划出任意两点间的最优路径,然后控制自己运动到目的地的过程,比较经典且常用的就是栅格地图,这种地图需要包含了地图中哪些地方不可通过,哪些地方可以通过的信息。

(3) 机器人的避障问题,虽然避障和机器人的导航比较类似,但是避障的过程更加注重局部的路径规划,以及对动态的障碍物的处理。

(4) 对三维地图的重建,实现与现实场景更加真实的效果。

以上是构建地图的重要因素以及不同分类地图的总结,其中栅格地图是最为

常用的地图，并在后期的不断发展中在许多机器人系统中得到应用，是使用较为成功的一种方法，该地图经常用于二维平面上移动机器人的导航等任务，栅格地图的工作原理是将整个工作空间分割为一系列的栅格单元，栅格单元的大小是可自行设定，根据测量值对于每一个栅格计算出存在被占用的可能性。每个栅格被赋一个 0 或 1 的值来表示栅格单元的状态：一般情况下，“1”表示被占据，表示被传感器“击中”占用了，若在一定的连续空间都被“击中”，那么该区域可能为障碍物；而“0”表示的是空闲，未被传感器“击中”，该单元可能为自由空间未被占用的状态，所以每个栅格都有一个概率值，来表示栅格被障碍物占据的可能性，用栅格形式表示环境地图，环境空间的分辨率与栅格尺寸大小直接关联，且所以栅格地图经常用于路径规划等问题。

对于特征地图以及拓扑地图，其中特征地图也可以称之为几何表述法，是由机器人采集环境中的几何特征数据，能够提取出点、线、多边形等几何元素表示空间环境中的特征，是一种对环境的稀疏描述，在全局坐标系下以具体的参数来确定物体或者障碍物的位置。而拓扑地图是根据环境结构来定义的一种紧凑的地图表示方法，由位置节点和节点之间的连接线组成，节点通常表示环境中的特定的地点，节点间的连线表示特定地点之间的路径信息。

综合这三类 2D 地图来说，栅格地图对于机器人的定位，避障等方面都具有一定的优势，所以在学术界被广泛使用，而拓扑地图以及特征地图对于机器人定位来说都是比较困难的，机器人必须能够有效的识别节点才能达到定位的要求。所以在本课题中使用栅格地图作为目标，实现二维和三维的栅格地图，给机器人后期的路径规划提供可行性。

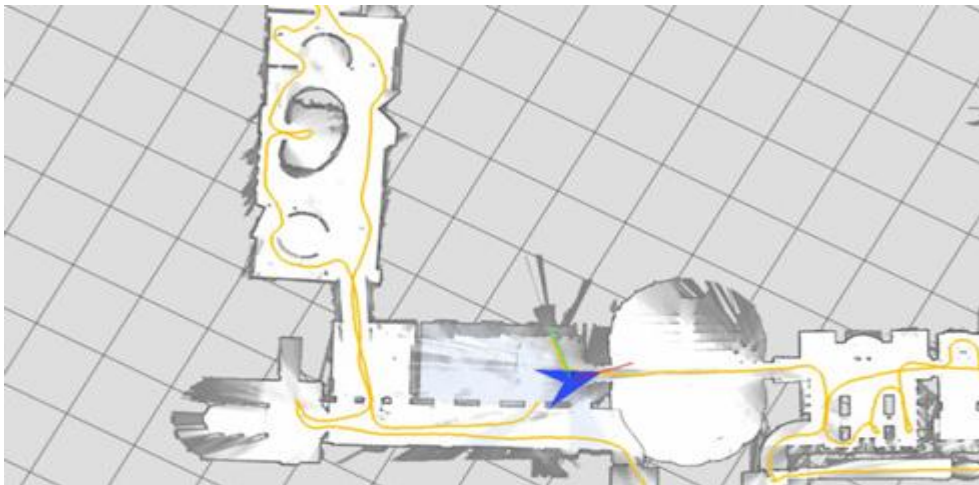


图 4-1 占用栅格地图

4.2 3D 环境的八叉树地图的介绍

八叉树是一种树状数据结构，通常用来表示三维空间信息。在一个八叉树中每个节点可以理解为一个正方体的体积，且每个节点有八个子节点，而所有子节点所表示的体积元素相加在一起是等于父节点的体积的。所以八叉树地图像是很多个小方块组成的如图 4-2-a。与二维占用栅格地图相似，每个方块都表示该空间被占据的可能性大小。因此通过八叉树地图可以查询某个空间内具体方块或点“是否可以通过”，从而实现不同层次的导航。总之，在环境较大时采用较低分辨率，而较精细的导航可采用较高分辨率。

用来表示三维空间的地图，比较经典且常用的便是 OctoMap[53]，那么 OctoMap 是基于八叉树结构的一个开源的 C++代码库，实现三维占据栅格地图的库函数，提供了机器人的数据结构以及多种映射算法，尤其是和机器人在三维空间中的路径规划问题，并且相比较使用点云表示的方式，更加节省空间。一般具有如下的优势：

- (1) 该地图具有 2D 占据栅格地图的性质，能够对三维的位置环境建立适当的环境模型，能够表示空间中该一定区域的是否被占用，能够表示自由空间和占用空间之间的区别，这对于在空间中运行的机器人是必不可少的重要信息，能够实现机器人在环境中的自主探索。
- (2) 具有一般地图的共性，能够接受和加入的新的传感器信息，即路标信息，对地图实现有效的更新，对周围环境的建模和更新是以概率的方式变化，这说明地图可以根据传感器测量的动态环境或者多个机器人的测量值对先前记录的地图有效的更新。
- (3) 该地图具有灵活的表达方式，地图的范围不必事先知道，而是根据地图的增长判断是否需要动态扩展，所以地图是可以多分辨率的，或者可以手动调节分辨率以适应不同的环境大小，同时也可以对局部地图使用精细的分辨率，实现高效的可视化。
- (4) 该种地图存储的方便性，可以在内存和磁盘上都能够有效的存储，并且占用空间比较小，具有压缩点云的特性。如图 4-2-b 是一个八叉树地图表示的长廊实例。

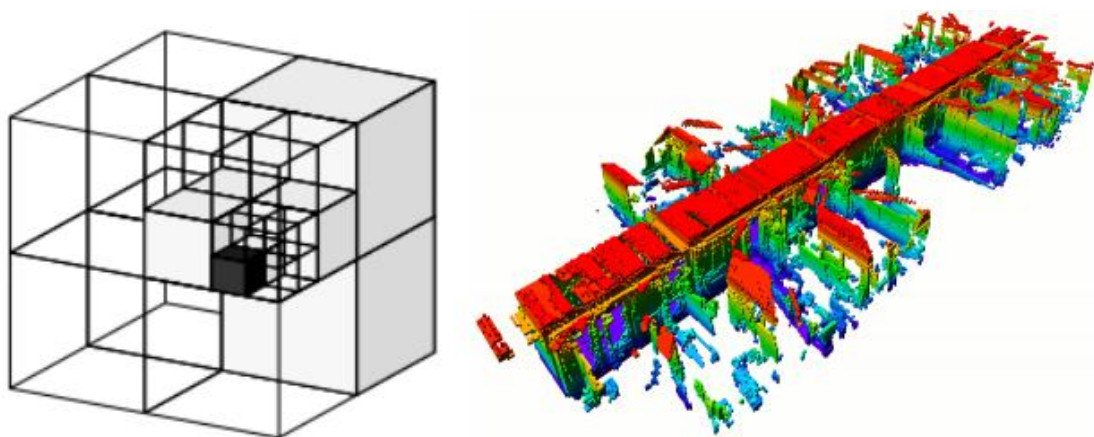


图 4-2 a) 八叉树表示示意图, b) 一个长廊的八叉树地图实例

八叉树地图是三维空间地图中最常用的一种地图, 不仅存储效果好, 且能够根据环境大小设置地图分辨率, 并对新插入数据也能够高效的扩展地图, 在三维空间机器人的导航领域也是一个研究热点[54-56]。

4.2 机器人系统的建模

4.2.1 运动学构型以及概率运动学

运动学可以表述为描述控制行为对机器人构型产生影响的微积分[46]。一个刚性移动机器人的构型一般使用六个变量来描述, 机器人的三维直角坐标和相对外部坐标系的三个欧拉角, 分别是横滚、俯仰、偏航, 由于本课题所研究的机器人是限制在平面环境中运动的, 所以其运动学状态是用三个变量描述, 称之为位姿。如图 4-3 所示的一个全局坐标系统下的机器人的位姿的示意图。一个平面上操作的移动机器人的位姿 (pose), 就是相对外部坐标系的二维平面坐标 (x , y) 和其方位角 (θ), 所以机器人的位姿用此向量描述 $(x, y, \theta)^T$ 。

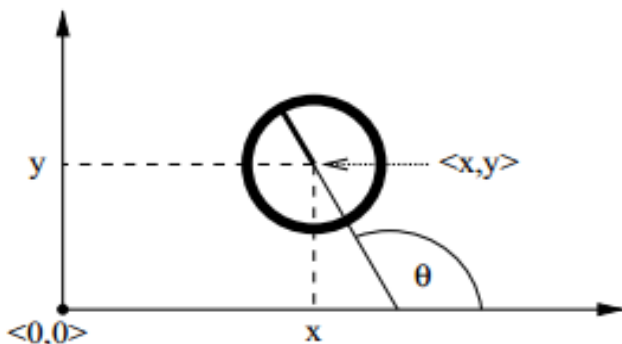


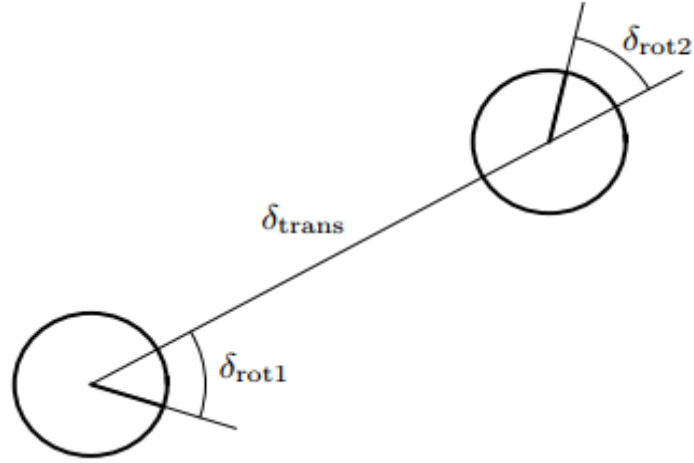
图 4-3 一个全局坐标系统中的机器人的位姿

使用的 Rao-Blackwellized 滤波定位算法时，需要建立合适的概率模型，主要是为机器人驱动和感知过程中存在的特定不确定性建立精确的模型。所以这里有两个重要的高斯模型对于粒子滤波 SLAM 方法十分重要，运动模型和测量模型。运动模型中在移动机器人的系统中起着状态变换模型的作用，运动模型一般有两种运动模型，分别是速度运动模型和里程计运动模型。一般情况下使用里程计运动模型，因为里程计模型往往比速度模型更精确，原因是很多机器人并不能通过计算机器人的轮子的旋转而获得精度等级相同的速度控制，但是里程计信息是在执行运动控制之后才会获得，因此里程计模型不能用于运动规划，而通常用于估计，运动模型由状态转换概率 $p(x_t | u_t, x_{t-1})$ 构成，其中 x_t 和 x_{t-1} 是机器人的位姿， u_t 是运动控制，此模型描述的是对 x_{t-1} 执行运动控制 u_t 后，机器人取得的运动学状态的后验分布。在实际应用中， u_t 一般是由机器人里程计给出，里程计模型使用相对运动信息，此信息由机器人内部里程计测量。

4.2.2 里程计运动模型

里程计通常是整合轮子的编码器信息得到，里程计运动模型就是用距离测量代替控制，虽然里程计会由于机器人的轮子存在漂移或者打滑存在一定的误差，但通常比速度模型更加精确一点，因为速度还要受到实际运动控制器和它粗糙的数学模型之间不匹配的影响，里程计只有在机器人移动之后才是可用的，所以适合用于给定的滤波算法，但是不能用于精确的运动规划和控制。

里程计模型使用的相对运动信息，该信息是由机器人内部里程计测量。具体的如：在时间 $[t-1, t]$ 内机器人从位姿 x_{t-1} 前进到 x_t ，所以运动信息 u_t 的表达式为 $u_t = \begin{pmatrix} x_{t-1} \\ x_t \end{pmatrix}$ ，为了说明相对距离， u_t 被转变为三个步骤的序列：旋转、平移，另一个旋转，如图 4-4，在时间 $[t-1, t]$ 的机器人的运动的里程计的测距模型

图 4-4 在时间 $[t-1, t]$ 的机器人的运动的里程计的测距模型

初始旋转 δ_{rot1} ，平移 δ_{trans} ，第二次旋转 δ_{rot2} ，这三个参数能够说明机器人的两个位置的相对运动，并可以由里程计的相对运动的统计量获得，在概率运动模型中，假定这三个参数被相对独立的噪声污染，根据数学上计算密度的基本算法，给出了如表 4-1 程序伪码：

motion_model_odometry (x_t, u_t, x_{t-1}):

$$\delta_{rot1} = \text{atan2}(\bar{y}' - \bar{y}, \bar{x}' - \bar{x}) - \bar{\theta}$$

$$\delta_{trans} = \sqrt{(\bar{x}' - \bar{x})^2 + (\bar{y}' - \bar{y})^2}$$

$$\delta_{rot2} = \bar{\theta}' - \theta - \delta_{rot1}$$

$$\hat{\delta}_{rot1} = \text{atan2}(y' - y, x' - x) - \theta$$

$$\hat{\delta}_{trans} = \sqrt{(x - x')^2 + (y - y')^2}$$

$$\hat{\delta}_{rot2} = \theta' - \theta - \hat{\delta}_{rot1}$$

$$p_1 = \text{prob}(\delta_{rot1} - \hat{\delta}_{rot1}, \alpha_1 \hat{\delta}_{rot1}^2 + \alpha_2 \hat{\delta}_{rot2}^2)$$

$$p_2 = \text{prob}(\delta_{trans} - \hat{\delta}_{trans}, \alpha_3 \hat{\delta}_{trans}^2 + \alpha_4 \hat{\delta}_{rot1}^2 + \alpha_4 \hat{\delta}_{rot2}^2)$$

$$p_3 = \text{prob}(\delta_{rot2} - \hat{\delta}_{rot2}, \alpha_1 \hat{\delta}_{rot2}^2 + \alpha_2 \hat{\delta}_{trans}^2)$$

Return p_1, p_2, p_3

表 4-1 里程计模型伪码

基于里程计信息计算 $p(x_t | u_t, x_{t-1})$ 的算法，其中控制 $u_t = (\bar{x}_{t-1}, \bar{x}_t)^T$ 给定， $\bar{x}_{t-1} = (\bar{x}, \bar{y}, \bar{\theta})^T$ 和 $\bar{x}_t = (\bar{x}', \bar{y}', \bar{\theta}')^T$ ，所以该算法是根据初始位姿 x_{t-1} ，并由机器人的里程计信息获得一对位姿 $u_t = (\bar{x}_{t-1}, \bar{x}_t)^T$ ，以及一个最终假设的最终位姿 x_t 作为输入，输出的是机器人在该位姿的概率 $p(x_t | u_t, x_{t-1})$ ，其中变量

$\alpha_1 \ \alpha_2 \ \alpha_3 \ \alpha_4$ 是指定机器人运动噪声的机器人的特定参数。

那么对于不同的参数的里程计的运动模型将会产生十分重要的影响,如图 4-5 给出了不同的误差参数 $\alpha_1 \ \alpha_2 \ \alpha_3 \ \alpha_4$ 的值对应的里程计运动模型的例子。a)给出了典型的分布, b)和 c) 分别表示非常大的平移和旋转误差

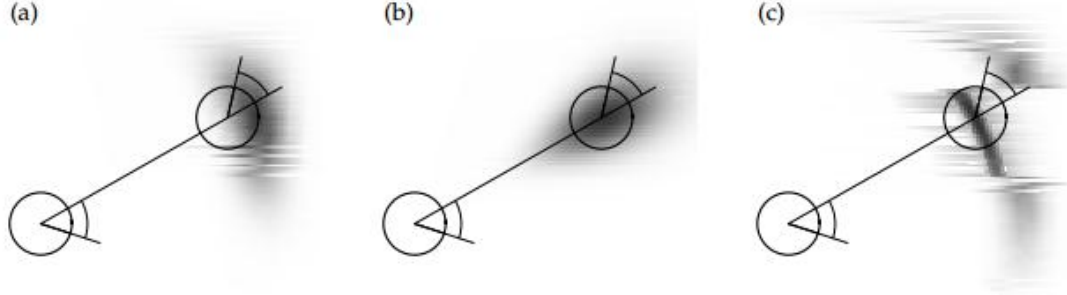


图 4-5 表示设置不同参数的里程计运动模型

4.2.3 传感器测量模型

环境测量模型是概率机器人模型中第二大重要模型,测量模型描述在客观世界生成传感器测量的过程. 模型的特性一般取决于传感器的种类, 比如成像传感器是通过投影几何学建立模型, 声纳传感器是通过描述声波和声波在环境表面上的反射来建立模型。在测量模型上可以定义一个条件概率分布 $p(z_t | x_t, m)$, 式中 z_t 是机器人的位姿, x_t 是机器人在 t 时刻的测量, m 是环境地图。在本课题中, 使用的是激光雷达测距扫描仪, 该扫描仪在之前的章节中已经介绍了是主动的发出一个信号, 并记录其激光回波, 相比较于声纳测距仪具有更加集中的测量束, 在激光工作时, 会产生一系列的测量值, 用 K 表示在一个 z_t 内的测量值的数目, 即有 $z_t = \{z_t^1, z_t^2, \dots, z_t^K\}$, 用 z_t^k 表示某一个独立的测量也就是一个测距值, 所以根据 $p(z_t | x_t, m)$ 为单一测量的可能性相乘得到的概率分解为

$$p(z_t | x_t, m) = \sum_{k=1}^K p(z_t^k | x_t, m) \quad (4-1)$$

从上式可以看出, 每个独立测量束噪声之间的独立性假设, 就像马尔可夫假设一样认为噪声在所有时刻上都是相互独立的。

在基于滤波的概率机器人领域测量模型主要有两种, 分别是测距仪的波束模型和测距仪的似然域模型, 对于测距仪的波束模型一般由四类误差测量引起, 这些误差包括小的测量噪声, 意外对象引起的误差, 以及由于未检测到对象引起的

误差和随机意外噪声。测量模型 $p(z_t|x_t, m)$ 是这四个测量误差的混合。在这里不再过多赘述，本节主要介绍本课题使用的基于测距仪的似然域测量模型。似然域模型的主要思想就是首先将传感器扫描的终点 z_t 映射到地图的全局坐标空间中，这样做就必须知道相对于全局坐标系，机器人的局部坐标系在哪里？以及机器人的传感器的光束 z_t 在何处？

令 $x_t = (x \ y \ z)^T$ 表示机器人在时刻 t 的位姿。在保持全局坐标系的视角下二维地图中，用 $(x_{k,sens} \ y_{k,sens})^T$ 表示与机器人连接的传感器局部坐标系的位置，用 $\theta_{k,sens}$ 表示传感器波束相对于机器人的航向的角度。根据三角变换将测量 z_t^k 的终点映射到全局坐标系下：

$$\begin{pmatrix} x_{z_t^k} \\ y_{z_t^k} \end{pmatrix} = \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x_{k,sens} \\ y_{k,sens} \end{pmatrix} + z_t^k \begin{pmatrix} \cos(\theta + \theta_{k,sens}) \\ \sin(\theta + \theta_{k,sens}) \end{pmatrix} \quad (4-2)$$

这些坐标只有当传感器检测到障碍物时才有意义。所以似然域测量模型简单的将最大距离读数丢弃了，下面说明这种测量模型的噪声和不确定性的来源。

- (1) 测量噪声，有测量过程引起的噪声使用高斯进行建模，在 x - y 空间是寻找环境中最近的障碍物，所以令 $dist$ 表示测量坐标 $(x_{k,sens} \ y_{k,sens})^T$ 与地图 m 上最近物体之间的欧式距离，由此得到传感器测量的概率可以由一个以 0 为中心的高斯函数表示：

$$P_{hit}(z_t^k|x_t, m) = \epsilon_{\sigma_{hit}}(dist) \quad (4-3)$$

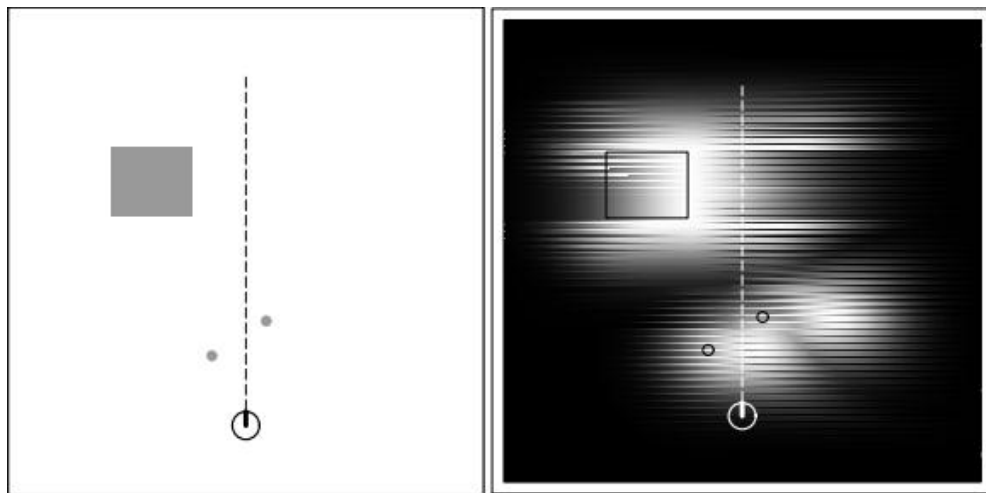


图 4-6 左边是环境实例，三个灰色区域为障碍物的环境实例机器人是在底部的圆圈表示得到的测量 z_t^k 右边是障碍物配置的似然域，颜色越深在该位置探测到障碍物的可能性越小

(2) 测量失败：在一些环境中，由于传感器限制，比如当激光测距仪检测到黑色吸光物体时，或者有些激光系统在强烈的阳光下测量物体时，都是发生检测失败，所以传感器失败的典型结果就是最大测量距离问题。传感器返回它的最大允许值 z_{max} ，这里用一个 z_{max} 为中心的点群分布 p_{max} 进行建模：

$$p_{max}(z_t^k | x_t, m) = I(z = z_{max}) = \begin{cases} 1 & z = z_{max} \\ 0 & \text{其他} \end{cases} \quad (4-4)$$

所以由这个分布式可以知道 p_{max} 是一个离散分布，其中 I 是一个指示函数，当其参数为真时取值为 1，否则取值为 0。

(3) 无法解释的随机测量：这里用一个均匀分布 P_{rand} 来建立模型：

$$P_{rand}(z_t^k | x_t, m) = \begin{cases} \frac{1}{z_x} & 0 \leq z_t^k < z_{max} \\ 0 & \text{其他} \end{cases} \quad (4-5)$$

所以综合以上的测量误差可以给出基于似然域的混合测量模型期望的概率 $p(z_t | x_t, m)$ 表示为三种分布之和 $z_{hit}P_{hit} + z_{rand}P_{rand} + z_{max}P_{max}$ 其中 $z_{hit}, z_{rand}, z_{max}$ 表示权值分配。如图 4-7 所示表示传感器的测量概率 $p(z_t | x_t, m)$ 描述为似然域模型。

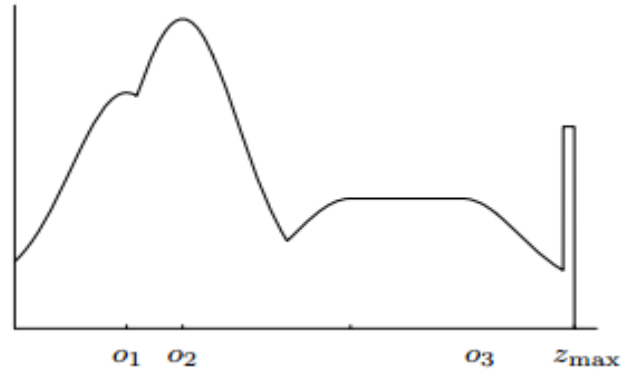


图 4-7 是将障碍物检测的似然描述成全局的 x-y 的坐标的函数，叫做似然域
那么根据上述的模型给出表 4-2 用似然域计算测量概率的算法：

```

Likelihood_field_range_finder ( $z_t, x_t, m$ )
 $q = 1$ 
for all  $k$  do
if  $z_t^k \neq z_{max}$ 

 $x_{z_t^k} = x + x_{k,sens} \cos \theta - y_{k,sens} \sin \theta + z_t^k \cos(\theta + \theta_{k,sens})$ 

 $y_{z_t^k} = y + y_{k,sens} \cos \theta - x_{k,sens} \sin \theta + z_t^k \sin(\theta + \theta_{k,sens})$ 

 $dist = \min_{x',y'} \left\{ \sqrt{(x_{z_t^k} - x')^2 + (y_{z_t^k} - y')^2} \mid \langle x', y' \rangle \text{ occupied in } m \right\}$ 

 $q = q * (z_{hit} * \text{prob}(dist, \sigma_{hit}) + \frac{z_{random}}{z_{max}})$ 

```

表 4-2 利用与最近物体的欧式距离计算测距仪的似然算法

此算法是假定不同传感器波束的噪声是相互独立的，首先判断检测传感器的额读数是否为最大距离读数，如果是则被舍弃，反之则计算 x - y 空间中与最近障碍物的距离，最后通过将一个正态分布和均匀分布混合得到似然结果。

第五章 RBPF 定位与构图的理论

5.1 粒子滤波基本算法

基于 Rao-Blackwellized 粒子滤波的 SLAM 的方法估计的方法，该方法的粒子滤波器使用有限个采样粒子代表概率密度分布，机器人所在位置的高概率区域含有高概率的粒子或在该地区的粒子的权重比较大。所以我们知道该种粒子滤波的 SLAM 方法以样本粒子的集合的方法去逼近位姿的近似的概率分布，采用一组带有权值的随机样本粒子来描述概率分布，在概率高的区域，粒子的密度就大，相反，在概率低的区域，粒子的密度就小。此时这种非参数化的粒子分布能够逼近机器人所在位置的概率分布。因此，基于粒子滤波的 SLAM 方法能够比较精确地表达基于观测量和控制量的后验概率分布。

本文设计的基于 Rao-Blackwellized 粒子滤波器是一种基于蒙特卡罗方法和递推贝叶斯滤波的方法，核心的思想是在贝叶斯滤波的基础上，利用蒙特卡罗积分来替代贝叶斯递推滤波中的复杂的积分运算，在贝叶斯估计的预测阶段，依靠经验分布产生一组随机样本集，也称之为粒子，在更新阶段，通过传感器的测量值调整粒子的位置和权值来更新分布，并将此分布作为下一步的经验分布。

在基于 Rao-Blackwellized 粒子滤波 SLAM 算法中，地图的特征采用独立的 EKF 实现。每个特征点都对应着一个单独的 EKF 滤波器，如图 5-1，假设用 M 个粒子用来估计机器人的路径估计用 $x_t^{[k]}$ 表示，那么每个粒子都对应着 N 个独立的 EKF 来估计的 N 个特征点，即理解为每个粒子都对应着一条路径和基于该路径上的 N 个特征点的高斯估计，并假设第 i 个特征带你的高斯分布的均值和方差为 $\mu_i^{[k]}$ 和 $\Sigma_i^{[k]}$ ， $[k]$ 表示粒子的索引。

	robot path	feature 1	feature 2	...	feature N
Particle $k = 1$	$x_{1:t}^{[1]} = \{(x \ y \ \theta)^T\}_{1:t}^{[1]}$	$\mu_1^{[1]}, \Sigma_1^{[1]}$	$\mu_2^{[1]}, \Sigma_2^{[1]}$...	$\mu_N^{[1]}, \Sigma_N^{[1]}$
Particle $k = 2$	$x_{1:t}^{[2]} = \{(x \ y \ \theta)^T\}_{1:t}^{[2]}$	$\mu_1^{[2]}, \Sigma_1^{[2]}$	$\mu_2^{[2]}, \Sigma_2^{[2]}$...	$\mu_N^{[2]}, \Sigma_N^{[2]}$
		\vdots			
Particle $k = M$	$x_{1:t}^{[M]} = \{(x \ y \ \theta)^T\}_{1:t}^{[M]}$	$\mu_1^{[M]}, \Sigma_1^{[M]}$	$\mu_2^{[M]}, \Sigma_2^{[M]}$...	$\mu_N^{[M]}, \Sigma_N^{[M]}$

图 5-1 M 个粒子代表的机器人的路径以及每个粒子对应的特征路标点

在 RBPF 解决 SLAM 的方法中，SLAM 问题被分解为两个部分：利用粒子滤波器进行定位的非高斯后验部分和利用卡尔曼滤波器进行全局地图更新的条件高斯部分，利用公式表达为

$$p(x_{1:k}, m | z_{1:k}, u_{1:k-1}) = p(m | x_{1:k}, z_{1:k})p(x_{1:k} | z_{1:k}, u_{1:k-1}) \quad (5-1)$$

式中 $p(m | x_{1:k}, z_{1:k})$ 为全局地图更新部分， m 表示全局地图信息， $p(x_{1:k} | z_{1:k}, u_{1:k-1})$ 表示机器人的定位部分，所以从这个公式我们可以将 SLAM 分为两个部分研究：机器人的定位和全局地图更新。所以 SLAM 中机器人定位的精确性和鲁棒性对全局地图的更新以及后期的状态都有着十分重要的影响。

5.2 从运动模型中采样

在已知数据关联的情况下，根据上述的粒子表格可以将粒子表示为

$$Y_t^{[k]} = \{x_t^{[k]}, \mu_{1,t}^{[k]}, \Sigma_{1,t}^{[k]}, \dots, \mu_{N,t}^{[k]}, \Sigma_{N,t}^{[k]}\} \quad (5-2)$$

在 RBPF-SLAM 算法的后验中有 M 个这样的粒子，根据 $t-1$ 时刻后验计算时刻 t 的后验，需要从上一时刻的粒子集 $Y_{t-1}^{[k]}$ 结合新的控制 u_t 和测量 z_t 产生新的粒子集合 $Y_t^{[k]}$ ，通过采样新位姿扩展路径后验，根据运动后验取出一个样本：

$$x_t^{[k]} \sim p(x_t | x_{t-1}^{[k]}, u_t) \quad (5-3)$$

式中 $x_{t-1}^{[k]}$ 为机器人位置在时刻 $t-1$ 的后验估计，存放在第 k 个粒子中，结果产生的采样 $x_t^{[k]}$ 被增加到粒子的临时集合中，同时还有以前位姿的路径 $x_{1:t-1}^{[k]}$ 。如图 5-2, 所示说明从一个单一初始位姿开始取得的位姿粒子的集合。

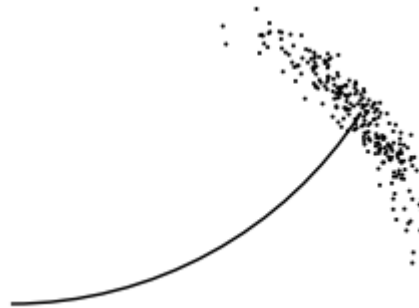


图 5-2 从概率运动模型取得的样本

上式只是根据最新控制 u_t 产生位姿 x_t ，而忽略了测量 z_t ，导致临时集合粒子不是根据期望的后验分布的事实，所以这里就涉及到了重采样技术，该方法是基于粒

子滤波的 SLAM 修正这种不匹配的通用技术，假设 Y_{t-1} 的路径粒子服从 $p(x_{1:t-1} | z_{1:t-1}, u_{1:t-1}, c_{1:t-1})$ 分布，此分布是一种渐进的修正近似。临时集合的路径粒子分布服从如下的分布：

$$p(x_{1:t}^{[k]} | z_{1:t-1}, u_{1:t}, c_{1:t-1}) = p(x_t^{[k]} | x_{t-1}^{[k]}, u_t) p(x_{1:t-1}^{[k]} | z_{1:t-1}, u_{1:t-1}, c_{1:t-1}) \quad (5-4)$$

因式 $p(x_t^{[k]} | x_{t-1}^{[k]}, u_t)$ 是采样分布，在式 5-3 中使用。将目标分布与当前时刻的测量值 z_t 与一致性 c_t 的关系有： $p(x_{1:t}^{[k]} | z_{1:t}, u_{1:t}, c_{1:t})$ 。我们知道重采样过程会导致目标分布与建议分布的不同，由此得到重采样的重要性系数由目标和建议分布的商给出：

$$w_t^{[k]} = \frac{\text{目标分布}}{\text{建议分布}} = \frac{p(x_{1:t}^{[k]} | z_{1:t}, u_{1:t}, c_{1:t})}{p(x_{1:t}^{[k]} | z_{1:t-1}, u_{1:t}, c_{1:t-1})} = \eta p(z_t | x_t^{[k]}, c_t) \quad (5-5)$$

经过对此式的变换与近似可以总结为经过测量更新完全相同的线性近似，得到的重要性权重用来从临时采样集合抽取更换 M 个新样本，以替换那些权重较低的粒子，通过重采样的过程，粒子是否保留下来与它们的测量概率成正比。

5.3 重采样技术

在标准的粒子滤波算法中由于先验分布比较容易得到并采样，因此将该分布作为重要密度函数，但是由于这种选择方式并没有将当前的测量值与重要性密度函数的选取相结合，所以该方法得到的样本分布不能更好的近似后验分布，特别是在状态转移概率密度分布分布明显偏离似然函数分布，或者似然函数明显呈现尖峰时，此时将产生较大的估计误差，似然函数与先验分布的关系如图：

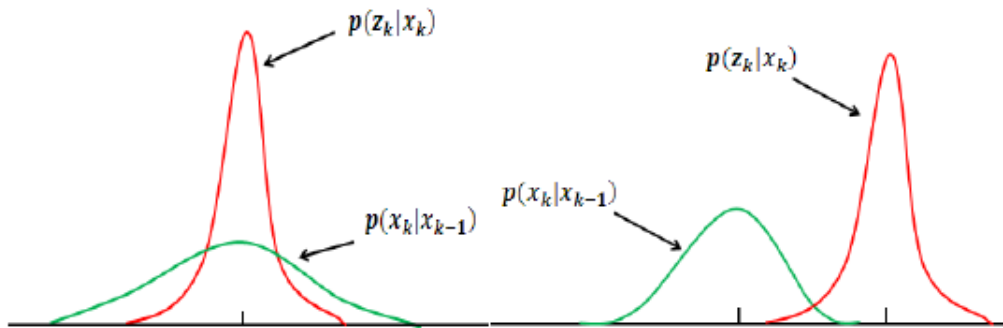


图 5-3 似然函数与先验分布的关系

由此图说明了在粒子滤波的过程中，随着算法迭代次数的增加，由先验分布和似

然函数的偏差可以看出绝大多数的粒子的权值不断的减小，甚至可以忽略不计，因此大量的计算都浪费在更新这些权重极小对估计后验密度函数几乎不起作用的粒子上，同时权重被分配到少数的粒子上，这就损失了粒子的多样性，而且估计的粒子集与实际的后验概率分布差异比较大，这就造成了粒子退化问题。

在解决粒子退化的问题上，应用比较多是重采样技术，其基本的思想就是通过对后验概率密度函数 $p(x_k|z_{1:k})$ 再进行 N 次采样，以产生新的样本集 $\{x_k^i\}_{i=1}^N$ ，并且将重采样后粒子的权重被设置为 $1/N$ ，重采样的负作用是引起样本枯竭的问题，同时会消耗很多时间用于计算重采样过程中。

5.4 更新观测到的特征估计

在 RBPF-SLAM 中基于测量值 z_t 和采样位姿 $x_t^{[k]}$ 更新特征估计的后验，该后验用均值 $\mu_{n,t-1}^{[k]}$ 和协方差 $\Sigma_{n,t-1}^{[k]}$ 表示，更新后的值连同新的位姿一起被增加到临时粒子集合中，准确的更新方差依赖特征 m_n 是否在 t 时刻被观测到。对于 $n \neq c_t$ ，表示没有观测到第 n 个特征，则建立的特征后验保持不变，说明不用顾及更新，那么对于观测到的特征 $n = c_t$ ，这种情况比较复杂，就需要用到贝叶斯准则，然后对其进行一些标准简化特征后验的表示为：

$$\begin{aligned} p(m_{c_t}|x_{1:t}, c_{1:t}, z_{1:t}) &= \frac{p(z_t|m_{c_t}, x_{1:t}, c_{1:t}, z_{1:t-1})p(m_{c_t}|x_{1:t}, c_{1:t}, z_{1:t-1})}{p(z_t|x_{1:t}, c_{1:t}, z_{1:t-1})} \\ &= \frac{p(z_t|x_t, m_{c_t}, c_t)p(m_{c_t}|x_{1:t-1}, c_{1:t-1}, z_{1:t-1})}{p(z_t|x_{1:t}, c_{1:t}, z_{1:t-1})} \end{aligned} \quad (5-6)$$

这里用粒子索引 k 重写：

$$p(m_{c_t}|x_t^{[k]}, c_{1:t}, z_{1:t}) = \eta * p(z_t|m_{c_t}, x_t^{[k]}, c_{1:t})p(m_{c_t}|x_{1:t-1}^{[k]}, c_{1:t-1}, z_{1:t-1}) \quad (5-7)$$

概率 $p(m_{c_t}|x_{1:t-1}^{[k]}, c_{1:t-1}, z_{1:t-1})$ 在时刻 t-1 由具有均值 $\mu_{c_t,t-1}^{[k]}$ 和协方差 $\Sigma_{c_t,t-1}^{[k]}$ 的高斯表示，在 t 时刻对新特征的估计也用高斯分布表示，在 RBPF-SLAM 中以 EKF 算法用来线性化感知模型 $p(z_t|m_{c_t}, x_t^{[k]}, c_{1:t})$ ，在此使用泰勒近似展开测量函数 h：

$$h(m_{c_t}, x_t^{[k]}) = \hat{z}_t^{[k]} + H_t^{[k]}(m_{c_t} - \mu_{c_t,t}^{[k]}) \quad (5-8)$$

在该式中函数 h 的非线性导致了后验为非高斯的，其中 $\hat{z}_t^{[k]} = h(\mu_{c_t,t-1}^{[k]}, x_t^{[k]})$ ， $H_t^{[k]}$ 是函数 h 的一阶导数，表示为 $H_t^{[k]} = h'(x_t^{[k]}, \mu_{c_t,t-1}^{[k]})$ ，由此式 5-6 即可以表示为：

$$p(z_t | m_{c_t}, x_t^{[k]}, c_{1:t}) \sim N(z_t; h(m_{c_t}, x_t^{[k]}), Q_t) \quad (5-9)$$

$$p(m_{c_t} | x_{1:t-1}^{[k]}, c_{1:t-1}, z_{1:t-1}) \sim N(m_{c_t}; \mu_{c_t,t}^{[k]}, \Sigma_{c_t,t}^{[k]}) \quad (4-10)$$

在这个近似下，特征 c_t 的位置的后验实质上是高斯的。新的均值和协方差可以使用标准的 EKF 测量更新得到：

$$\begin{aligned} K_t^{[k]} &= \Sigma_{c_t,t-1}^{[k]} H_m^{[k]} Q_t^{[k]-1} \\ \mu_{c_t,t}^{[k]} &= \mu_{c_t,t-1}^{[k]} + K_t^{[k]} (z_t - \hat{z}_t^{[k]}) \\ \Sigma_{c_t,t}^{[k]} &= (I - K_t^{[k]} H_m^{[k]}) \Sigma_{c_t,t-1}^{[k]} \end{aligned}$$

5.5 未知数据关联

在粒子滤波的 SLAM 中，粒子会依赖自己的局部数据解决关联问题，在数据关联中时刻 t 的数据关联是基于现有的数据决定变量 c_t 的问题。在实际问题中，机器人观测到环境中的特征，根据该特征相对这些特征的实际位姿，这些测量值对应着地图中的不同特征，采用比如最大似然技术是一种有效的数据关联技术，该算法使用多个粒子，因此能基于每个粒子确定一致性，所以滤波不仅在机器人的路径上采样，而且在可能有数据关联决策上进行采样。每个粒子维持数据关联变量的局部集合用 $\hat{c}_t^{[k]}$ 表示，在最大似然数据关联里，每个 $\hat{c}_t^{[k]}$ 由最大化测量值 z_t 的似然来决定：

$$\hat{c}_t^{[k]} = \underset{c_t}{\operatorname{argmax}} p(z_t | c_t, \hat{c}_{1:t-1}^{[k]}, x_{1:t}^{[k]}, z_{1:t-1}, u_{1:t})$$

最大似然技术使得估计地图特征成为可能，对所有已知的地图，如果似然低于阈值 P_0 ，便使用最大似然创建并增加地图新特征。

那么在本课题中，使用的栅格地图，当一个特征被观测到时，它存在的对数概率将会增加一个数值，该数值使用标准贝叶斯公式计算，同样的，如果一个特征没有被观测到，这样的信息将会引起特征存在的变量减少一个固定的数值，当该变量减少到低于一定的阈值时，该特征将从粒子列表中移除。

5.6 占用栅格地图构建算法

占用栅格地图构建算法是根据给定的数据计算整个地图的后验概率， $P(m | z_{1:t}, x_{1:t})$ ，其中 m 为地图， $z_{1:t}$ 为直到时刻 t 的所有测量值， $x_{1:t}$ 为用所

有机器人位姿定义的路径，最常用的占用栅格地图是二维平面地图，这种二维平面地图描述的是三维世界的二维切片，而且二维地图通常是机器人平面导航的选择，占用栅格地图技术可以推广到三维，在三维地图中我们用八叉树的形式表示空间中的位置是否被占用，这种八叉树地图同样可以应用到三维空间中运动的机器人，比如飞行器，海底探测机器人，装有机械臂的机器人等。

首先介绍关于占用栅格地图的算法的介绍， m_i 表示第 i 个栅格单元，占用栅格地图将空间分割为有限多个栅格单元 $m = \{m_i\}$ 每个 m_i 与一个二值占用变量相对应，该变量指示出该单元是否被占用，如果该栅格被占用为“1”，未被占用为“0”， $P(m_i)$ 表示了该栅格单元被占用的可能性。标准的占用栅格方法将构建地图这一问题划分为一些独立的问题，即为所有的栅格单元 m_i 建立后验概率 $P(m_i|z_{1:t}, x_{1:t})$ 每一个这样的问题都是一个静态的二值问题，这种方法不能表达相邻单元之间的联系，所以整个地图的后验概率用边缘概率的乘积近似 $P(m|z_{1:t}, x_{1:t}) = \prod_i P(m_i|z_{1:t}, x_{1:t})$ ，那么占用栅格地图构建算法应用了对数占用概率的表达方式，即 $l_{t,i} = \log \frac{P(m_i|z_{1:t}, x_{1:t})}{1 - P(m_i|z_{1:t}, x_{1:t})}$ ，对数的表达的的优点是可以避免 0 和 1 附近数值的不稳定性，由此计算后验的概率为 $P(m_i|z_{1:t}, x_{1:t}) = 1 - \frac{1}{1 + \exp\{l_{t,i}\}}$ 。

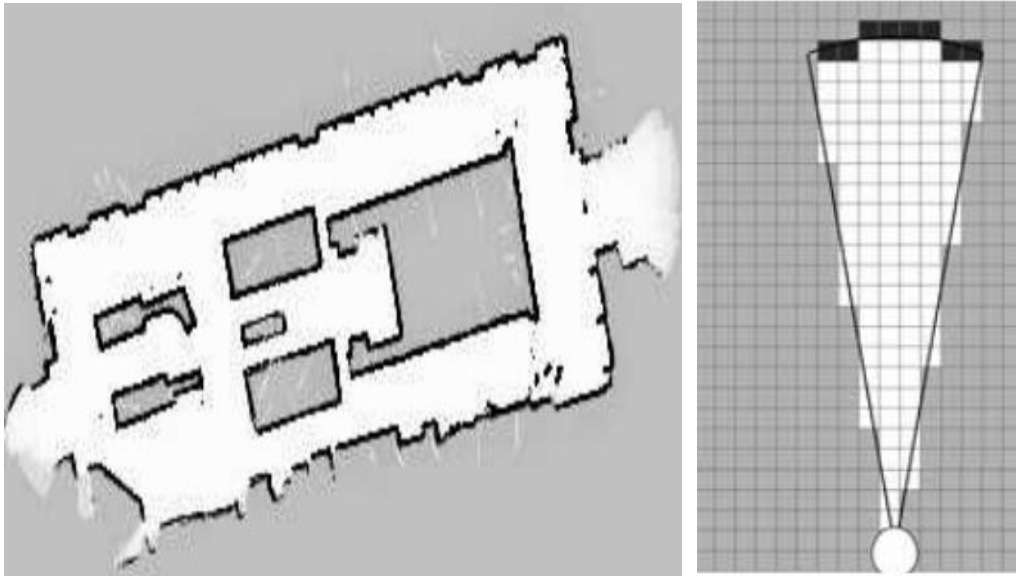


图 5.4 左图是占用栅格地图实例，右图是栅格地图细节处放大后，图中每个栅格的暗度对应着该栅格被占用的可能性

5.7 本章小结

本章对使用 Rao-Blackwellized 粒子滤波算法定位进行了全方位的解读，该

方案是基于有效的对机器人的路径估计，而路标估计之间相互独立，由贝叶斯公式可以对粒子滤波 SLAM 问题对后验概率分布的估计 $p(x_{1:k}, m|z_{1:k}, u_{0:k-1}, x_0)$ 进行分解：

$$\begin{aligned} p(x_{1:k}, m|z_{1:k}, u_{0:k-1}, x_0) &= p(m|x_{0:k}, z_{1:k}, u_{0:k-1})p(x_{1:k}|z_{1:k}, u_{0:k-1}, x_0) \\ &= p(x_{1:k}|z_{1:k}, u_{0:k-1}, x_0) \prod_{i=1}^M p(m_i|x_{1:k}, z_{1:k}) \end{aligned}$$

在这个等式中 $p(x_{1:k}|z_{1:k}, u_{0:k-1}, x_0)$ 表示机器人的路径估计， $\prod_{i=1}^M p(m_i|x_{1:k}, z_{1:k})$ 表示机器人的环境地图估计，所以将 SLAM 过程分解为已知路径下的地图的构建问题。假设已知粒子集 $\{x_{1:k}^{(i)}, w_k^{(i)}\}_{i=1}^N$ ， k 时刻的控制输入 u_k 和 $k+1$ 时刻的观测值 z_{k+1} ，则粒子滤波的步骤如下

- (1) 初始化：设置初始化参数，有机器人系统状态的初始值 \hat{x}_0 ，协方差矩阵 P_0 ，系统过程噪声协方差矩阵 Q_0 ，测量噪声协方差矩阵 R_0 ，从初始分布中采样最初的 N 个样本 $\{x_0^{(i)}, w_0^{(i)}\}, i = 1, 2, \dots, N$
- (2) 预测：预测机器人的位姿基于运动模型，由控制输入这里一般采用里程计模型用来预测机器人 $k+1$ 时刻的位姿，并计算每个粒子在 $k+1$ 时刻的位姿的预测均值和方差。
- (3) 数据关联问题：获取 $k+1$ 时刻的观测值，并将其与 k 时刻每个粒子的估计观测值进行数据关联，这些关联过程都是相互独立
- (4) 获取建议分布：对于每个粒子代表机器人的位姿，基于每个粒子各自的观测值，利用 EKF 算法计算每个粒子位姿估计的均值和方差，并以此均值和方差构建一高斯分布函数作为需要求取的重要性概率密度函数，该概率密度函数就是每个粒子的建议分布 $q(x_{k+1}|x_k^{(i)}, z_{k+1}, u_k)$ 。
- (5) 估计机器人路径轨迹：按照的步骤计算 $k+1$ 时刻用来表示机器人后验概率分布 $p(x_{1:k+1}|z_{1:k+1}, u_{0:k}, x_0)$ 的粒子集 $\{x_{1:k+1}^{(i)}, w_{k+1}^{(i)}\}_{i=1}^N$ 。
 - a. 从建议分布中采样： $x_{k+1}^{(i)} \sim q(x_{k+1}|x_k^{(i)}, z_{k+1}) \quad i=1, \dots, N$
 - b. 计算粒子的非正则权值： $w_{k+1}^i(x) \propto w_k^i(x_{k-1}) \frac{p(z_{k+1}|x_{k+1}^i)p(x_{k+1}^i|x_k^i)}{q_k(x_{k+1}^i|x_{0:k}^i, z_{1:k+1})}$
 - c. 归一化权值： $w_{k+1}^i(x) \propto w_k^i(x_{k-1}) \sum_j \hat{w}_{k+1}^j$

d. 重采样：计算粒子集的 N_{eff} ，如果 N_{eff} 小于给定阈值，则对粒子集

$\{x_{1:k+1}^{(i)}, w_{k+1}^{(i)}\}_{i=1}^N$ 进行重采样，得到新的粒子集 $\{x_{1:k+1}^{(i)}, \frac{1}{N}\}_{i=1}^N$ ，否则粒子集保持不变，

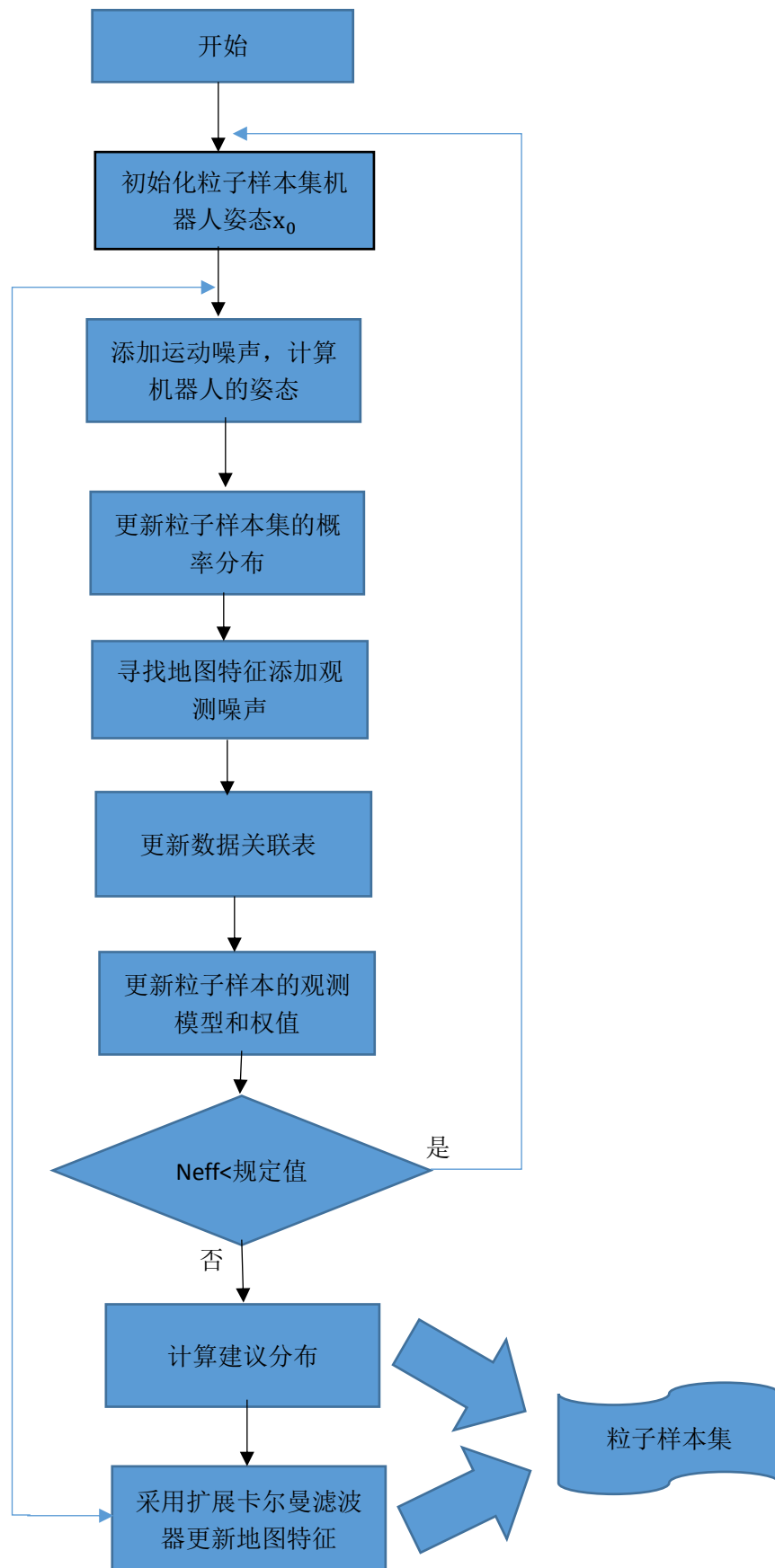
(6) 更新地图估计：根据 $\prod_{i=1}^M p(m_i|x_{1:k}, z_{1:k})$ 公式我们摘掉环境地图的估计已经被分解为 M 个独立的特征估计问题，而对于每个独立的特征估计，EKF 既可以满足精度要求并在效率上有一定的优势所以地图的后验概率分布 $p(m_i|x_{1:k+1}, z_{1:k+1}, u_{1:k+1})$ ，采用 EKF 算法进行估计。地图估计表示为 $\{\mu^1, P^1, \dots, \mu^M, P^M\}$ 在这个式中 μ^i 表示环境地图中第 i 个特征的高斯均值， P^i 表示环境地图中第 i 个特征的方差，如果某特征在此静态地图被观测到，那么该特征采用 EKF 算法，更新方程分别是

$$\begin{aligned}\mu_{k+1}^i &= \mu_k^i + K_{k+1}^i [z_{k+1}^i - h(\mu_k^i)] \\ P_{k+1}^i &= P_k^i - K_{k+1}^i S_{k+1}^i (K_{k+1}^i)^T\end{aligned}$$

进行更新，如果没有在地图中被观测到，那么其均值和方差便等于上一时刻的值。

(7) 如果系统一直未停止，则返回步骤 2 继续执行。

如流程图所示。



第六章 实验验证与分析

随着科技的进步,各类机器人的需求量越来越大,无论是在服务行业还是工业应用领域,通过机器人替代人类来减少成本,提高服务质量和效率是未来各行业的趋势,因此在一些特定场景下,构建三维环境模型是必要且重要的任务,通过构建出的地图完成指定任务是对机器人的基本要求。在国内外研究者们一般采用里程计,激光雷达,超声波,以及视觉传感器来对机器人进行定位和构图。基于激光雷达一般可以获取室内平面的二维栅格地图,再利用定位信息,对深度相机获取的三维环境的点云信息进行构建三维地图。

本文提出的是一种结合两种传感器各自优势的同时构建二维和三维地图的 SLAM 方法,基于 Rao-Blackwellized 粒子滤波的同时定位的构建二维和三维地图的方法,可以说是融合各自的优势构建室内三维地图。总体思路如下:基于 ROS 机器人操作系统下的 Gmapping 包实时的构建二维栅格地图的同时,提取出定位信息结合深度相机获取更加丰富的三维环境信息,在机器人的移动过程中,通过粒子滤波的方法获取机器人在不同时刻的位姿,实现精确的定位,那么此定位信息是一个不断变化的位姿矩阵信息,此时对于深度相机采集到的点云信息,我们采用栅格化处理,因为栅格化处理不仅可以降低数据量更能够提高栅格点云的拼接速度,所以根据定位获取的位姿变换矩阵完成相邻三维栅格点云地图的拼接,最终完成室内场景的三维地图的构建,与此同时,二维栅格地图在构建的同时又反馈了机器人的定位信息。

6.1 实验平台

本实验的移动平台是 Clear path Robotics 公司的 Turtlebot[57]机器人,该移动平台具备了基本的移动功能,传感器和控制系统。传感器有 Kinect1.0,单轴陀螺仪 ADXRS652,内置 32 个传感器,并在 ROS 操作系统中开源了底层控制代码,控制系统是使用 ubuntu14.04 操作系统的联想笔记本,ROS 的版本是 Indigo 版本,本课题中将结合使用思岚科技的激光雷达产品 RPLIDAR 同时构建二维栅格地图和八叉树地图。



图 6-1 turtlebot

本次实验中以 turtlebot 为移动机器人平台，利用 Rplidar 传感器、Kinect 1.0 传感器，IMU，Arduino、笔记本电脑等设备。图 6-2 描述了实验中硬件系统框架。其中我们使用的笔记本是联想 T440p，CPU 是酷睿 i5，2.6GHz, 16G 内存。

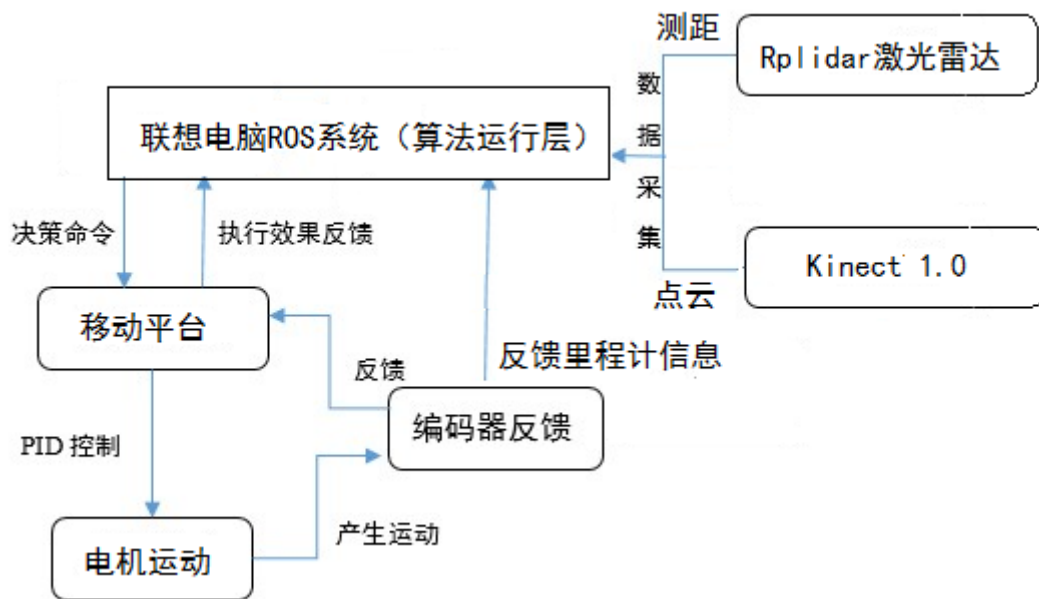


图 6.2 硬件系统框架

6.2 Gmapping 算法定位

Gmapping[58]算法的原理在之前的章节中已经有所介绍，它适用于机器人在平面上移动的场景，栅格地图作为机器人二维环境的度量方法，是基于构建二维栅格地图中最为经典和常用的方法。在本课题中，该算法将不仅仅是构图并同时获取定位信息辅助构建三维地图，Gmapping 算法是根据 Rao-Blackwellized 粒子滤波的应用，那么该方法主要是解决栅格地图构建下的 SLAM 问题，根据上文

所述，该算法不仅要获得机器人在运动过程中里程计的控制信息 $u_{1:k-1}$ ，同时需要获取外部激光雷达获取的测距信息 $z_{1:k}$ ，以及机器人的运动轨迹 $x_{1:k}$ ，这里的环境地图为 m ，因此为了机器人位姿的联合后验概率：

$$p(x_{1:k}, m | z_{1:k}, u_{1:k-1}) = p(m | x_{1:k}, z_{1:k})p(x_{1:k} | z_{1:k}, u_{1:k-1})$$

通过联合后验概率公式的因式分解，问题化解为求解机器人的运动轨迹和根据运动轨迹估计地图后验，那么计算机器人运动轨迹的部分是通过机器人的里程计信息结合外部环境的测距信息估计机器人的位姿，对于已知机器人位姿信息后，问题就很简单的理解为，知道机器人位姿后的构图问题，计算出环境地图的概率。这种使用粒子滤波的方法估计机器人的位姿和地图的联合后验的方法，每个粒子都对应着一个机器人的位姿。并不断在机器人运动过程中的测量值更新粒子的权重，并淘汰权重比较低的粒子，这种将前一时刻的测量值结合里程计信息融合的方法增加了机器人定位准确性，但同时也增加了计算量，所以一般在使用该方法时，会采用自适应算法进行重采样以减少运动过程中粒子消退的问题，并提高粒子的置信度，得到可靠的机器人的后验置信度。总结基于栅格地图的粒子滤波的方法，主要有根据运动模型采样粒子集，基于当前粒子集结合观测模型更新粒子集的权重，接着更新栅格地图，也就是根据已知的位姿信息计算每个栅格被占用的可能性。

该算法的核心思想是维护粒子的集合，每个粒子都包括了采样的机器人的路径信息，同时也包括了地图，也就是说每个粒子都拥有它自己的地图，然后根据每个粒子的地图信息进行测量似然估计，计算出粒子的重要性权重。所以粒子的地图相似性越大，权重越大，代表机器人在该位置的可能性越大。如图 6.3

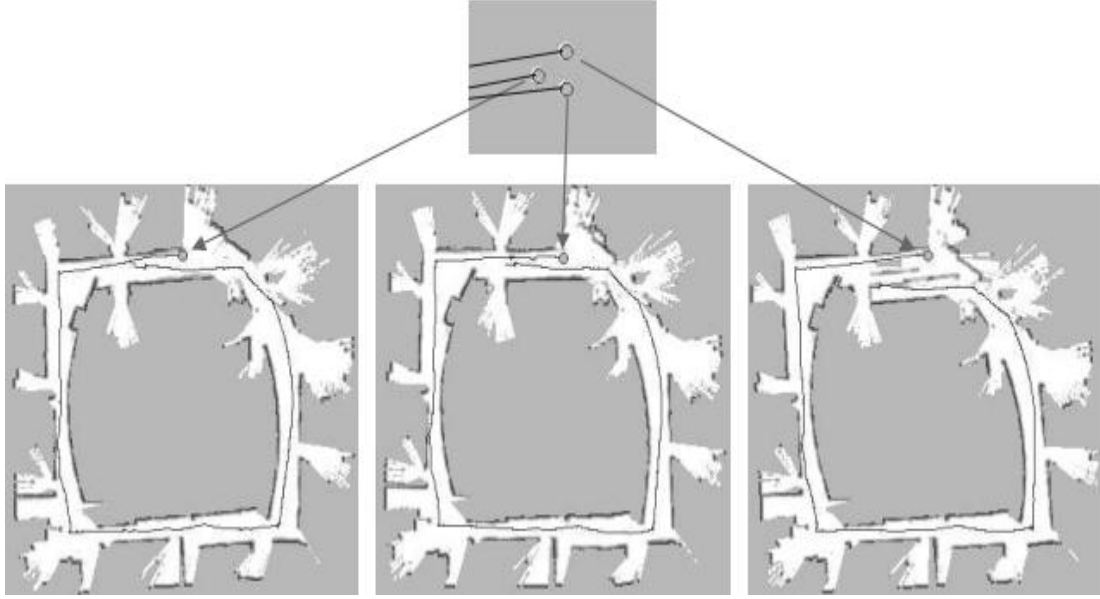


图 6.3 基于栅格的 Rao-Blackwellized 粒子滤波的地图，每个粒子都有自己的地图，基于给定粒子的地图之间的测量似然估计，计算粒子的权重。

使用粒子滤波的思想，使得在每个粒子的基础上估计数据关联变量成为可能，每个粒子能基于不同的数据关联，所以该算法可以说提供一种高效的解决 SLAM 里的数据关联的问题。有效的解决 EKF, Graph-SLAM 采用的单一的数据关联决策问题，为了解决随时间推移高效的更新粒子和地图，这里所用的地图是一种树形结构的表示法，这可以使得算法的更新的复杂度从线性关系降低到对数关系，能够使得粒子共享相同的地图部分。

6.3 三维栅格地图的构建

在基于 ROS 的 Gmapping 算法定位的作用下，机器人会同时产生定位信息即机器人的位姿转换矩阵，以及具有路标信息的栅格地图，在实验中我们采用键盘控制的方法，控制机器人的移动速度以及扫描地区，对获取的不同时刻机器人位姿信息 x_1, x_2, \dots, x_k ，可以得到相邻机器人的位姿转换矩阵：

$$x_t = R * x_{t-1} + t$$

其中 R 是旋转矩阵， t 是平移向量，在这里为了保证传感器与机器人是同步的位姿变换矩阵，需要将 kinect 传感器固定在机器人平台上，所以在此处计算出的变换矩阵同样是 Kinect 的位姿变化，那么根据旋转矩阵和平移向量，得出的相邻时刻位姿转换矩阵 T ：

$$T = \begin{bmatrix} R_{3 \times 3} & t_{3 \times 1} \\ O_{1 \times 3} & 1 \end{bmatrix} \in R^{4 \times 4}$$

根据变换矩阵 T ，随着机器人的运动过程中 Kinect 获取点云数据不断的完成拼接，形成一幅完整的三维栅格地图。

6.4 不同场景实验结果的对比

在课题中使用 ROS 实现此系统，对于在本课题中使用的 ROS 开发包的基本介绍以及设置：

- (1) 对于 turtlebot 的建模以及底层的驱动和该机器人的底层里程计信息的解算都在 ROS 的 turtlebot-indigo 开发包中，启动 turtlebot 机器人的 launch 文件是 minimal.launch，该文件是设置机器人控制层与主控电脑的通讯串口以及一些电量检测等等，并加载 turtlebot 的模型 URDF 文件等等基本信息如图 6-4。设置订阅主题用于控制机器人的运动，以及发布里程计信息。

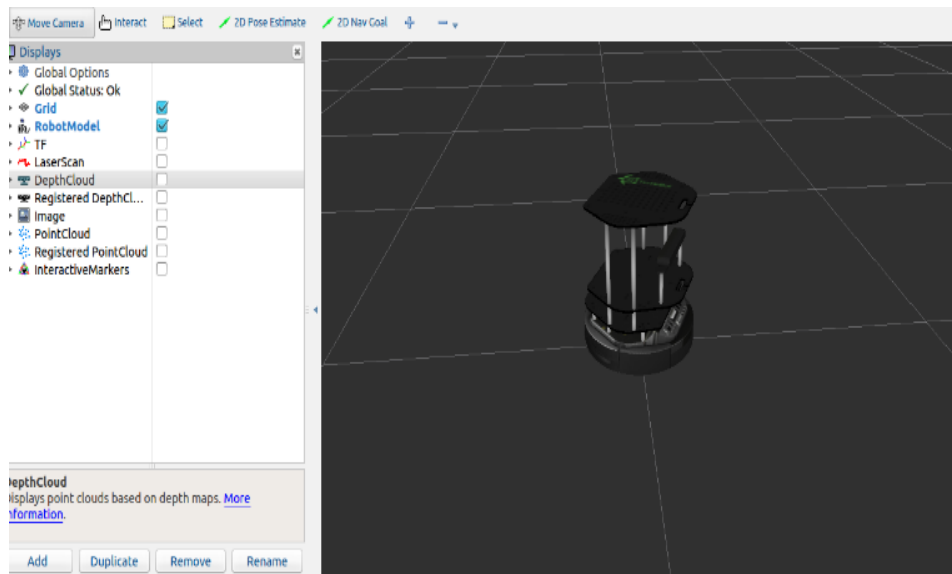


图 6.4 turtlebot 模型

- (2) 对于激光雷达 rplidar 也是有着对应的驱动开发包 rplidar_ros[59], 开发包主要是设置激光传感器与主控电脑之间的串口设置，根据串口端口号为/dev/ttyUSB0, 串口波特率设置为 115200bps，该部分的 launch 文件截图如图 6-5，该开发包会自动读取激光数据并处理为一定的数据格式可供后期处理。

```

<launch>
  <node name="rplidarNode"          pkg="rplidar_ros"  type="rplidarNode" output="screen">
    <param name="serial_port"        type="string" value="/dev/ttyUSB0"/>
    <param name="serial_baudrate"    type="int"    value="115200"/>
    <param name="frame_id"           type="string" value="base_laser_link"/>
    <param name="inverted"           type="bool"   value="false"/>
    <param name="angle_compensate"    type="bool"   value="true"/>
  </node>
</launch>

```

图 6.5 RPlidar 驱动包的 launch 文件

图 6.6 为节点流程图展示了各个节点之间的关系,其中 serila_node 节点是连接机器人与电脑端的节点,不仅仅需要传输 ROS 发送的指令信息,而且还需要不断的读取里程计信息,并结合 rplidarNode 节点读取的观测数据,输出到基于粒子滤波的 SLAM 算法中。

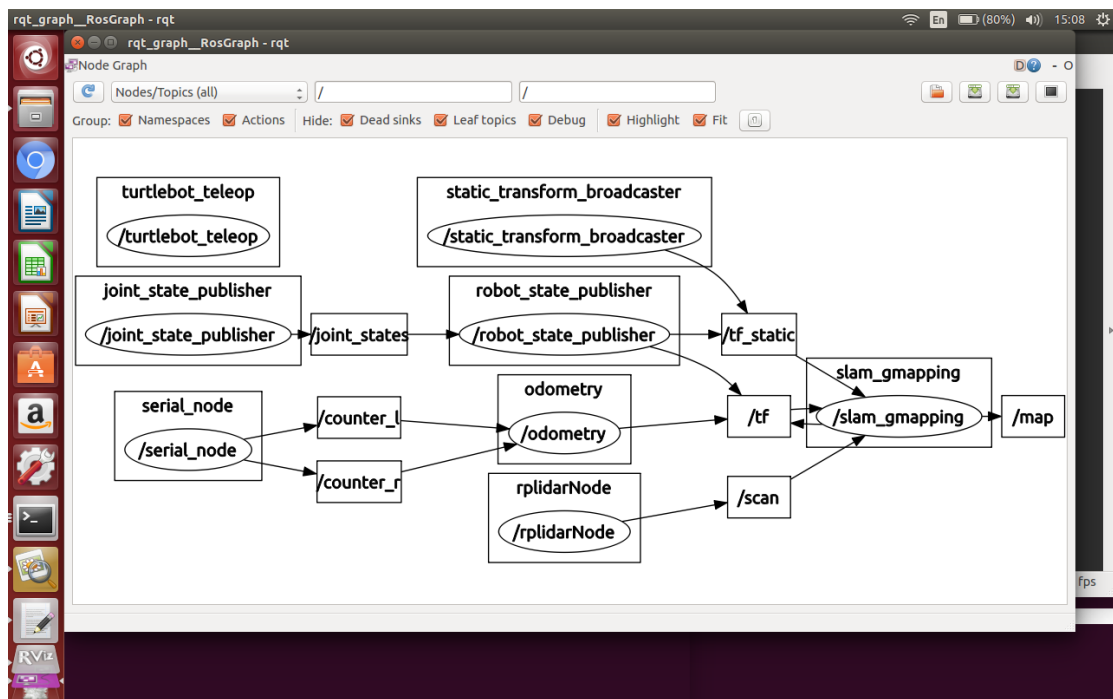


图 6.6 为移动机器人与 ROS 系统的节点流程图

- (3) 启动激光雷达的节点之后,就需要同时启动 Kinect,同时开启 Gmapping 节点同时订阅两传感器的数据进行机器人的定位与构图,其中最重要的便是在要设置好 Gmapping 的参数问题,如下图 6-7 所示,截取了部分关于 Gmapping 使用过程中的各种参数的设置。其中设置了初始化的粒子数,运动模型和测量模型的参数,以及栅格地图的最小分辨率等参数。

```

<node pkg="gmapping" type="slam_gmapping" name="slam_gmapping" output="screen">
  <param name="scan_topic" value="$(arg scan_topic)"/>
  <param name="base_frame" value="$(arg base_frame)"/>
  <param name="odom_frame" value="$(arg odom_frame)"/>
  <param name="map_update_interval" value="5.0"/>
  <param name="maxUrange" value="7.9"/>
  <param name="maxRange" value="8.0"/>
  <param name="sigma" value="0.05"/>
  <param name="kernelSize" value="1"/>
  <param name="lstep" value="0.05"/>
  <param name="astep" value="0.05"/>
  <param name="iterations" value="5"/>
  <param name="lsigma" value="0.075"/>
  <param name="ogain" value="3.0"/>
  <param name="lskip" value="0"/>
  <param name="minimumScore" value="50"/>
  <param name="srr" value="0.01"/>
  <param name="srt" value="0.02"/>
  <param name="str" value="0.01"/>
  <param name="stt" value="0.02"/>

```

图 6.7 Gmapping 参数设置的 launch 文件

- (4) 最后就是关于 OctoMap 的 ROS 的开发包的设置,并设置好 Kinect,rplidar,以及机器人的中心三者之间的 tf 关系,因为在 octomap_server 的开发包中,利用激光雷达定位信息 tf,并直接处理获取的点云数据,获取三维八叉树地图,该 Launch 文件中设置了点云的分辨率为 0.05m,以及传感器的测量似然函数的参数,如下图 6-8 所示

```

<node pkg="octomap_server" type="octomap_server_node" name="octomap_server">
  <remap from="cloud_in" to="$(arg cloud_in)"/>
  <param name="resolution" value="0.05" />
  <param name="sensor_model/hit" type="double" value="0.7" />
  <param name="sensor_model/miss" type="double" value="0.4" />
  <param name="sensor_model/min" type="double" value="0.12" />
  <param name="sensor_model/max" type="double" value="0.97" />
  <param name="sensor_model/max_range" type="double" value="3.0" />

```

图 6-8 OctoMap 参数设置

通过 ROS 的工具可以很方便的查看节点之间数据的流向,如图 6-9 为机器人在运行过程中各个节点的数据流程图。

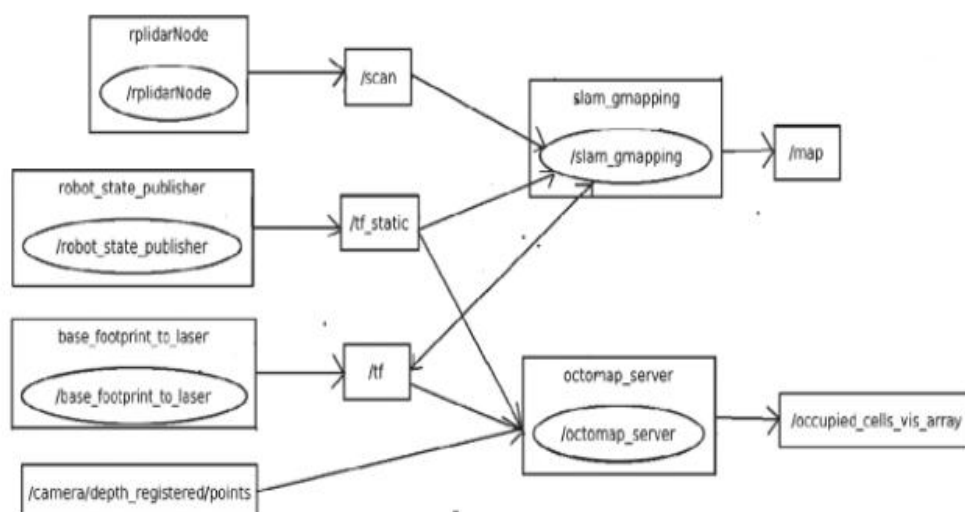


图 6-9 节点流程图

本课题为了检测系统的性能，选择了两种截然不同的场地，场景一是复杂的室内环境，为汕头大学人工智能与机器人实验室，场地中拥有各种椅子、桌子、小隔间等杂乱的实验器材设备，如图 6-10。

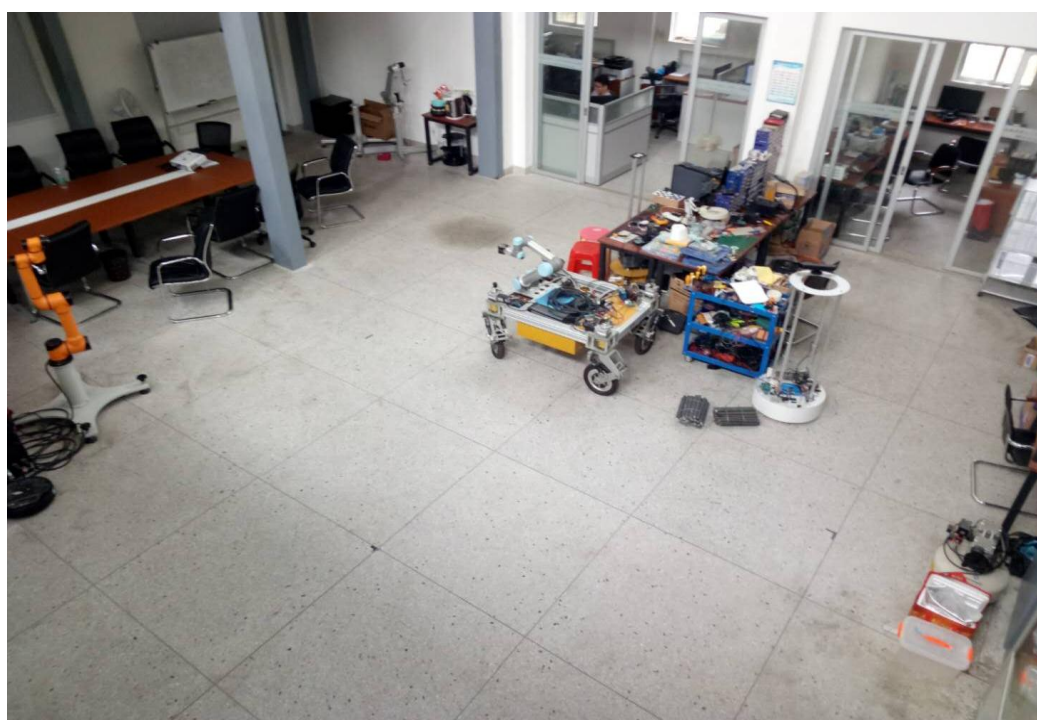


图 6-10 复杂的实验室场景

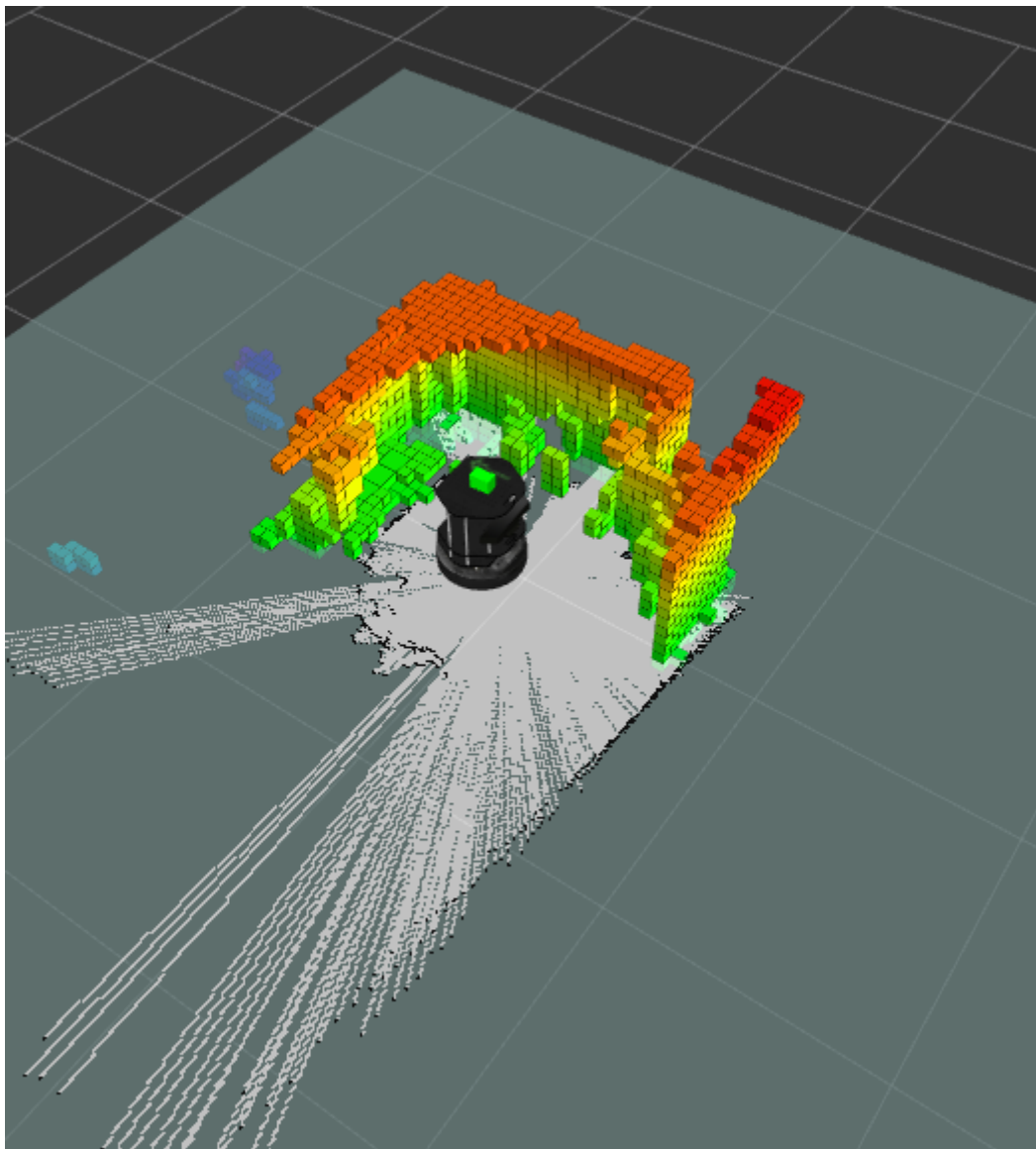


图 6.11 移动机器人开始进行同时构建二维和三维栅格地图

对于图 6-10 所示复杂且凌乱的室内场景中，本系统表现出一定的鲁棒性并能够构建出可以很好的反应真实三维环境的地图，图 6-11 为移动平台刚开始运行构建地图时的地图，其中有些三维栅格噪声点，并不影响实验结果，最终能够构建出如图 6-12 所示的二维栅格地图和 OctoMap，并在 rviz 中可视化的结果。从该构建的图中我们可以看出二维占用栅格地图以及 OctoMap 的分布，能够对比看出实际场景中的物体的大致位置，说明得出的地图具有一定的真实性，图中对于二维地图中蓝色为机器人的可行区域，在边界处显示为黑色表示这二维地图的不可行区域，或者理解为此处被障碍物占用的概率比较大，那么结合 OctoMap 地图，可以看出对于不同的高度具有不同的颜色分布，从颜色上可以区分出机器人的周围三维环境中的物体的高度信息，并且说明了该三维地图是随着高度的增加，

颜色变为红色，是不可行的区域。

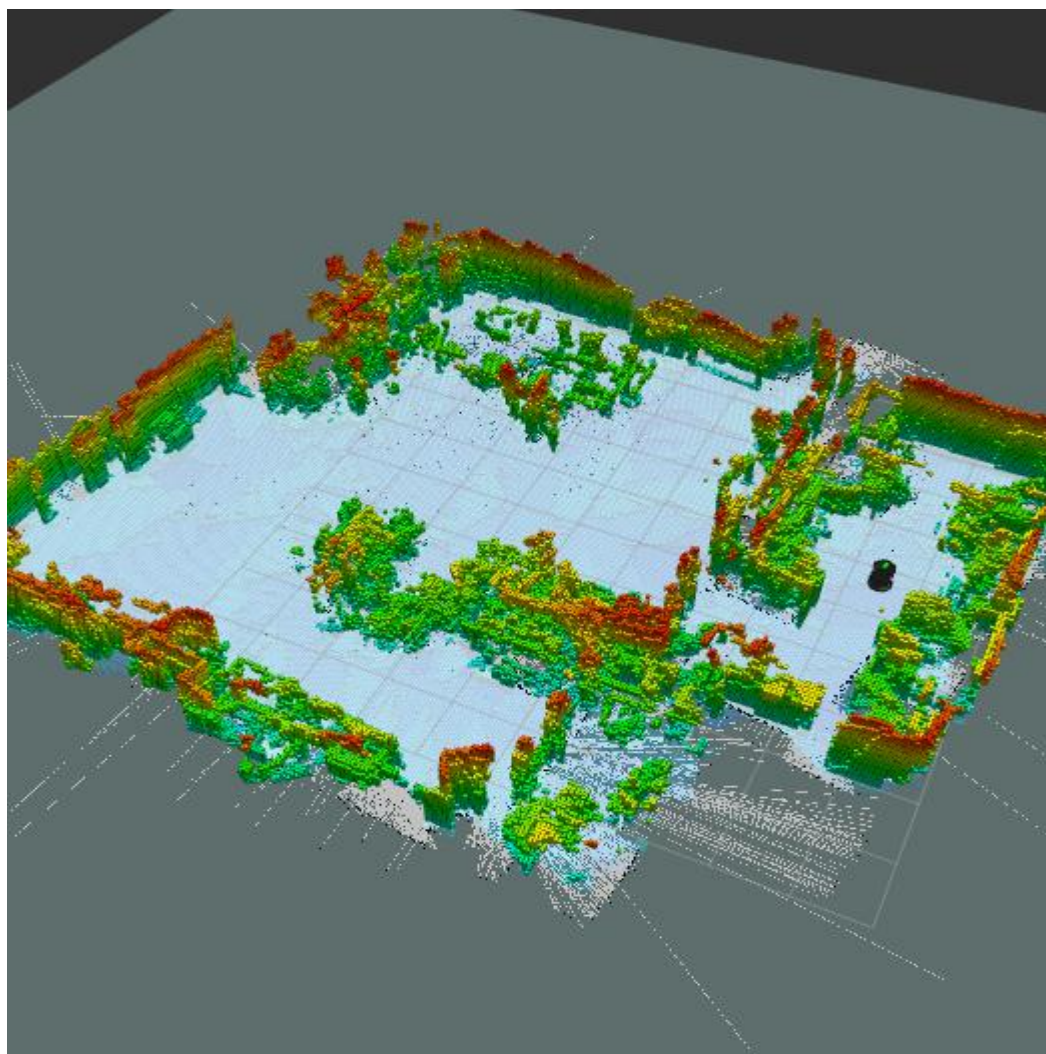


图 6-12 复杂实验室环境构建的二维与三维占用栅格地图

由于在粒子滤波的 SLAM 中，对于大场景的构建，就需要计算机具有足够的计算量，并能够适当的调整粒子的数量，以适应不同的实验场景，对于如图 6-13 所示的实验场景，是汕头大学科技楼四楼的长廊，长度大约为 300m, 宽度约 2m, 并在不同区域的宽度有所变化，对于这种大场景中的实验，该系统也表现出很好的稳定性。



图 6-13 右图是实验的长廊的场景 左图是本系统构建的地图

图 6-14 展示了变换视角后的地图，可以清晰的展示出墙壁的位置，并可以根据三维的八叉树地图的颜色区分墙壁的高度，同时地图可以应用在平面上移动机器人的碰撞检测，以及对于在室内空间中移动的机器人的路径规划，或者对安装在移动平台上的机械臂的三维空间的碰撞检测和路径规划等问题。

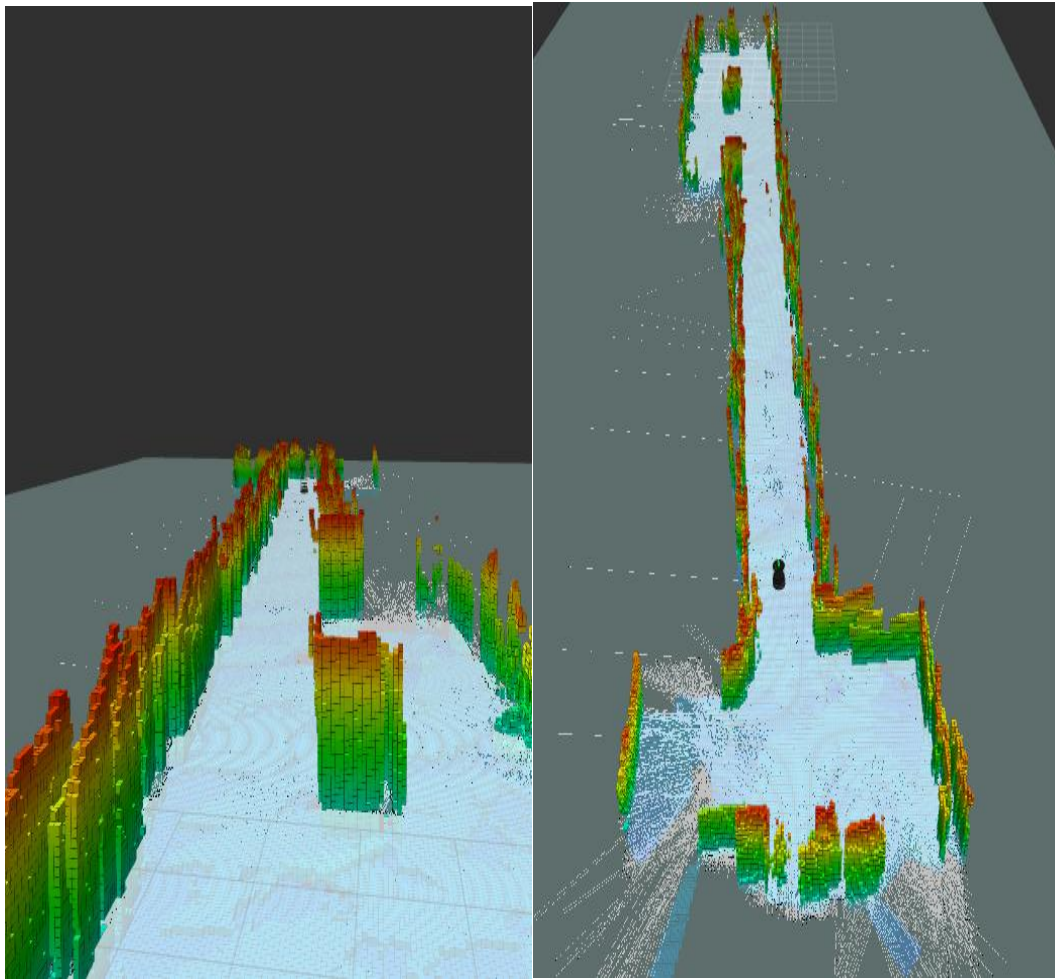


图 6.14 变换角度后观察的二维和三维栅格地图

在之前的分析中也说明了，粒子滤波在大场景中，随着构建地图的增减，特征点的增加，系统的运算的复杂负荷呈指数增加，所以这就容易系统的运算速度导致，定位失败，例如图 6-6，是在长廊的实验场景中，由于机器人的定位失败导致原本应该是一面墙壁，但是机器人却在地图中显示穿过了墙壁，定位失败将导致很严重的结果，由于机器人不能在此系统下重新根据观测值重定位，并重新构建新的地图，这将导致整个地图的失败。

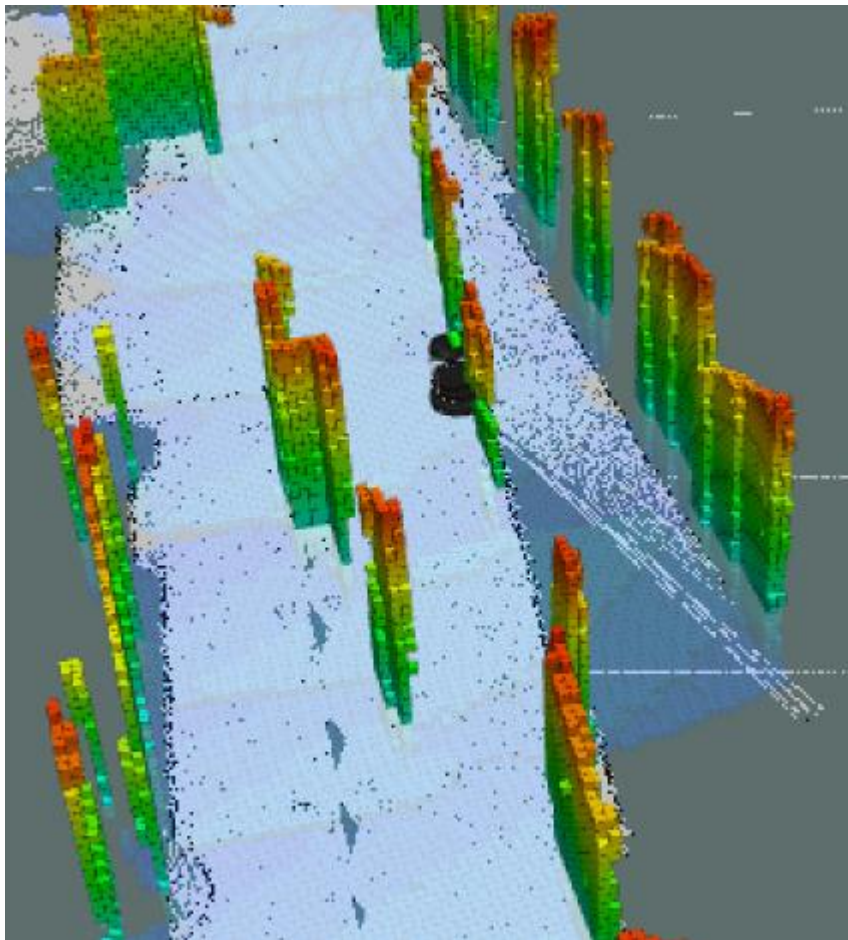


图 6-6 机器人定位失败导致构建地图错乱

6.5 回环检测

在闭环问题中，是指机器人在运行的过程中通过某一未知区域，需要通过路标的方式或者数据关联的问题解决机器人是否在之前的运动过程中来到这个位置，这就是在 SLAM 中的回环检测问题，一般情况下，有基于位姿图的优化方法，在位姿图的方法中，说明位姿图的节点是运动过程中机器人的位姿，连接节点的边是位姿之间的约束关系，也就是位姿间的变换矩阵，通过构建这种位姿图限定节点之间的相互约束关系，来减小位姿之间的误差，提高准确性。

在基于视觉的 SLAM，使用词袋用来存储之前数据中的关键特征，这是由前端视觉里程计提取图像特征点，并将特征点存储为词袋的形式，以供给后端优化以及回环检测部分，在后端优化中通过不断的对比新采集数据中的关键特征，来检测机器人是否回到了之前的位置，但是在本课题是基于粒子滤波的方法并没有严格意义的回环检测的工作，但是在 RBPF-SLAM 算法中通过粒子集合的多样性来维持相关性，所以在实验中地图的闭合环路的能力依靠粒子数的大小，在不同的

实验场景中，需要根据实际情况，改变粒子的数目增强本系统的闭合回路能力，采样集合粒子的多样性导致了更好的闭环性能，因为新的传感器测量值能够进一步影响之前的机器人的位姿。所以在该算法中使用的改进的建议分布确保更少的粒子在重采样中被淘汰，但是这一步也会增加系统的运算负荷，导致粒子滤波的 SLAM 在大场景中无法维持长距离的相关性，这也是粒子滤波算法应用在 SLAM 领域的弱点，所在在比较大场景中，计算量逐渐的增大，就容易导致定位失败，与此同时，这种方法没有重定位的能力，所以一旦地图构建失败将无法重新给机器人定位。但是在程序中通过改进的建议分布，提高粒子之间的相关性，同时保持粒子的多样性是十分重要的。

6.6 小结

本章主要是介绍实验的过程以及对各个开发包的详细的配置的流程，从实验的结果来看，该工作不仅发挥了激光雷达用于机器人定位的优势，并结合深度相机获取的点云实时的生成三维八叉树地图，可用于三维空间中机械臂的碰撞检测以及路径规划，本课题中将机器人的系统状态的后验概率分解为路径估计部分也就是所谓的机器人的定位，和以路径估计为基础的地图估计部分，其中路径估计也就是定位部分使用粒子滤波算法，地图估计部分使用 EKF 算法，并将第一部分粒子滤波定位信息结合深度相机构建三维地图，如此可以能够获得更丰富的三维环境地图，并结合两种地图可视化显示出更丰富的地图效果。

第七章 总结与展望

本文是涉及了一种基于粒子滤波的方法并结合当下研究者最常用的两种传感器构建 SLAM 的问题，粒子滤波器定位也可以称之为蒙特卡罗定位，粒子滤波的核心思想是用一组滤波器来估计当前移动机器人处于某位置的概率可能，其中每个粒子可以都可以代表机器人位置的可能性，之后使用观测值对每个粒子进行加权传播，从而使最有可能位置的概率越来越高。利用激光定位的粒子滤波的方法具有定位精确的优点，融合深度相机获取更为丰富的三维物体信息，同时构建二维和三维的地图，以便用于装配机械臂的移动平台的路径规划和机械手的空间路径规划等问题。

基于粒子的 SLAM 算法在实现的时候主要要考虑：地图表示的方法，一般需要根据实际场景去选择，信息感知问题，就是要选择在合适的场景下的传感器，在本课题中地图的选择都是占用栅格地图都是为了能够在室内场景中应用地图导航，而在传感器的选择上，不仅仅使用了一种传感器而是结合两种传感器的优势生成更加精确并能反应周围环境的地图，数据关联问题：数据关联是对两个环境特征在不同时刻的检测进行匹配，用来确定他们是否对应环境中的同一个特征，数据关联性的计算主要用于新观测环境特征的检测，环境特征的匹配，地图匹配这三个任务。

参考文献

- [1] 李人厚, 宋青松. 自主移动机器人导论[M]. 西安: 西安交通大学出版社. 2013.
- [2] 罗荣华, 洪炳铭. 移动机器人同时定位与地图创建研究进展[J]. 机器人, 2004 (26): 183-186
- [3] Durrant-Whyte. H, Bailey. T, simultaneous localization and mapping part I[J], IEEE Robotics and Automation Magazine, September 2006, 13(2): 99-120;
- [4] J.J. Leonard, H.F. Durrant-Whyte, Mobile robot localization by tracking geometric beacons, IEEE Transaction on Robotics and Automation, Vol. 7, pp: 376-382, 1991
- [5] Borenstein J, Everett B, Feng L. Where am I? Sensors and methods for mobile robot positioning[C]. Michigan: The University of Michigan, 1996: 1-60.
- [6] Leonard J J, Durrant-Whyte H F. Directed sonar sensing for Mobile Robot Navigation[C]. Boston. MA: Kluwer Academic, 1992, 1-30.
- [7] Google. Project tango. URL [https:// www. Google. Com / a-Tap / project tango /](https://www.google.com/a-tap/project-tango/)
- [8] Jafri R. A GPU-accelerated real-time contextual awareness application for the visually impaired on Google's project Tango device[J]. Journal of Supercomputing, 2017, 73(2): 1-13.
- [9] Mitsou N, Tzafestas C. Maximum Likelihood SLAM in Dynamic Environments[C]// IEEE International Conference on TOOLS with Artificial Intelligence. IEEE, 2007: 152-156.
- [10] Santos J M, Portugal D, Rui P R. An evaluation of 2D SLAM techniques available in Robot Operating System[C]// IEEE International Symposium on Safety, Security, and Rescue Robotics. IEEE, 2014: 1-6.
- [11] Kohlbrecher S, Stryk O V, Meyer J, et al. A flexible and scalable SLAM system with full 3D motion estimation[C] IEEE International Symposium on Safety, Security, and Rescue Robotics. IEEE, 2011: 155-160.
- [12] Giorgio Grisetti, Cyrill Stachniss, and Wolfram Burgard. Improving Grid-based SLAM with Rao-Blackwellized Particle Filters by Adaptive Proposals and Selective Resampling, In Proc. of the IEEE International Conference on Robotics and Automation (ICRA),
- [13] Giorgio Grisetti, Cyrill Stachniss, and Wolfram Burgard. Improved Techniques for Grid Mapping with Rao-Blackwellized Particle Filters, IEEE Transactions on Robotics, Volume 23, pages 34-46, 2007
- [14] K. Konolige, G. Grisetti, R. Kummerle, B. Limketkai, R. Vincent, Efficient Sparse Pose Adjustment for 2D Mapping, In Proc. of Int. Conf. on Intelligent Robots and Systems (IROS), Oct. 2010.
- [15] E. B. Olson. Real-time correlative scan matching. In International Conference on Robotics and Automation, pages 4387-4393, 2009.
- [16] Bruno Steux and Oussama El Hamzaoui: tinySLAM : a SLAM Algorithm in less than 200 lines of C code, Accepted for the International Conference on Control, Automation, Robotics and Vision (ICARCV), 2010
- [17] Durrant-Whyte H, Roy N, Abbeel P. A Linear Approximation for Graph-Based Simultaneous Localization and Mapping[C]// Robotics: Science & Systems VII, University of Southern California, Los Angeles, Ca, Usa, June. DBLP, 2011: 41-48.
- [18] Hess W, Kohler D, Rapp H, et al. Real-time loop closure in 2D LIDAR SLAM[C]// IEEE International Conference on Robotics and Automation. IEEE, 2016: 1271-1278.
- [19] Davison A J, Reid I D, Molton N D, et al. MonoSLAM: real-time single camera SLAM[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2007, 29(6): 1052-1067.

- [20] Klein G, Murray D. Parallel Tracking and Mapping for Small AR Workspaces[J]. Proc.of IEEE & Acm Int.sympo.on Mixed & Augmented Reality, 2007:1-10.
- [21] 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.
- [22] Welch G. An introduction to the Kalman filter[J]. Course Notes of Acm Siggraph, 1995, 8(7):127-132.
- [23] Castellanos J A, Martinez-Cantin R, Tard, et al. Robocentric map joining: Improving the consistency of EKF-SLAM[J]. Robotics & Autonomous Systems, 2007, 55(1):21-29.
- [24] JULIER S J, UHLMAN J K, DURRANT-WHYTEH F. A New Method for the Nonlinear Transformation of Means and Covariance in Filters and Estimators[C]. IEEE Transactions on Automatic Control, 2000, 45(3):477-482.
- [25] Grisetti G, Kummerle R, Stachniss C, et al. A Tutorial on Graph-Based SLAM[J]. IEEE Intelligent Transportation Systems Magazine, 2011, 2(4):31-43.
- [26] MOURIKIS A I, ROUMELIOTIS S I. A multi-state constraint Kalman filter for vision-aided inertial navigation Proceedings of the 2007 IEEE International Conference on Robotics and Automation (ICRA). Roma, Italy, 2007.
- [27] MOURIKIS A I, ROUMELIOTIS S I. A dual-layer estimator architecture for long-term localization Proceedings of the 2008 Workshop on Visual Localization for Mobile Platforms at CVPR. Anchorage, Alaska, 2008.
- [28] LEUTENEGGER S, FURGALE P, RABAUD V, et al. Keyframe-based visual-inertial slam using nonlinear optimization Proceedings of 2013 Robotics: Science and Systems (RSS). Berlin, Germany, 2013
- [29] Zhou G, Fang L, Tang K, et al. Guidance: A visual sensing platform for robotic applications[C]// IEEE Conference on Computer Vision and Pattern Recognition Workshops. IEEE, 2015:9-14.
- [30] Chen, Liang-Chieh, et al. "Semantic image segmentation with deep convolutional nets and fully connected crfs." arXiv preprint arXiv:1412.7062(2014).
- [31] Noh, Hyeonwoo, Seunghoon Hong, and Bohyung Han. "Learning deconvolution network for semantic segmentation." Proceedings of the IEEE International Conference on Computer Vision. 2015.
- [32] Qinrui Yan, "Monocular Vision based Object Level Simultaneous Localization and Mapping for Autonomous Driving ", Master Thesis of KU Leuven, Belgium, Sep 2016
- [33] F. Endres, J.Hess, D. Cremers et al. An Evaluation of the RGB-D SLAM System[C].In Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA), 2012.
- [34] Kümmerle R, Grisetti G, Strasdat H, et al. G2o: A general framework for graph optimization[C] IEEE International Conference on Robotics and Automation. IEEE, 2011:3607-3613.
- [35] J. Stuckler and S. Behnke. Model learning and real-time tracking using multi-resolution surfel maps[C]. in AAAI Conf. on Artificial Intelligence (AAAI), 2012.
- [36] Endres F, Hess J, Sturm J, et al. 3-D Mapping With an RGB-D Camera[J]. IEEE Transactions on Robotics, 2017, 30(1):177-187.
- [37] Kalman R E. A new approach to linear filtering and Prediction Problem. Journal of Basic Eng(ASME), 1960, 82D:95-108
- [38] Bucy R S, Renne K D. Digital synthesis of nonlinear filter[J], Automatic, 1971, 7(3):287-289

- [39] S.J.Julier, J.K.Uhlmann, H.F.Duxrant-Whyte."A New Approach for Nonlinear Transformation of Means and Covariances in Filters and Estimators,"IEEE Trans.Automatic Control,2000,vol.45,march.477-482.
- [40] Geoffrey T, Lindsay K, Iterated Extended Kalman Filter[J]. Advanced Robotics, 2006, 26: pp.185-187.
- [41] Tully S, Moon H, Kantor G, et al. Iterated filters for bearing-only SLAM[J]. Proceedings-IEEE International Conference on Robotics and Automation, 2008:1442-1448
- [42] Doucet A, Freitas J F G D, Gordon N J. Editors: Sequential Monte Carlo Methods in Practice[J]. Springer-Verlag, New York, 2001.
- [43] Gordon, N.J.Salmond,D,J. and Smith,A.F.M,Novel approach to nonlinear/nongaussian state estimation,IEEE Proceeding-F,1993,140(2):107-113.
- [44] 胡士强, 敬忠良.粒子滤波算法综述.控制与决策[J],2005,20(4):361-365
- [45] G. Grisetti, C. Stachniss, and W. Burgard, "Improved techniques for grid mapping with rao-blackwellized particle filters," IEEE Transactions on Robotics, pp. 34-46, 2007.
- [46] S. Thrun, "Probabilistic robotics," MIT Press, 2006
- [47] Smisek J, Jancosek M, Pajdla T. 3D with Kinect[C]// IEEE International Conference on Computer Vision Workshops. IEEE, 2011:1154-1160.
- [48] Willow Graage 公司网站.<http://www.willowgarage.com/>[2013-5-12].
- [49] 张建伟, 张立伟, 胡俊等, 开源机器人操作系统—ROS[M],北京: 清华大学出版社, 2009.
- [50] Alberto Elfes. A Sonar-Based Mapping and Navigation System[J]. The Robotics Institute Carnegie-Mellon University Pittsburgh, 1986: 1151-1156.
- [51] 李群明, 熊蓉, 褚健.室内自主移动机器人定位方法研究综述 [J].机器人 2003(25-6): 560-56.
- [52] O. Aycard, F. Chappillet, D. Fohr, and J. Mari. Place Learning and Recognition Using Hidden Markov Models. In Proceedings of the IEEE International Conference on Intelligent Robots and Systems. 1997: 1741-1746.
- [53] Hornung, Armin, Wurm, et al. OctoMap: an efficient probabilistic 3D mapping framework based on octrees[J]. Autonomous Robots, 2013, 34(3):189-206.
- [54] Marzat J, Moras J, Plyer A, et al. Vision-based localization, mapping and control for autonomous MAV: EuRoC challenge results[J]. 2015.
- [55] Liu D, Pan R, Zhang Y. Research of SLAM for Indoor Environment based on Kinect[C]// The Workshop on Advanced Research & Technology in Industry Applications. 2016.
- [56] Jadidi M G, Gan L, Parkison S A, et al. Gaussian Processes Semantic Map Representation[J]. 2017.
- [57] <http://www.turtlebot.com>.
- [58] http://wiki.ros.org/slam_gmapping
- [59] <http://wiki.ros.org/rplidar>

致 谢

三年研究生时光飞逝而过，即将收拾行囊离开校园踏入新的生活。在此，我想真诚的感谢那些曾经帮助过我的人。

我要感谢我的导师范衡教授，他学识渊博、待人和蔼，严谨求实的科研风格深深的影响着我。感谢他对我循循诱导，他善于培养学生的科研能力，让我在科研的路上迅速成长。感谢他对我在学习上和生活中的支持和信任，通过让我做项目、当助教、搞助管锻炼我解决问题的能力，使我的研究生活过得充实饱满。

感谢朱贵杰博士师兄、李文姬博士师兄、容毅标博士师兄、赵雷师兄和谢红辉师兄、肖杨师兄等的帮助，是他们在学术上给予我诸多的指导，为我树立了学习的榜样。感谢李冲、王宇鹏、方毅、王继彪等实验室的小伙伴，是他们陪伴着我一起快乐的学习。

感谢我的两位同门研究生们，感谢我的同班同学和我室友，是他们陪伴着我一起学习一起生活。

最后，我要感谢我的父母和家人，这些年来是他们一直支持着我的学业，让我在学习上没有任何顾忌，他们是我学习的最大动力。

姚利

2018 年 4 月 汕头大学

攻读硕士学位期间主要的工作成果

发表论文

- [1] A SLAM with Simultaneous Construction of 2D and 3D maps Based on Rao-Blackwellized Particle Filters(Accepted by The 10th International Conference on Advanced Computational Intelligence 2018)
- [2] 3D mapping of Multi-Floor Buildings based on sensor fusion (accepted by International Conference on Industrial Informatics-computing Technology)

申请专利

- [1] 范衡, 姚利, 朱贵杰, 李冲. 一种智能停车库中存取车路径规划的方法、装置及存储介质, 201711155275.2 (在审)
- [2] 姚利, 范衡, 朱贵杰. 一种物流产品自动计数及控制设备-201720027159. (有权)